www.tutorial4us.com

Tutorial4Us

College Projects

```
void main()
{
 int no;
 clrscr();
 printf("Hello");
}
```

Basic Program

C Tutorial

JAVA Programming

tutorial4us.com

A Perfect Place for All Tutorials Resources

Core Java | Servlet | JSP | JDBC | Struts | Hibernate | Spring

Java Projects | C | C++ | DS | Interview Questions | JavaScript

College Projects | eBooks | Interview Tips | Forums | Java Discussions

For More Tutorials Stuff Visit

www.tutorial4us.com

A Perfect Place for All Tutorials Resources

Advance Java

(K V Rao (Notes)

www.tutorial4us.com

5/10/09

In java programming scale there are three modules 1. J2SE / JSE = java2 flatform standerdedition / desktop app / two-tire app

2. J2EE / JEE = Java2 flatform Enterprice edition

3. J2ME / JME = java2 flatform micro eddition

1. J2SE / JSE :- (Latest version 7.0 name as dolphin)

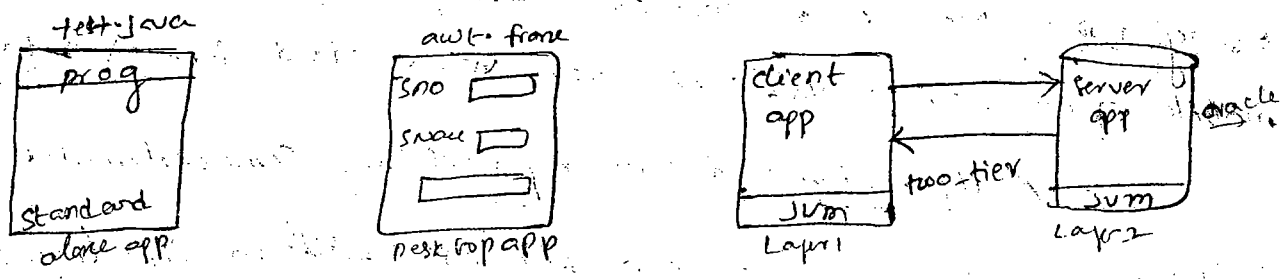
J2SE Module is given to develop standalone, Desktop, and two-tier applications.

- The application can contain only one layer is called standaloneapplication.

- The GUI standaloneapplication is called desktop applicat eg awt.

- The Application at n tires two layers residing in the same computers & two diff computers is called two tier applicat

- A layer is a logical partition in an application.



Alternate technology for J2SE module is VB, .Net, PB (power builder), D2k (Developer 2000)

2. J2EE / JEE :- Version 6 web application, Enterpriseapp, n-tier app
eg (website) (credit/debit cards) (enterprises applicati)

J2EE module is given to develop website, enterprise application and n-tier application.

The application not contains largescale, complex and heavyweight, applicatio logic is called enterprise application

the application has multiple layers is called n-tier app

J2SE is the base module for J2EE, J2ME module.

- Servlets, JSP, EJB, JMS (Java messaging service), JNDI (Java Naming and Direct Interface), Java mail and etc are the J2EE technologies.

- Most of the java based projects we will developed by using J2EE technologies

3. J2ME/JME :- to Develop micro app, mobile app, (micro = 10^{-6}, nano = 10^{-9})

If devices are running and taking decission automatically i.e called artificial intelligent for this we need to embeded programs in to electronic chips and attach these clips to various types of devices to develop these programs by using java we need to take the support of J2ME module using J2ME Module we can also develop mobile applications like mobile games, simcards software, value added services and etc.

- Fully automated washing mecline, Semi automated washing mecline are the examples of artificial Intelligence based devices.

6/10/09

Reflection API :- (working with java.lang.reflect package)

In any language if an API

C → API → set of functions, comes in the form of files

C++ → API → set of functions, classes available in files

Java → API → set of classes & Interfaces comes in the form of packages.

What is an API?

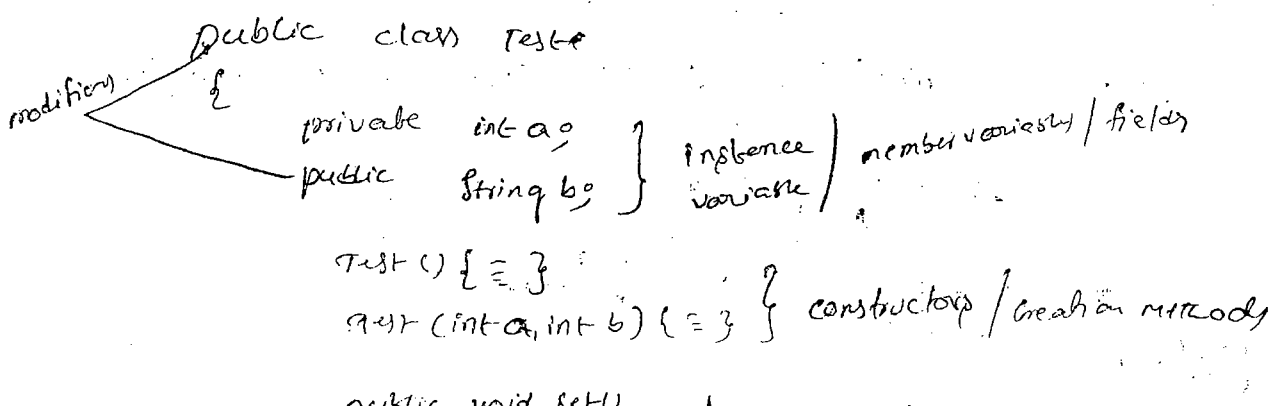
:- API provides base for the Application program to develop certain language base/technology based Applications. In java language API means set of classes and interfaces that com in the form of packages. awt API, IO API, Utility API, reflection JDBC API, and etc are the API's in java programming language every API is given to develop certain purpose applications usin Language & technology. awt API is given to develop java bas GUI applications. working with awt API is nothing but d worki with class & interfaces of java.awt pack, java.awt.event packag

- All java API will done along with J2SDK software in the form of rt.jar file (jar file is java based compressed file like in zip file)

- API is very usefull to develop user defined API & to develop application, ~~java API is are very used~~

- Reflection means mirror image in order to gather interna details even java class & interface dinamically & programatically we take the support of reflection API. java applications developed

- working with Reflection API is nothing but developing java applications by using classes & interfaces of java.lang.reflec package



The member variables of a java class are technically called as fields.

- In java prog language there is no word called access specifier all special keywords public, private, protected, final, synchronise, transient, static and etc are called modifiers/access modifiers.

- java.language default package of java language is, sub packages are not default pack so java.lang.reflect pack is not default pack of java.

- How to gather details certain java class/interface without using reflection API Based programming ?

- there are 3 approches for this

1. by using JavaP (Java profiler)

~~javap java.lang.System~~ javap java.lang.System
javap java.awt.Button
javap java.lang.Runnable

2. by using API Documentation

Note :- We need to download by API documentation saparately from java.sun.com website in the form of jar file.

③ By opening source code of java API classes and interfaces.

- use src.zip file of J2SDK software installation.

Reflection API
(or)
Software project

s/w products
- classbrowser
- GUIBuilder
- Debugger
(very useful to develop)

service sector projects.

Software company creates new s/w technology

Ex IBM, microsoft, micro system etc

- client organizations give project to s/w development compi for development. Ex: LIC, SBI, etc

- s/w Develop compines gives s/w technologies given by s/w vendors to dev s/w projects for client organisations Ex: Batavies, CTS, wipro, bots etc

- s/w Develop company is developing s/w project exclusively for certain client organization then it is called Service Sector projects. The project is developed as a common product for multiple client organizations then it is called s/w produ Ex tally, ms office etc

7/10/09

s/w products :-

- Reflection API is use-full to develop s/w products that gathers and shows details of a java class through hint box is called classBrowser.

- The tool that allows to develop GUI applications through drag and drop operations is called GUI Builder application.

- The tool that allows to trace the flow of execution of a java application is called Debugger.

- Using Reflection API we can perform the following activites.

1) Gathering info details about given java class or interface like class name, super class name, membervariable name, method names and el

2) Creating - of a java class whose class name is not known until (create a java class object without using new keyword

3. Call java method without knowing method name and argument details until runtime.

4. Creating an array without knowing array name, type, size details until runtime.

5. Get & set values for number variables without knowing their names and types until runtime.

- In order to develop reflection API based application to gather internal details about given java class & interface we need to load that class & interface from the memory to java application, to do this java application uses

forName(-) method of java.lang.Class

Eg: Class.forName("java.lang.String"); (interface; can't create object or instance)
Class.forName("java.lang.Runnable");
Class.forName("ABC");

Prototype of: Class.forName(-);

public static Class.forName(String name) throws ClassNotFoundException.

→ Factory method → Class c = Class.forName("java.lang.String"); (points to loaded class or interface)
→ Throws checked Exception.

c is object of java.lang.Class
it is not object of java.lang.String

- Class.forName() loads given class & interface from memory at run time into java application and returns object of java.lang.Class this object represents or points to the class or interface i.e loaded, so using this object we can gather all the internal details about that class & interface (which is loaded from the memory)

Ex:- Thread t = Thread.currentThread(); for factory method
String s = String.valueOf();
Calendar c = Calendar.getInstance();

- The java method of a java class i.e capable of constructing it's own java class object is called Factory method. generally factory methods are static methods.

8/10/09

- If a java method return type is classname that method returns either object of that class or the object one of it's sub classes.

- java method return type is interface name the method returns one implementation class object of that Interface.

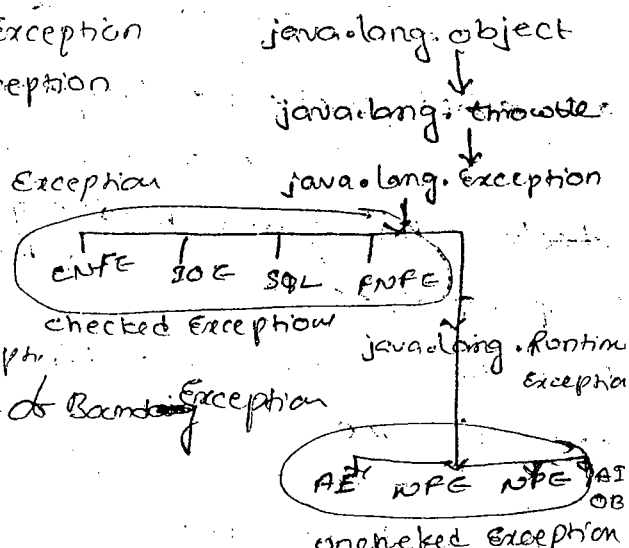
→ what is the use of factory method.?

- when java class is having only private constructors we can't create object of that class, out side the class by using new keyword. in this situation in take the support of factory method to create object of this kind of classes.

Ex java.lang.Class is having only private constructor. so to to create object of this class to depend upon factory method called forName().

- what is dibberence blw checked Exception & unchecked Exception

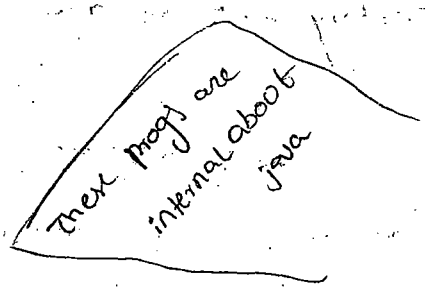
CNFE = classNotFoundException
IOE = InputOutputException
SQLE = SQLException
FNFE = FileNotFound Exception
AE = ArrayException
NFE = NumberFormat NotFound Excep
NPE = NullPointer Excep
AIOOBE = ArrayIndexOutOfBound Exception



→ the immidiate sub classes of java.lang.Exception class are called checked Exception classes.

If a java method throws checked exception we must catch and handle the exception by using try, catch blocks are

we need to declare the exception to be thrown by using throws keyword otherwise compile time syntax will be generated.



Eg1:-
```
class Test
{
  P S V M (String K[])
  {
    try
    {
      class c = class.forName("java.lang.String");
    }
    catch (ClassNotFoundException cnf)
    {
      SOP (cnf.toString());
    }
  } //main
} //class
```

Eg2:-
```
class Test
{
  P S V M (String K[]) throws ClassNotFoundException
  {
    class c = class.forName("java.lang.String");
  } //main
} //class
```

- The sub classes of java.lang.RuntimeException class are called unchecked Exception classes.

If a java method throws uncheckedException ~~check~~ catching and handling that exception during method called is purely optional

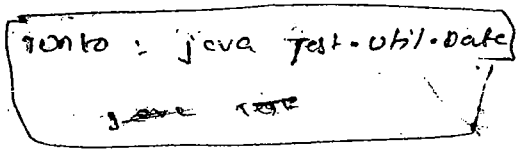
Note:- Though checked exception will be raise at runtime not at compile time.

- It is always recomended to catch and handle the exception while working with java statements that throws exception because this process avoids abnormal program termination. It is alway recomended to avoid declaring the Exception to be thrown by using throws keyword because

this process can't stop abnormal program termination when Exception is risd.

class.forName() throws classNotFoundException when specified class/interface is not available in the memory to loa

Q:- class test
{
psvm (String k[]) throws Exception
{
Class c = class.forName(~~java.lang.String~~)(k[0]);
Sop (k[0] + " is Interface ?" + c.isInterface());
}
}



Q:2 class test
{ psvm (String k[]) { try
{
Class c = class.forName(k[0]);
Sop (k[0] + " is interface ? =" + c.isInterface());
}
catch (classNotFoundException cnf)
{
Sop (cnf.String());
}
} // main
} // class.

Q — when java dose not support pointers why dose throws null pointer Exception?

— when a programmer call method on a reference variable that holds null the application throws java.lang.NullPointerExceptio

the word pointer in this Exception class name is no way related with C, C++ pointers.

It represents regular dictionary meaning says method is called on a variable that holds null.

Test t;
t.abc(); // Throws NullPointerException.
Test t = new Test();

Φ → write a prog to get superclass name of given class.

* when object of java.lang.class points to certain class or interface to get that class name or interface name through object of java.lang.class we need to call _getName()

codes:- 1. Load the given class from memory in to java application

Class c = Class.forName("java.lang.String"); (forName points to c)

2. call getSuperclass() on [c] obj

Class sc = c.getSuperclass();

[sc] is the object of java.lang.class pointing to superclass of java.lang.String class which is referred by [c] object.

3. Points the name of superclass

String sname = sc.getName();

Note:- To gather basic internal details about given java class or interface object of java.lang.Class is sufficient but to gather advanced details we need to work with classes and interfaces of reflection API (java.lang.Reflect package)

```
// Appl.java
/* prog to get Superclass name to given class
   author : Team-5
   version : 1.0 */
public class Appl
{
    public static void printSuperclassName(String name)
    {
        try
        {
```

```
// Load the class from the memory to java app
    class c = class.forName(name);
// call getSuperclass
    class sc = c.getSuperclass();
// print superclass name
    String cname = c.getName(); // gives given class name
    String sname = sc.getName(); // give superclass name of given class
    SOP (cname + " extends " + sname);
}//try
catch (ClassNotFoundException cnf) // to handle known exception
{
    cnf.printStackTrace();
}
catch (Exception e) // to handle unknown exception
{
    e.printStackTrace();
}
}//method
public static void main (String K[])
{
    printSuperclassName(K[0]);
}//main
}//class.
```

Integer (super)
↓
Number (sub)

~~college~~ (super class)
↓
~~student~~ (derived class)

```
// javac App1.java
// java App1 java.lang.String (exec1)
// java App1 java.lang.Integer (exec2)
// java App1 java.lang.Number (exec3)
// javap App1 java.lang.Integer (procedure)
```

```
//javap App1 java.util.Date
// java App1 App1
```

→ write a java Application to print All the classes of inheritence hirarchy for a given class.

java.lang.Object (C)
↑ extends
Test (C)
↑ extends
Demo (C)
↑ extends
XYZ (C)
↑ extends
ABC (C)

java.lang.Object (C)
↑ extends
java.lang.Number (C)
↑ extends
java.lang.Integer (C)

java.lang.Object (C)
↑ extends
java.awt.Component (C)
↑ extends
java.awt.TestComponent (C)
↑ extends
java.awt.Textfield (C)

code :- 1. Load the class from the memory

```
Class c = Class.forName(java.lang.String)
```

(point)

2. Get Intermediate superclass of a given class

```
Class sc = c.getSuperclass();
```

3. Print all the classes and interfaces hierarchy

```
while(sc != null) {
    String sname = sc.getName();
    SOP("<-" + sname);
    c = sc;
    sc = c.getSuperclass(); }
```

// App2.java

```
/* prog to get All the classes of inheritence hirarchy for a given
   class. author: Team-S ; version : 1.0 */

public class App2
{
    public static void PrintHierarchy(String name)
    {
        try
        {
            Class c = Class.forName(name);
            Class sc = c.getSuperclass();
            SOP("classes of hierarchy for "+name+" class are");
            while(sc != null)
            {
                String sname = sc.getName();
                SOP("<-" + sname);
                c = sc;
                sc = c.getSuperclass();
            } // while
        } // try
        catch (ClassNotFoundException cnf);
        {
            cnf.printStackTrace();
        }
        catch (Exception e)
        { e.printStackTrace(); }
```

```
} // method
public static void main(String k[]).
{
    printHierarchy(k[0]);
} //main
} // class.
```

// javac App2.java

// java ~~Long~~ App2 java.long.Integer , java App2 java.applet.Applet, java App2 java.awt.TextFi

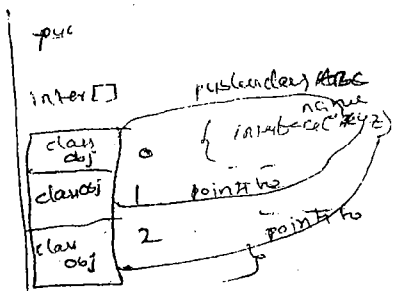
Ø

→ what is the diff b/w exception object.toString and exception object.printStackTrace?

- Exception object.toString just print the exception class name when exception is raised the exception object.printStackTrace() gives the route cause of the exception rising by providing elaborated details regarding the exception i.e raised, every java method execute using stack method when exception is raised during method execution the method stack will be filled up with messages, the exception object.printStackTrace method print all these stack messages

→ wit a program to print all the interfaces implimented by a given class.

12/10/09



-code :-

1. Load the class from memory

class [c] = class.forName("ABC");

2. call getInterfaces() on object [c]

class inter = c.getInterfaces();

3. print all interfaces names

```
for(i=0; i<inter.length; i++) {
    String iname = inter[i].getName();
    SOP("It is"+iname); }
```

... the above code will array is array of type java.lang.Class that means all objects in this array are class called class object.

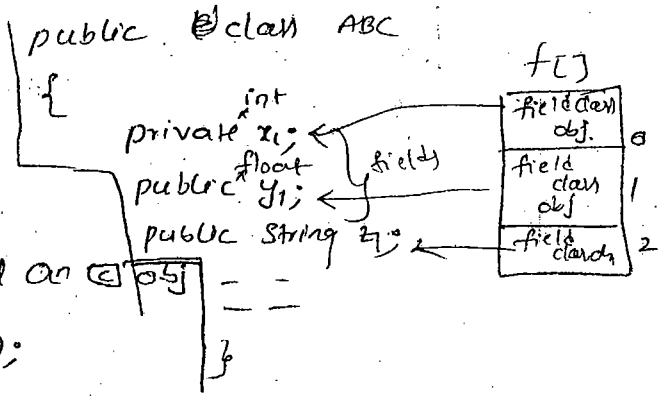
- Object of java.lang.Class can point to concrete class, abstract class, interface and data type.

```
// App3.java
/* prog to get all interfaces of given class
   version: 1.0
   author: Team.S */
public class App3
{
  p s v ~~String~~ listInterfaces(String ~~KE~~ name)
  {
    try
    {
      // load the class
      class c = class.forName(name);
      // call getInterfaces()
      ~~call~~
      class inter[] = c.getInterfaces();
      // print interfaces
      for(int i=0; i<inter.length; i++)
      {
        String iname = inter[i].getName();
        SOP("IF IF" +iname);
      }
    }
    catch(classNotFoundException cnf)
    {
      ~~SOP~~(
        cnf.printStackTrace();
    }
    catch(Exception e)
    {
      e.printStackTrace();
    }
  p s v m (String KE[]);
  {
    listInterfaces(KE[0]);
  }
}
```

// javac App3.java ; java App3 java.lang.String; java App3 java.awt.Button

— wrt a program to gather all the member variables of a (fields) given class.

— To represent number variable of a java class dinamically from java application we need to use object of java.lang.reflect.Field class.



code

① load the class from memory

class c = class.forName();

② call getDeclaredfields[] method on obj

field f = c.getDeclaredfield();

class in java.lang.reflect prog

③ print member variables fields

```
for(int i=0; i<f.length; i++)
{
    String ftype = f[i].getType().getName(); — java.lang.class
    String fname = f[i].getName(); — java.lang.field
    sop(fname + " " + ftype);
}
```

// String s1 = "type"
int x = s1.substring(0,2).length;

// java App4.java

/* prog to get all member variables (or) fields of given class

version : 1.0

author : Team-B */

```
java.lang.reflect.*;
public class App4
{
    p s v printfields(String name)
    {
        try
        {
            // load the class
            class c = class.forName(name);
            // call getDeclarefields
            field f[] = c.getDeclarefields();
            // print field details.
```

```
for(int i=0; i<f.length; i++)
{
    String fname = f[i].getName();
    String ftype = f[i].getType().getName();
    int x = f[i].getModifiers();
    // SOP(x); ==
    String mod=null;
    if(Modifier.isPublic(x))
        mod = "public";
    String mod=null;
    if(Modifier.isFinal(x))
        mod="final";   (or) mod = mod+"final";
    String mod=null;
    if(Modifier.isStatic(x))
        mod="static";   (or) mod = mod+"static";
    String mod=null;
    if(Modifier.isPrivate(x))
        mod="private";   (or) mod = mod+"private";
    String mod=null;
    if(Modifier.isProtected(x))
        mod="protected";   (or) mod = mod+"protected";
    }
        SOP("\t\t"+mod+" "+ftype+" "+fname);
}
catch(classNotFoondException cnf)
{
    cnf.printStackTrace();
}/* catch(Exception e){ e.printStackTrace(); } */
p s v m(String k[])
{
    printFields(k[0]);
}
}
```

only one modifier provided

All the modifiers that are provided

// javac App4.java
// java App4 java.lang.String
// java App4 java.lang.Integer

- Modifier is the java class available in java.lang.reflect package this class is usefull to get the modifiers of a given java class, member variables.

13/10/09

→ wt is the diff blw getfields() & getDeclaredfields() ?

- getfields() gives information about public member variables that are available in the given class and it's super classes in the hierarchy.

- getDeclaredfields() gives information about public and non public member variables that are available in the given class

→ write a program to gather methods details of a given java class

code snippet

1. load the class from memory

```
class c=class.forName("Test");
```

2. call getDeclaredMethods() methods

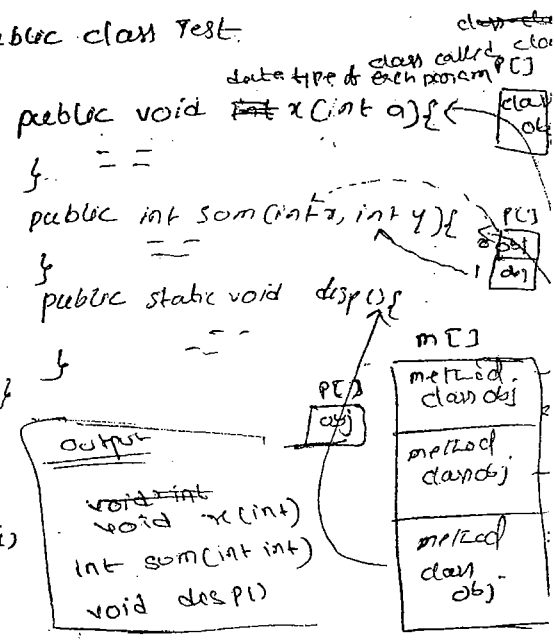
```
Method m[]= c.getDeclaredMethods();
```

(available in java.lang.reflect pack)

3. print methods Details.

```
for(int i=0; i<m.length; ++i)
String sname = m[i].getName();
String rtype = m[i].getReturnType().getName();
class P[] = m[i].getParameterTypes();
SOP(rtype + " " +mname+ "(");
for(int k=0; k<m.length; ++k)
{
    SOP(P[k].getName()+" ");
}
SOP(")");
}
```

```
public class Test
{
public void x(int a){
}
public int som(int x, int y){
}
public static void disp(){
}
}
```



// App5.java

```
/* prog to get method details in java class.
   version:1.0;
   author Team-5; */

public class App5
{
    p s v printMethods (String name)
    {
     try
      {
        class c=class.forName(name); // load the class from mem
        //call getDeclaredMethods()   -- gives public & non public method any information about source
        Method m[]=c.getDeclaredMethods(); (Another method is
                                              getMethods())
                                              -- All the public methods available given class
        for(int i=0; i<m.length; ++i)
        {
           String mname=m[i].getName();
           String rtype = m[i].getReturnType.getName();

           class p[] = m[i].getParameterTypes();
           int x= m[i].getModifiers();
           s o p (rtype+" "+mname+"(");

         /* for (int k=0; k<m.length; ++k)
            {
         *       s o p (p[k].getName()+" "); *
            }
            s o p (")");   */

           String mod="";
           if(modifier.isPublic(x))
              mod= "public";

           if (modifier.isPrivate(x))
              mod= mod+"private";

           if(modifier.isProtected(x))
              mod= mod+"Protected";

           if(modifier.isStatic(x))
              mod= mod+"Static";
```

```
if(modifier.isFinal(x))
    mod = mod+"final ";
if(modifier.isAbstract)
    mod=mod+"abstract ";
if(modifier.isSynchronized(x))
    mod = mod+"synchronized ";
S.o.p(mod+" "+rtype+" "+mname+")");
for(int k=0; k<p.length; ++k)
{
    S.o.p(p[k].getName()+" ");
}
S.o.p(")");
} // outer for
} // try
catch(exception e)
{
  & e.printStackTrace();
}
}
P.S.v.m(String k[])
{// printMethods(k[0]); ---- //
} // main
}
```

```
String s1=new String()
Class c1=s1.getClass();
S.o.p(c1.getName());
--
Class c2=System.class
S.o.p(c2.getName());
```

// javac Apps.java

// java APPS java.lang.Integer; java APPS java.lang.Runnable

javap java.lang.System; javap ~~is~~ APPS APPS

-getMethods() gives information about all the public methods aval in the given class and available in the classes of inheritance hierarchy

-getDeclaredMethods() gives details about both public & non public methods available in the given class

Q → In how many ways we can create object of java.lang.Class ?

There are three ways ① By using Class.forName()

```
Class c = Class.forName("Test");
```

2. By using getClass() of java.lang.object class

```
String s1 = new String();
String c = s1.getClass();
```

here object "c" point to java.lang.String class

```
Button b = new Button();
Class c = b.getClass();
```

here object "c" point to java.awt.Button class

3. By using .class keyword

```
Class c = String.class;
```

here object "c" point to java.lang.String class

```
Class c = Button.class
```

here object "c" point to java.awt.Button class

Note :- most regularly used technique to create the object of java.lang.Class is Class.forName()

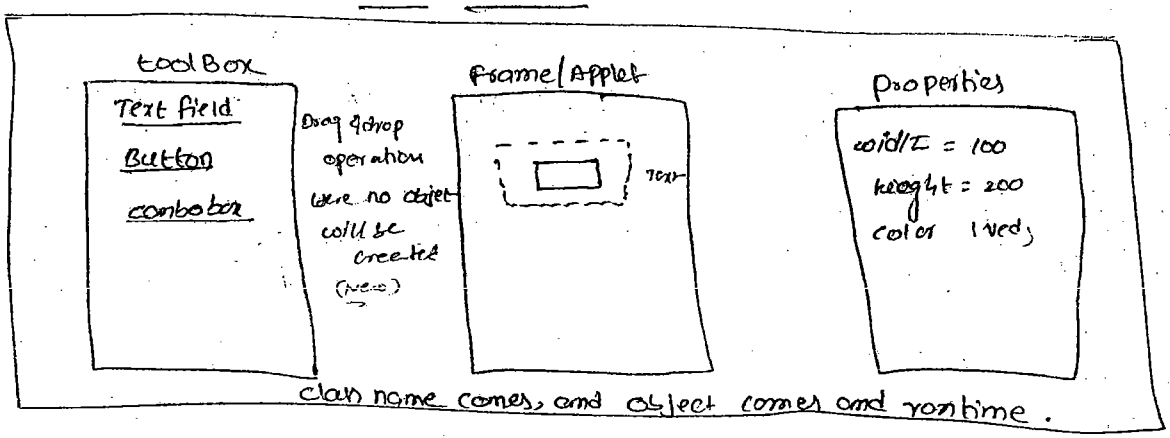
14/10/09

→

```
class Test
{
    ~~p s v m (String k[])~~
    {
        int a=10;
        int b=20;
    public void disp()
    { S O P ("a="+a+" b="+b); }
    }
    P S V m (String k[])
    {
        Test t=new Test();
        t.disp();
    } }
```

- The new keyword / operator creates object of the given class at runtime but expects presents of that java class at compiletim that means we can use new keyword to can create object of a java class whose class name is known at compiletime we can't use new keyword to create object of a class that comes at runtime as dinamic value.

GUI Builder



- In GUI builder applications the class name will be clicked from toolbox at runtime, it's object will be created when drag & drop operation will done at runtime, in this situation 'new' keyword is not suitable for object creation.

- 'new' keyword is not suitable to create the object of java class whose name is known only at runtime, the application like GUI builder demands this kind of object creation so we need to use some special technique given by reflection API to create object of java class without using 'new' keyword when class name is given dynamically at runtime.

```
public class Test
{
    int a;
    int b;
(constructor) Test() {
    SOP("zero arg calculation"); }
    public String toString() {
    return "a=" +a+ "b=" +b; } }
```

(O)

```
class Test {
    int a = 10;
    float b = 1.35f;
    public Test() {
    SOP("a=" +a+ "b=" +b+
        "constructor : Test"); }
    public String toString() {
    return "a=" +a+ "b=" +b;
    }
```

① Load the class from memory to runtime.

```
Class c = Class.forName(args[0])
```

(points to → Test)

② class newInterface (constructor)
newInstance() method

```
Object obj = c.newInstance();
```

It is a method available in class

③ print data of the object obj

```
S.O.P("obj." + obj.toString());
```

— In the above code c.newInstance() takes class name pointed by object c and creates object of that class (here test class object)

newInstance() is available in java.lang.class

//

//objTest.java

/* prog to create object of a java class whose name is not known until runtime */

```
public class objTest
{
  P S V M (String k[])
  {
    try
    {
      // load the class
      class c = class.forName(k[0]);
      // call newInstance()
      Object obj = c.newInstance();
      // print data of object
      S.O.P(obj.toString());
    }
    catch (classNotFoundException cnf)
    {
      cnf.printStackTrace();
    }
    catch (InstanceException ie)
    { ie.printStackTrace(); }
```

```
        catch(Exception e)
        {
            e.printStackTrace();
        }
    }
// javac objTest.java
// java objTest Test
// java objTest java.awt.Rectangle[x=0,y=0,width=0,height=0]
// java objTest java.lang.NumberFormatException: For input string: "Test"
                    at java.lang.Runnable
```

when "newInstance()" method is used to create object of an abstract class or an Interface the method throws java.lang.InstantiationException

Q → how many ways are to create object of a java class?

(1) By using "new" keyword

```
Test t=new Test()
Data d=new Data()
```

(2) By using factory() method

A method in a java class i.e capable of creating it's own java class object is called factory method. since

Eg1:- `class c=class.forName(-)` : here object [c] is created using factory method forName()

(3) Eg2:- `Thread t=Thread.currentThread()`

Eg3:-
```
public class Test
{
    public static Test mymethod() // factory method.
    {
        Test t=new Test();
        return t;
    }
}
Test t=Test.mymethod();
```

③ By using method of a class that returns other class object.

Ex:- Integer i1 = new Integer(10);

String s = new String();

String s = i1.toString();

Eg:- class Test

{

public ABC mymethode() // special method

{

ABC ab = new ABC();

return ab;

}

}

Test t = new Test();

ABC a1 = t.mymethode(); } here create object of another class. not a factory method

inheriting duplicate object

④ By using cloning operation

use clone() of java.lang.object()

(Preparing new object is called cloning & it may security)

java.lang.cloneableInterface

- cloning operation is allow only on those object that are clonable java object is called clonable object when the object of that class implements java.lang.clonable Interface. this Interface does not contain method so it is called marker interface but implementation of this Interface is mandatory to add clonable behaviour on java class object.

String s = new String("ok"); // s object data is "ok"

String s1 = s.clone(); // s1 object data is "ok"

⑤ By using newInstance() of java.lang.class

class c = class.forName("java.lang.String");

Object obj = c.newInstance(); // creates object of java.lang.String class.

⑥ Through deserialization

The process of capturing data from a file and constructing new object using that data is call deserialization

- the process of capturing data from an object and writing that data to a file is called Serialization.

JDBC

15/10/09

persistance :- the process of storing data in perminent place and using that data in multiple applications is called persistance.

→ the file & s/w that can store data parmenently & that can manage data is called persistance store.

Eg:- Files, DataBase s/w

- Insert, Update, Delete & Read operations performed on the data c persistance stores are called persistance operations & logic used for it is called persistance logic.

Java App $\xrightarrow[\text{(java.io package)}]{\text{IO streams}}$ Files (Flat files)

→ Java application uses java.io.package based I/O streams to interact with persistance store called file.

- Limitations with files as persistance store.

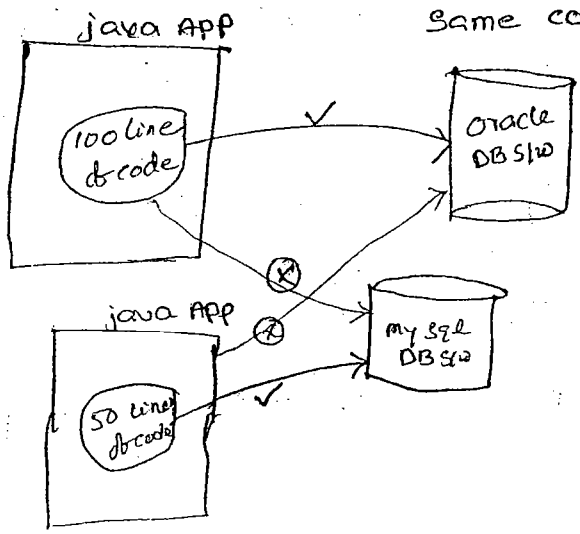
1. No data Sequirity
2. can't maintain huge amount of data
3. No queri Language support.
4. Mearching (searching) & comparison operations are complex operations.
5. Report genaration is complex process

Java App $\xrightarrow{\text{JDBC-API}}$ (java.sql, javax.sql pack) DB s/w {oracle, sybase,

Working with classes & interfaces of java.sql package, javax.sql package is nothing but working with JDBC.

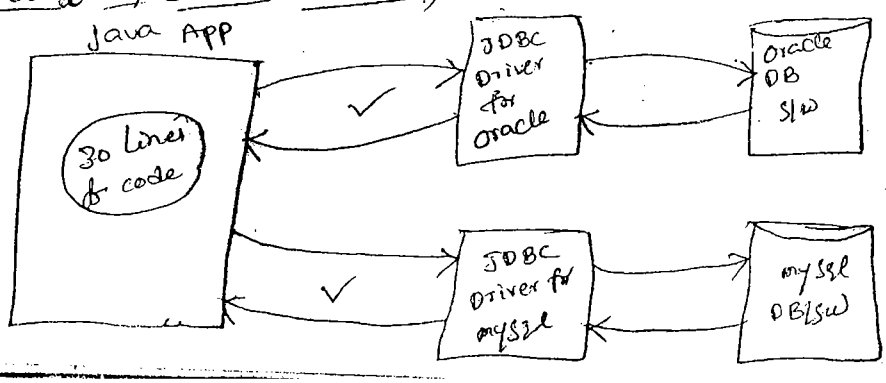
- To Solve the limitations of files it is recomended to take data base s/w as persistance storage.
- In large scale java application, medium scale java Application it is recomanded to work with ~~files or~~ Data Base s/w as persistance store. In small scale java App like mobile App which recomanded to use files as persistance store.
- API of any language & technology provides base for the programmer to develop the applications.
- JDBC API provides set of classes & interfaces as base for the programmer to develop java application interacting with Database s/w.

Approch1 (with out using JDBC Driver)



Same code can't interact with other Data base s/w.
one java s/w interact with one database s/w.

Approch2 by (using JDBC.)



using same functionality of Both s/w.

A java Application can interact with DB s/w in two appr

① Approch1 (without using JDBC)

② Approch2 (using JDBC)

In Approch1 java App interacts with database s/w and directly so the code writtening java App becomes specific to the DB s/w. and can't be use to interact with another DB s/w.

"JDBC Driver is a Bridge s/w b/w Java App and DB s/w it converts java call to DB calls & DB calls to java calls.

- All JDBC Drivers are given based on common rules and guidlines of JDBC specification given by sunmicroSystems.

A specification is a document contains rules and guidlin to develop the s/w, JDBC specification contains rules & guidlines to develop JDBC Drivers for DB s/w.

In Approch2 java App is using DB specific JDBC Driver to interact with DB s/w. since all JDBC Driver for different DB s/w are given based on common JDBC specification the way we work with all JDBC Driver in our java application is similar.

Q→ wt is JDBC?

"JDBC is an API specification that contains rules, guidlines and classes, interfaces to develop JDBC drivers and to make java Application interacting with DB s/w".

Every DB s/w needs a sapoarate JDBC Driver but all these JDBC Drivers are given based on common JDBC specification.

There are 2 types of specifications

1. Open specification.

2. propritary specification.

After Designing rules & guidlines of specification if they are kept in public market and ~~alone~~ allowed all ~~any~~ company to develop s/w based on specification. then that specification is called open specification

Ex: JDBC specification
ODBC "
Servlet " and etc

If s/w are given based on specification by a company who as prepaid the specification and not allowing other companies to develop the s/w based on the specification then that specification is called propritary Specification

Ex: .Net Specification

- S/w Vendor company means a company who develops Designs new s/w.

Ex: Micro soft, Sonmicro system etc

Since JDBC is an open specification we can expect JDBC Drivers from different/vendors

1. Sonmicro systems.
2. DB vendors.
3. CAR parking vendors

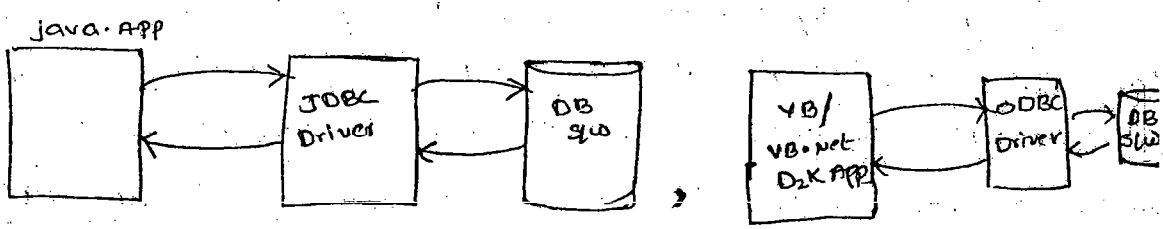
Note: It is allways recomended to work with JDBC Drivers supplied by DB vendors.

~~16/6/10~~ 16/10/10

Q → What is ODBC ?

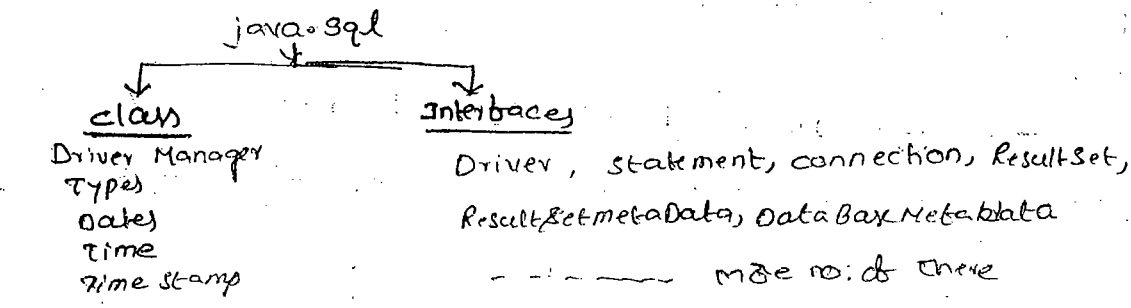
ODBC is an API specification containing rules, Guidlines to develop ODBC Drivers. Each ODBC Driver is Specific to one DB s/w. Similarly each JDBC Driver is Specific to one DB s/w.

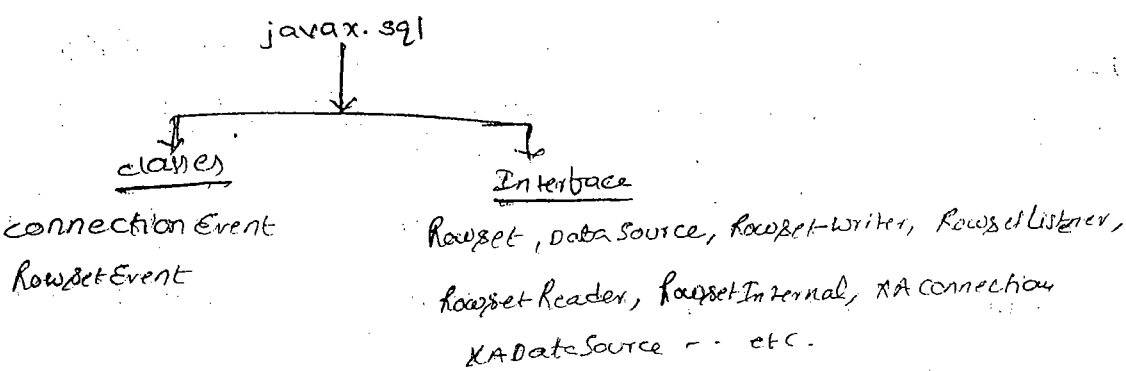
- java applications use JDBC Drivers for database interaction. Similarly other then java applications use ODBC Drivers for DB interaction.



Q → When all s/w technologies are using ODBC Drivers for DB interaction why dose java application expect saparately JDBC Driver ?

- ODBC Drivers are given based on 'c' language utilities pointers, but java dose not support pointers. so java application can't use ODBC Drivers. ODBC Drivers are arrived to the market before the arrival of JDBC Drivers.





Implementing interfaces of various packages of JDBC API is not the responsibility of programmer it is the responsibility of the vendor companys how give JDBC Drivers. These vendor companys provides lot of classes implementing various interfaces of JDBC API packages (java.sql, javax.sql) all this class together is called JDBC Driver

— Sun microSystem gives JDBC as a specification containing rules & guidlines to develop JDBC Driver s/w s. The methods declared in the interfaces and (abstract()) Declared in the abstract class of JDBC API packages are rules of JDBC specification concrete() definations available in classes and abstract classes of JDBC API packages are called guidlines of JDBC specification

— since all JDBC Drivers of different DB s/w are given based on common JDBC specification rules and guidlines the way we worked with all this JDBC Driver from our java application is going to be much similar.

— Every JDBC Driver is identified with it's Driver classname. The java class that implements java.sql.Driver interface is called JDBC Driverclass.

- SunmicroSystem gives a built in JDBC Driver along with J2SDK software. This built in JDBC Driver class is having JDBC Driver class called sun.jdbc.odbc.JdbcOdbcDriver (package name . classname)

- A Built in JDBC Driver of J2SDK s/w provides lots of classes implementing interfaces of JDBC API packages.

- Every JDBC Data Base comes having set of classes that are implementing various interfaces of JDBC API packages. But These class names will change based on the JDBC Driver s/w but will never use these implementation class names in our java JDBC programs.

19/10/09

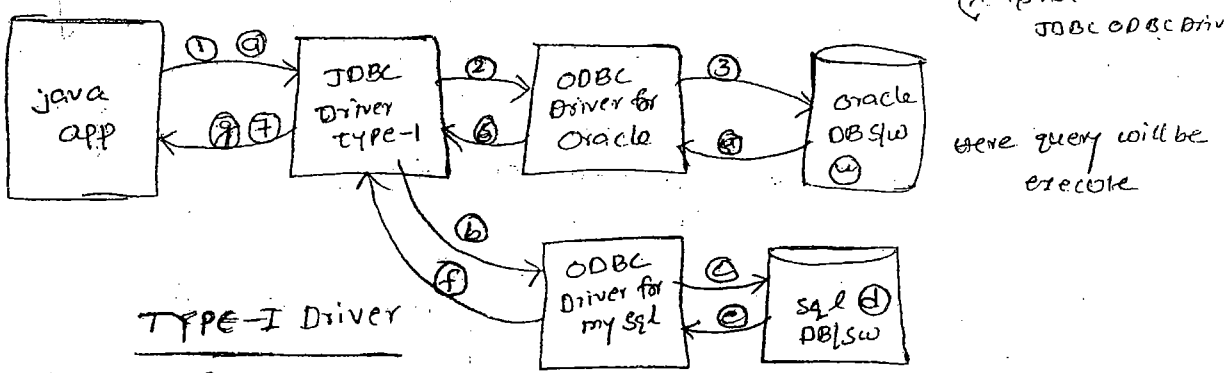
→ Vendors, s/w companys can develop JDBC Drivers based on any mechanism like

TYPE I
TYPE II
TYPE III
TYPE IV & etc

Only SunMicroSystem is giving type-I mechanism based JDBC Driver. TYPE-I mechanism based JDBC Driver is design to interact with DB s/w throw an appropriate ODBC Driver.

The built in JDBC Driver that comes with JSDK s/w installation is Type-I Mechanism based JDBC Driver.

(it is not full flexible JDBC ODBC Driver)



TYPE-I Driver

- SunMicroSystem as given type-I mechanism based JDBC Driver in the initial days of versions of java this Driver is design to interact with any data base specific ODBC Drivers. So relationship b/w JDBC driver type-I & ODBC Drivers is 1:n.

JDBC Driver type-I is not specific to any DB s/w, but Each ODBC Driver is specific to one DB s/w

- JDBC Driver type-I is bridge b/w java application and ODBC Driver s/w

→ Every ODBC Driver is identify with it's ~~Driver~~ DSN (Data Source Name)

There are 3 types of DSN :-

1. User DSN (visible current login user of windows)
2. System DSN (visible to all the users of windows)
3. File DSN (sharable in the network)

Every JDBC Driver is identified with it's driver class name

Note :- Windos operating system instalation gives lot of ODBC Drivers.

Procedure to create DSN for ODBC Driver for oracle.

start button arrow
↓
Settings arrow
↓
control pannel
↓
administrative tools
↓
Data sources arrow
↓
User DSN tag arrow, ~~add arrow~~
↓
add arrow
↓
microsoft ODBC for oracle
↓
finish
↓
Kasi (Any name)
↓
username scott
password tiger.
(Discription & Server is optional) required for centralized oracle s/w instalation.
↓
OK (DSN created)

DSN

→ Standard steps to follow while developing JDBC Based java application.

Step.1 :- Load JDBC Driver class and register the driver class Obj. with ~~Driver Manager Service~~ DriverManager Service.

② Establish the connection with DB s/w (Establish the connection.)

③ Create statement object and send sql queries to DB.

④ make SQL query execute in the DB s/w.

⑤ Gather result from DB s/w and process that result

⑥ close connection with DB s/w---

→ DriverManager is a special service to manage JDBC Drive and to establish connection with DB s/w by using the JD Drivers.

Every JDBC Driver has to be register with DriverMan Service. for this we need to place JDBC Driver class object in DriverManager Service.

→ the java.sql.DriverManager class Represents DriverManager Servi

→ Establish connection with DB s/w from java application is nothing but creating communication channel b/w java application and DB s/w.

→ Statement Object is a corier services to send sql queries to DB s/w, to make them executing DB s/w and to brin result from DB s/w.

→ Once work with DB s/w is compleated it is recomended to close the connection b/w java application & DB s/w.

→ the DriverManager Service chooses an appropriate JDBC Driver from the set of register JDBC Drivers and uses a driver establis the connection with DB s/w.

*

→ wrt prog to establish connection oracle DB s/w by using type-1 mechanism based JDBC Driver (built in JDBC Driver of J2SDK s/w).

```
// conntext.java
import java.sql.*;
public class connText
{
    P S V m (String x[]) throws Exception
    {
        //Load driver class and register that driver class obj with
          DriverManager service
        Class c=Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");   ①
        S O P("jdbc driver class is loaded register with DriverManager
                                                        service");
        Connection con = DriverManager.getConnection("jdbc:
odbc:kasi", "scott", "tiger");
                ②          ③       ④
*       S O P("class named conn", conn.getClass().getName());
        if(con!=null)
        S O P("connection is establish");
        else
          S O P ("connection is not established)
    } // main
} // class.
```

① jdbc Driver classname of SunmicroSystem supply built in jdbc Driver.

② Data Base url having DSN.

③ Data Base user name.

④ Data Base password.

→ Setup required :- Any version of J2SDK s/w, Any version of oracle DB s/w.

→ for compilation :- javac conntext.java

→ Execution :- java conntext

20/10

Connection conn = DriverManager.getConnection("jdbc:odbc:kavi", "scott", "tiger");

url: jdbc:odbc:kavi
- jdbc:odbc — protocal
- kavi — Data source Name (DSN)

jdbc = major protocal
odbc = Miner protocal.

→ Protocal is a set of roles followed by two parties that wants to participating communication.

There are two types of protocals

① Network level protocal.

Eg:- TCP/IP

② Application level protocal

Eg:- HTTP, SMTP, jdbc:odbc etc

Network level protocals are given to get communication b/w two physical computers.

Application level protocals are given to get communication b/w two s/w are s/w applications.

jdbc:odbc is the application level protocal to get communication b/w jdbc Driver type-1 & odbc driver. in that jdbc is major protocal and odbc is sub (minor) protocal.

Q → Connection conn = DriverManager.getConnection("jdbc:odbc:DSN", "scott", "tiger")

- In the above statement how can you say "conn" is an object when java.sql.Connection is an Interface?

- In the above statement conn is not the object of java.sql.connection interface it is the object of the class that implement java.sql.connection interface.

When java method return type is an interface name method does not return object of the interface method actually returns one implimentation class object of that interface.

→ The getConnection() of DriverManager class return type is java.sql.Connection that means this method definition returns one implementation class object of Connection interface.

→ when interface ~~difference~~ refers implementation class object the reference variable becomes object of implementation class. Similarly the above statement "conn" is refering implementation class object of java.sql.Connection interface returned by "DriverManager.getConnection()" so "conn" is called object of a class that implements java.sql.Connection interface. but the implementation class name will change based on the jdbc Driver ~~they~~ we use.

→ To know classname of any object dynamically use object.getclass.getName

→ To know classname of jdbc connection object write following step

```
S o P ("classname of conn obj", conn.getclass.getName());
```

run: sun.jdbc.odbc.JdbcOdbcDriver.java

→ We can manually create jdbc Driver class object and we can register that object with DriverManager service by using "DriverManager.registerDriver()"

```
Sun.jdbc.odbc.JdbcOdbcDriver od=new sun.jdbc.odbc.JdbcOdbcDriver();
DriverManager.registerDriver(od);
```

```
Connection conn=DriverManager.getConnection("jdbc:odbc:DSN","scott",
                                            "tiger");
```

- In the above code there is no need of Class.forName() that loads jdbc Driver class.

→ ~~object~~ constructor is object level one time execution block where as static block is class level one time execution block when jvm loads class in to application from memory the static block of class executes in this static block in general initialize static member variables of a class.

→ constructor executes when object of a class is created in constructor both static & non static data is visible. whe multiple objects are created by a class a static block of class executes only for one time but constructor of the class executes for multiple number of times.

```
/* class Test
{
  static
  { SOP("static block of Test class"); */
```

⇒ When programmer creates object for a java class (first obj) the static block of class constructor of class will execute. The remaining object creation only constructor will execute.

Ex:-
```
class Test
{
    static
    {
        s o p ("static block of Test class");
    }
    public Test()
    {
        s o p ("constructor of Test class");
    }
}
public class App1
{
    p s v m (String a[]) throws Exception
    {
        Test t = new Test();
        Test b = new Test();
        Test t3 = new Test();
        Test t4 = new Test();
    }
}
```

javac Test.java
java Test
output: static block of Test class
constructor of Test class
" "
" "
" "

1 → what happens internally when jdbc Driver class is loaded into the java application by using class.forName() ~~like this?~~ follow class.forName("jdbc.odbc.JdbcOdbcDriver"); ? ("sun.jdbc.odbc

Any:- class.forName() loads Driver class from memory → this process automatically executes static block of jdbc Driver class

→ this static block contain code to create jdbc Driver class object and code to register Driver class object with DriverManager service by using DriverManager.registerDriver(). due to this programmer is not bothered about registry jdbc Driver explicitly with DriverManager service.

→ the static block of sun.jdbc.odbc.JdbcOdbcDriver class will looks as shown bellow.

```
static
{
   JdbcOdbcDriver jdbcodbcdriver=new JdbcOdbcDriver();
   try
    {
      DriverManager.registerDriver(jdbcodbcdriver);
    }
    catch (SQLException sqlexception)
    {
       sqlexception.printStackTrace();
    }
} // static block
```

SQL queries

(Data Retrive language) Select(DRL) — a bunch of statements will be store

public ResultSet executeQuery(String qry) throws SQLException

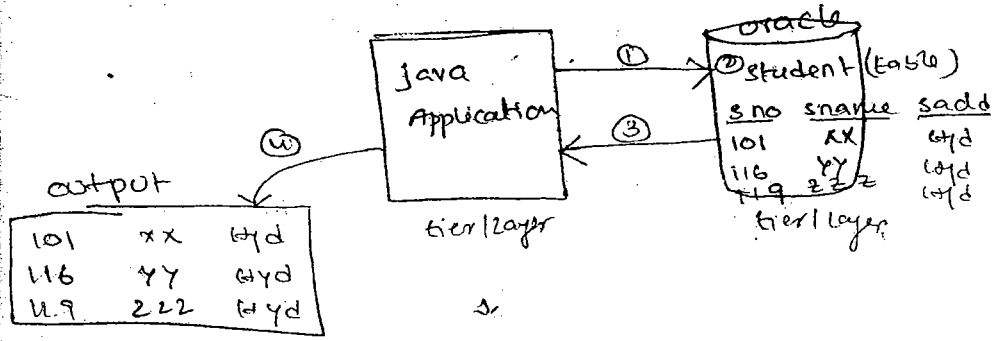
non Select(DDL, DML)

public int executeUpdate(String qry) throws SQLException.

→ The select SQLqueries Return bunch of records selected from DB table. all this records can be store in a single java object called ResultSet Object, to execute select SQLqueries use executeQuery().

→ Non-select sqlqueries return an integer number indicating the no: of records that are effected, to execute these queries use executeUpdate()

→ ResultSet object is a object of a class that implements java.sql.ResultSet interface, ResultSet object store bunch of records given by select SQLquery, Statement object means is a object of a class that implements java.sql.Statement interface.



→ wrt a prog to select all the records of a DB table (oracle 9i)

Setup:- Any version of J2SDK 9i (J2SDK 1.5)
Any version of DB 9i (oracle)
DSN for ODBC Driver for oracle
Table is student.

SQL> create table student(sno number, sname varchar2(20), sadd varchar2(20));

SQL> table created.

```
sql> insert into student values (101, 'xx', 'Hyd');
    1 row created

sql> insert into student values (116, 'yy', 'Hyd');
    1 row created

sql> insert into student values (119, '222', 'Hyd');
    1 row created.

sql> select * from student

  sno    sname    sadd
  101    xx       Hyd
  116    yy       Hyd
  119    222      Hyd
```

* All JDBC Applications are two-tier applications.

code snippet :-

① Load JDBC Driver class

```
class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
```

② Establish connection with DB s/w

```
Connection conn = DriverManager.getConnection("jdbc:odbc:DSN",
                                              "scott", "tiger");
```

③ Create Statement Object

```
Statement st = conn.createStatement();
```

(Statement — Interface; object of a class that implements java.sql.Statement)

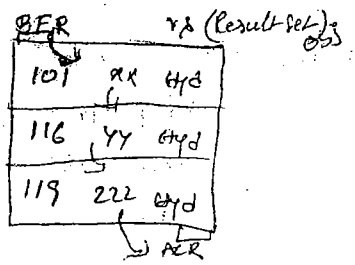
④ Send SQL query to DB s/w, execute it in DB s/w and get Result from DB s/w.

```
ResultSet rs = st.executeQuery("select * from student");
```

(Before first Record)

⑤ process ResultSet and display Result.

```
while(rs.next()){
    rs.next();
    int no = rs.getInt(1);
    String name = rs.getString("sname");
    String add = rs.getString(3);
    SOP(no + " " + sname + " " + sadd);
}
```



⑥ close JDBC connection with DB s/w. also close other JDBC objects.

```
rs.close();
st.close();
conn.close();
```

→ Every ResultSet object contain two procedores

1. BFR (Before first Record)
2. ALR (After Last ")

@ * when the ResultSet object is created the record pointer points to BFR position by default.

- To move the record pointer next position in the resultset object nextmethod on ResultSet object this method returns true if record pointer is move to valid position, this method returns falls when the record pointer was moved to ~~invalid~~ Invalid position like BLR, ALR position.
- To need values from ResultSet object call getxxx() by passing either columnname(s) columnnumber as argument value.
- It is always recomended to throws JDBC object it reverse order of their creation.

→ most of the methods avalible in JDBC API, throws java.sql.SQLException.

```
// SelectTest.java
/* prog to get all the records of a DB table
   version : 1.0
   author : Team-S ; */

public class SelectTest
{
    P S V M (String x[]) throws Exception
    {
        Class.forName ("sun.jdbc.odbc.JdbcOdbcDriver");
        SOP("Driver class is loaded and registerwith DriverManager services");
        Connection conn = DriverManager.getConnection ("jdbc:odbc:DSN",
                                                       "scott", "tiger");
        SOP (" connection establish with DB Slw");
        Statement st = conn.createStatement ();
```

~~ResultSet~~

```
ResultSet rs = st.executeQuery("select * from student");
SOP("query executed result will come");
while(rs.next())
{
    int no = rs.getInt(1);
    String name = rs.getString("name");
    String add = rs.getString("sadd");
    SOP(no+" "+name+" "+add);
}// while
rs.close();
st.close();
conn.close();
SOP("jdbc objects are closed");
}// main
}// class
// javac SelectTest.java
// java SelectTest
```

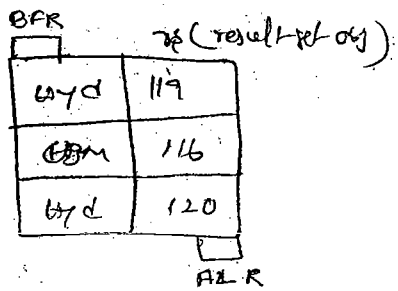
"JDBC is avalible in J2SDK s/w so to run above application basic J2SDK setup enough"

* Java JDBC are not capable of reading on commited data of a DB Table by default * that means Java JDBC application can need commited data of the table.

→ In Realworld projects type-1 based JDBC Driver will not be used they or use type-4 or type-3 with Type-4 mechanism based JDBC Drivers.

→ While working with SQL select query it always recomanded to count the column index in the order they are selected from the sql query, not in the oder they are available in DB table.

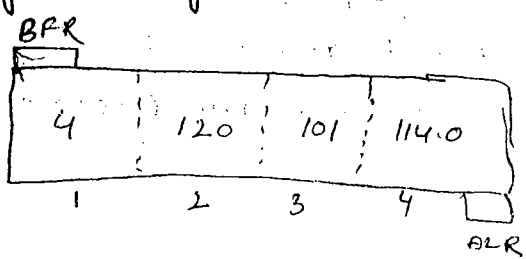
```
ResultSet rs = st.executeQuery("select sadd, sno from student");
S O P ("query Executed and result has come");
while(rs.next())
{
    String addrs = rs.getString(1);
    int sno = rs.getInt(2);
    S O P (addrs + " " + sno);
}
```



Note :- It is always recomanded to get data from ResultSet object by using column names as argument value of rs.getXXX() methods.

→ when select query contains agrigate fonctions then the result store in resultset object must be retrived by using column indexes as rs.getXXX() methods argument values.

```
ResultSet rs = st.executeQuery("select count(*), max(sno), min(sno),
                                avg(sno) from student");
if(rs.next())
{
    S O P ("count of rows is " + rs.getInt(1));
    S O P ("count of max value " + rs.getInt(2));
    S O P ("count of min value " + rs.getInt(3));
    S O P ("avg is " + rs.getFloat(4));
}
```



→ while executing select sql query with conditions it is recomanded to collect condition values to outside the java application to make java application has flexible application a java application can gather by input values in 2 ways

1. As command line arguments like args[0], arg[1] & etc.

2. As values collected from keyboard by using streams like buffer reader. fileinput stream etc.

① Eg:- // read condition values are command line arguments.

```
int min=Integer.parseInt(args[0].trim());
int max=Integer.parseInt(args[1].trim());

ResultSet rs=st.executeQuery("select * from student
                    where sno>="+min+" and sno<="+max);
while(rs.next())
{
    int sno=rs.getInt("sno");
String name=rs.getString("sname");
String add=rs.getString("sadd");
 s o p ("no+" "+name+" "+add);
}
```

→ Example of getting condition values of select query from keyboard.

```
// read condition value from keyboard.
BufferedReader br=new BufferedReader(new InputStreamReader
                                    (System.in));
S O P ("Enter starting student number");
int min=Integer.parseInt
String tno=br.readline();
int min=Integer.parseInt(tno.trim());
S O P ("Enter end student number");
```

```
tno=br.readLine();
int max=Integer.parseInt(tno.trim());
ResultSet rs=st.executeQuery("select * from student where
                    sno>="+min+" and sno="+max);
while(rs.next())
{
    s o p(rs.getInt("sno")+" "+rs.getString("sname")+" "
                    +rs.getString("sadd"));
}.
```

~~when you~~

→ when you don't know to retrive that column values from Resultset object take the support of rs.getString()

→ JDBC allows to work with subquery, query inside the query is called subquery to pass condition value of main query based on other query execution we take the support of subquery

Eg:- `ResultSet rs=st.executeQuery("select * from student where sno=(select max(sno) from (student)");`

```
while(rs.next())
{
    s o p(rs.getString("sno")+" "+rs.getString("sname")+" "
                    +rs.getString("sadd"));
}.
```

```
Select * from student where rownum>=1 and rownum<=3;
```

→ Every table of oracle contains one ~~pse~~ pseudo colomn called rownum as invisible colomem holding record numbers.

we can use oracle pseudo column to select table rows based on their row numbers are record numbers.

JDBC allows to work with this pseudo column in SQL query preparation.

Eg: ResultSet rs = st.executeQuery("select * from student where rownum >=1 and rownum <=3");

```
while(rs.next())
{
    SOP(rs.getInt("sno") + " " + rs.getString("sname") + " " + rs.getString("sadd"));
```

23/10

→ java JDBC Applications are frontend applications developed for non-technical end user interact with DB s/w and to manipulate DB table data

→ Anything that is part of application and not visible to endusers (operators of the application) is called Backend s/w. all DB s/w are Backend s/w

→ Anything i.e part of the application and used by end users to interact with backend s/w is called frontend, the backend s/w responds for the request given from frontend.

→ Java JDBC applications are frontend applications to interact with backend s/w called DB s/w.

→ Any application that contains two layers are uses to JVM (or) uses 2 s/w is called two tier applications. since java application interact with DB s/w by using JDBC the JDBC applications are called two tier applications.

Q→ what is the difference b/w performing operations on DB from sql prompt & from java application.?

Ans:- only Technical programmers can perform data base operations from sql prompt. non technical programmers (or) end user can't to do this work.

The JDBC applications developed by the java programmer can be used by end users are non technical programmers to perform operations on DB. with out knowing about java language & SQL queries.

prog
→ write a java application to delete records of a table

Note: Delete query is a non select query so use "execute update()" method. to execute this query

code snippent

1. Load JDBC Driver class

```
class.forName("son.jdbc.odbc.JdbcOdbcDriver");
```

2. Establish the connection with DB s/w.

```
connection con=DriverManager.getConnection("jdbc:odbc:DSN"
                                           "scott","tiger");
```

3. Create statement object

```
Statement st=con.getCreateStatement();
```

4. Send query to DB s/w and make DB s/w execute query

```
int result=st.executeUpdate("delete from student where
                            sno=110");
                        (or) sno>110");
```

5. Print the result

```
if(result!=0)
    sop(result+" no: of record are deleted")
else
    sop("no records of deleted");
```

6. close JDBC connection other objects.

```
st.close();
con.close();
```

→ "finally" block execute irrespective of exception raised in try block. it is always recomanded to place the code of releasing non java resources like connection w/r DB s/w connection with files in finally block

→ In JDBC application development it is recomanded to close all JDBC objects in finally block.

→ To avoid java.lang.NullPointerException in an application development it is always recomanded to call java method by ensuring that the object is not pointing to null value

```
Ex: Test t
    t.xyz(); wrong (NullPointerException)

    Test t
    if(t!=null)
      t.xyz(); ✓
```

Program

```
//DeleteTest.java
/* prog to delete records of a table */
import java.sql.*;
import java.io.*;
public class DeleteTest
{ public static void main(String [C])
  {
    Connection con=null;
    Statement  st=null;
    BufferedReader br=null;

    try
    { // read condition value from keyboard:
      String tno=null;
      br=new BufferedReader(new InputStreamReader(System.in));
      S.o.p("Enter sno to delete record");
      if(br!=null)
        tno=br.readLine();

      int no=Integer.parseInt(tno.trim());
      // write JDBC code...
      class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
      con=DriverManager.getConnection("jdbc:odbc:DSN","scott","tiger");

      if(con!=null)
        st=con.createStatement();
      int result=0;
      if(st!=null)
        result=st.executeUpdate("delete from student
                                 where sno="+sno);
```

```
if(result!=0)
    s.o.p(result+"number of records are deleted");
else
    s.o.p("no records are delete");
}//try
catch(ClassNotFoondException cnf)
{
    ~~s.o.p(~~
    cnf.printStackTrace();
}
catch(SQLException se)
{
    se.printStackTrace();
}
catch(Exception e)
{
    e.printStackTrace();
}
finally
{
    try
    {
        if(br!=null)
            br.close();
    }
    catch(IOException ioe)
    {
        ioe.printStackTrace();
    }
    try
    {
        if(st!=null)
            st.close();
    }
    catch(SQLException se)
    {
        se.printStackTrace();
    }
    try
    {
        if(con!=null)
            con.close();
    }
    catch(SQLException se)
    {
        se.printStackTrace();
    }
}//finally
}//main
```

//javac DeleteTest.java
//java DeleteTest

→ Variables declared inside the try block will not be visible after the try block so declared the variables before the try block & use them inside & after the try block

→ nonselect queries send by JDBC application to DB s/w will always execute having "auto commit mode enabled".

24/10/09

Note:- It is always recomanded to add JDK home\bin directory as first value of path environment variable of mycomputer properties.

Procedure: mycomputer → properties → advanced tab → Environment variables → new

variable name: PATH

variable value: D:\java\jdk1.5.0_10\bin;. ← current directory

→ java JDBC applications execute non select queries in DB s/w by having "auto commit" mode, inorder to disable in auto commit mode on DB s/w from java applications use "con.setAutoCommit(false);"

→ Inorder to commit the query execution result that are executed after disable autocommit mode use "con.commit();", to rollback query execution use "con.rollback();".

Ex code:-

```
con.setAutoCommit(false);
if(con!=null)
    st=con.createStatement();
```

```
int result=0;
if(st!=null)
    result = st.executeUpdate("delete from student where sno=");
con.rollback();(or) con.commit();
```

Prog

→ wit a prog to insert recording to DB table by reading record values f n keyboard.

code snippet:-

① read values from keyboard and store them in to variable.

no → variable holding
name → keyboard data (Input Data)
addr →

② establish connection with DB s/w using type-1 driver

③ create Statement object

```
Statement st = con.createStatement();
```

④ prepare query & execute query by using executeUpdate();

```
String qry="insert into values(101,'raja',25,'hyd')";
String qry = "insert into stud values
                ("+no+","+name+","+addr+")";
sop(qry);
int result=st.executeUpdate(qry);
```

⑤ process the result

```
if(result==0)
    sop("record is not inserted");
else
    sop("record is inserted");
```

⑥ close JDBC object

```
st.close();
con.close();
```

→ since insert query is non select query use executeUpdate() to send query to DB s/w and to execute query in DB s/w.

// InsertTest.java

```
import java.sql.*;
import java.io.*;

public class InsertTest
{
    P S V m (String K[]) throws Exception
    {
        Connection cop;
        Statement st;
        int no;
        String name, addr;

    // Load JDBC driver class and Driver class will register Driver manager service
    class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    // Establish connection
    Connection con = DriverManager.getConnection("jdbc:odbc:DSN",
                                                 "scott", "tiger");
    // read input values from keyboard
    ButterReader br = new ButterReader(new InputStreamReader(System.in));
    S O P ("Enter student number");
    int no = Integer.parseInt(br.readline());
    S O P ("Enter student name");
    String name = br.readLine();
    S O P ("Enter student address");
    String addr = br.readLine();
    // create JDBC object
    Statement st = con.createStatement();
    // prepare query
    // String qry = "insert into student values (100, 'Kasi', 'Hyd')"; Example
    String qry = "insert into student values ("+no+",'"+name+"','"+addr+"')";
```

```
// Execute query
    int result = st.executeUpdate(qry);
// process result
    if(result==0)
        sop("record not inserted");
    else
        sop("record inserted");
// close JDBC objects
    st.close();
    con.close();
} //main
} //class.
```

// javac InsertTest.java

// ~~Intest~~ java InsertTest.

* To insert multiple records at a time we use for loop after creating BufferReader object. (pot entair logic way same)

alter table student add constraint x primary key (sno);

Q → what happens if you try to read data from resultset object once it is closed?

Exception will be thrown (java.sql.SQLException having message SQLException having message resultset is closed

Q → wt happens if you try to read data from resultset object when record pointer is pointing to BFR & ALR position?

java.~~long~~sql.SQLException will be thrown having message invalid cursor state

Note :- for most of the problem that are raised during runtime of jdbc code execution, the java.sql.SQLException will be thrown having different messages.

* when statement object is closed internally closes the related resultset object that are created based on this statement object.

— when jdbc connection object will closed it closes the internal jdbc statement, Resultset object.

Intigrated Development Environment. 25/10

→ IDE s/w provide the environment i.e require java J2EE applications.

→ Developing java application by using IDE s/w is an easy task and also recomanded to do, A

- NetBeans
- MyEclips
- JBuilder
- JDeveloper
- JCreator

} Java based IDE s/w's

RAD (Rational Application Developer) e.t.c.

- **NetBeans**
  - type: IDE s/w for developing java J2EE Application.
  - version: 5.x (J2SDK 1.4), 6.x (J2SDK 1.5/1.6)
  - vender: sunmicrosystem (java s/w)
  - → to download s/w: www.Netbeans.org
  - → for FAQS: www.NetBeans.org
  - opensource s/w

procedure to Develop JDBC Application By using NetBeans IDE :-

Step1

① Lanch NetBeans IDE and create java project

② file → Newproject → java → java Application → projectnam

myproj7

finish

③ Add JDBC Application to applied the record of the table in the project.

Myproject → Right click → New → java classes → class name: updateTest → Finish.

J2EE

Shotcut :-

Psvm + tab space → public static void main

Psf + tab → public static final

sout + tab → system.out.println

Alt + Shift + i → to get import statement

For more shotcut key reffer Help → Keyboard Shotcuts

```
// updateTest.java
import java.io.*;
import java.sql.*;

public class updateTest
{
    p s v m(String x[]) throws Exception
    {
        s o p ("main method");
        // Read input values from key board.
        BufferedReader br= new BufferedReader(new InputStreamReader(System.in));
        s o p ("Enter Student number");
        int no= Integer.parseInt(br.readLine().trim());

        // write JDBC code.
```

```
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection con=DriverManager.getConnection("jdbc:odbc:DSN",
                                           "scott","tiger");
Statement st=con.createStatement();
// String qry="update student set sadd='delhi' where sno=101";
String qry="update student set sadd='"+sadd+"' where sno="+sno;
S.O.P(qry);
int result=st.executeUpdate(qry);
if(result==0)
    S.O.P("no records are updated");
else
    S.O.P(" records are updated");
st.close();
con.close();
}//main
}// class
```

④ Execute the java application:

Rightclick on source file java application → run file

How to pass commandline arg to a java application i.e running in NetBeans IDE

Rightclick on project → Arrow properties → Arrow Run → main class [Updaterec] Arguments [101, London]

Note:- commandline arguments passed as space saparated values

myproject → Rightclick on project → Run

→ Create table, Drop table & etc are called DDL queries

Since these are non select query use executeUpdate() for queries execution.

→ In company level DB team will design table and also perform other DDL operations

→ java programmer is not responsible to perform DDL operations from java applications.

```
import java.io.*;
import java.sql.*;
public class DropTest
{
    p s v m (String KI) throws Exception
    {
        // Read input value from keyboard
    BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
     S op ("Enter table to delete");
      String tab=br.readLine().trim();

      // write JDBC code
     class.forName("son.jdbc.odbc.JdbcOdbcDriver");
    Connection con=DriverManager.getConnection("jdbc:odbc:DSN",
                                    "scott","tiger");
    Statement st=con.createStatement();
    ~~S o p~~ st.executeUpdate("drop table "+tab);
            S o p ("table droped");
        st.close();
        con.close();
    }// main
```

→ JDBC application is performed non-select operation then the application will not contain Resultset object

→ JDBC Driver type-1 industry standards to develop realworld projects..

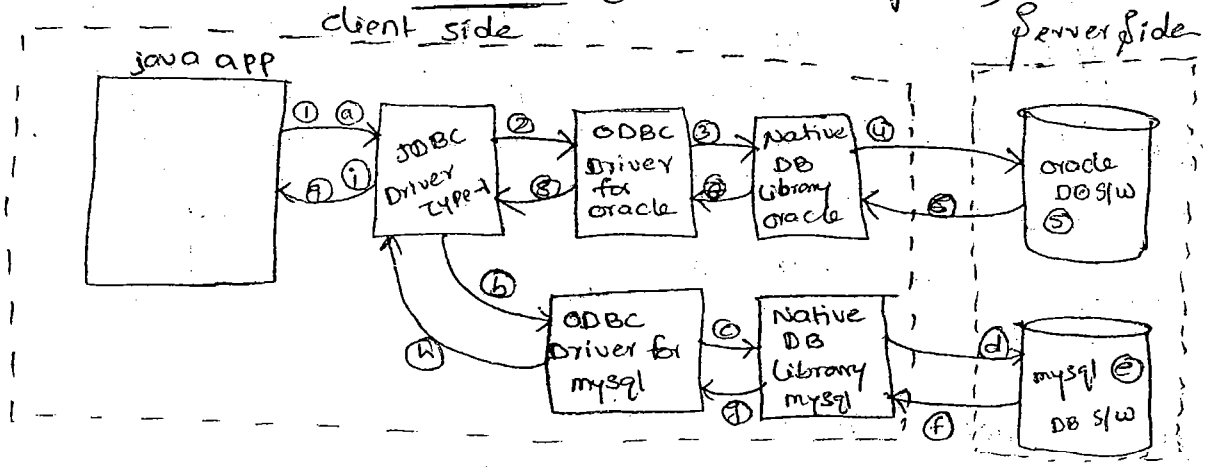
26/10

→ There are 4 different types of mechanisms to develop JDBC Drivers. they are.

1. TYPE-1
2. TYPE-2
3. TYPE-3
4. TYPE-4

Software vendors different mechanisms to develop JDBC Drivers for various DB s/w.

TYPE-1 (JDBC_odbc_bridgeDriver)



→ Vendor DB Library / Native DB Library helps JDBC Driver / ODBC Driver to locaed communicate with DB s/w.

→ Vendor DB Library code will be availble in the Language using which DB s/w is developed. Every DB s/w contains it's own vender DB library and it will be install along with DB s/w instalation.

→ Vendor DB library for oracle will be there in 'c' Language because ~~the vender DB library~~ the oracle s/w is is also developed by using 'c' Language

→ Only Sun micro System is given type-1 mechanism based JDBC Driver as JDBC Driver of J2SDK s/w.

Advantages :- Since odbc Drivers are available almost for all DB s/w you can use JDBC Driver type-1 to interact with any DB s/w.

Since JDBC Driver type-1 is built in JDBC Driver of J2SDK s/w there is no need of arranging JDBC Driver Saparately.

Disadvantages :- * Performence of JDBC Driver type-1 is very poor.

* Since vendor DB library & ODBC Drivers are Required at the client side the type-1 JDBC Driver is not suitable

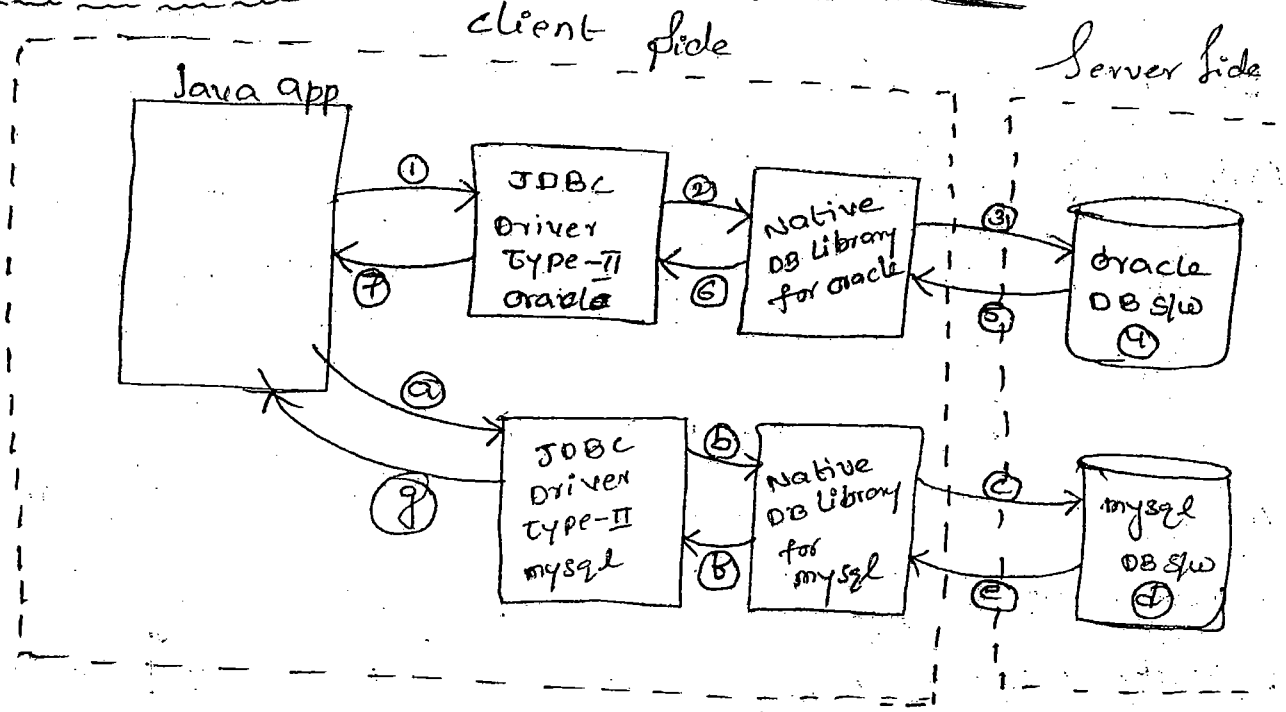
ⓐ For Applet to DB communication

ⓑ For internet based applications

* Becose of poor performence JDBC Driver type-1 is not suitable for large scale applications.

Note :- untrusted applet can't interacted with files available in the mechine for security purpose by default all applets are untrusted applets.

TYPE-II :- (Native API / partly Java Driver)



→ JDBC Driver type-II is given to interact to the DB S/w through Native DB Library.

JDBC Driver type-I is given to interact to the DB S/w through ODBC Driver. where as these ODBC Driver uses Native DB Library to communicate with DB S/w

Advantages :- JDBC Driver type-II gives quit good performence compare to type-I Driver.

- JDBC Driver type-II Dose not need any ODBC Driver.

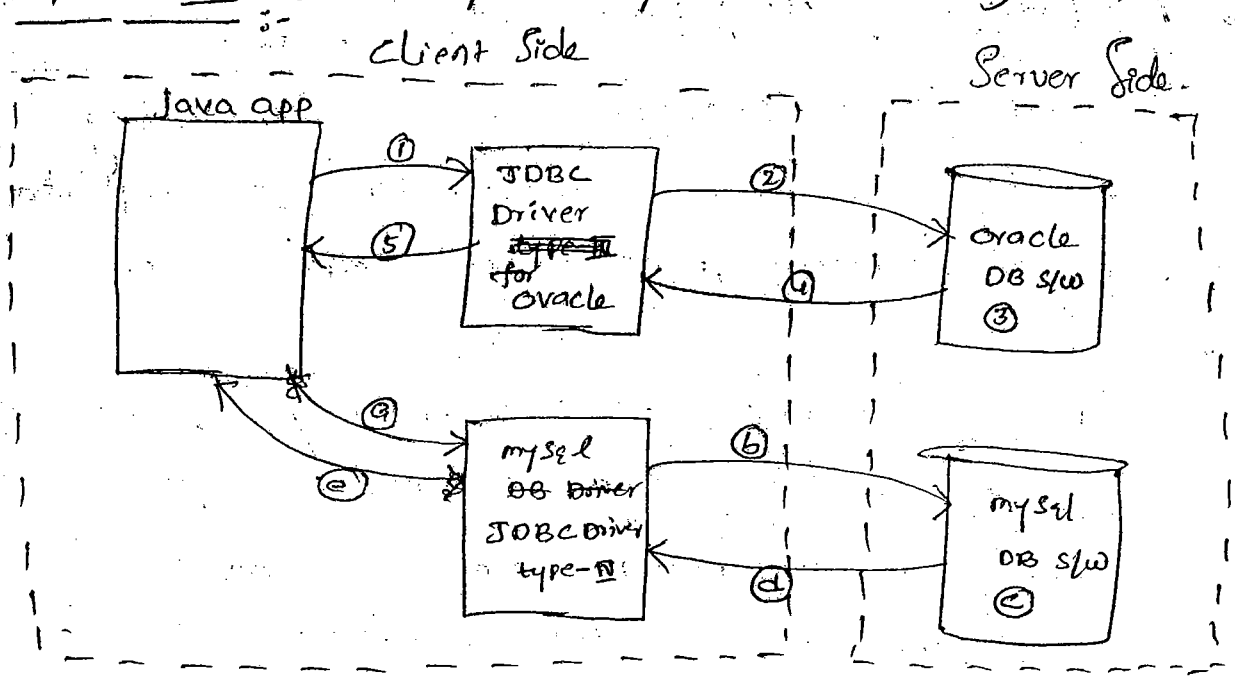
disAdvantages :- Since Native DB Library Prequire at client side JDBC Driver type-II

ⓐ Not Suitable for internet application

ⓑ Not Suitable for applet to DB application.

- For Every DB S/w there is a Need of Saparate JDBC Driver type-II.

TYPE-IV (Native protocal / All java Driver) :-



→ JDBC Driver type-IV is design to interact with DB s/w Directly.

Advantages :- Since type-IV Driver is developed totally in java language this Driver executes Platformindipendency

Type-IV JDBC Driver can be downloaded ~~from~~ from internet dinamically.

- JDBC Driver type-IV use bettor perfomence when compare to other JDBC Drivers.

- Since ODBC Drivers & vendor DB Library is not required on the client mechine.

- JDBC Driver type-IV Suitable for large scale applica and also use applet to DB s/w communication.

- JDBC Driver type-IV is suitable for large scale application.

Disadvantage :- For Every DB there is a need of saparate JDBC Driver type-IV.

→ TYPE-III :- (Netprotocal)

It is not JDBC Driver it is a protocal

Type-III is a protocal i.e is given having set of rules to get the communication b/w java application and JDBC connection (type-III is not a regular JDBC Driver).

JDBC connection pool is a factory that contains set of readily available JDBC connection object.

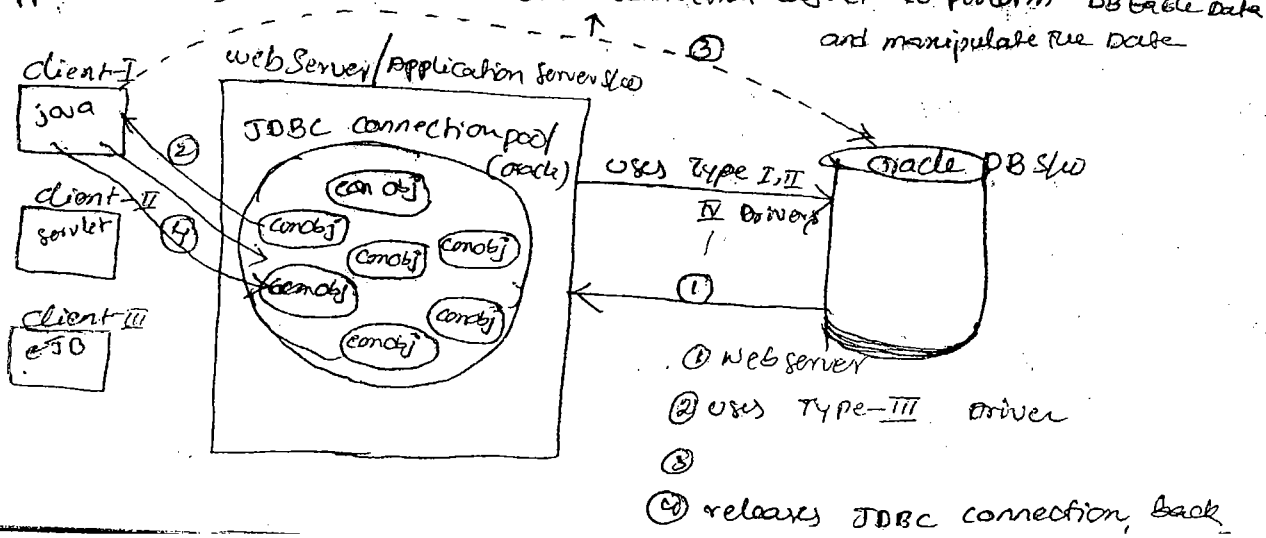
JDBC connection pool for oracle is all JDBC connection of that will represent connectivity with oracle DB s/w.

JDBC connection pool for mysql means all JDBC connection object in that pool represent connectivity with ~~oracle~~ mysql DB s/w.

Advantages of working with JDBC Connection working pool

Programmer is not responsible to create, manage and destroy JDBC connection object, All these operation will be taken care by JDBC connection.

We can use minimum number of JDBC connection object to give service to maximum number of client applications



S – Select	C – Create (Insert)
C – Create	U – Update
U – Update	R – Read (Select)
D – Delete	D – Delete

27/10

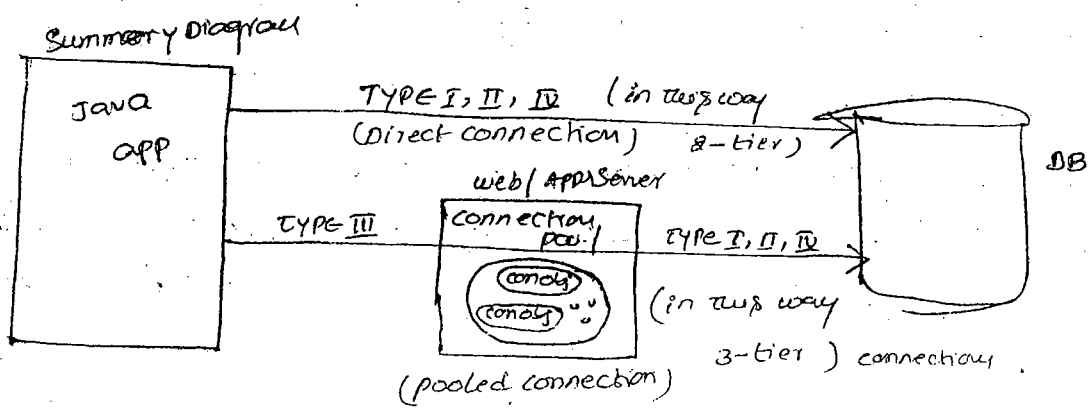
→ Respect to the Diagram

① webServer/ application Server uses Type I, II, IV Driver to interact the DB s/w and to create JDBC connection object in JDBC connection pool.

② client application uses type III protocal (Net) to interact the JDBC connection pool and to get JDBC connection object from connection pool.

③ client application uses this JDBC connection object to create other JDBC object and to manipulate table data.

④ client application realeases JDBC connection object back to JDBC connection pool. So that connection object become ready to give service to other clients.

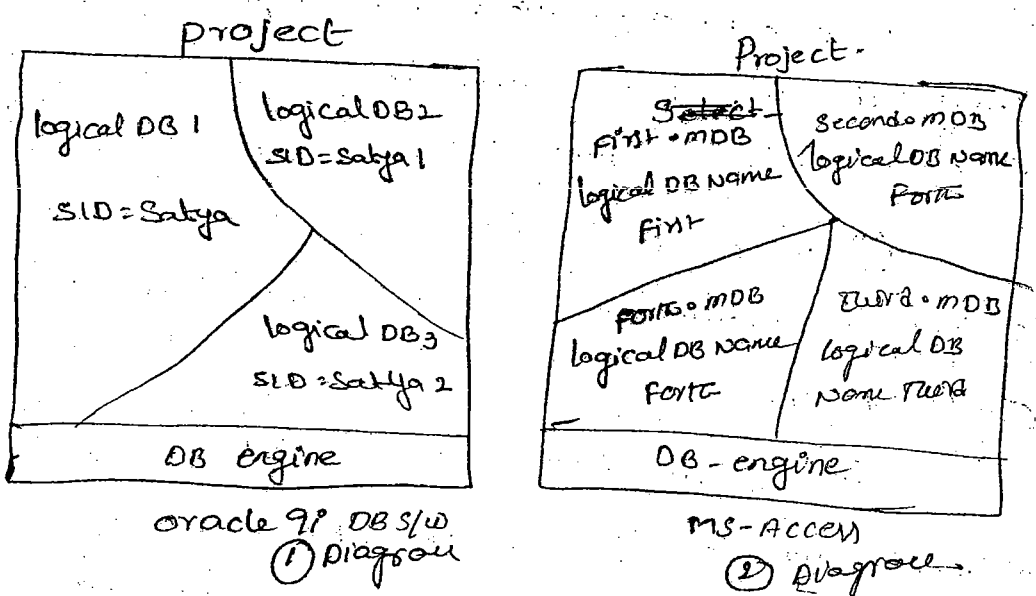


– The JDBC connection object i.e created by a programmer manually is called Direct connection.

– The JDBC connection object i.e is collected from JDBC connection pool is called pooled connection object

Y → wt is the JDBC Driver that is utilised in your project?

If the project is 2-tier / Desktop application use JDBC Driver type-IV, If the project is web application (website) on 3-tier (or n-tier application use JDBC Driver type III, IV. Here JDBC Driver type-IV will be used create JDBC connection objects in JDBC connection pools. If the IV Type-IV Driver will be used as protocol to get JDBC connection object from JDBC pool.



oracle 9i DB S/W
(1) Diagram

MS-Access
(2) Diagram

* → A logical partition created in physical DB S/W is called Logical DB. Each logical DB is identified with its name.

- If logical DB allows multiple users & application to manipulate table data concurrently & simultaneously then that DB S/W is called multiuser DB S/W, other wise that DB S/W is called Single user DB S/W.

- In Diagram (1) oracle 9i is physical DB S/W, Satya 1, 2, are the logical DB names in oracle Each logical DB is identified with it's SID.

- In Diagram (2) MS-access is the physical DB S/W. First.mDB, Second.mDB, Third.mDB, Forth.mDB files are logical DBs and Filename (.mDB) file acts as logical DB name.

→ when oracle DB s/w is installed one default logical DB. to know SID (service ID) of this logical DB.

① Select * from global_name;

SATYA.US.ORACLE.COM

② open \oracle\ora92\network\admin\tnsnam.ora ↓

SERVICE_NAME=Satya

③ controlpannel → administrative tool → Services

OracleServiceSATYA

~~How to create multiple logical participate in data~~ ms access

- oracle, mysql, are the multiuser DB s/w, MS-Access is the single user s/w.

- If multiple users of a company uses same oracle 9i DB s/w multiple logical DB it will be created in that oracle DB s/w for multiple project or one per project basis.

→ In order to work with given any JDBC Driver any DB s/w by any vendor. we need to know the following minimum requirments

① JDBC Driverclass Name

② JDBC URL (or) DB URL

③ DB userName

④ DB Password

⑤ jar file that represent JDBC Driver s/w.

1) driver name : type 1

driverclassname = sun.jdbc.odbc.JdbcOdbcDriver

db URL = jdbc:odbc:DSN

db password = scott/tiger

Vendor = Sun micro Systems.

remarks: comes with J2SDK s/w installation as built in JDBC driver.

② driver name = oci8 drive (oracle connection interface) type-II

Driver for oracle from oracle corporation

Driver classname = ~~jdbc.odbc~~ oracle.jdbc.~~odbc~~driver.OracleDriver

DB URL = jdbc:oracle:~~thin~~ oci8:@<sid>

Vender = oracle corporation

Remarks = comes with oracle DB s/w installation in the form of jar files

classes111.jar (Oracle 8i)

class12.jar (Oracle 9i)

ojdbc14.jar (Oracle 9i/10g)

jdbc:oracle:oci8 is protocal,

main protocal sid is logical DB name in oracle DB s/w

③ Driver name = oracle thin driver (type-IV jdbc driver for oracle from oracle corporation)

Driver class name = oracle.jdbc.driver.OracleDriver

db URL = jdbc:oracle:thin:@<host>:1521:<sid>

~~vendors~~ = jdbc:oracle:thin --- protocal

main protocal, sub protocal, sub sub protocal

<host> the host name or IP-address of a computer where oracle DB s/w is installed

<sid> logical DB name of oracle DB s/w

Vender = oracle corporation

remarks = comes with oracle DB s/w installation in the form of jar.files

class111.jar (Oracle 8i)

class12.jar (Oracle 9i)

~~class~~

ojdbc14.jar (Oracle 9i/10g)

~~jdbc:oracle:oci8 is protocal~~

→ Every s/w installed in a computer will reside in the logical position of computer, this logical position is called port and will be identified port no's

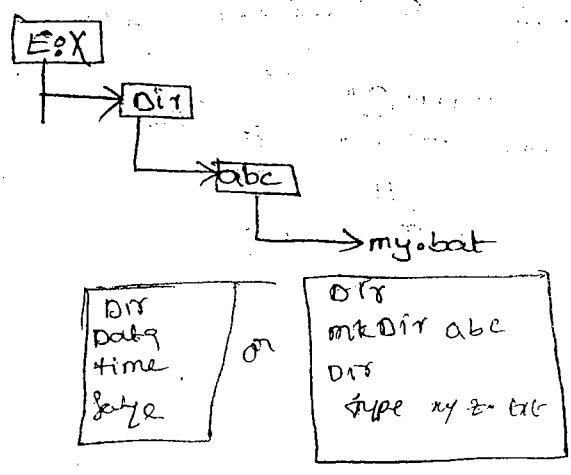
- oracle DB s/w resides on port no 1521 by default
- Internet explorer on ~~port~~ resides on portnumber 80 by default. IP-address of computer / host name of a computer + (plus) port number is called socket-number. A socket is a communication end point to interact with s/w of the computer.

→ classes111.jar, classes12.jar & classes ojdbc14.jar files are available in oracle_home/ora92/jdbc/lib folder of oracle 9i s/w installation.

Q&A

Q → what is different b/w path, classpath Environment Variables?

→ Inorder to make .cmd files, .batch files, .exe files of certain Directory visible & executable in any directory of computer add the directory location of these files to path environment variable.



E:\>Dir\abc>my.bat ✓

E:\Dir>my.bat (X)
E:\xyz>my.bat (X)
C:\abc>my.bat (X)

problem

problem statements

Solutions :- add E:\Dir\abc folder in PATH Environment Variable

Mycomputer → Properties → Advanced tab → Environment Variables →

```
E:\Dir\abc> my.bat (v)
E:\Dir > my.bat (v)
E:\xyz> my.bat (v)
e:\xyz> my.bat (v)
E:\Dir\xyz> my.bat (v)
```

Variable name: PATH
Variable value: E:\Dir\abc
OK close.

=

javac, javap, java — etc tools in J2SDK are exe files available in J2SDK_home_Dir\bin Directory

to make this exe files and tools visible & executable in every directory on computer we need to keep directory location of its exefiles in PATH Environment variable.

variable name PATH

Variable name E:\Dir\J2SDK1.4.2_04\bin;.

; is the saparator b/w values added in mycomputer Environment variables. Don't use semicolon(;) as beginner(or) terminator.

- New values added to mycomputer environment variable will not be visible in old command prompts. they are visible only with new command prompts. which are open after adding values to environment variables.

classpath :-

E:\abc
Test.java
```
package P1;
public class Test{
public void disp(){
S.O.P("Test:disp()");
}
}
```

E:\abc> javac -d . Test.java
E:\abc → P1 → Test.class

main Application
O:\demo\App1.java
```
import P1.Test;
public class App1{
P.S.V.M(String k[]){
Test t=new Test();
t.disp();
}
}
```

O:\demo> javac App1.java (X)
can't resolve Symbol / P1, Test, disp()

Solution :- add the Directory when [P] package is avalible (E:\abc>) in classpath environment variable.

Mycompoter → properties → advanced tab → Environment variables
→ Enter variables & values

Variable name : classPATH
Variable value : E:\abc;.;

[OK] [cancel]

D:\demo> javac appl.java ⊗ D:\java appl (V)

If java application uses new classes & interfaces that are not part of JSDK s/w and current directory then those new classes & interfaces related directory (or) jar file must be added in classpath environment variable.

- when java application uses third party API's (other than JSDK API & current directory classes, interfaces) then the third API Related directory / are jar files must be added in the classpath environment variable.

***

→ path Environment variable can be use in for java applicatio & outside the java application. classpath environment variable must be used for java application.

Note :- my computers, user environment variable are specific to windows login user compoter

- mycomputer system environment variable are visible for all the windows user of a compoter

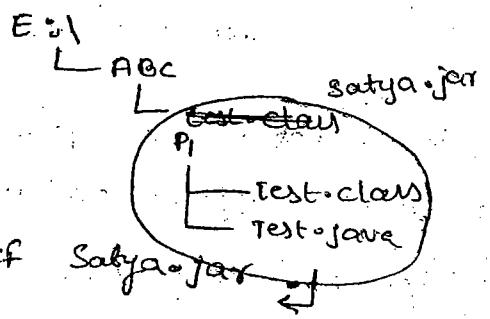
- it is always recomanded to add new values in the begining of exting values for my compoter environment variable values.

- path, classpath java_home & etc are the my compoter environment variables.

cf — input to file.
|
create jarfile

29/10

→ Preparing the jar file representing the user defined packages.



E:\ABC> jar cf Satya.jar .

c = create jar file
f = use input file

jar is a built in tool of J2SDK s/w in the above example jar tool prepares satya.jar file by combining every thing of the current directory

To add jar file in the class path:-

Variable name: class path.

Variable value: E:\ABC\Satya.jar; other value;.

Note:- It is always recomanded to add new values in Environment variables at the begening of the existing values.

it is not recomanded to set Environment variable values from command prompt because they will go of once command prompt is closed. Always recomended to use mycomputer - property based environment variables.

- jar file mean java archieve file which combine multiple files in to single file like windows zip file.

- Generally third party s/w vendor supply their API's in the form of jar files. Oracle corporation is giving OCI driver s/w, thin Driver s/w in the form of jar file called classes12.jar(or) ojdbc14.jar(or) classes111.jar &

inorder to work with thin JDBC driver we need to keep one of the aboves jar files in the classpath.

Procedure develop java JDBC Application by using oci driv

(Oracle corporation supplyed JDBC Driver for oracle)

① know JDBC Driver class name & DB URL of oci JDBC Driver.

② Driver class name is : "oracle.jdbc.driver.OracleDriver"

DataBase URL is : "jdbc:oracle:oci8:@ satya", "scott", "tiger"

③ java JDBC application as shown belloo. by using above details

```
// SelectTest.java
import java.sql.*
public class SelectTest
{
  p s v m (String xc[]) throws Exception
  {
     class.forName("oracle.jdbc.driver.OracleDriver");
  Connection con=DriverManager.getConnection("jdbc:oracle:oci8:@
                                              satya","scott"
                                              ,"tiger");
     Statement st= con.createStatement();
     ResultSet rs = st.executeQuery(Select * from Student);
  while (rs.next())
  {
    S O P (rs.getString(1)+ " "+ rs.getString(2)+ " " rs.getString(3));
  }
  //
     rs.close();
     st.close();
     con.Close();
  }
}
```

-add oracle home/ora92/lib/classes12.jar file in class path environment variable.

Procedure:-

mycomputer → Properties → advanced tab → Environment variable → variable name: class path

variable value: D:\oracle\ora92\jdbc\lib\classes12.jar; other values; → ok → ok → ok

-compile & execute java application.

Procedure to develop java jdbc application by using oracle thin Driver (oracle corporation supplied jdbc driver type-II for oracle).

steps ① Follow all the steps of oci Driver as it is but change DB URL as shown bellow.

```
connection con=DriverManager.getConnection("jdbc:oracle:thin:
                @localhost:1521:satya","scott","tiger");
```

- To Represent a computer beeing from same computer use the word called local host.

connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:satya","scott","tiger");

jdbc:oracle:thin → Protocal

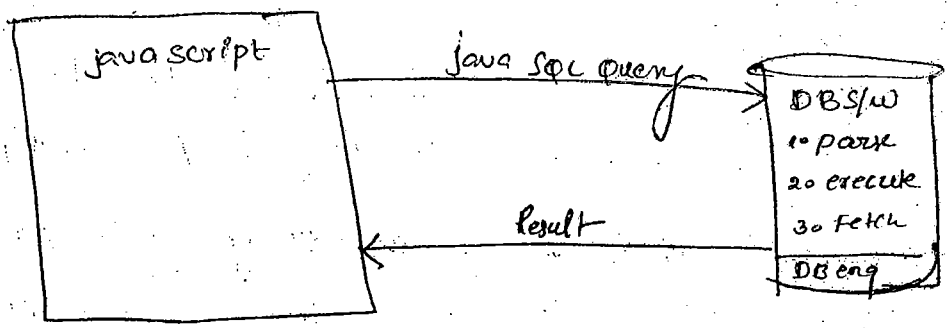
@localhost → hostname/Ip address of computer where oracle DB s/w installed

1521 → ~~Protocal~~ port no of oracle DB s/w

satya → SID

scott → DB user name

tiger → DB password



In JDBC they are three types of statement objects

1. Simple statement obj (java.sql.Statement)
2. PreparedStatement (java.sql.PreparedStatement) ↑ extends
3. CallableStatement obj (java.sql.CallableStatement) ↑ extends

→ SimpleStatement object means it is the object of the class that implement java.sql.StatementInterface.

→ PreparedStatement object means it is the object of the class that implements java.sql.PreparedStatementInterface

→ CallableStatement object means it is the object of the class that implements java.sql.CallableStatementInterface.

→ when java application sends sql query to DB s/w DB s/w performs following three operations on the query

1. parse
2. execute
3. Fetch

- In parse operation the query will be splitted in to token & Syntax query will be verified

- In execute operation the sql query will be executed and result will be generated storing in the buffer.

- In Fetch operation Result will be gather from buffer and will be send to client application.

Limitations of Simple Statement Object :-

① Simple Statement resides 'raw' query to 'database s/w So when you send same query to DB s/w for multiple no: of time with same or diff values. the query will be passed, executed output will be fetch for multiple no: of times in these operations parsing same sql query for multiple no: of times is unnecessary.

② preparing SQL query for simple statement object by using variables is ~~unnecessary~~ unnecessary complex. to overcome these problems use sql query as precompiled query the query that goes to DB s/w without values, becomes passed query and allows client application you to multiple values to execute multiple times is called precompiled query

③ the JDBC object prepared Statement is capable of representing precompiled query available on the DB s/w. and allows to set the values for query and also allows to execute the query

— the prepared statement object which can work with precompiled query reduces when network traffic b/w java application & DB s/w.

* — when you want to execute same query for multiple no: of times with same or different values it is recommended to make that sql query as precompiled query & work with preparedStatement Object.

- In Railway ticket Reservation System, Bus ticket Reservation System kind of applications use precompiled queries and preparedStatementObjects

procedure to work with PreparedStatement Obj:-

① Prepare SQL query placeholders (?)

String query = "insert into student values(?,?,?)";

Note:- Each '?' symbol is called placeholder parameter (or) positional parameter. This parameter allows to set the value for the query after query goes to DB s/w and becomes parsed query in the DB s/w.

② Create PreparedStatement Object by making SQL query as precompiled query

PreparedStatement ps = con.prepareStatement("query");

↓

this method makes given sql query as precompiled query

ps = this object represent precompiled query available in DB object.

③ Set values for place holder parameters (?) of precompiled query

ps.setInt(1, 452);
ps.setString(2, "kavi")
ps.setString(3, "hyd");

Note:- PreparedStatement Object allows the programmer to set value for sql query in java style.

④ executeQuery

int result = ps.executeUpdate();

in DB s/w

Note:- the above statement execute SQL query that is represented by preparedStatement Object.

⑤ To execute query for multiple times; repeat the steps in ③ & ④

⑥ close PreparedStatement Object.

ps.close();

[ro] ⇒ wrt a program to insert multiple student Details collected from the user into a DB table.

```
// psttest.java
import java.io.*;
import java.sql.*;
pt
public class psttest
{
   P S V M (String k[]) throws Exception
   {
      BufferedReader br=new BufferedReader(new InputStreamReader
                                          (System.in));
      S.O.P ("Enter total no.of students");
      int n=Integer.parseInt(br.readLine());
     // write JDBC code

      class.forName("oracle.jdbc.driver.OracleDriver");
 connection con=DriverManager.getConnection("jdbc:oracle:thin:
                                  @localhost:1521:satya","scott","tiger");
  PreparedStatement ps=con.prepareStatement("Insert in to
                                  student values(?,?,?)");
      //read student Details
for(int i=1;i<=n;i++)
{
     S.O.P ("Enter" +i+" student Details");
     S.O.P ("Enter student no");
     int sno=Integer.parseInt(br.readLine());
     S O P ("Enter student name");
     String sname=br.readLine();
     S O P ("Enter student address");
     String saddr=br.readLine();
    // set values to placeholder parameters (?) query
```

```
}
}
```

Note

DB

tim

Pre

will

ID

the

jar

or

```
PS6 cnt
    PS.setInt(1, sno);
    PS.setString(2, sname);
    PS.setString(3, saddr);
  // execute query.
int  res=ps.executeUpdate();

  if(res!=0)
      SOP(i+"student record is inserted");
      else
       SOP(i+" student record is not inserted");
  } //for

  // close jdbc objects

  br.close();
  PS.close();
  con.close();
 }
}
```

Note :- In order to execute SQL query only for one time in DB s/w use simple-Statement object.

In order to execute sql query for multiple no: of times with same & different values by making query as precompiled query use preparedStatement object

- values kept in mycomputer Environment variables will not be visible in the projects that are created in IDE s/w & its applications placed in IDE s/w's uses third party API (other than JDSDK API's) the API Related jar files should be added saparately to the project library or class path

-In Netbeans IDE the procedure to add jar file

expand project → rightclic library → Add jar → Browse & select the jar.

If java application executing from netbeans IDE is using oracle thin Driver then the thin driver related classes 12.jar file are equalent jar file must be added saparately to the librarys of the projects.

MYSQL

31/10

MySQL
- type: DB s/w
- version: 4.x
- vendors: mysql (Sonmicro system)
- open source s/w
- default port no: 3306
- download s/w: www.dev.mysql.com

→ mysql is the most popular open source/free DB s/w

procedure to instal & activate mysql DB s/w :-

- Instal mysql DB s/w using its setup file.

Go to mysql home/bin folder and lanch winmysqladmin tool which starts mysql s/w and vendors traffic signal in the system tray on the task bar

→ MySql front is a GUI tool perform operations on DB s/w (mysql) this tool is like sql+ tool of oracle DB s/w.

Procedure to create Logical DB in mysql & Add tables in the Logical DB s/w :-

Launch mysql front
↓
user : admin
password : admin
↓
connect
↓
admin@localhost
↓
Create DB
(DB Name)
↑
Satydb → select satyadb → data base tag → Right click → Create

netable:
tablename : product → add following fields/columns.

P no int(5)
P name varchar(20)
P qty int(4)
↓
Create
↓
Right click on product table.
↓
insert record & start adding the records

NOK : In mysql DB s/w each logical DB's is identified with it's Logical DB name like Satyadb shown above.

→ wrt a java application to interact with mysql DB s/w by using Jdbc Driver type-1

Step-1 : Install odbc driver for mysql and create DSN for it

NOK : A windows s/w will not give odbc Driver for mysql by default.

Procedure to Create DSN

controll pannel
↓
administrative tools
↓
Data sources
↓
User DSN
↓
Add
↓
mysql odbc Driver
↓
finish
↓
DataSourceName mysql DSN
— —
— —
DB name: satyadb (logical DB name)
user : admin
pwd : admin
↓
Test data source
↓
OK

→ Develop java application as shon bello by using Jdbc driver type-I.

```
import java.io.*;
import java.sql.*;
{
   public class Test SelectTest
   {
      p v s m(String K[]) throws Exception
      {
         class.forName("son.jdbc.odbc.JdbcOdbcDriver");
Connection con=DriverManager.getConnection("jdbc:odbc:DSN", "admin",
                                                  mysql
                                                  "admin");
      Statement st=con.createStatement();
      ResultSet rs=st.executeQuery("select * from product");
```

```
        while (rs.next())
        { sop(rs.getString(1)+" "+rs.getString(2)+" "+rs.getString(3));
        }
        rs.close();
        st.close();
        con.close();
    }
}
```

> javac SelectTest.java
> java SelectTest

→ The mysql organization supplyed type-IV mechanism based jdbc Driver for mysql is called Jconnector Driver this Driver will not come with mysql DB s/w installation has to be installed separately by extracting zipfile called mysql-

mysql/
↓
connector/connection

mysql-connector-java-3.0.8-stable.
version.

Jconnector Driver

name : Jconnector
vendor : mysql org
for which DB : mysql DB
open source Jdbc Driver s/w
Driver mechanism : TYPE-IV
Driver class name : org.gjt.mm.mysql.Driver
URL : jdbc:mysql://<logical DB name>
(jdbc: — protocal)
jar file that represents jdbc Driver :

mysql-connector-java-3.0.8-stable-bin.jar

The above jar files comes when mysql-connector-java-version-stable-zip file is extracted

Step 1 :- Develop java application as shown bellow by knowing JConnector Jdbc DriverManager.

```
import java.io.*;
import java.sql.*;

public class SelectTest
{
    P S V M (String k[]) throws Exception
    {
      class.forName("org.gjt.mm.mysql.Driver");
Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306://satyadb","admin","admin");
    Statement st=con.createStatement();
    ResultSet rs=st.executeQuery("select * from product");

    while(rs.next())
      {
        sop(rs.getString(1)+" "+rs.getString(2)+" "+rs.getString(3));
      }
    rs.close();
    st.close();
    con.close();
  }
}
```

→ add mysql-connect-java-version-stable.zip file extraction home/mysql-connector-java-version-stable/mysql-connector-java-version-stable-bin.jar file in the classpath.

→ Compile & Execute the java App :-

Note: when jdbc Drivers of existing java application is change the following three details will be changed

(1) jdbc Driver classname

(2) DB URL

(3) jar file ~~when~~ that needs to be added in the classpath

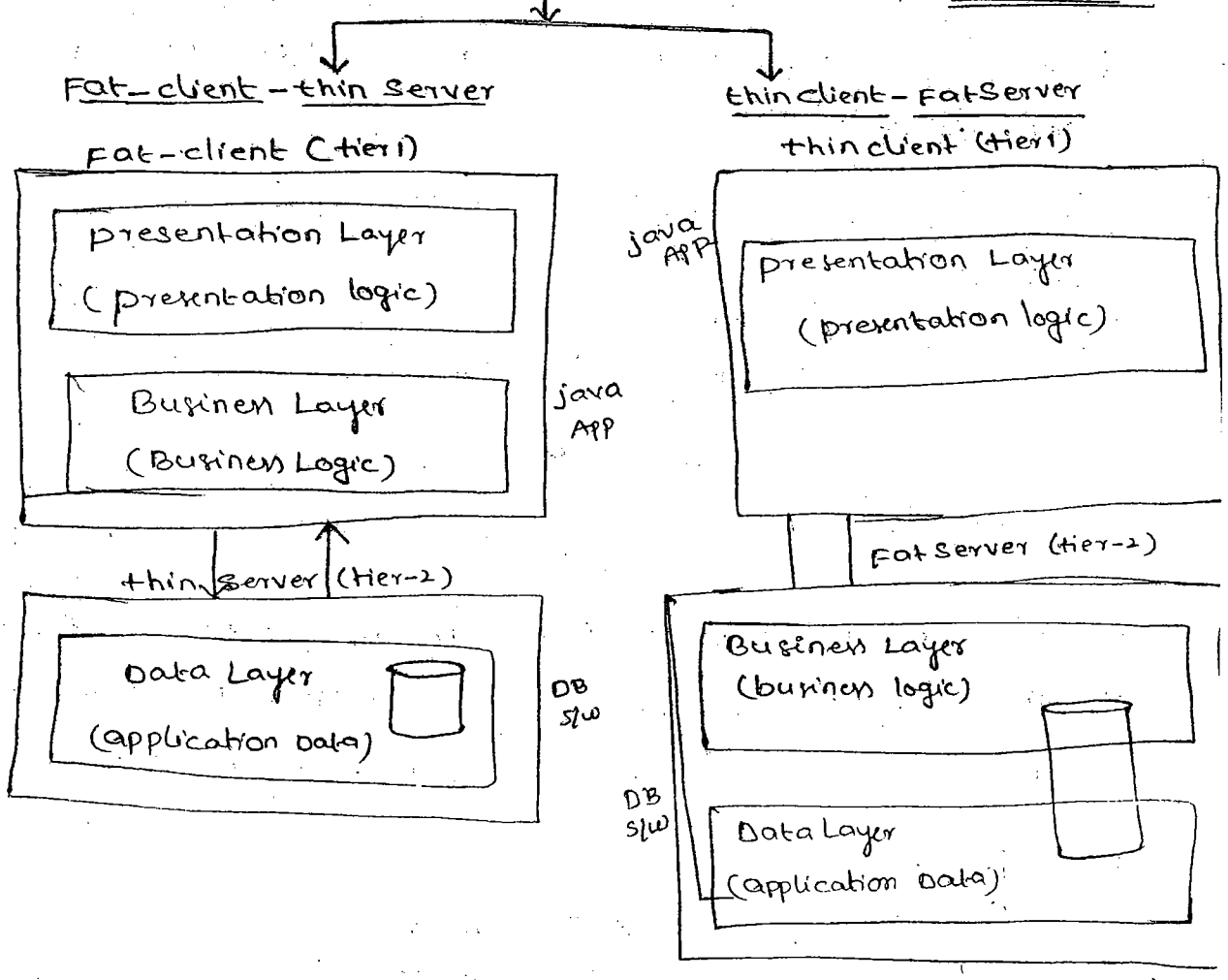
*

Note :- Oracle thin driver, oci driver are given by DB vendor oracle corporation where as jconnector driver for mysql is given by third party vender

→ mysql front TOAD PL/SQL developer etc are the tools given to interact the DB s/w in GUI mode

2/11/09

two tier application



→ The logic i.e capable of designing graphical user interface Reading values from keyboard and presenting result on the monitor is called Presentation logic.

→ The main logic of the application is called business logic the logic uses input values and generates the result.

→ The layer that holds/ stores application data is called data layer.

→ The jdbc applications that we have developed so far are called Fat client and thin server application.

In order to keep business logic on DB s/w we need to keep business logic in the form of stored, PL/SQL procedures and stored PL/SQL Functions.

→ In order to call PL/SQL procedure and function of DB s/w being from java application we take the support of callable statement object jdbc application.

→ PL/SQL procedure does not returns a value where as PL/SQL function returns value.

→ PL/SQL function procedure contains parameter in 3 modes.

① In - input mode (Default)

② out - output mode

③ Inout - Input & output mode.

→ we can collect result from PL/SQL function either as return value are from out parameters. The results of PL/SQL procedure must be gathered by using out parameters.

```
y = x * x;        x → in mode
                  y → out mode

x = x * x;        x → inout mode.
```

→ Procedure to develop PL/SQL procedure in oracle DB s/w.

① Launch SQL prompt of oracle

SQL>ed

create or replace procedure my-proc-1 (no in number,
name out varchar2) as
begin
Select name into name from emp where emp.no=:n
end;

↳ save and execute the procedure.
(SQL>/) procedure created.

To see errors SQL> show errors;

→ code snippet to use callable statement object:-

① prepare query having place holders, which call PL/SQL procedure.

String qry="{call my-proc1(?,?)}";

Note:- the above syntax is standard jdbc syntax to call any PL/SQL Function or procedure.

② Create callable statement object pointing to PL/SQL procedure.

CallableStatement cs=con.prepareCall(qry);
↳ represents my-proc1 procedure.

③ Set values for In parameters by using setxxx(-)

cs.setInt(1,7499);

④ Register out parameters of PL/SQL procedure with jdbc types

cs.registerOutparameter(2, TYPES.VARCHAR)
↳ Jdbc Datatype.

→ The Jdbc Datatype is the bridge data types b/w java data types and underLaying database s/w data types.

→ Jdbc Datatypes helps jdbc drivers to convert java notation data to DB s/w notation data and vice versa.

java Datatype	jdbc datatype	DB s/w Datatype (oracle)
int	Types.INTEGER	NUMBER (Number)
java.lang.String	TYPES.VARCHAR	varchar, Varchar2
float	TYPES.FLOAT	Number
java.sql.Data	TYPES.DATE	data

→ public static final member variables of a java classes/Interfaces are called constant.

→ ALL jdbc Data types are constants of java.sql.Types classes.

⑤ make database s/w to execute the procedure

```
cs.execute();
```

⑥ call getxxx(); on callable statement object to gather result stored in out parameters after executing PL/SQL functions and procedures.

```
String res = cs.getString(2);
                          └ out parameter number.
```

⑦ Repeat step-3, 5 and 6 for multiple times execution of PL/SQL procedure

⑧ close callableStatement object

```
cs.close();
```

```
// cstest.java
import java.sql.*;
// import java.io.*;
public class cstest
{
    P v s m (String k[]) throws Exception
    {
        class.forName("son.jdbc.odbc.JdbcOdbcDriver");
connection con=Drivermanager.getConnection("jdbc:Oracle:
            thin:@localhost:1521:DSN","scott","tiger");
        String qry="{call my_proc1(?,?)}";
    CallableStatement cs=con.prepareCall(qry);
        cs.SetInt(1,7499);
        cs.registerOutparameter(2,Types.VARCHAR);
        cs.execute();
        String res=cs.getString(2);
        S.O.P("the result is"+res);
        cs.close();
        con.close();
    }
}
```

→ In order to centralize the business logic and persistance logic of an application PL/SQL procedures and fonctions are usefull. If project contains multiple modules. If all modules are looking to use some SQL queries based persistanc logic it is recomanded to centralize SQL queries on DB s/w in the form of PL/SQL procedures (or) Fonctions.

→ Conclusion on statement object:-

In order to execute sql query only for one time use simple statement object in order to execute some sql query for multiple no: of times with same. (or) different values use prepared (or) Statement object.

In order to centralize persistance logic use CallableStatement object which can call PL/SQL procedure or function of DB s/w.

3/11

→ Procedure to call PL/SQL functions of oracle from java jdbc application.

① prepare PL/SQL Function in oracle DB s/w.

```
SQL> ed
  └→ Create or replace function my-fx1 (no. number,
name out varchar) return number as bsal number;
begin
    Select ename into name from emp where
                                emp no=no;
    Select sal into bsal from emp where empno=no;
    return bsal;
end;
```

→ Return value related parameter of a PL/sql function must be treated as out parameter because this also given to gather result given by PL/SQL function.

② Develop java jdbc application as shown bellow by using callable statement object.

```
import class java.sql.*;

public class csset
{
   Pvsm (String K[]) throws Exception
   {
      Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@
        localhost:1521:DSN","scott","tiger");
   // Prepare query to call PL/SQL statements.
   String qry="{ ? = call my-fn(?,?) }";
   CallableStatement cs=con.prepareCall(qry);
   // Set value for in parameter.
      cs.setInt(2,7499);
   // register outparameter with jdbc Datatype
      cs.registerOutparameter(3,Types.VARCHAR);
      cs.registerOutparameter(1,Types.INTEGER);
   // execute PL/SQL function
      cs.execute();
   // Get Result from out parameters
      int res1=cs.getInt(1);
      String res2=cs.getString(3);
   // Display result.
      S.O.P("Employee sal is"+res1);
      S.O.P("Employee name is"+res2);
```

→ Always count place holder parameter or positional parameter of a query in the order they are opening in the query not in the order their parameters are placed in PL/SQL procedure/function.

Note:- For related example applications on prepare statement, CallableStatement object refer application ⑤ to application ⑧ of booklet.

META-DATA :-

"Data about data is called metadata" using meta data concepts we gather details about details. Using jdbc meta data operations we can gather Limitations & capabilities are underlaying data base s/w.

→ jdbc allows '3' modes of metadata operations.

① Data base meta Data (single word) → gives Limitations and capabilities of underlying DB s/w.

```
DatabaseMetaData dbmd = con.getMetaData();
```

DatabaseMetaData means it is the object of a class that implements java.sql.DatabaseMetaData interface.

② ResultSetMetaData → gives details about table like column names, data types and --- etc which is represented by ResultSet object

```
ResultSet rs = st.executeQuery("select * from emp");
ResultSetMetaData rsmd = rs.getMetaData();
```

→ ResultSetMetaData object means it is the object of a class that implements java.sql.ResultSetMetaData interface

③ parameterMetaData → gives details about parameters (?) avalible in query (preparedStatement, callableStatement)

```
PreparedStatement ps = con.prepareStatement("select * from student where sno=? and sno=?");

ParameterMetaData pmd = ps.getParameterMetaData();
```

→ parameter metadata object means it is the object of a class java.sql.ParameterMetaData interface

→ jdbc metadata operations are given to gather details about db s/w, tables, PL/SQL functions, procedure parameters being from java applications.

→ For Example application on DatabaseMetaData refer Application-16 in Booked.

→ while working with DatabaseMetaData object any get method called on that object returns null (or) zero means the jdbc Driver has not implemented those methods properly

→ Different jdbc Drivers will give different details based on their implementation.

→ To gather multiple details regarding underlying DB s/w we need to use DatabaseMetaData object and call various getter methods i.e xxx() methods on that object.

→ MetaData programming is always helper programmer for programmer in the process of performing main persistance operations.

→ If there is a requirment of working with new DB s/w from Java applications we can gather details about that database s/w by using DatabaseMetaData object.

ResultSetMetaData

5/11

Using ResultSetMetaData object you can print complete details about DB table, i.e represented by ResultSet object.

```
Statement st = con.createStatement();
ResultSet rs = st.executeQuery("select * from student");
ResultSetMetaData rsmd = rs.getMetaData();
int colcnt = rsmd.ColumnCount();
for(int i=0; i<=count; i++) // Print col names
{
   S.O.P(rsmd.getColumnLabel(i) + "\t");
}
 SOP();
for(int i=1; i<colcnt; ++i) // Print col Data types
{
   S.O.P(rsmd.getColumnTypeName(i) + "\t");
}
S.O.P("\n\n");
while(rs.next())
{
   S.O.P(rs.getString(1) + " " + rs.getString(2) + " " + rs.getString(3));
}
S.O.P("table name is " + rsmd.getTableName(1));
S.O.P("schema name is " + rsmd.getSchemaName(1));
S.O.P("col is writable " + rsmd.isWritable(1));
}
```

→ We can print column values of a DB values without knowing column count when ResultSetMetaData object is used as shown bellow.

→ use case 1:

```
Statement st=con.createStatement();
ResultSet rs=st.executeQuery("select * from dept");
ResultSetMetaData rsmd=rs.getMetaData();
int colcnt=rsmd.getColumnCount();
while(rs.next())
{
  for(int i=1; i<=colcnt; i++)
  {
    S.O.P(rs.getString(i)+"\t");
  }
  S.O.P();
}
```

→ parameter metaData object helps the programmer to know about place holder parameters(?) available in the query represented by prepared statement object, callable statement object.

→ for example application on parameterMetaData refer application-16, page no: 87

→ most of jdbc available in market are not providing support for parameterMetaData programming i.e most of drivers are not implementing jdbc specification completly.

Oracle, Sybase, MS-Access, My-SQL and -- etc are called conventional DB s/w's. where as MS-Excel, Note paid files and -- e.t.c are called non conventional DB s/w's.

→ performing print operations directly on conventional main DB s/w's of projects kills the performence of main DB s/w and perform print operations on them.

→ The ICICI bank project uses oracle as main DB s/w and transfers quarterly data to non-conventional DB s/w MS-Excel to print ~~quarterly~~ Quarterly account statement of customers.

→ To interact with all non conventional DB s/w we need to use Jdbc Driver type-II.

→ In MS-Excel s/w we can create multiple workbook file (.xls files) Each workbook acts as one logical DB. Sheets in work book are called tables. First row names / value in sheet are called column names.

→ MS-Excel s/w :- (Physical DB s/w)

→ college.xls (Logical DB)

- Sheet1 (table)
- Sheet2 (table)
- Sheet3 (table)

Sheet1

SNO	SName	SADD
110	Raja	Hyd
119	Ramesh	Hyd
116	Ramana	Hyd

Col names

TABLE ROWS

* Procedure to make java application interacting with MS-Excel (as DB s/w):

① Create MS-Excel work book college.xls and add data in sheet1 and rename sheet-1 to student.

② create DSN for "Microsoft Excel Driver"

Start
↳ Settings
↳ control pannel
↳ Administrative tools
↳ Data Sources
↳ Add
↳ Microsoft Excel Driver
↳ Finesh

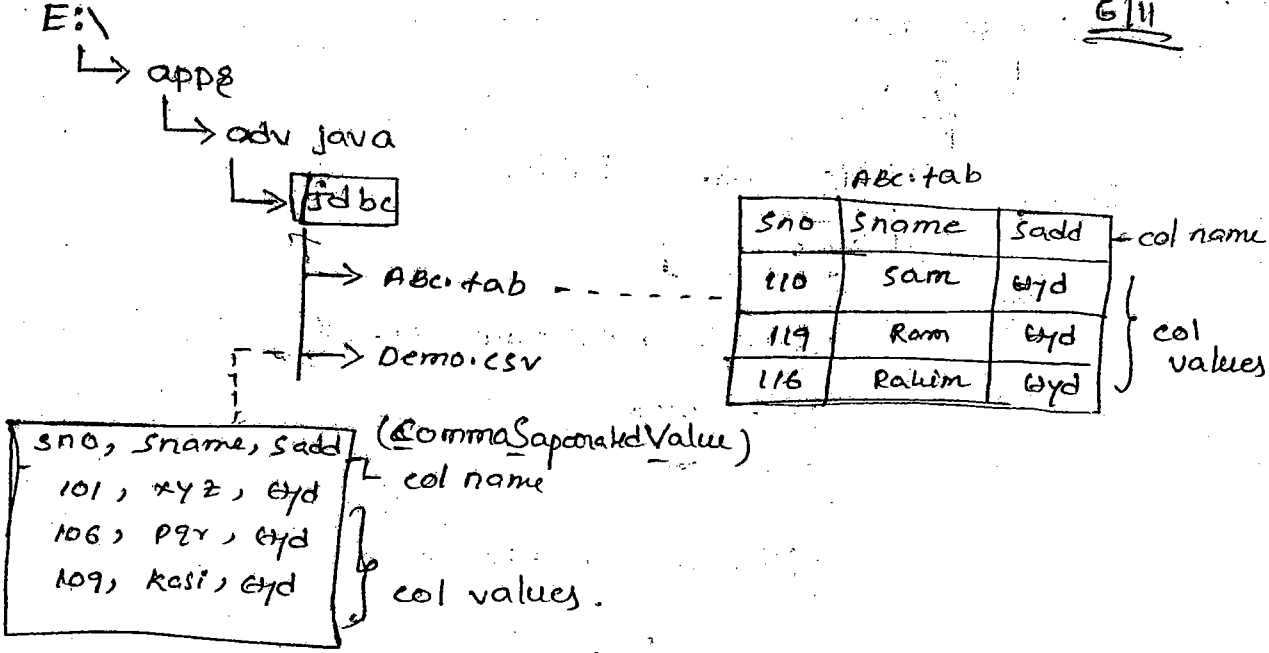
DataSourceName: xlsdsn

Select workbook:
↳ Browse & select
↳ college.xls
↳ OK → OK

TABLE ROWS

→ write jdbc program as shown bellow by using jdbc Driver type-I.

```
// ExcelTest.java
public class ExcelTest
{
  P S V m (String k[]) throws Exception
  {
    class.forName("son.jdbc.odbc.JdbcOdbcDriver");
Connection con=DriverManager.getConnection ("jdbc:odbc:
                                              xlsdsn");
    Statement st=con.createStatement();
    ResultSet rs=st.executeQuery("select * from [Student]");
    while(rs.next())
    {
      S.O.P(rs.getString(1)+" "+rs.getString(2)+" "+rs.
                                              getString(3));
    }
    rs.close();
    st.close();
    con.close();
  }
}
```



Note :- By taking files data as DB Data we need to treate files as DB tables, the Directory where files are avalible as logical DB and first row values as column names.

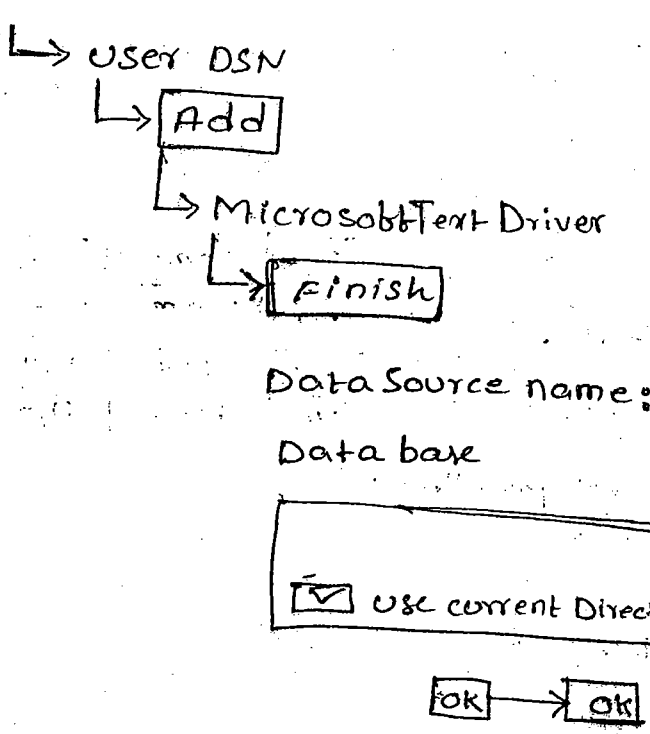
- procedure to develop java jdbc application that reads Data from text files.

① prepare text files in a Directory and place formated data (;) comma Saparated and , tabSpace saparated or etc.

Rebbers above files :-

→ Create DSN for microsoft odbc Driver text files.

start
→ Settings
→ control pannel
→ Administrative tools
→ Data Sources
→ User DSN



→ write jdbc program as shown bellow by using jdbc Driver type-I.

```
// TxtTest.java
import java.sql.*;
public class TxtTest
{
  p s v m (String a[]) throws Exception
  {
    class.forName ("sun.jdbc.odbc.JdbcOdbcDriver");
Connection con=DriverManager.getConnection ("jdbc:odbc:
                                              txtdsn");
  Statement st=con.createStatement();
  ResultSet rs=st.executeQuery("Select * from abc.tab");
  while (rs.next())
  {
    s o p ("rs.getString(1)+" "+rs.getString(2)+" "
                                +rs.getString(3));
  }
  rs.close();
  st.close();
con.close(); } }
```

→ If odbc Driver of text file is not able to recognise first row value as column values we need to do that work explicitly by using various options of dsn Creation process for this.

control pannel
↳ Data Sources
↳ txtdsn
↳ configure
↳ options >>
↳ *.tab
↳ Define format
↳ abc.tab
↳ ☑ column name header
↳ format [tab delimited ▼]
↳ Guess
↳ sno
Data type : Integer
sname.
Datatype : Langchar
sadd
Datatype : Langchar
↳ modify
↳ ok

→ we can move data b/w two conventional DB s/w (or) conventional, non-conventional DB s/w (or) two non-conventional DB s/w.

→ Write a java application to migrate the data from Excell sheet to table in oracle DB s/w?

- Refer Application (13) in booklet page: 24, 25

* use commercial type-II, type-IV mechanism based jdbc drivers of non-conventional DB s/w to perform non-select operations on non-conventional Database softwares

MS-Access

type : Single user DB s/w

version : 11.x

Vendor : Microsoft

commercial s/w

logical DB : .mdb file

Use jdbc Driver type-I along with "microsoft odbc Driver" for MS-Access.

Start
↳ Programs
↳ MS-office
↳ ms-Access 2003
↳ create new file

New

☑ Blank data base

↳

File name: "satyadb.mdb"

↓ (Save in any folder)

create table in Design view

File name	Datatype
eno	Numaric
ename	text
eadd	text

↓

ctrl + w

table name: employee OK

↓

Select emp

↓

Enter records

↓

ctrl+w

* → Procedure to develop java application interacting with MS-Access db s/w by jdbc Driver type-I

① create DSN for Microsoft ms-Access Driver (odbc Driver)

Control pannel

↳ administrative tools

↳ Data Sources

↳ User DSN

↳ [Add]

Microsoft Access Driver

↓

[finish]

DataSourceName : accdsn

↓

Browse & select above .mdb file

↓

[create] (It not possible to go Create)

satyadb.mdb

↳ [OK]

↳ [OK]

→ Develop java jdbc application interacting with table of MS-Access DB s/w.

```
import java.sql.*;

class AccessTest
{
  P V S m (String k[]) throws Exception
  {
    class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection con=DriverManager.getConnection("jdbc:odbc:DSN");
  Statement st=con.createStatement();
  ResultSet rs=st.executeQuery("select * from emp");
  while(rs.next())
  {
    S O P(rs.getInt(1)+" "+rs.getString(2)+" "+rs.getString(3));
  }
  rs.close();
  st.close();
con.close();
}
}
```

- To establish connectivity with single user DB s/w use DriverManager.getConnection method that takes one argument. Similarly we to establish connectivity with multiuser DB s/w use DriverManager.getConnection method that takes three parameters.

- There are three overloaded forms of DriverManager.getConnection method ① public static Connection getConnection(String url, String user, String pwd)

② P S connection getConnection(String url, String users, String pwd)

③ P S C getConnection(String url, properties prop);

Batch processing :-

→ So for our java applications are sending a query to DB s/w execute query in DB s/w and gather in a result from DB s/w that means we are sending & executing query as individual query.

→ when java application wants to send multiple related query DB s/w for execution it is recommended instead of sending these queries separated to DB s/w for multiple times, it is recomanded to combine multiple queries into single unit (batch) and recomanded to execute individually in DB s/w, gather the result as batch from java application this process is called batch processing & batch updation.

→ Batch updation reduces network round trip s/w java application & DB s/w

Sample code :-

```
con.setAutoCommit(false);
Statement st = con.createStatement(); // Add services in batch
st.addBatch("insert into student values(101,'raja','hyd')");
st.addBatch("update student set sname='jai' where sno=120);
st.addBatch("delete from student where sno=120");
// Any no: of nonSelectQuery
// execute queries of batch
int res[] = st.executeBatch();
```

res	
1	0
2	1
0	2

```
S.o.p("total no: of records are updated"); // S.o.p(" "+total);
int total=0;
for(int i=0; i<res.length; i++)
    total = total + res[i];
S.o.p(total);
```

we must add only the select query this Batch

The process of combining related operations in to single query and executing them by applying to do every they are notify principle

the process of combining with check amount, from source account, deposit amount into destination account operations into single unit and executing them by applying do everything or nothing principle for transfer money operation is called transaction

— Batch processing of jdbc helps the programmer to apply transaction support on set of related jdbc queries.

Logic of Transaction

```
Boolean flag = true
for(int i=0; i<res.length; i++)
{
    if(res[i] == 0) {
        flag = false;
    } break;
}
if (flag == true) {
    con.commit();
}
else
{
    con.rollback();
}
```

* only batch processing doesenot perform transaction management we need to write some additional logic to get the effect of transaction management on the results of batch processing

Sample code of Batch processing enabled with transaction management.

— For Example Application of Batch processing + transaction management refer application no. of pageno - 29

Q → can I use execute method on statement object to execute Select, non select Queries?

yes possible.

— Execute method returns true if query execution result is resultset object (Select queries). execute method returns ~~integer value~~ false if the given Select SQL query execution returns an ~~in~~ integer value (non Select queries).

After executing query by using execute method if the query returns true ResultSet object then use st.getResultSet method gather that ResultSet similarly use st.getUpdateCount method to gather the integer result given by query

Ex:-
```
Statement st = con.createStatement();
boolean flag = st.execute("update student set sname = 'uuu' where sno = 567");
if(flag)
if(flag==true)
{
    ResultSet rs = st.getResultSet();
    while(rs.next())
    {
        SOP(rs.getInt(1)+" "+rs.getString(2)+" "+rs.getString(3));
    }
    else
    {
        int rowcount = st.getUpdateCount();
        SOP("no. of rows that are updated "+rowcount);
    }
}
```

8/11 jth Birthday

→ jdbc connection pool is a factory that contains set of readily available jdbc connection object. before actually being used. the advantages of working with jdbc connection pool are.

① By using less amount of jdbc connection object we can use service to more clients.

② creating, managing distroy of jdbc connection object. will be taken care by jdbc connection pool. so programmer & client applications are free from these operations.

- The jdbc connection object i.e created by programmer explicitly is called direct jdbc connection object. The jdbc connection object i.e collected from jdbc connection pool is called pooled jdbc connection object.

Note: It is always recomanded to work with pool jdbc connection object.

- All web, Application server like tomcat, weblogic and e.t.c allows to create a manage jdbc connection pool.

Here type-III diagram protocol

- In this Diagram we are using jdbc type-II in combination with other jdbc driver to make client application getting jdbc connection object from jdbc connection pool. most of the projects in a industry use jdbc driver type-III, type-IV in their projects development. they use type-II as a driver to create jdbc connection objects in jdbc connection pool. and type-III as protocal to communi-cate with jdbc connection pool from client applications.

→ jdbc Data source object represent jdbc connection pool so each jdbc connection object is jdbc connection pool that should be access jdbc connection object. to provide global visibility jdbc data source object it must be register with in a global pace called "Naming registry s/w" Like RMI Registy, DNS, & e.t.c, having petname or aliasname (JNDI Name). Naming registy is a s/w that can manage objects, Object refferences, having nick name, alies names. java applications use JNDI API (javax.naming package) to interact with naming registy s/w.

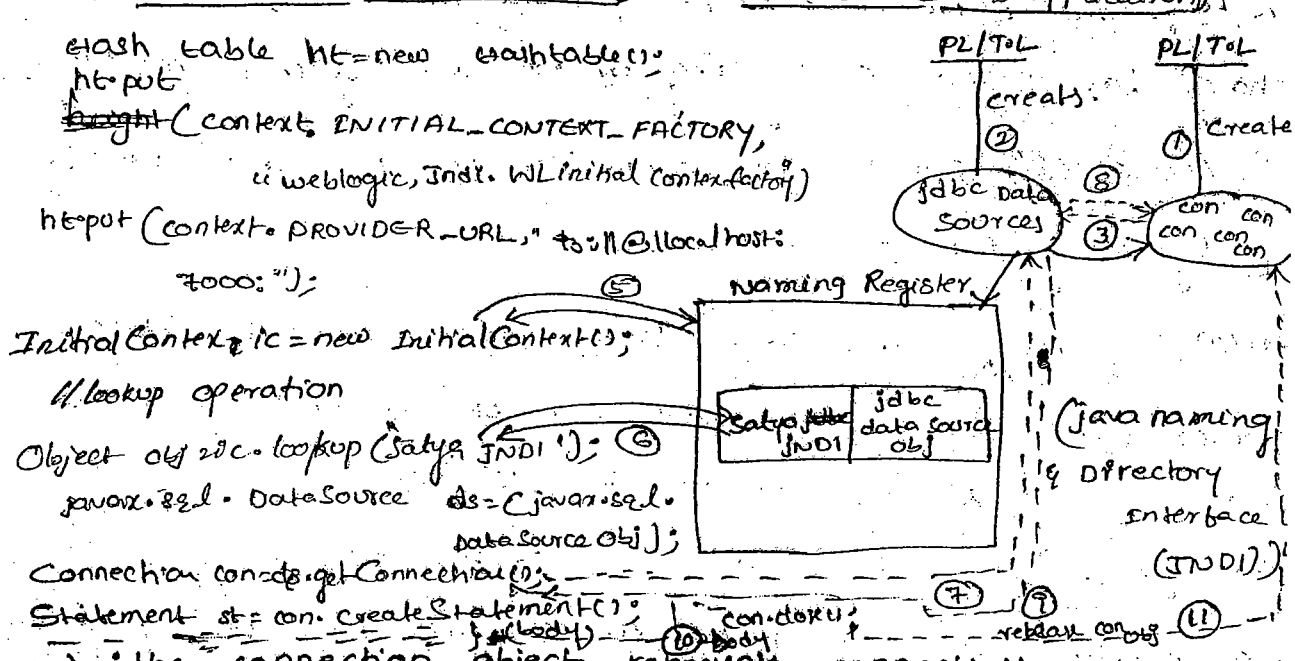
java app → JDBC API → DB s/w (Oracle, Sybase, MS-Access)

java app → JNDI s/w → Naming Registy s/w (RMI registy, DNS, OpenLDAP)

Note Every web, application server s/w comes with building naming, directory registy s/w in web logic s/w the built in namily registy s/w name is web logic directory registory.

Client side (Standalone Java app) Serverside (web application)

```
Hash table ht=new Hashtable();
ht.put
(Context.INITIAL_CONTEXT_FACTORY,
 "weblogic.jndi.WLInitialContextFactory")
ht.put(Context.PROVIDER_URL,"t3://localhost:
7000:");
InitialContext ic = new InitialContext();
// lookup operation
Object obj = ic.lookup("satya JNDI");
javax.sql.DataSource ds=(javax.sql.
DataSource obj);
Connection con=ds.getConnection();
Statement st = con.createStatement();
```



→ jdbc connection object represents connectivity to DB s/w & to create this jdbc connection object 4 details are require

① Driver class name

② URL

③ user name

④ password.

Initial context object represents connectivity with naming registry s/w to create this object two details are required

① Initial context factory class name (jdbc driver class)

② provide URL (similar to DB URL)

These two details should be store as the values of String constant placed in javax.naming.Context interface

— the details require to create initial context object are called JNDI properties (in above diagram) the process of getting object/object reference from naming, directory registry s/w through lookup operation is called based on nick name is called lookup operation.

Note :- The Initial context factory names provider URL values will change based on the naming Registry we use. Respect to the Diagram

① T.L Objects create jdbc connection pool for certain DB s/w By using type-IV JDBC Driver

②③④ :- T.L/RL creates Jdbc Data source, links that data source with jdbc connection pool and registers jdbc datasource with Naming Registry s/w for global visible

⑤ Client application creates InitialContext Object representing the connectivity with naming Registry s/w by using JNDI property

⑥ client application gets jdbc datasource object from Naming Registry through lookup operation.

⑦⑧⑨ Client application gets one jdbc connection object from connection pool through DataSource object.

⑩ Client application uses that connection object to create other jdbc objects and to manipulate Database Data.

⑪ Client application releases jdbc connection object back to Jdbc connection pool. by calling con.close();

procedure to create jdbc connection pool, Jdbc Data Source in web logic 8.1 s/w.

① Launch web logic s/w & administration console.

start
→ programms
→ BEA weblogic platform 1.8
→ Examples
→ web logic Server Examples
→ Lanch weblogic Server Examples

↳ Administrative console

↳ user name : weblogic

pwd : weblogic

signin

② Create Jdbc connection pool for Oracle.

Admin console

↳ Services

↳ Jdbc

↳ connection pools

↳ configar new Jdbc connection

↳ DataBase type Oracle

↳ Database Driver Oracle thin Driver

↳ continue

↳ name : mypool (Any name)

Service ID (SID) Database name : satya

Host name : localhost: 1521

DB username : scott

pwd : tiger

conform pwd : tiger

↳ continue

↳ test driver configaration

↳ Create & deploy

③ create ~~database~~ a Jdbc DataSource and link with above pool.

Administration console

↳ services

↳ Jdbc

↳ DataSources

↳ configar new jdbc Datasource

↳ name: myds (Any name)

JNDI name: Satya (nick name object Jndi)
Jndi

↳ continue

↳ pool name: my pool 1 1

↳ continue

↳ create

Note:- The moment we click the above create button Jdbc Data Source object will be register with RMI registry having Jndi name SatyJndi

[Satya → Right click → new jndi tree
click on Jndi Satyajndi
pool that Satyajndi is registered with RMI Registry]

Java application that collects connection object from above Jdbc connection pool to interact with DB s/w:-

weblogic.jar

```
import java.sql.*;
import java.util.*;
import javax.naming.*;
public class SelectTest
{
  P v s m (String KI) throws Exception
  {
    // prepare Jndi properties.
      Hashtable ht=new Hashtable();
      ht.put(Context.INITIAL_CONTEXT_FACTORY,"weblogic.jndi.
                                    WLInitialContextFactory");
      ht.put(Context.PROVIDER_URL,"t3://localhost:7001");

    // Create InitialContext
       InitialContext ic=new InitialContext(ht);
    // lookup operation.
      DataSource ds=(DataSource)ic.lookup("Satyajndi");
```

```
// get connection object from connection pool
Connection con=ds.getConnection();
Statement st= con.createStatement();
ResultSet rs = st.executeQuery("select * from student");
while(rs.next())
{
  s.o.p(rs.getInt(1)+" "+rs.getString(2)+" "+rs.getString(3));
}
con.close();
rs.close();
st.close();
con.close();
}
```

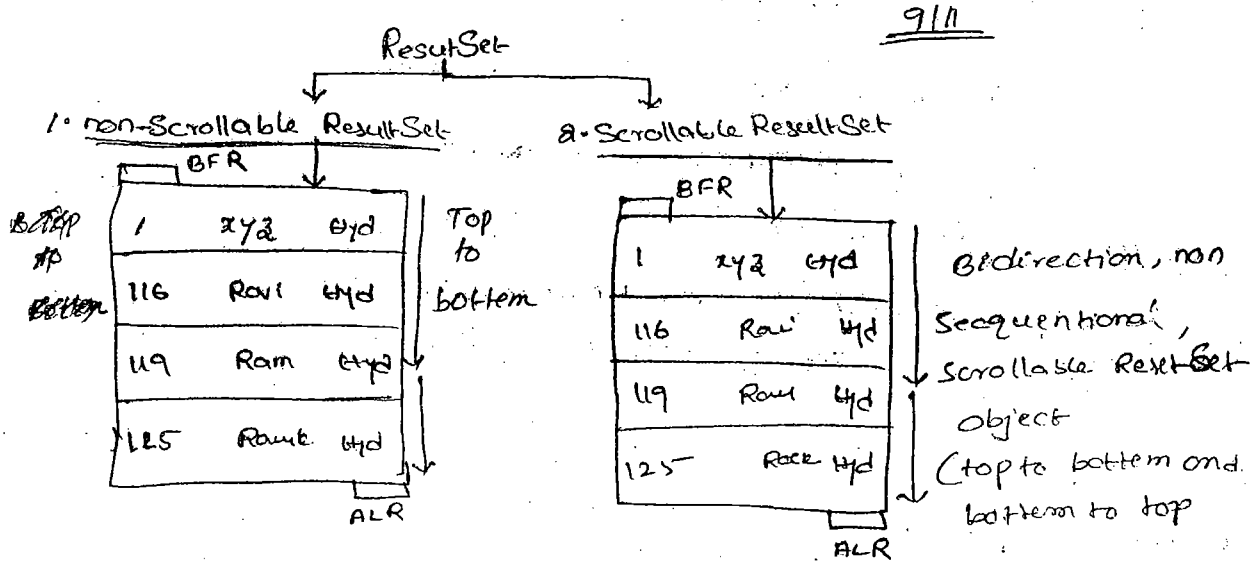
D:\bea\weblogic81\server\lib\weblogic.jar

contain ~~WLInitialfactory~~ WLInitialContextfactory.class

Set this ~~classpath~~ on CLASSPATH Environment variable.

NOTE :- To execute this above program add .weblogichome\weblogic 81\server\lib\weblogic.jar to the class path to recognise InitialContextfactory.class called weblogic.jndi.WLInitial contextfactory class.

9/11



- A ResultSet object that allows to access the records only in one direction i.e top to bottom is called non-Scrollable (or) unidirectional (or) Sequential access ResultSet object.

the resultset object that we have worked so for non-scrollable ResultSet object.

- the ResultSet object that allows to access the record bidirectional, randomly is called scrollable (or) bidirectional (or) non-sequential ~~object~~ access ResultSetObject.

Scrollable ResultSet Object use better performence that compare to non-scrollable ResultSetObject.

to create non-scrollable ResultSet object :-

```
Statement st=con.createStatement();
ResultSet rs=st.executeQuery("select * from student");
```

here "rs" is non-scrollable ResultSet object.

to create Scrollable ResultSet Object :-

```
Statemet st=con.createStatement(type, mode);
ResultSet rs=st.executeQuery("select * from student"); (there are integer parameters)
```

here "rs" is scrollable ResultSet object.

the posible values for type (int constant)

ⓐ ResultSet.TYPE_SCROLL_SENSITIVE

ⓑ ResultSet.TYPE_SCROLL_INSENSITIVE

the posible values for mode (int constant)

ⓐ ResultSet.~~TYPE~~CONCUR_UPDATABLE

ⓑ ResultSet.~~TYPE~~CONCUR_READONLY

note:- the posible values of TYPE, mode parameters are integer constants paced in ResultSet interface.

- public static final ~~member~~ variable of a java class/interface is called "constant"

- To create Scrollable ResultSet object the jdbc Statement object has to be prepare having type & mode values.

Methods to navigate on non-scrollable ResultSet object :-

next() ---> moves record pointer next Record
getRow() --> Gives the position of current record.

Methods to navigate through Scrollable ResultSet object :-

next()
previous()
Last()
first()
afterLast()
beforeFirst()
getRow()
absolute (+/- position)
relative (+/- position)
isFirst()
isLast()

Q --> what is the diff b/w absolute() & relative()?

A - absolute() moves the record point in ResultSet with respect to First record or Last record.

positive number indicate forword direction with respect to first record,

-ve number indicate reverse direction with respect to Last record.

- ~~Relative()~~ - relative() moves the record pointer with respect to current position in forword, reverse direction

+ve indicate forward direction
-ve number indicates reverse direction

→ write a program to print data of ResultSet object in both directions.

```
// ScrollTest.java

import java.io.*
import java.sql.*;

public class ScrollTest
{
   P S V m(String K[]) throws Exception
   {
      class.forName("son.jdbc.odbc.JdbcOdbcDriver");
Connection con=DriverManager.getConnection("jdbc:oracle:thin:
                                @localhost:1521:DSN","scott","tiger");
Statement st=con.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,
                                ResultSet.CONCUR_READ_ONLY);
ResultSet rs=st.executeQuery("select * from student");
S O P("TOP ---> Bottom");
while(rs.next())
{
   S O P(rs.getInt(1)+" "+rs.getString(2)+" "+rs.getString(3));
}
S O P("Bottom ---> TOP");
while(rs.previous())
{
   S.O P(rs.getInt(1)+" "+rs.getString(2)+" "+rs.getString(3));
}
rs.absolute(3);
S O P(rs.getRow()+" Record is "+rs.getInt(1)+" "+rs.getString(2)+" "
                                +rs.getString(3));
rs.relative(2);
S O P(rs.getRow()+" Record is "+rs.getInt(1)+" "+rs.getString(2)+" "
                                +rs.getString(3));
rs.relative(-3);
S O P(rs.getRow()+" Record is "+rs.getInt(1)+" "+rs.getString(2)+" "
                                +rs.getString(3));
```

```
rs.absolute(-4);
SOP(rs.getRow()+" Record is "+rs.getInt(1)+" "+rs.getString(2)+" "
rs.last();                                    +rs.getString(3));
SOP(rs.getRow()+" Record is "+rs.getInt(1)+" "+rs.getString(2)+
                                              " "+rs.getString(3));
}
}
```

Note:- We can create Scrollable ResultSet in 4 modes

1. As "Sensitive" ResultSet object
2. As "Insensitive" " "
3. As "Readonly" " "
4. As "updatable" " "

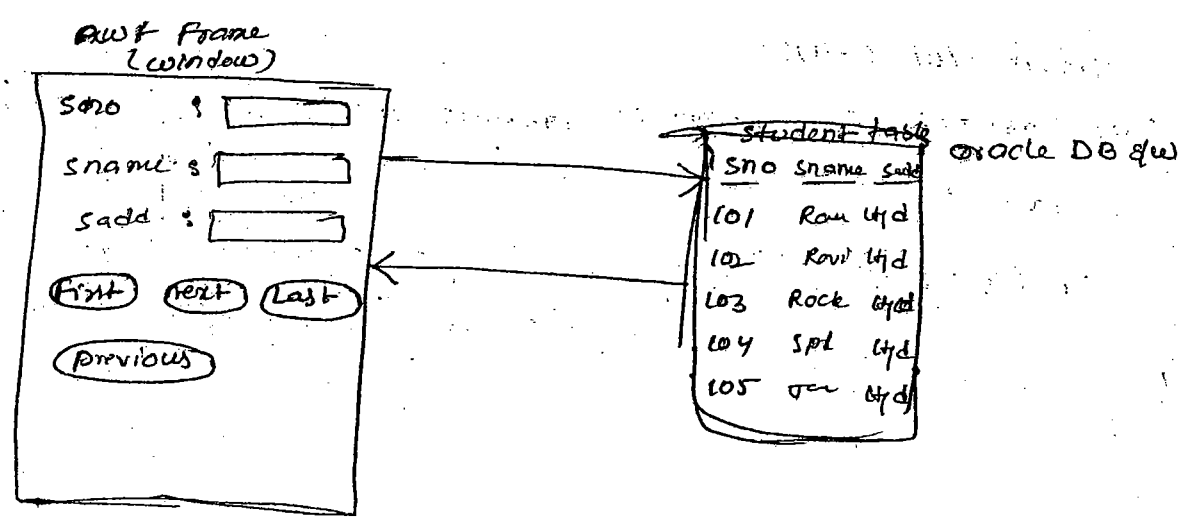
→ Any java/JEE Application can interact with DB s/w by taking support from jdbc code.

→ All types of jdbc Drivers support the scrollable behaviour of ResultSet object.

→ In order to make GUI java application interacting with DB s/w place jdbc code in awt/Swing application

→ while developing large scale application it is always recomended to place jdbc object like connection, statement in one time execution block like constructor or static blocks and write the code of using these objects for table data manipulation. in repeatedly executing block like java methods are Event handling methods.

→ constructor is object level one time execution block where as static block is class level one time execution block.



awt setup

container : Frame window

components : Label - 3, textfield - 3, Button - 4

Layout manager : FlowLayout

Event : : ActionEvent
EventListener

EventListener : ActionListener

- the above application is an awt frame based desktop application interacting with DB s/w.

- To develop above application we need scrollable ResultSet object.

→ For the source code of above application Refer Application - (10) of the booklet page no: (21) to (23).

10/4

- we can develop GUI java applications as Desktop/windows applications by using javaw tool, GUI java applications are nothing but awt/swing applications. javaw is a built in tool of jdk s/w

procedure :- Create short cut any window of windos o.s

like right click in window → new → shortcut → ~~text~~ → javaw - scrollframe → GUI java application.

↳ next
type name for the shortcut
[MyApp]
↳ finish.

Open the properties of above shortcut
↳ shortcut tab
↳ start in : E:\app\adw.java\jdbc
↳ location of the above given application.
↳ choose shortcut key
↳ choose icon
↳ apply
↳ ok

[character user interface.

Note :- we can't make CUI make applications as window application.

Procedure to develop application and (scrollFrame) by using Netbeans IDE's based on swings :

① create java project by Launching netbeans IDE

File
↳ new project
↳ General
↳ java app
↳ next
↳ my project (projName)
↳ finish

② Create JFrame in the project

right click on project
↳ new
↳ JFrame : class name : scrollFrame
↳ finish.

③ Design the Frame having components.

④ Add following code in the source code of application.

Note write bellow code after registring action events on the button from design mode

☞ rightclic on Button design mode
↳ Events
↳ Action
↳ ActionPerformed.

→ write following code in make connection method and call that method From constructor

```
public void ~~make~~ makeConnection
import java.sql.*;
public class ScrollFrame extends javax.swing.JFrame
{
    ResultSet rs=null;
    public ScrollFrame() {
    InitComponents();
    makeConnection();
    }
public void makeConnection()
{
        page no: 22  Line no: 627 to 641
}
}
```

Add following ~~query~~ code jButton1 ActionPerformed().

```
try    rs.
{
    rs.first();
    jTextField1.setText(rs.getString(1));
    "   2.      "   (2));
    "   3.      "   (3));
}
catch (Exception e)
```

```
    {
        e.printStackTrace();
    }
}
```

Add following code in JButton2 Action performed method

```
try
{
    if(rs.isLast())
    {
        rs.next();
        jTextField1.setText(rs.getString(1));
        "    "  2.    "    (2));
        "    3.    "    (3));
    }
}
catch (Exception e)
{
    e.printStackTrace();
}
```

Add Following in JButton3. Actionperformed method

```
try
{
    if(rs.isFirst())
    {
        rs.previous();
        jTextField1.setText(rs.getString(1));
        "    2.    "    (2));
        "    3.    "    (3));
    }
}
catch (Exception e)
{
    e.printStackTrace();
}
```

→ Add following in JButton4. ActionPerformed method

```
try
{
    rs.last()
    jTextField1.setText(rs.getString(1));
    "    2.    "    (2));
    "    3.    "    (3));
}
catch(Exception e) { e.printStackTrace(); }
```

Add classes12.jar file to the libraries of the project.

Extends project.

↳ rightclick on Libraries

↳ Add jar

↳ classes.jar

Run the project

rightclick on source code of Project

↳ run file.

////

→ wt is the different b/w Sensitive ResultSet object, In Sensitive ResultSet Object?

- When ResultSet object representing a DB table from c java application if the modifications done in DB table are immidiatly reflecting ResultSet object. then the resultSet object is called has sencitive ResultSet object. if not reflection then the resultSet object called Insencitive object.

- TO create Sensitive ResultSet object

```
Statement st=con.createStatement(ResultSet.TYPE_SCROLL_
                                 SENSITIVE, ResultSet.CONCURE_UPDA
                                 BLE)
ResultSet rs=st.execute Query("select * from student");
```

TO creative Insensitive ResultSet Object

```
Statement st=con.createStatemet(ResultSet.TYPE_SCROLL_INSENSIT
                                VE, ResultSet.CONCURE_UPDATABLE)
ResultSet rs=st.execute Query("select * from student");
```

By example application of Sensitive ResultSet object

Refhere App:15 page no: 86

→ JDBC Driver Type-one given by sunmicro systems supports Sensitive ResultSet object but dosenot supports Insensitive ResultSet object

- oracle oci driver, oracle thin driver, J connection pointer & mysql support only insensitive ResultSet objects Don't support Sensitive ResultSet object.

Insensitive objects because there is no commonality of the JDBC Drivers towards the support of these ResultSet objects.

→ Wt is the difference b/w Read only ResultSet object and and updatable ResultSet object?

Ⓐ:- If the modifications done in ResultSet object are immediately reflecting in DB table represented by resultset object then the ResultSet object is called updatable ResultSet object. otherwise the ResultSet object is called Readonly ResultSet object.

→ Statement st = con.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE, ResultSet.CONCURE_UPDATABLE).

ResultSet rs = st.~~createStatement~~executeQuery("select * from student");

To create Read_only ResultSet object :-

Statement st= con.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE, ResultSet.CONCUR_READ_ONLY)

ResultSet rs= st.executeQuery("select * from sno");

→ By using updateble ResultSet object we can perform insert, update, delete operations on DB table i.e represented by ResultSet object with out using sql server.

→ // Read the records

```
while(rs.next())
{
  S.O.P(rs.getInt(1)+" "+rs.getString(2)+" "+rs.getString(3));
}
```

// To insert the records.

rs.moveToInsertRow(); // creats an empty record in ResultSet

rs.updateInt(1,456);
rs.updateString(2,"rajesh");
rs.InsertRow(); // insert the records in to table.

} Sets data to empty Records.

// To update the existing record.

rs.absolute(2);
rs.updateString(2,"remesh");
rs.updateRow();

// To delete existing record

rs.absolute(2);
rs.deleteRow();

Limitations with updatable ResultSet object :-

① dose't allow criteria based non select operations.

② dose't allow bulk operations (non select operations DB table)

→ For example application in resultset object refer [App:17]

[on page no: 27, 28]

→ JDBC Driver type-I given by sunmicrosystems represents both updatable, Read-only resultSet objects, where as Jdbc, oci, oracle thin driver, Jconnector Driver dose't support updatable resultSet. They support only Read-only ResultSet object.

→ It is always recomanded to avoid working with updatable ResultSet object. because all Jdbc driver are not supporting updating ResultSet object in a common way.

→ A standard principle s/w industry is don't hardcode (type) any value in an java application that are changable in the feature collect these values from outside the application to make java application as flexible application.

→ To make java jdbc application as possible application it is recomanded to connect 4 values of the application from

② DB database url.

③ DB database user name.

④ JDBC password.

→ A textfile that maintains entries in the form of key=value pairs is called properties value.

→ Developing JDBC application as flexible application with the support of properties value.

① prepare to properties file having details about jdbc driver.

```
# details.txt

driver = sun.jdbc.odbc.JdbcOdbcDriver
url = jdbc:odbc:oracle.
DBuser = scott.
password = tiger.
```

→ write a java application by using details of properties file.

```
import java.io.*;
import java.sql.*;
import java.util.*;

public class filetxt
{
    P s v m(String x[]) throws Exception
    {
        FileInputStream fis = new FileInputStream("  ");
        Properties p = new Properties();
        p.load(fis); // load values from properties file
                     element
        String s1 = p.getProperty("driver");
        "   s2 =   "   ("url");
        "   s3 =   "   ("DBuser");
        "   s4 =   "   ("scott");
        class.forName(s1);
```

```
Statement st = con.createStatement();
ResultSet rs = st.executeQuery("select * from student");
while(rs.next())
{
  s.o.p(rs.getString(1) + " " + rs.getString(2) + " " + rs.getString(3) +
                                                    " " + rs.getString(4));
}
rs.close();
con.close();
st.close();
}
}
```

→ java.util.*; is a map data structure and it's a of java.util.hash table class. the element values of data structor can be gathered from outside properties file.

12/11

→ It is always recomanded to insert data values as string values in the DB table columns. because string type data values do not allow comparison blw two dates and then. database operations. To overcome this problem insert data values in the table, in data types column. Directly has data values in the pattern i.e supported by DB sql date of joining, date of birth, date of course completion values are generally will be taken as data values.

→ If you pass java.sql.Date class object to jdbc Driver representing the data values the jdbc driver takes the responsibility inserting date values into DB column. in the angle it required for DB s/w.

Date class?

→ java.util.Date class object represents date, time value that is usefull in normal java application. where as java.sql.Date class object represents sql date values to insert date value in data. data types columns of DB s/w.

Note:- java.sql.Date is the sub class of java.util.Date.

Date patterns means
- DD-mm-YY
- mm-DD-YY
- YY-mm-DD

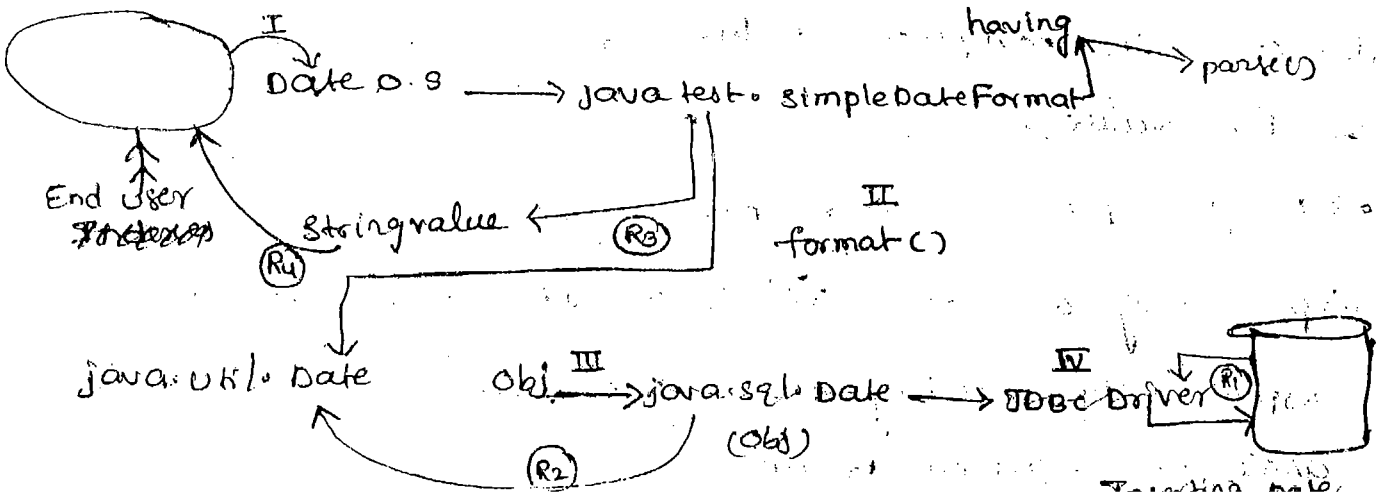
} Date patterns

→ Each DB s/w is having specific date patterns that it supports

- Oracle Date pattern is: mm-DD-YY
  12-NOV-09
- SQL Date pattern is: mm-DD-YY
  12-11-09

→ the jdbc programmer should work with common logic who files not specific to any DB s/w to insert and retrive DB values.

Processor to insert Date and value



in his own pattern. II for using parse() method of simple Date format class String date value will be converted into java.util.Date class object.

- III) java.util.Date class object is converted into java.sql.Date class object

- IV) JDBC Driver inserts Date value of java.sql.Date class object in DB table in the angle(pattern) i.e required DB s/w.

→ To convert Date class obj into java.util.Date class format

```
import java.sql.*;
import java.util.*;
import java.text.*;
{
   public class testapp
   {
//to convert string date value to java.util.Date class object
String d1="2009-10-20"; //YY-MM-DD
SimpleDateFormat sdf1= new SimpleDateFormat("YY-MM-DD");
java.util.Date ud1= sfd1.parse(d1);
S.O.P (ud1.toString());
//to convert java.util.Date obj to java.sql.Date obj
        ms=ud1.getTime();
    java.sql.Date sqd1= new java.sql.Date(ms);
    S.O.P ("sqd1.toString());
```

- In the above code ud1 is java.util.Date and this ud1.getTime() method returns

To date and time value represented by java.util.Date class obj only.

→ the parse() of java.text.SimpleDateFormat class is capable of converting string date value to java.util.Date class object

→ If given string date value pattern is yy-mm-dd, we can convert that date directly in to java.sql.Date class object with out converting into java.util.Date class object

→ If the given string Date value is in other that yy-mm-dd pattern we need to convert 1st in to java.util.date class object then into java.sql.Date class object.

Converting yy-mm-dd pattern date value to java.sql.Date object

```
String d2 = "1991-11-30"; // yy-mm-dd
java.sql.Date sqd2 = java.util.Date.valueOf(d2);
S.O.P (sqd2.toString);
```

(R1) java application reads date value from DB through jdbc Driver in the form of java.sql.Date class object.

(R2) convert java.sql.Date object to java.util.Date class object

(R3) format() method of simple date format class converts java.util.Date class object date value to string date value

(R4) End user Resources date value in the from of string value.

// to convert java.sql.Date object to java.util.Date object

```
java.util.Date ud2 = (java.util.Date) sd1;
S.O.println(ud2.toString());
```

// to convert java.util.Date object to string date value.

```
SimpleDateFormat sdf2 = new SimpleDateFormat("dd-mm-yyyy");
String d2 = sd2.format(ud2);
```

note :- All jdbc privent are giving support to work with date values

the above given processors to insert, retrive date values are not specific to any DB s/w's.

date values from ResultSet object the date pattern updates will be decided based on the Jdbc driver we use.

→ for oracle oci thin driver Date pattern is mm-DD-YY

→ for Jdbc Type-I : YY-mm-DD.

→ for oracle DB s/w the Date pattern of Date values is DD-mm-YY.

→ DB access in mysql DB s/w the Date pattern is YY-mm-DD format

Rowsets :-

we can send any there java objects (even the s/w) that are Serializable ~~table~~. to make java object as Serializable object one class of the object must implements to java.io.Serializable interface.

→ To overcome this problem we work with Rowset object which are extension to rs object and also ~~object by~~ Serializable object by default.

→ Rowset allows programer to develop Jdbc application based on setter(), getter() methods.

→ Rowset object is a object of access that implements java.sql.Rowset interface.

→ s/w vendors are implementing Rowset in 3 forms.

① cached Rowset → disconnected Rowset → insensitive Rowset object.

② JDBC Rowset → connected Rowset → Sencitive Rowset ~~default for~~ gather ~~data from xml files~~ object.

③ Web Rowset → connected Rowset → useful to gather data from xml files

→ oracle corporation is giving Support of Rowset's in the form of 3 classes.

① Oracle cached Rowset (o) ---> as cached Rowset

② OracleJdbcRowset (o) ---> as jdbc Rowset

③ Oracle web Rowset (o) ---> as web Rowset

→ All these three classes are avalible in ocrs12.jar and for this jar file ojdbc14.jar independent jar file.

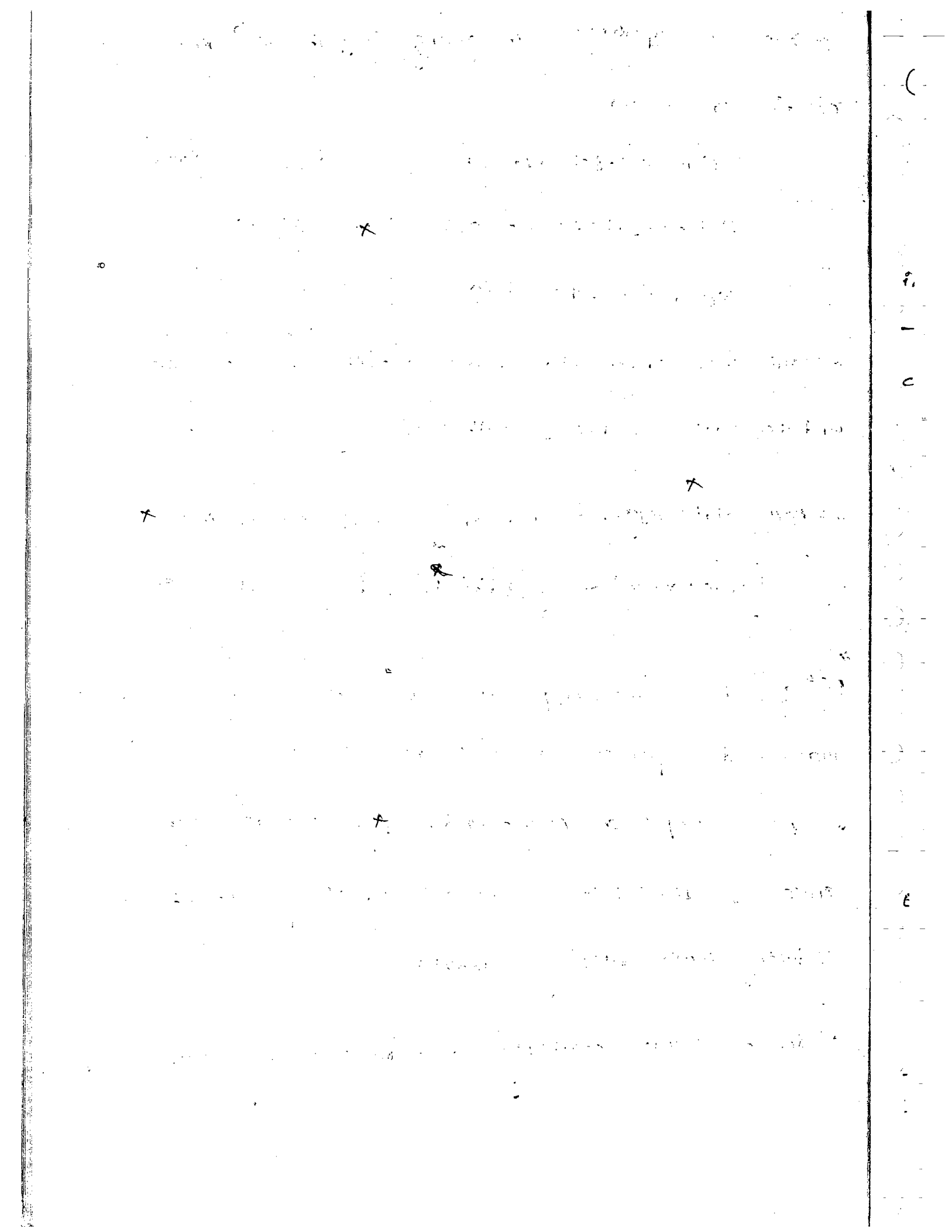
→ Both ojdbc14.jar, ocrs12.jar files are avalible in

Oracle Home\ora92\jdbc\lib folder (~~ojdbc12.jar~~)

Note :- when java application uses IIIrd party API Related main and dependent jar files must be added in the classpath

→ For examples on Rowset refer application 18 & 19 of booklet since as JDBC Drivers are not supporting Rowset s/w industry avoids using ResultSet.

→ Rowset are the capability of performing only select operations



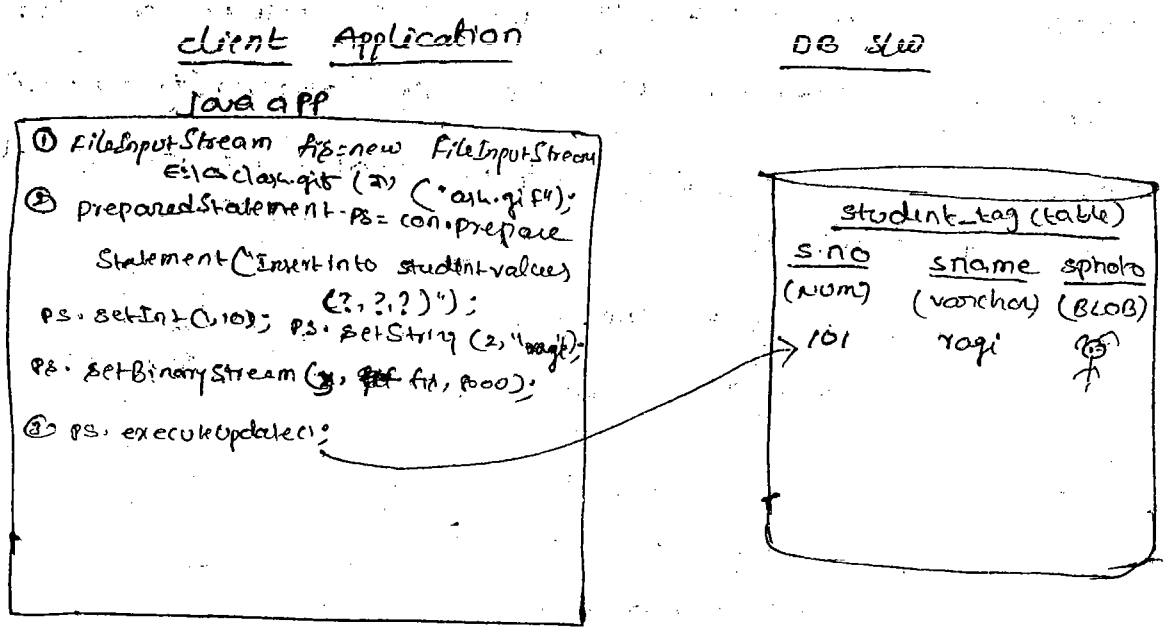
Large object files (LOF)

(Character) CLOB — TXT files, Doc files, RTF files

(Binary) BLOB — avot files

- By talking column data type as LOB we can insert different types of files in table columns as values.

- Oracle DB s/w provides blob, clob data types to allow columns storing large objects as values.



E:\abc
↳ ash.gif

→ To set value to DB table column whose data type is BLOB use setBinaryStream(-,-,-) (or) setBLOB(-,-,-)

→ Similarly to set value to table column whose data type is

Sample code to insert photo graph into DB table column.

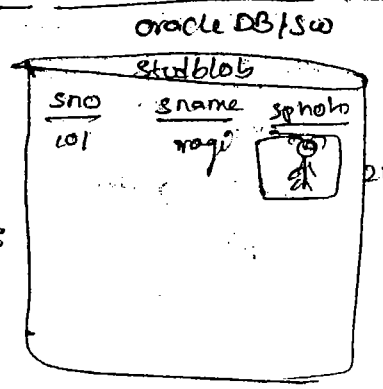
For Example application on inserting and retriving photo graph an application:(22) page no:(31)

- Bubber is a temporary memory that stores data and reduces round trip b/w source ~~data~~ & destination resources. while transfering the data

- To read data from BLOB type table column use getBynary Stream () getBLOB(), lly to read column values from getCLOB type table column we use getCharacterStream() method (or) getCLOB().

Sample code for photo retival operation

java app client app



① statement st=con.createStatement();
ResultSet rs=st.executeQuery("select * from student");

② FileInputStream fis=rs.getBinaryStream(3);

③ FileOutputStream fos=new FileOutputStream("D:\asclan.gif");

④ byte[] bubber=new byte[1024];
int count=0;
while((count=fis.read(bubber))!=-1) // reading content of the bubber 1024 times.
{
fos.write(bubber,0,count);
}

→ For complete code refer photoretriver.java of application:22 pageno:32

→ In realword no one stores larg objects directly in table column because this process will degrated a performance. DB. Instead of they store large object related files in certain directory of computer &

columns.

— To work with Jdbc connection pool of oracle without taking support from webservers & Application server the oracle corporation apply jdbc connection pool to use in standlone applications this pool comes along with oracle 9i installation in the form of jar file.

→ Oracle-Connection-pool DataSource object represents a standalone Jdbc connection pool supplied by oracle DB

```
// connpoolTest.java.
import java.io.*;
import java.sql.*;
import javax.sql.*;
public class connPoolTest
{
  p s v m(String x[]) throws Exception
  {
    OracleConnectionPoolDataSource ds=new OracleConnectionPoolDataSource();
    ds.setDriverType("thin");
    ds.setServerNames("localhost");
    ds.setPortNumber(1521);
    ds.setServiceName("satya");
    ds.setUser("scott");
    ds.setPassword("tiger");
    pooledConnection pcon=ds.getPooledConnection();
    Connection con=pcon.getConnection();
    Statement st=con.createStatement();
    ResultSet rs=st.executeQuery("select * from student");
    while(rs.next())
    {
      sop(rs.getInt(1)+" "+rs.getString(2)+" "+rs.getString(3));
    }
```

ojdbc14.jar file in the class path.

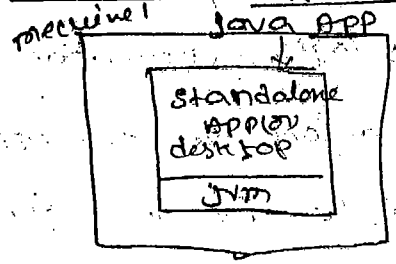
** — when jdbc program is executing querys by disabling auto commit mode the querys that are kept in Save point can be commited or rollback saparetly irrespective of other querys execution

Ex:- After disabling autocommite mode if 10 query are executing by Jdbc application if we want to rollback 5 queries and if we want to commit remaining 5 queries then we need to prepare Savepoint in jdbc application so that the query is placed in Savepoint ~~will be executed~~ can be commited or rollback without worried about the other queries executing data.

Savepoint object needs it is a object of a class that implements javax.sql.Savepoint Interface.

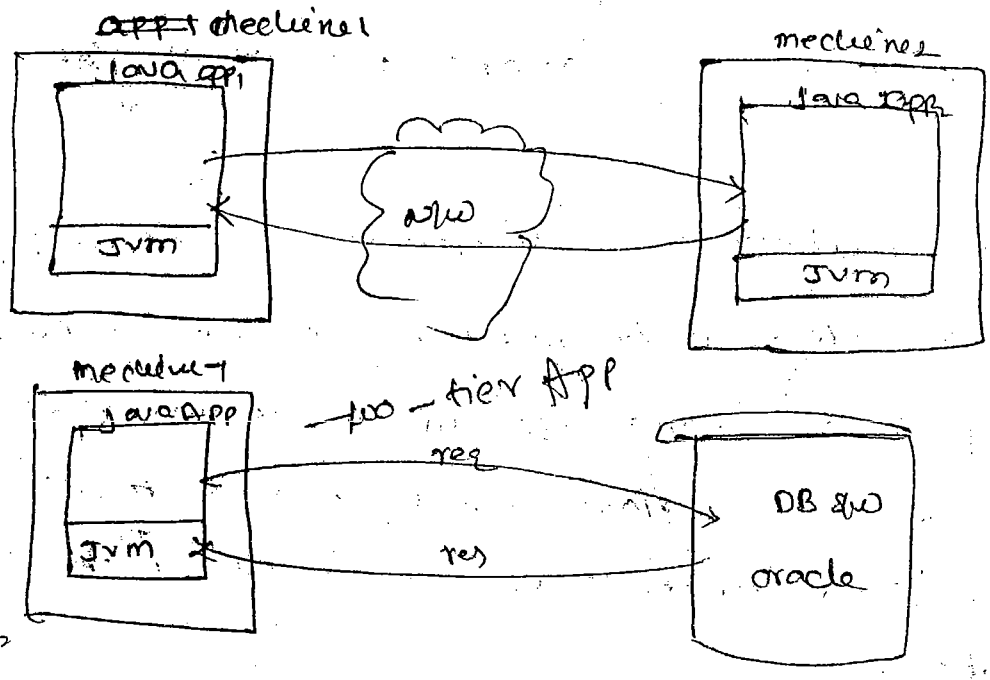
```
import java.sql.*;
import javax.sql.*;

public class SavepointTest
{
    p s v m (String args[]) throws Exception
    {
        class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection con = DriverManager.getConnection("jdbc:~~odbc~~ oracle : thin:
                                        @localhost:1521:","scott","tiger");
        con.setAutoCommit(false);
        Statement st = con.createStatement();
        SavePoint svpt = con.setSavepoint("a1sp");
        ~~Savepoint svpt = st~~ executeUpdate("update student set name=
                                        'Hy' where sno=101");
                  svpt = st.
        con.rollback(svpt);
        con.commit();
    } // main
} // class.
```

Standalone application (or) one tier App



Servlets

two-tier App:



→ Standalone and desktop applications are visible & accessable only in a computer where it is placed. All core java application expect socket programming, Applet are basically standalone/ Desktop applications. The GUI mode standalone applications are called desktop applications.

→ java socket programming application are jdbc applications are called two tier App. two tier App means Application contain two layers These two layers can reside in the same computer are in two different computers. Each layer of the application represents a logical position of the application having separate logic are sw.

much ristricted. So the business logic, application, data & application resources of these standalone, two tier applications are not globally accessable & not visible for new, unknown client.

- To overcome this problem develop applications as web App / web site so the resources will act as global resources.

Q → wt is the difference blw web Application & website?

A web application i.e hosted on the internet nlw by purchasing domain name space in internet nlw is called website. website provide global visibility to client to access it's resources because clients can use internet nlw as the channel of communication.

A → web application contain webpages to generate these web pages we use web resource programms. these web resource programms are capable of generating both static and dynamic web pages. the web resource that generates static web page (fixed content) is called static web source programming.

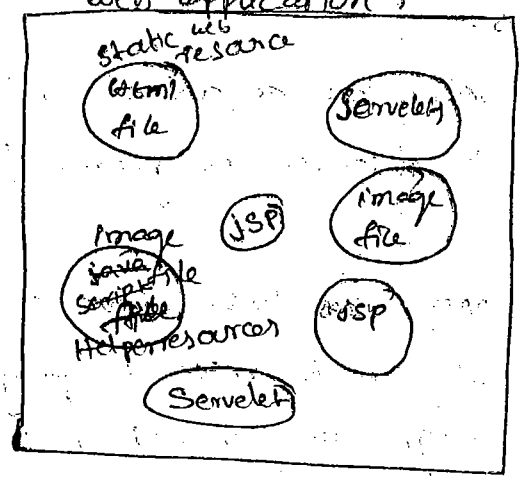
Eg: HTML file,

The web resource program i.e capable of generating dynamic web pages (dynamic content) is called dynamic web ~~page~~ resource program.

Eg: Servelet, JSP & etc.

Dynamic web source programs generate web page content based on time of request generation and based on the input values given to the request.

→ A web application is a collection of both static & dynamic web resource programs.



* Servelets, JSP's are Dynamic web resources
* Html is static web resources
* ~~java script~~ file is Image help resources

java script, Image files are helper web resources other main resources of web application like Html, Servelet & etc

→ To execute standalone java applications jvm, jre is required. By

→ to execute Applets & Html files browser s/w is required By to execute java based dynamic web resource program of a web application like Servelets, JSP's we need webServer s/w.

→ java based webserver s/ws internally used Jre & Jvm technologies.

→ A web server is a peace of s/w i.e capable of listining Http request continuously. manages web application & executes web resource programms dynamically when they are requested from web client (browser windows)

→ A web server s/w provides the necessary ~~setting~~ i.e required to execute web resource programs of web application and delivers their results response to browser window (client).

executing web resource programs of web applications which are like fishes.

Java based webservers are capable of managing & executing java based web resource programs like servelet & JSP's

web application developer uses both java & non java web technologies to develop web resource programmer of web application. but programmer is not responcible to develop web server s/w's they are avalible as readymade s/w in the market

Ex tomcat ~~web~~ resin, and etc

The process of "moving" developed web application to a web server is called "deployment" similarly the process of "removing" web application from web server is called "undeployment".

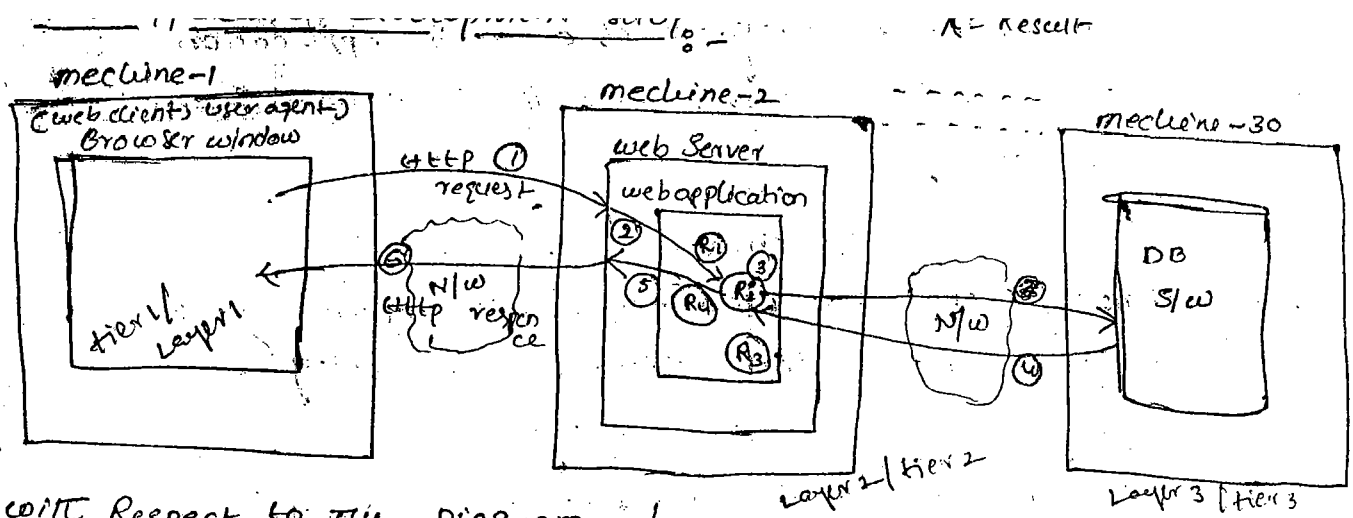
note programmer develops web application having web resources by using web technologies and deployes that web application in ~~tomcat~~ web server then web server manages and execute web resource programmes when they are requested by clients then delevers results as responce to client (browser).

17/11

→ To develop standalone java application JSDK s/w is required.

→ To develop java based & jdbc based two-tier Application JSDK s/w, Jdbc Driver and DB ~~s/w~~ s/w are required

→ To develop & execute web application we need web browser, web server, web technologies to develop web resources and DB s/w are required.



! With Respect to the Diagram browser window gives http request to a web resource of web application

2. web server takes this request and passes to appropriate web resource of web application.

3. the business logic and request processing logic of web resource program execute (in R3) (R3 (3))

4. web resource program interacts with DB s/w if necessary

5. Result
~~Responce~~ of web resource program goes to web server

6. web server sends this result to browser as http Responce.

⇒ Based on the type of the response content that a web resource program generate. there are two types of web resources

① Static web resource.

② Dyamic " " .

→ Based on the place where web resource program executes there are two types of web resource programms

1. client side programs.

2. Server side " .

• In web resource program of web application comes to browse for execution then it is called "~~Static web resource~~" program (or) "client side web resource program"

executes with in the web server when it is called "Server side web resource program".

Ex Servlets, JSP

→ The technologs that are used to develop client side web resource programs are called "client side technologies".

Ex html, Java script technologies

→ The technology that are used server side web resource program are called "Server side technologies"

Ex Servlets, JSP/ASP.net technologies. ASP;

- we can use client side technologies (html technology in combination with any server side technologies.

Browser s/w's :-

Internet Explorer - - -> micro soft (2) priority
Netscape navigater ——> Netscape (1)
Opera - ——> opera s/w
Gogle chrome ——> Google
Hot java ——> Red Hat
mozilla fire Hot ——> mozille s/w (3)

web server s/w's :-

Tomcat ——> apache (1) (Java based)
(Java web server) jws ——> sun micro system (")
(personal web server) pws ——> m8
Internet Information Server IIS ——> ms (3)

Jetty ——> Adobe (java based)
~~retrig~~ resin web server ——> Resin w/s (") (2)
Apache web server ——> Apache (")

All application Servers (Extension of web servers) :-

web logic ——> BEA system (oracle corporation) (1) java based
web sphere ——> IBM (java based) (2)
JBoss ——> Apache (") (3)

JRUN - - - - - > macro media (Adobe) (")

OlogAS - - - - > oracle corporation (")

GlassFish ——————> sun ms(Oracle corporation (") (4)

NOTE:- Every application server s/w can be used as web server. In web server we can deploy and manage only web application where as application server we can deployed manage web application and EJB components.

client side technologies :-

Html - - - - - -> world wide web consoritation (w_3c)

java script ——————> net scape company

Vb script ——————> micro soft

AJAX ——————> Adaptive path

(Asynchronous java script and extension)

Server side technologies :-

CGI ——————> open community

(personal homepage ← PHP ——————> Apache. (4/8)

for hipertext processing) ASP ——————> micro soft

(Active server page)

asp.net ——————> (") (5/4)

(Server side java script) SSJS ——————> netscape

Servlet ——————> sun microsystem (7) (java based)

JSP ——————> (" (3) (1) ")

→ To develop java based web application the following s/w are required

① Any Browser s/w

② java based web server / application server.

③ Any client side technologies

④ Server side technologies (Servlet / JSP)

All server side technologies are not compatable with on web servers. asp, asp.net are compatable with pws, IIS Server.

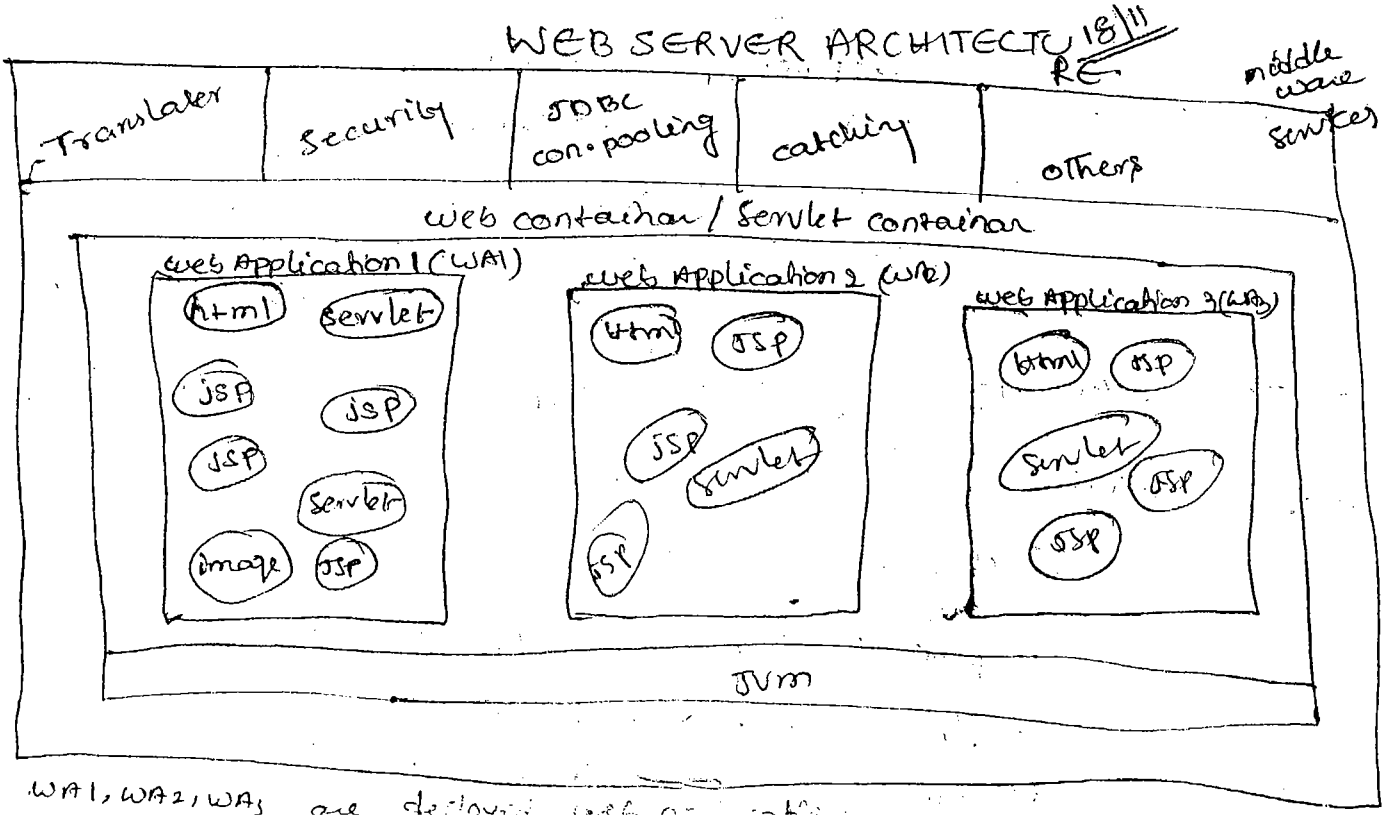
- CGI, PHP programs are compatable with apache web server.
- Servlet, JSP programs are compatable with java web servers.

DB S/W:-

Oracle (oracle corporation) (1)
sybase (sybase)
MS-Access (microsoft)
My-sql (sunms) (2)
Post gre sql (opencommunity) (3)
DB/2 (IBM) (4)

→ web application development needs multiple technologies utilization, we can develop web application either as two-tier application (without DB S/W) (or) as three-tier Application (with DB S/W).

- the main logic of the web resource program that processes the given request and generates result to web server to send as responce is called request processing logic or business logic.



WA1, WA2, WA3 are deployed web applications

→ In one web server we can deploy and manage zero or more web applications.

Server.

→ In one web Application we can see one or more web resources.

ver.

→ web container is the built in s/w of web server at comes

eg:

with web server s/w installation.

→ Addition services that are configurable on our applications are called "middleware services". middleware services make applications as perfectly & activate.

Eg: transaction service, security service & e.t.c

NOK → middle ware services are not minimum logics of the Application development. they are additional logics that are configa

hon → ble on the applications.

Responcibilities of web Server:

① Receives Http request from client.

② Delivers Http responce to client.

③ provides additional services that are configarable on the deployed web applications.

④ provides environment to deploy & undeploy web application.

idle

are

vices

⑤ provides environment to perform administrative operations on the server.

~~Responcibilities of web container~~

Responcibilities of web container:-

① provides JVM, ~~JRE~~ jre

② provides setup i.e required to execute java based server side web resource program like servlets & JSP.

③ provides environment to manage web resources of web application.

tomcat

- type: webserver
- version (5.0) (J2SDK(1.4)), version (5.5) (J2SDK(1.5)), version (6.0) J2SDK(1.6)
- vendors: Apache
- open sorce s/w
- name of web containar/servlet containar
- Default port no: 8080
- default username and password are tomcat Administration
  username: Admin
  pwd:

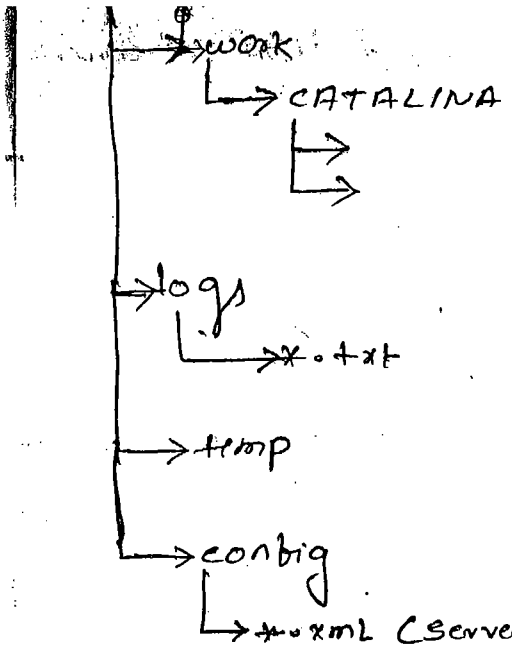
Procedure to Installing configar Tomcat 5.5 s/w:

Step.1: Install tomcat server by using setup file. during this installation choose.

1. Installation folder
2. J2SDK Home directory
3. port number
4. Admin login: username and password

Step 2) understand the diretory structure of tomcat server. that comes after installation

- D:/
  - tomcat 5.5
    - bin directory
      - *.exe (tomcat.exe)
    - common
      - Lib folder
        - *.jar (servlet_api.jar, JSP_api.jar)
    - Server
      - Lib
        - *.jar
    - Shared
      - Lib
        - *.jar
    - web-apps
      - *.war
      - Div1, Div2, Div3



D:\tomcat 5.5 — called tomcat home directory

→ to start tomcat web server

tomcat Home
→ bin
→ tomcat5.exe file

→ Servlet, JSP's are part of J2EE. But J2EE is not a installable s/w. So the web server that are given based on Servlet, JSP supply jar file, representing servlet, JSP API. in tomcat web server.

~~tomcat Home / common / lib / servlet-api.jar, JSP-api.jar~~
common
lib

tomcat Home / common / lib / servlet_api.jar, JSP_api.jar

represents Servlet API, JSP API, respectively.

~~Step3~~ All deployed webapplications of tomcat server will be available

tomcatHome/webapps in the form of directorys & jar files

Start tomcat server :-

Start
→ programs
→ Apache tomcat 5.5
→ configure tomcat
→ Startup tipe manual

goto tomcat Home / bin/ and use tomcat 5.exe file to start the server.

steps → open Home page of tomcat server

http://localhost:2020 (portnumber)

open the browser & type all "URL" Administration.

2020 is port-number

localhost : host name where tomcat server is install.

to shoted to shut down tomcat

start
→ programs
→ apache tomcat 5.5
→ ~~started~~ configure tomcat
→ stoped
→ OK

Note :- when tomcat server is started to starts a dæmon process. The process that runs continuesly is called ~~dæmon~~ daemon process.

How to change port-number of tomcat server i.e already installed.

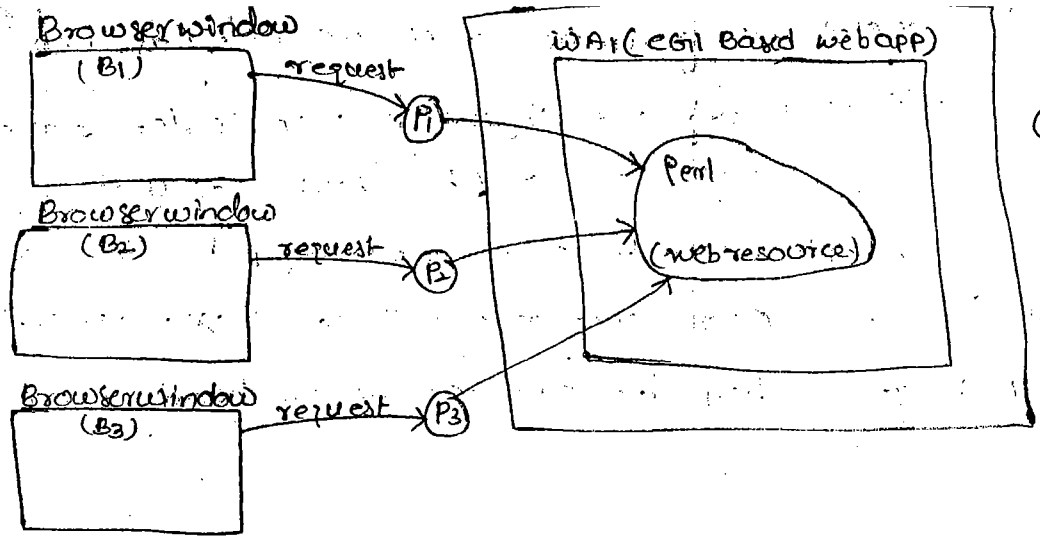
goto tomcat-Home/ conf/ server.xml file, chang port attribute value of first connector tag → restart the server.

→ If multiple s/w's installed in a computer uses same port number then, there is a posibility of getting "port number" clash. to overcome this problem change the port-number of s/w. by changing port-number of s/w it is recomanded to use numbers the range of 10252 to 65535 because the port number 121024 are busy with operating system services

Server side technology

Processbased
CGI

ThreadBased
Servlets
JSP

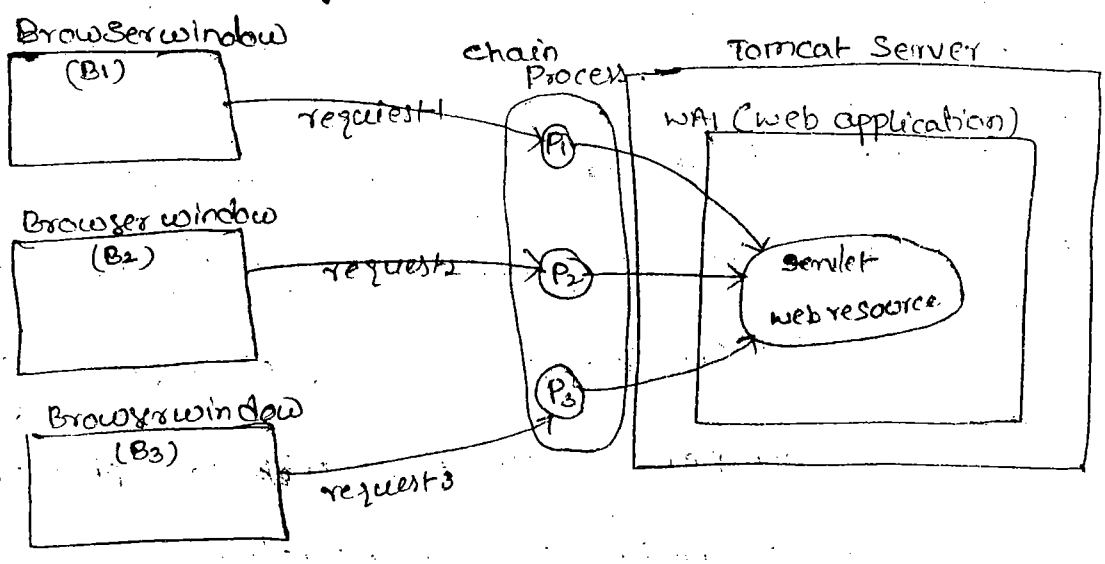


P1 P2 P3 are O.S processes.

→ when request comes to 'perl' program of CGI based web application one operating system process will be started representing that request so when multiple requests are comming to CGI base web applications multiple operating system process will be started since context switching blw two operating systems process takes time the performence of CGI based web applications is always poor. CGI based web applications are not scalable applications that means when more clients are increased for web applications there will be preformence degradation.

→ the process of transfer control blw two procesn or two Threa is called control jumping or context switching.

→ control jumping time is quite high blw two O.S processes.



resources of web application only one object of that servlet classes will be created and multiple Threads will be started on that object representing multiple requests since control jump b/w Threads of a process takes less time performence of the Thread based serverside programms will be good when compare to process based server side program.

→ The web applications that are developed based on thread based serverside resources are scalable web applications.

Note:- If application gives good performence Irrespectively of increase or decrease in no: of clients is called is scalability.

→ when web server or applications server is started the doman process that runs continuesly is called "chainprocess".

Defination of Servlet:-

① Servlet is a server side technology to develop java based dynamic web resources of web application.

② Servlet is a server side technology which can extends the functionality of Http Servers (web application server).

③ Servlet is a java object that runs based single instance multiple threads principle.

Note:- When multiple requests are given to the web resource called Servlet only one object of Servlet class will be created and multiple threads will be started on that object representing multiple requests on one per request bases this concept make Servlet as single instance multiple threads technology.

→ For related basics on servlet Introduction refer page 7 of booklets.

→ Servlet components are WODA (write once Deploy any where) components that means the servlet component is deployable in all the web servers and application servers.

→ Sun micro system has given servlet, JSP as specifications having rules & guidlines to develop web container s/w's these specification contains API's servlet, JSP programms are web resource programs of java web applications developed by using servlet API, JSP API since web containers have based on servlet, JSP, specification the web container can execute these servlet, jsp programs are server side programs.

→ Servlet OR API means working with classes and interfaces avalible in javax.servlet, javax.servlet.http package.

→ API provides base for the programmer to develop applications or resources based on certain technologies.

→ Servlet API provides base for the programmer to develop java based server side web resource of web application.

→ according to java API is nothing but set of classes and interfaces in the form of packages.

→ JSP API comes as javax.servlet.jsp pkg
javax.servlet.jsp.tagext pkg and etc.

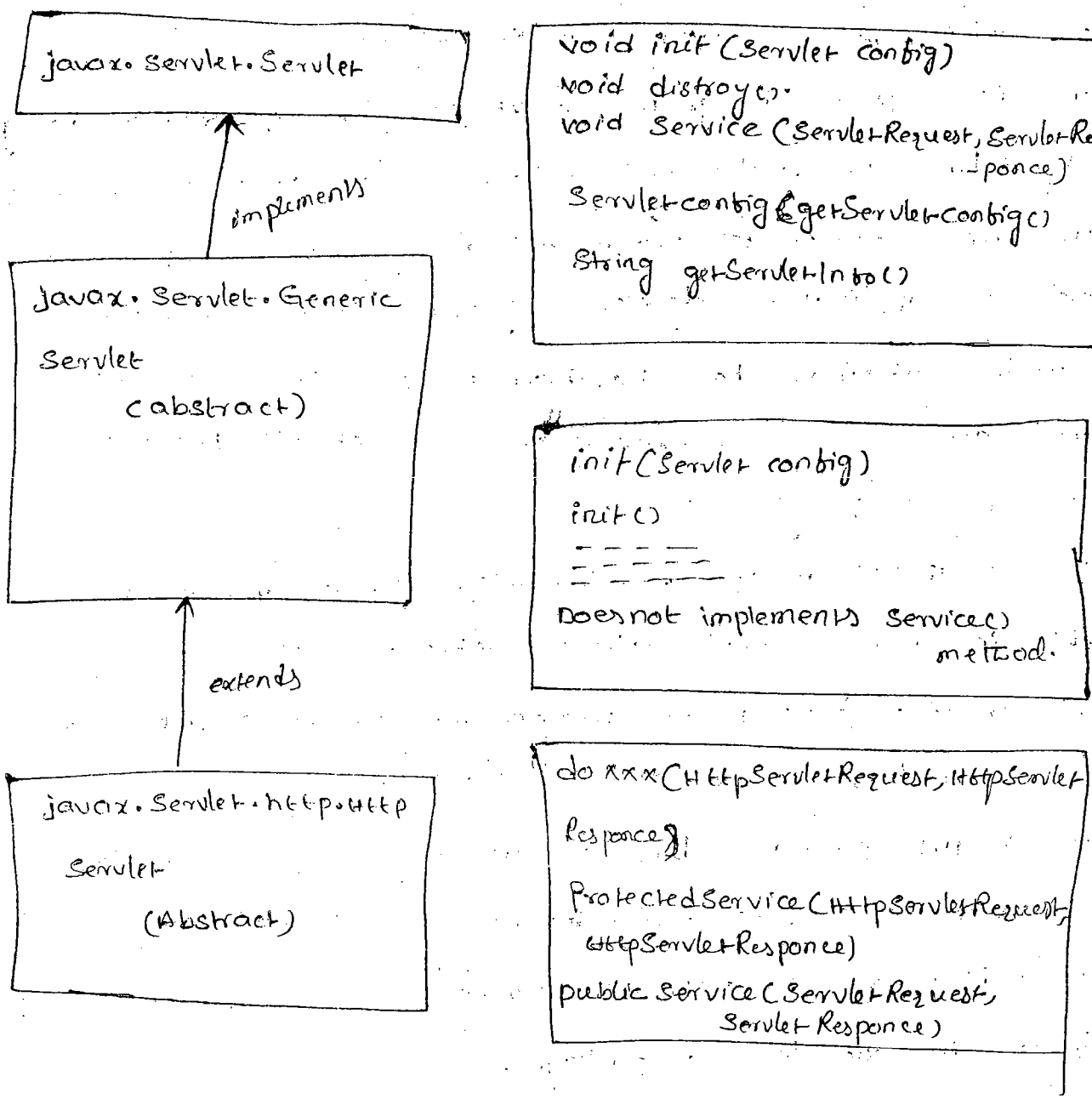
NOTE :- JSP based API is given based on servlet API.

→ Since servlet, JSP API are not part of J2SDK s/w and since servlet jsp technologies are not directly installable s/w's the different servers like web application servers are giving servlet, JSP related jar files.

JSP-API.jar files respectively

In weblogic server Servlet & JSP APIs comes in the form of weblogic.jar files.

20/11



→ javax.servlet.http.HttpServlet class is containing all the methods as concrete methods but the class is given as abstract class.

→ In order to make logic and data of a class accessable only through subclasses take that class as abstract class containing all the methods as abstract methods.

→ Note:- In java abstract classes can contain only abstract methods (or) only concrete methods (or) mix of both.

→ once servlet webserver is given to web server/web container as java class creating object that class managing that object executing methods of that object, Destroying that object and ---- etc operations is the responsibility of web server programmer is not responsible to perform all these operations.

→ There are 3 approches to develop servlet web resources of a web application.

① take a java class implementing javax.servlet.Servlet interface and provide implementation for '5' methods of that interface

② Take a java class extending from javax.servlet.GenericServlet class and provide implementation for service() method of that class.

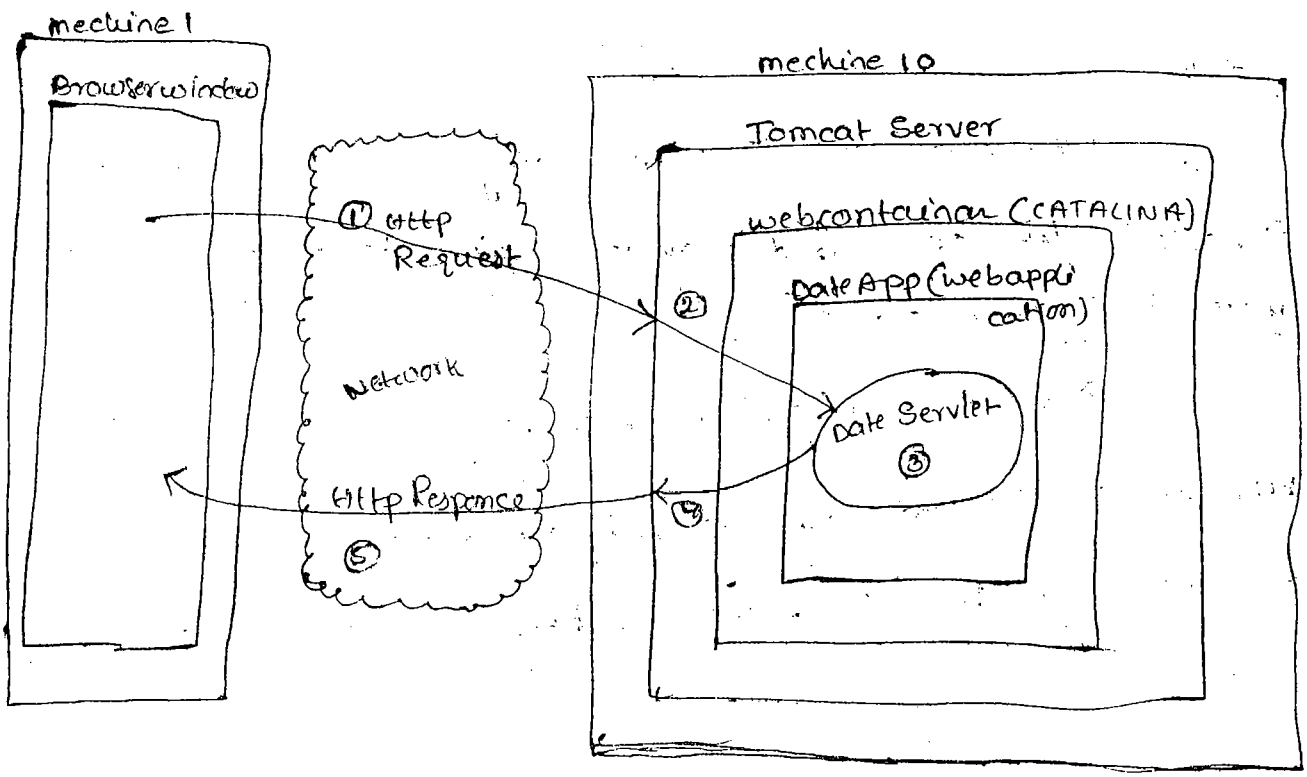
③ Take a java class extending from javax.servlet.http.HttpServlet and override one of the two service() methods (or) one of '7' doxxx() methods of HttpServlet class.

→ How many types of Servlets are there?

A :- Saying that there are 2 types of servlets. GenericServlet, HttpServlet is wrong Answer. There is a potibility of developing 'n' types of Servlets like HttpServlets, smtpServlets, Ftp servlets and etc since all webservers avalible in the market are designed as Http Server programmers prepfer working with Http Servlet.

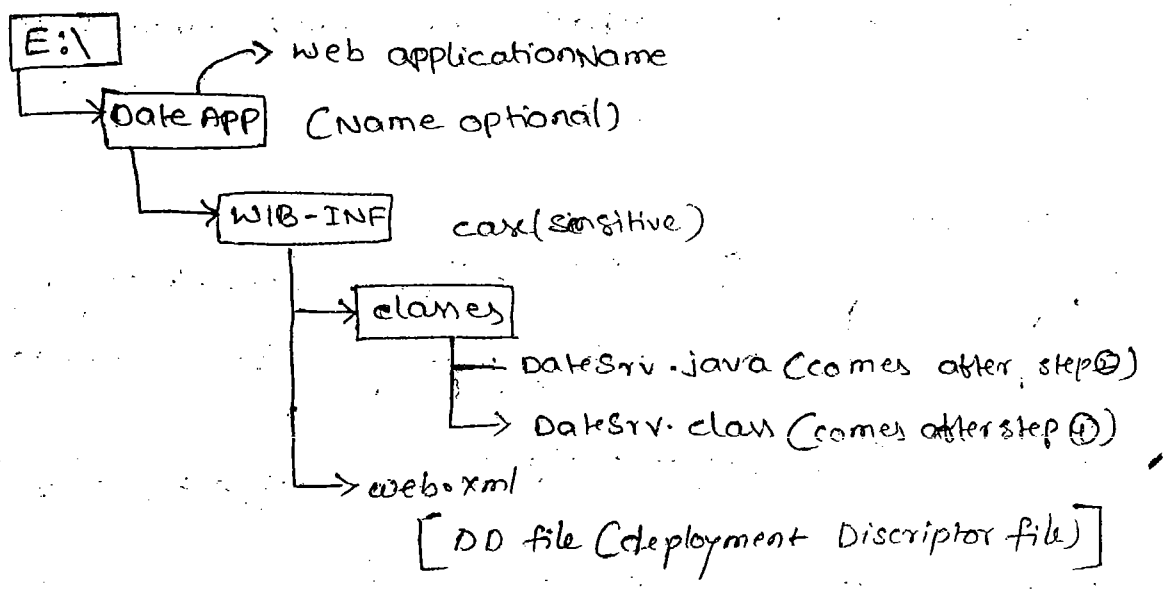
→ javax.servlet.GenericServlet class is not a saparate type of Servlet because it contains common properties of all protocal specific Servlets.

→ A Servlet placed in a webserver will be identified through it's url pattern not by using it's classname this fecility is given to improve Security inorder to give request to a servlet web resource of web application we must use url pattern of Servlet



Note :- When DateSer servlet is Requested it send Date and time of server as responce to browser.

step1 ① create Deployment Directory structure (staging Directory of web application)



Note :- the above deployment directory structure kept in E:\DateApp is common for all the Server the source code file and compiled code file of servlet web resource will be place in

WEB-INF/classes folder.

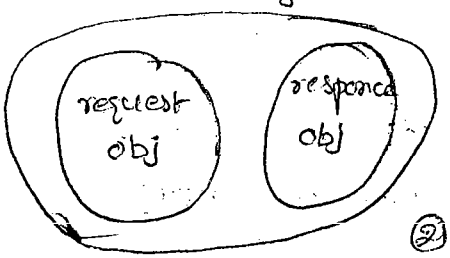
step2 ② Develop Source file of servlet (DateSrv.java) and save that file in WEB-INF/classes folder of web applications deployment directory structure.

Note ① :- since web container has to create object of DateSrv class by loading from WEB-INF\clases Folder of deployed web application the DateSrv class must be taken as "public" class the object of a class is created webServe

class.

② All the Servlet class of web application must be taken as "public" class then only they will be visible for web container.

Note:- Every Servlet object contains two objects when it is required from client.



① request object:- responsible to gather data of request given by a browser window.

② responce object:- this object is responsible to deliver result generated by Servlet as responce to browser through web server for every request comming to Servlet one saparate set of requests, responce objects will be created.

→ To place business logic or request processing logic in Servlet define service method this method gets web server created Request, Responce objects as argument values. Every time when the servlet is requested.

```
public void service (ServletRequest req, ServletResponce res)
{       throws ServletException, IOException
    ---
    ---  // Business logic
    ---
}
```

→ mime (multipurpose Internet mail extention)

```
// DateSer.java
import javax.servlet.*; // for ServletException(c), ServletRequest(I), ServletResponse(I)
import javax.servlet.http.*; // for HttpServlet class
import java.io.*; // for PrintWriter class, IOException(c)
import java.util.*; // for Date(c)

public class DateSer extends HttpServlet
{
  public void service(ServletRequest req, ServletResponse res) throws
                                    ServletException, IOException
  {
    // Set response content type (mime)
      res.setContentType("text/html");
    // get printwriter stream object, pointing to response object.
      PrintWriter pw=res.getWriter();
    // write business logic (request processing logic)
      Date d=new Date();
      String result=d.toString();
    // write result back to browser
    pw.println("<b><center> Date and time is: "+result+"</center></res>");

    // close stream obj
     pw.close();

  } // main service (-,->)
} // class
```

→ In the above code `res.setContentType("text/html")` statement makes the servlet you send response text to browser in the form of html code.

object so by using pw.println() is send responce content to responce object
↳ this responce object sends to webserver
↳ this web server sends the responce to browser.
↳ browser displays responce to on document area for end user.

→ Request obj "req" is not the object of 'ServletRequest' interface it is the object of a class that implements javax.servlet.ServletRequest interface

→ Responce obj "res" is not the object of javax.servlet.ServletResponce interface it is the object of the class that implement ServletResponce interface.

Both these implementation classes will be given by underlying web server. Since these classname will be change based on webserver in use, we never highlet them in our servlet programming to make our servlets executable in all the servers.

Step:3
(3) → Add tomcathome/common/lib/servlet-api.jar file in the class path

Note:- our Servlet source file is using servlet api and Servlet api in tomcat server is comming in the form of Servlet-api.jar file

variable name: CLASS PATH
variable value: tomcat/common/lib/servlet-api.jar; other APIs

Step(4) → compile source file code of the servlet (DateSrv.java)

Step(5) E:/DateApp/WEB-INF/classes/DateSrv.java

→ Configure DateSrv servlet in web.xml file having url pattern.

```
<servlet>
    <servlet-name>ABC</servlet-name>
    <servlet-class>DateSrv</servlet-class>
</servlet>

<servlet-mapping>
    <servlet-name>ABC</servlet-name>
    <url-pattern>/first</url-pattern>
</servlet-mapping>

</web-app>
```

→ ABC is the logical name of servlet

→ DateSrv is the fully qualified java class i.e acting as servlet.

→ /first url pattern of the servlet.

NOTE :- the servlet logical name given in servlet tag and servlet mapping tag must match

→ web.xml file of web application is called deployment discriptor file or configaration file. when web application is deployed in the web server the webserver reads web.xml as first file and gets informa about servlet web resources configured in web.xml

→ web.xml contains configaration of servlet web resources having url patterns

→ All the servlet web resources of web application must be configu in web.xml file.

→ It is always recomended to develop web.xml file of your application based on example web.xml file.

→ Deploy the web application in tomcat server. copy E:\DateApp folder to tomcat-home/webapps folder.

→ start tomcat server start
↳ programs
↳ apache tomcat 5.5
↳ configar tomcat
↳ start
↳ ok

DateSrv servlet

open browser window

→ type http://localhost:4040/Dateapp/first

① ② ③ ④ ⑤

output

Date and Time nov 21 10:14:55 IST 2009

① Protocal to communicate with webserver from browser

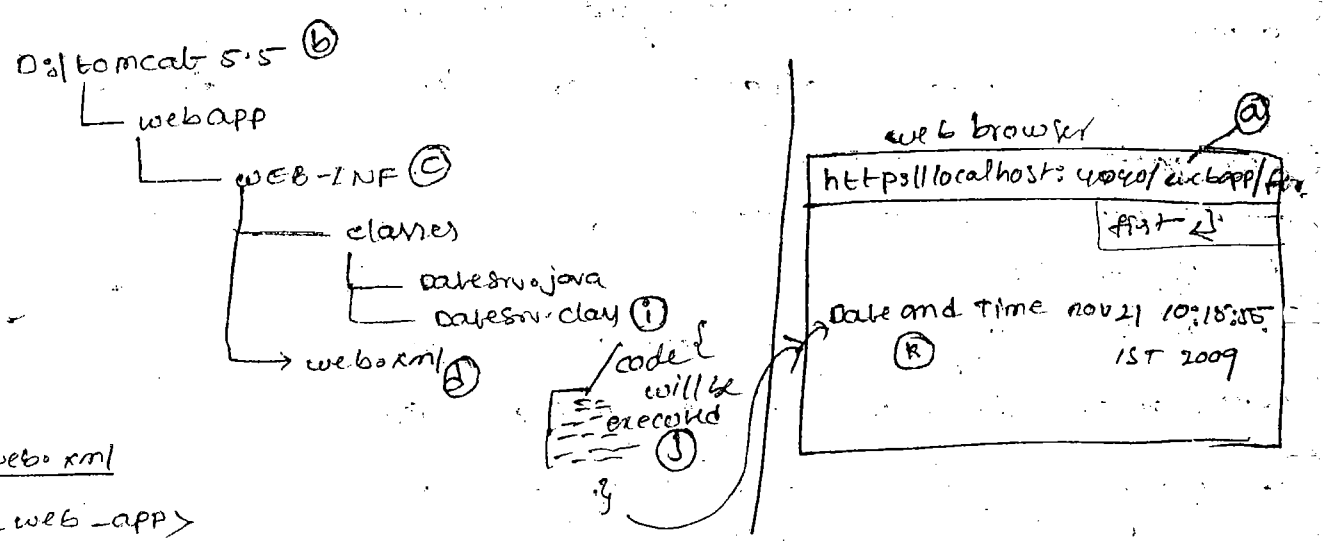
② Host name of a computer where webserver is installed.

③ port number of server. (web server)

④ Dateapp name of the web application or context path or context root of the web application

⑤ first url pattern of the target Servlet DateSrv.

Ø → Can you explain the flow of execution that takes place from request to responce generation of using a servlet-?



web.xml

```
<web-app>
    <servlet>                                   ⑨
        <servlet-name> ABC </servlet-name>
        <servlet-class> DateSrv </servlet-class>   ⓗ
    </servlet>
    <servlet-mapping>                           ⓕ
        <servlet-name> ABC </servlet-name>
        <url-pattern> /first </url-pattern>
    </servlet-mapping>                          ⓔ
</web-app>
```

Respect to the above case:-

① End user generates request to servlet of srv DateSrv by typing url from the browsers address bar

②③④⑤ based on the <hostname:portnumber> details placed in url the request will locate webserver(tomcat).

⑥ based on the webapplication name (Dateapp) placed in url the request will locate Dateapp webapplication placed in webapps folder of tomcat server.

⑦ :- Server receives the request and uses web.xml file Dateapp web application to locate the resource (DateSrv servlet)

⑧⑨⑩⑪ :- Based on the url pattern typed in the request url (/first) the webserver will get logical name (ABC) of DateSrv Servlet. by using this logical name the class name of DateSrv Servlet will be retrived.

⑫ web server/webcontainer loads servlet class (DateSrv) from the memory and creates servlet class object (if it is already object available that will be use).

⑬ webserver/webcontainer calls service() method of servlet and execute request processing logic that is there to generate result.

⑭ Servlet generated result goes to browser as responce Through webserver.

→ Any modification done in source code of the servlet will be reflected after the recompilation of servlet and reload of the web application.

→ To reload the web application

Open tomcat home page → tomcat manager → submit username and password → choose the webapplication (Dateapp) → Reload.

write following two lines of code in the service method of servlet.

pw.println("
 classname of req object is" + req.getClass().getName());

pw.println("
 classname of res object is" + res.getClass().getName());

In tomcat server implementation classes of various interfaces belonging to servlet api are available

tomcat-home/server/lib folder

Don't highlet their class name in servlet programming because they will change based on the server we use.

→ To prove that the servlet is a single instance, multiple threads component try to print servlet object hashcode and name of the thread for every request that comes to servlet in this situation all the request we will get same hash code having different thread names.

To prove that write following code in the service() method of servlet.

pw.println("hashcode of current servlet object is " + this.hashCode());
pw.println("current request related threadname is " + Thread.currentThread().getName());

NOTE: java object is identified with its hashcode value where as thread is identified with its thread name

→ messages placed in pw.println will come on to the browser window as responce. messages placed in s.o.p statement of servlets will come on the server console as debugging statements. (conformation statements)

```
    {
      Thread.currentThread().sleep(10000);
    }
    catch(Exception e)
    {
      e.printStackTrace();
    }
pw.close();
```

≡

```
ServletOutputStream sos = res.getOutputStream();
sos.println("<b>hello from servlet</b>");
```

Q*

→ what is the difference b/w printWriter Stream obj and ServletOutput Stream object?

- Both the Stream obj will point to responce object and can send results content of the servlet as responce to the browser.

- printWriter is a character Stream object so it is very usefull to send character data as responce to browser.

```
printWriter pw = res.getWriter();
pw.println("<h1> hello </h1>");
```

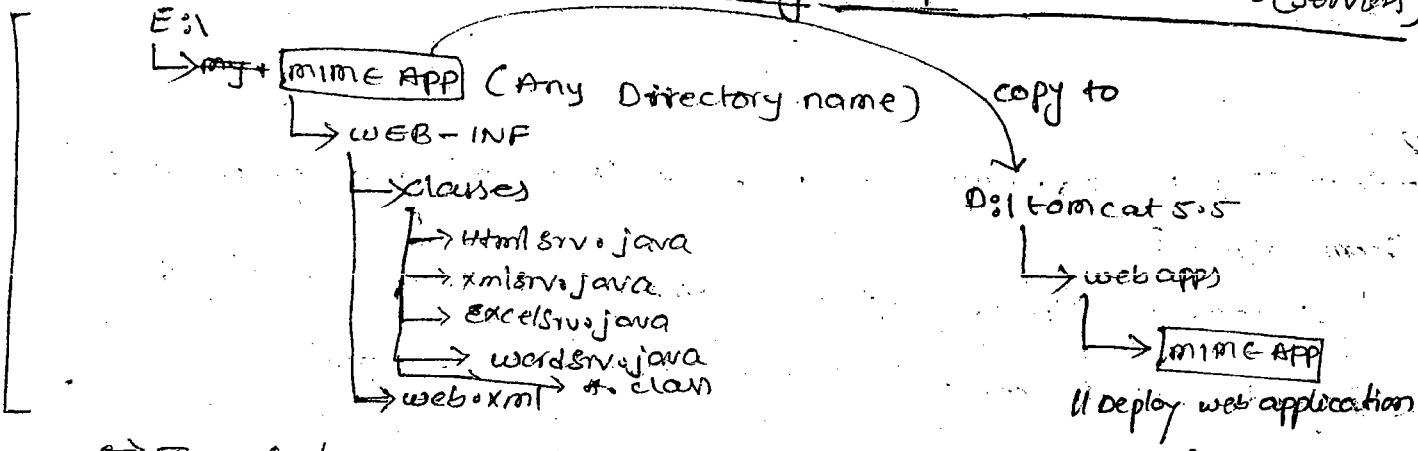
→ ServletOutputStream is a byte Stream so it is very usefull to send binary data like images audio, vedio files as respon to the browser. this Stream can also be used to send text responce.

```
ServletOutputStream sos = res.getOutputStream();
sos.println("<b>hello</b>");
```

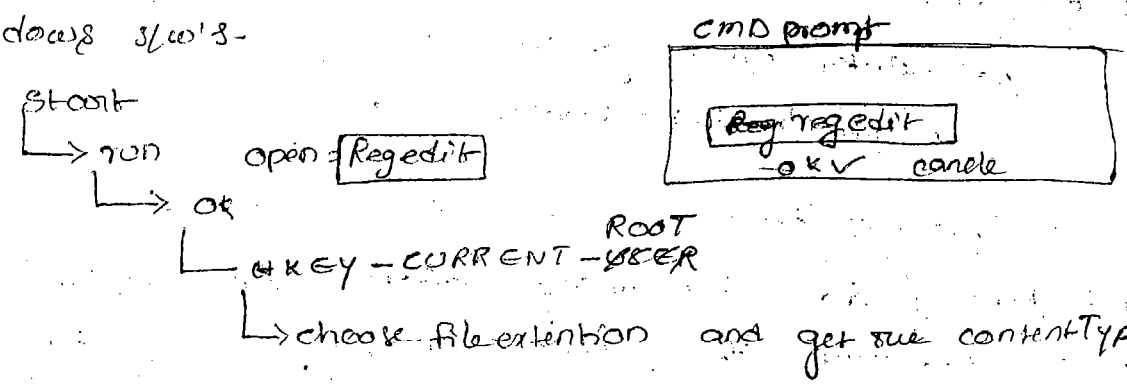
→ By changing response contentType of Servlet webresource (MIMETYPE) we can make web resource generate a response to browser in different formats.

→ When multiple Servlet web resources are there in web application all these servlets must be configured in web.xml file having unique url patterns.

= Develop a web application having multiple web resources (Servlets)



→ In windows O.S we can take the support from REG EDIT tool to know the MIME TYPES / contentTypes of various windows s/w's-



Ex
- .doc = Application/msword.
- .gif = image/gif
- .html = text/html
- .bmp = image/bmp
- .PPT = application/vnd.ms-powerpoint
- .XLS = application/vnd.ms-excel
- .xml = text/xml

Note: printWriter is the most popularly use stream object in Servlet programming to delever responce to the browser

Source code of above App:-

// Htmlsrv.java

```
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
import java.io.*;
public class Htmlsrv extends HttpServlet
{
    public void service(HttpServletRequest req, HttpServletResponse
    {       res) throws ServletException, IOException

        PrintWriter pw=res.getWriter();
        res.setContentType("text/html");
        pw.println("<table border="1" bgcolor="red">");(2)  <table border='1' color='red'>
        pw.println("<tr><th> party </th> <th> seats </th> <th> Remarks</th
        pw.println("<tr><td> T.D.P </td> <td>35/150 </td> <td> poor performence </td> </tr>");
        pw.println("<tr><td> CONGRESS() </td> <td> 40/150 </td>
                    <td> poor performence </td> </tr>");
        pw.println("<tr> <td> LOKSATTA </td> <td> 65 </td> <td> good performence </td></tr>
        pw.println("<tr> <td> PRP </td> <td> 20 </td><td>
        pw.println("</table>"); improved performence </td> </tr>");
    }
}
```

// Excelsrv.java

```
public class Excelsrv.java
{
    public void service(-  -  -)
    {   PrintWriter pw=res.getWriter();
        res.setContentType("application/vnd.ms-excel");
```

```
import statements;

public class xmlsrv extends HttpServlet
{
    p v service(HttpServletRequest req, HttpServletResponce res),
                throws ServletException, IOException
    {
        PrintWriter pw = res.getWriter();
        res.setContentType("text/xml");
        (Same as above)
    }
}
```

```
// wordsrv.java
    import statements;
public class wordsrv extends HttpServlet
{
    p v service(HttpServletRequest req, HttpServletResponce res)
        throws ServletException, IOException
    {
        PrintWriter pw = res.getWriter()
        res.setContentType("application/msword");
            (Same as above)
    }
}
```

E:\mimeApp\WEB-INF\classes> javac *.java

E:\mimeApp\WEB-INF\classes>

```
<web-app>
    <s>
        <s-n> ABC </s-n>
        <s-c> htmlsrv </s-c>
    </s>
    <s-m>
        <s-n> ABC </s-n>
        <url-p> /Url </url-p>
    </s-m>
```

```
    <s>
        <s-n> ebg </s-n>
        <s-c> excelsrv </s-c>
    </s>
    <s-m>
        <s-n> ebg </s-n>
        <url-p> /Eurl </url-p>
    </s-m>
```

```
<servlet>
<s-n> = <servlet-name>
<s-c> = <servlet-class>
<s-m> = <servlet-mapping>
<url-p> = <url-pattern>
```

```
<s-n> abhi </s-n>
<s-c> xmlsrv </s-c>
</s>
<s-m>
  <s-n> ghu </s-n>
  <url-p> /xurl </url-p>
</s-m>
```

```
<s-n> xyz </s-n>
<s-c> wordsrv </s-c>
</s>
<s-m>
  <s-n> xyz </s-n>
  <url-p> /wurl </url-p>
</s-m>
</web-app>
```

→ deploy the mimeApp

→ contentType of any s/w is called mimeType.

changes done in web.xml file will be reflected without reloading of the web application.

→ can i give multiple url patterns for a servlet.?

(A) yes it is possible.

```
<web-app>
  <s>
    <s-n> ABC </s-n>
    <s-c> Html5srv </s-c>
  </s>
  <s-m>
    <s-n> ABC </s-n>
    <url-p> /Hurl </url-p>
  </s-m>
  <s-m>
    <s-n> ABC </s-n>
    <url-p> /Htmurl </url-p>
  </s-m>
  <s-m>
    <s-n> ABC </s-n>
    <url-p> /Hturl </url-p>
  </s-m>
</web-app>
```

→ the above Servlet Html5Str. can be requested by using any one of the above 3 url patterns they are /Hurl / Htmurl / Hturl

→ what happen in multiple Servlet Configured in web.xml are having same url patt?

(A) A last servlet i.e configured in the list of multiple servlets that are having same url pattern

note: It is always recommended to configure multiple Servlets are web-application by having unique url patterns.

→ Servlet web resource will be compiled from command prompt by using javac compiler where as Servlet web resource will be executed by web container of webserver.

25/11

→ protocal is a set of rules followed by two parties which wants to participate in communication.

There are two types of protocals

① Network level protocal.

② Application level protocal.

→ N/w level protocal contains set of rules to get the communication b/w two physical computers

Ex TCP/IP.

→ Application level protocal contains set of roles to get the communication b/w two s/w applications (or) two s/w's

Ex Http

Q→ wt is the diff b/w normal text & Hypher text?

A→ Normal text allows only sequential reading and don't contain hyperLink Ex; notepad text, word. document text.

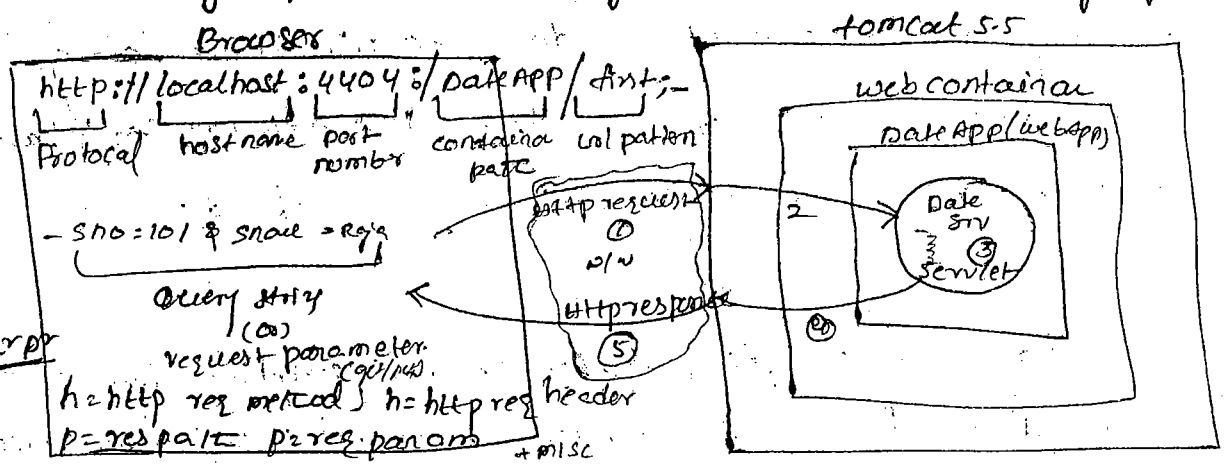
- the text that allows non sequencial reading. random reading is called hypher text. hypertext contain hyper link.

Ex text of webpages.

→ Http @ is a application level protocal i-e given to transfer hypertext b/w browser & web server. s/w.

Html is a markup language that is given develop hypertext based web pages.

The language that uses tags is called markup language.



→ when browser generate request to the web resource of web application by typing url in the address bar of the browser lot of details will go to web resource & web server along with the http request. we can remember these details as hrpr details

→ the http request is composed with multiple details given by end user given by browser s/w.

→ A typical structure of http request contains following:

ⓐ http req method, path of the web resource

ⓑ http req header names/values

ⓒ Req data in the form of req param

ⓓ misslaneous information

Ex: of http req structure

path of the web resource.

Protocal & it's version

http request method → POST /DateApp/furl : http/1.1

http request headers:
- Host: www.Nataraj Soft.com
- User agent: mozilla/IE 5.0
- Accept: text/html, text/xml, image/jpeg, application/msword, */*;
- Accept language: English-US
- Accept Encoding: gzip/distate
- Accept Charset: ISO-6491
- keepAlive: 300
- connection: keep alive

<< blank line >> } - blank line (the req is there)

can send only limited amount of data to the server along with the request whereas the post req methods send bulk http request and send unlimited amount of data to the server.

Q → wt is diff b/w HttpRequestParameters & HttpRequestHeaders?

(A) → HttpRequestParameter contain input values of web resource given by enduser from browser window. requestparameters names & values are user debine they can be duplicated.

• HttpRequestHeaders contain details of browser as values like browser s/w name language supported by browser and etc. HttpRequestHeader names are fixed but values will be change based on the browser s/w we use. HttpHeader names are unique, values in HttpHeaders will be generated by browser.

In the above diagram sno, sname are RequestParameters

Note:- we can see HttpRequest without RequestParameters but every request contains requestheaders.

- List of HttpRequestHeader names:-

• Accept	• if modify-since	• keep-Alive.
• Accept-encoding	• Referer	
• connection	• User-agent	
• cookie	• Accept-charset	
• HOST	• Accept-Language	

→ For disription about various HttpHeader Refer page nos 12 and 13 in booklet

→ How much time the connection b/w browser & webserver should exist when webserver is not generating responce or error for the given request will be there in HttpRequestHeader called connection.

→ Inorder to gather various details of HttpRequest being from Requested web resource called servlet we need to work with HttpServletRequest object.

By using HttpServletRequest object we gather following details from HttpRequest:

① HttpRequestParameter names & values

② HttpRequestHeader names & values.

③ Miscellaneous information about HttpRequest.

Different ways of Reading HttpRequestParameter values being from Servlet

Ex: Url :-

http://localhost:4040/DateApp/first?sno=101&sname=raja&coure=java

(requestparameter)

Approch 1

```
String s1 = req.getParameter("sno"); // 101
String s2 = req.getParameter("sname"); // raja.
String s3 = req.getParameter("coure"); // java
```

Note - In approch no:1 we must know the name of the RequestParameter inorder to get its value.

- If multiple values are get for request parameter only first value will come

Approch 2

```
Enumeration e = req.getParameterNames(); // Get all the names of Req param.
while(e.hasMoreElements())
{
    String name = (String) e.nextElement(); // given name of Req param
    String val = req.getParameter(name); // val of the Req param
    S.O.P("name+" "+val);
}
```

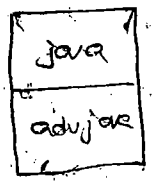
sname raj
course java

Approach gives all the names & values of RequestParameter if RequestParameter has multipleParameter only first value will be given.

Approach3 :-

26/11

```
String s[] = req.getParameterValues("course");
for(int i=0; i<s; i++)
    pw.println(s[i] + " ");  // java adv java
```




```
String s1 = req.getParameterValues("course")[0];  // java (or) [1] = adv java
```

Different approaches of gathering HttpRequest Header values being from the Servlet :-

Approach:1 :-

```
String s = req.getHeader("user-agent");  // gives browser s/w name
String s1 = req.getHeader("accept-language");  // gives languages that are supported by browser s/w (like english, telugu, french etc)
String s2 = req.getHeader("accept");  // give mime type that are supported by browser s/w
String s3 = req.getHeader("accep-encoding");  // gives encoding techniques that are supported by browser s/w.
```

Note: In this approach no:1 we must know the HttpRequest Header name in order to get the value.

Approach:2 :-

```
Enumeration e = req.getHeaderNames();
while(e.hasMoreElements())
{
    String name = (String) e.nextElement();
    String val = req.getHeader(name);
    pw.println(name + " " + val);
}
```

3rd approach we will get all the names and values of HttpRequest header that are set by the browser in HttpRequest.

→ since HttpRequest names are unique names i.e no (2nd), 3rd approach together RequestHeader values.

→ we can change Accept-language RequestHeader value of HttpRequest by using browser settings. in InternetExplorar.

tool menu
↳ Internet options
↳ languages
↳ add
↳ choose language

In netscape

In netscape
Edit menu
↳ preferences
↳ Languages
↳ Add
↳ choose language

→ To gather various information about HttpRequest being from servlet

// By using methods of javax.servlet.ServletRequest(i) ~

```
pw.println("<br> server name " + req.getServerName()); //localhost
pw.println("<br> protocal is " + req.getProtocal()); // HTTP/1.1
"  ("<br> scheme " + req.getScheme()); // http
"  ("<br> port no " + req.getServerPort()); //4040
"  ("<br> character encoding " + req.getCharacterEncoding()); //UTF
"  ("<br> content length " + req.getContentLength()); //-1
"  ("<br> client machine IP address " + req.getRemoteAddr()); //127.0.0.1
"  ("<br> client machine hostname " + req.getRemoteHost()); //127.0.0.1
```

⑥ pw.println("
 req method " + req.getMethod()); // GET/POST

" "
 path info " + req.getPathInfo(); // extra info sent by browser during file upload

" "
 query string " + req.getQueryString(); // sname=vasa & sno=101

" "
 req URI " + req.getRequestURI(); // /DateApp/frm2

" "
 req URL " + req.getRequestURL(); // http://localhost:4040/DateApp/frm2

URI = Uniform Resource Indicator.

→ By overiding base class methods in derived class the derived class methods should have same name and signature but can have same(or) excess access specifier

The javax.servlet.http.HttpServlet class is having two service methods

1st service method

① public void service(ServletRequest req, ServletResponse res) throws ServletException, IOException.

2nd service method

② protected void service(HttpServletRequest req, HttpServletResponse res) throws ServletException, IOException.

→ By overriding 2nd service method of predefined HttpServlet in its subclass we can take either public(or) protected as access specifier of subclass method.

27/11

→ When web resources of web application sends HttpResponse to the browser through web server or application server the Response contains multiple details including the response content as Response body. we can remember these details as SC# Response details S = Http Status code (100 to 599)

(100 to 199) indicates info

(300 to 399) ⟶ Redirection.
(400 to 499) ⟶ Incomplete web resource.
(500 to 599) ⟶ Server number.

C indicates Responce content (or) Responcebody

H indicates HttpResponce Headers.

→ SCA: Responce status code indicates the status of request-processing that is happen in the web resource.

→ when webserver sends response to browser successfully then the HttpResponce contains status code in b/w 200 to 299

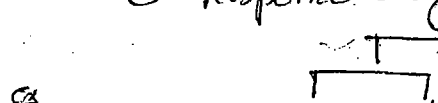
- If webserver is Redirect in the request from the original requested web resource to another web resource then the browser gets 300 to 399 value as responce status code.
- If the requested web resource is not configured properly are developed incompletly where the browser get error status 400 to 499.
- If underlaying webserver or application server is having problem then browser gets error status code in b/w 500 to 599.
- Responce body is nothing but responce content generated by web resource like code placed in pw.println statements in Servlet component.
- ResponceHeaders Header values provide instruction to browser in the process of displaying responce content on the browser window as web pages.

from browser, one connection will be created in blw browser and server. once Request related responce goes to browser from webresource through web server then the connection will be closed. The typical structure of Http Responce contains

① miscellaneous information including responce status code

② Http Responce Headers.

③ Responce body



Request Header
- set-cookie: JSESSION ID: 148937ABCA179BA
- content-type: text/html
- " -length: 1354
- Date: Fri feb 2008 8:40:34 Gmt
- Server Apache tomcat/5.0
- connection-close();

<blanck space>

Responce content
- <Html> <body> <h1> welcome to servlet </h1>
- we are attending basic of servlets
- </body> </Html>

→ List of Http Responce Headers :- are location

location, Refresh, set-cookie, cache-control (or) pragma, content-encoding, content-length, content-type, last-modified, Server, Date, connection and etc.

→ For related information on Http Responce, status code and Responce Headers Refer page no: (14) to (16) in our booklet.

Automatically refreshed

```
res.setHeader("refresh","10");
```

→ How to make webpage generated by servlet Refreshing the content at regural intervels.

Ⓐ:- use HttpResponce Header called refresh by writing following code in the web resource of web application (servlet)

```
res.setHeader("refresh","10");
```

place above code in service method of servlet so the responce webpage genarated by servlet will be refresh automatically after every '10' sec.

Note: this concept is used very popularly while designing the web resources which displays live cricket scores and stock market share values.

Q → How to prevent browser for not preserving responce of web resource in the buffer.?

Ⓐ:- By default browser stores output of every resource in the buffer and uses that buffer content across the multiple same request given to that web resource. this reduces network probit bcoz of and n/w round trips browser & Server. But this buffering is not good if web resource is genarating new responce for every request. so in order to prevent this buffering happning in the browser keep send additional instruction to browser by using ~~cache~~ cache-control or pragma responceheader to set their Headers values write following code in the service method of servlet.

```
res.setHeader("cache-control","no-cache"); // for
```

(or) HTTP/1.1 ~~page~~ base web servers.

```
res.setHeader("pragma","no-cache"); // for HTTP/1.0
```

based server.

→ Buffer is a temporary memory. Buffer is also called as cache, buffering / caching is the process of storing result in a temporary memory and using that result across the multiple request.

Q → How to know the browser s/w name being from servlet using which the servlet is requested.

```
String brname=req.getHeader("user-agent");
pw.println("browser name"+brname);
```

use HttpRequest Header as shown bellow in the service method of servlet.

pw.println("browser name" + brname);

→ by knowing browser s/w name the web resource can format the responce according to the browser settings. like eleminating unsupported tags and adjesting the look & style tags.

28/11

→ Html files of web application are capable of generating static web pages where as Servlet, JSP web resources of web application generate Dynamic web pages. these many days we have pass input values to servlet web resource from browser by appending query stream in the URL of browsers address bar.

http://localhost:9090/DateApp/first?sno=101&sname=raja

↓

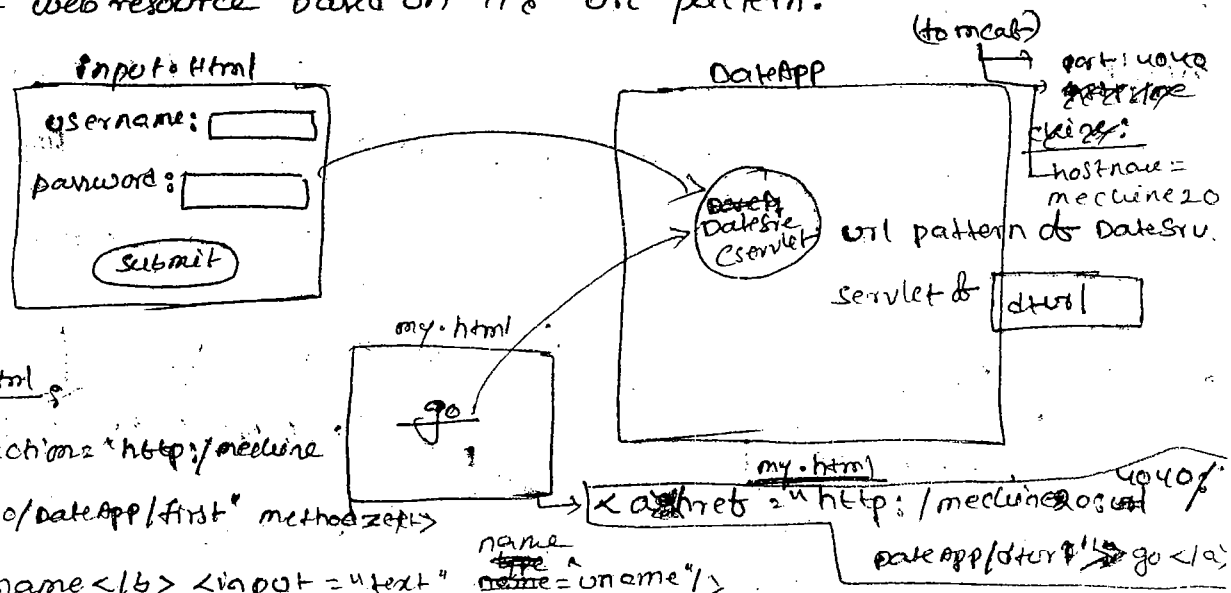
Query string request parameter names & values.

→ passing input values to web resource by typing query string in the address bar of the browser is not user-friendly operation. the non technical end user can't do this work properly. to overcom this problem design GUI by using Html forms and Hyperlinks for end users to generate the request. "the visitor of a website or web application is called end user."

→ the Html form based request given to web resource of web application can send the request having input values (taken from data as request parameter values)

→ the hyperlink generated request to a web resource can't send input values along with the request.

→ the other web resources of web application identities are Servlet web resource based on it's url pattern.



input.Html

```
<form action="http://mecline20:4040/dateapp/first" method="get">
<b> uname </b> <input type="text" name="uname"/>
<b> password </b> <input type="text" name="password"/>
<b> </b> <input type="submit" value="send"/>
</form>
```

my.html

```
<a href="http://mecline20:4040/dateapp/dtsrv">go</a>
```

→ Html, servlet communication means generating request-Servlet from form (or) Hyperlink of Html file.

→ Html form is very useful to collect input values from end user and to generate request web resource by having those input values.

to identify the servlet & to pass the request to the servlet.

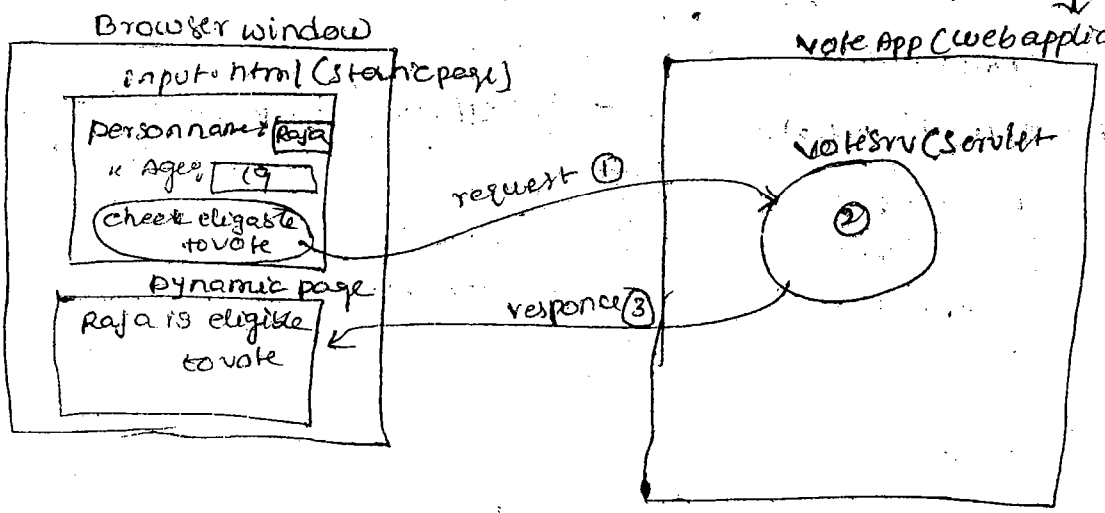
my.html

```
<a href="http://machine20:4040/dateApp/durl"> go</a>
```

→ When html form page generates request to web resource the form component names (text box names) like uname, pwd), becomes request parameter names and the form component values text box values like raja, hyd becomes request parameter values.

→ Since doXxx() methods are given based on http protocol standards it is recomanded to use them to place business logic in our HttpServlet. When form page generates request having HttpRequest() method get it is recomanded to procen that request by overriding doGet() method in our Servlet. By & use doPost() method to procen the request whose HttpRequest() method is POST.

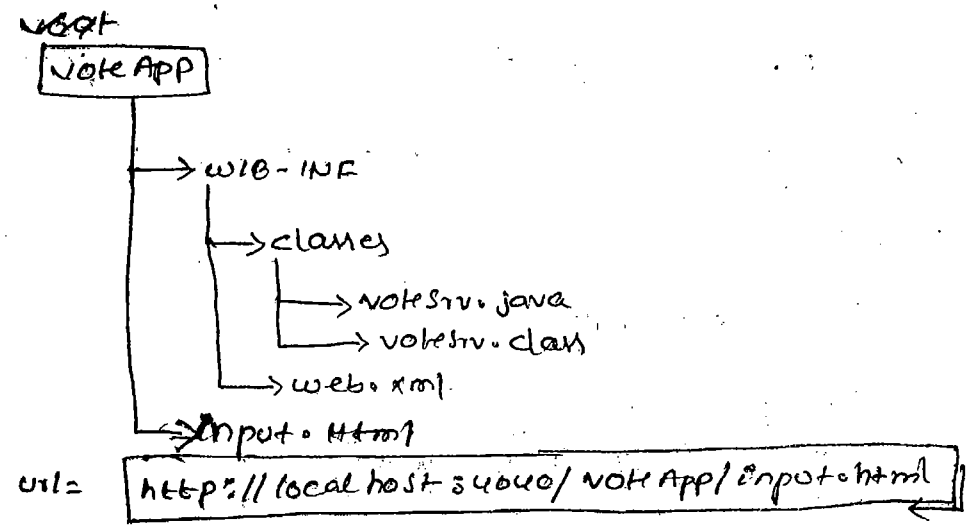
→ prog Application Develop hello voteApp. program



→ In a web application the html files and web resources must be kept outside the WEB-INF folder generally under context path directly. Directory

→ html file based web resource is identified with it's filenam

→ Html file based web resource is need not to be configured in web.xml file



input.html

```
<b> input. Html </b> <br>
<form action="http://localhost:4040/voteApp/vturl" method="get">
<b> person name </b> <input type="text" name="pname" /> <br>
<b> person age </b> <input type="text" name="page" /> <br>
                    <input type="submit" value="check vote eligibility">
</form>
```

Note :- To read from form data from form components of form page servlet must use Request.getparameter method having form component name as Argument value.

voteSrv.java

```
//voteSrv.java
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class voteSrv extends HttpServlet
{
    public void doGet(HttpServletRequest req, HttpServletResponse res)throws
```

```
// read from form data from form page ("input.html")
    String s1=req.getParameter("pname");
    String s2=req.getParameter("page");
// here pname, page are the form form component names
    int age=Integer.parseInt(s2.trim());
// write b.logic
    String msg=null;
    if(age>=18)
      msg="Eligible to vote";
    else
      msg="wait"+(18-age)+" years to vote";
// write response
    PrintWriter pw=res.getWriter();
    res.setContentType("text/html");
pw.println("<font color=red><center>"+msg+"</font></center>");
pw.println("<br><a href="input.html">back</a>");
  } // doGet
} // class
```

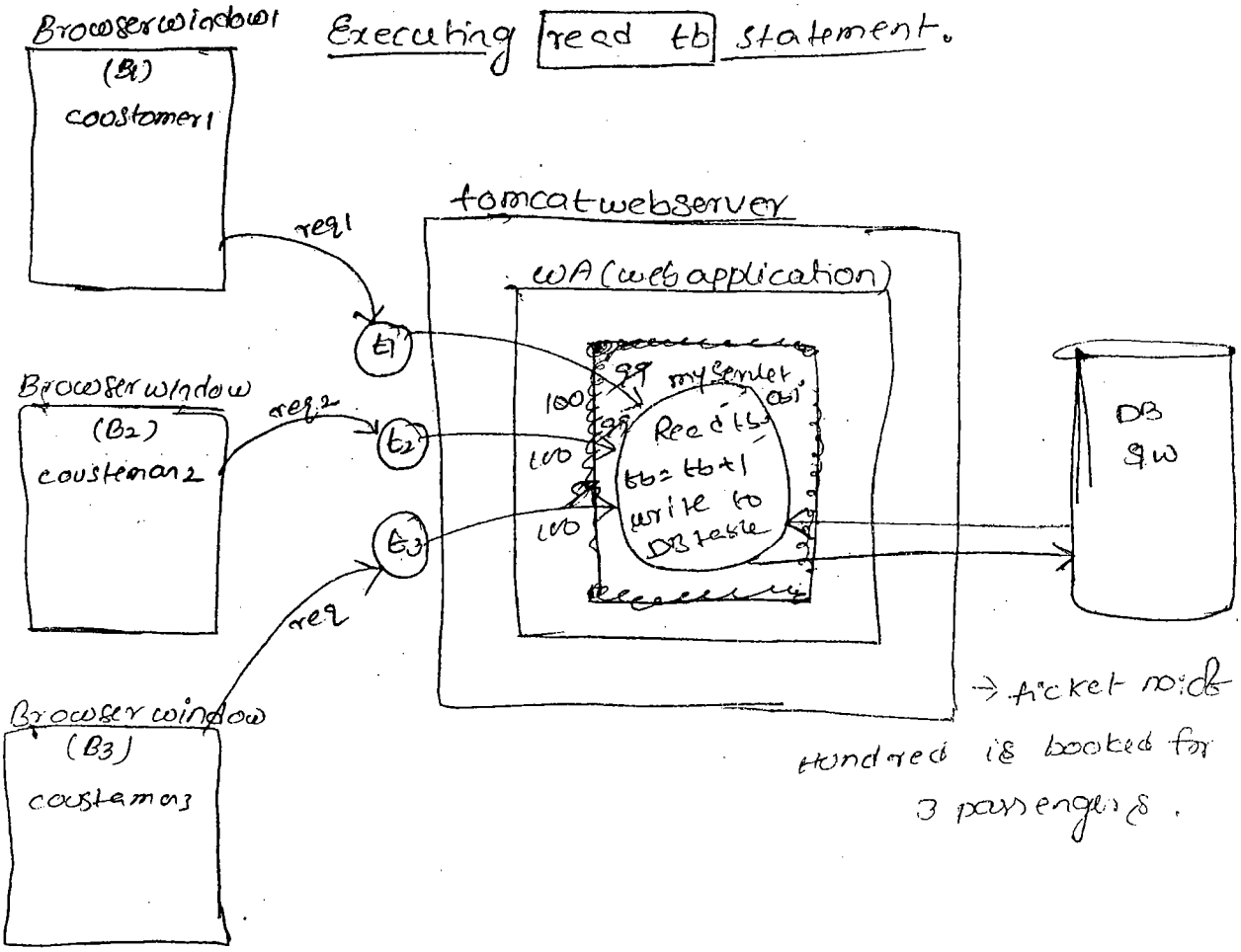
web.xml

```
<web-app>
    <servlet>
    <s-n>vt</s-n>                                    (**)
    <s-c>vote.srv</s-c>
    </servlet>
    <s-m>
      <s-n>vt</s-n>
      <url-pattern>/vturl</url-pattern>
    </s-m>
</web-app>
```

where multithreads are acting on single java object simultaniously / concurrently / Interleaving then there is a possibility of getting Data corruption in the object this makes object as non thread safe object

** Servlet is a single instance multiple thread component That means when multiple ~~requests~~ request are given to Servlets simultaniously / concurrently / parallely multiple threads will be started on a Servlet obj representing multiple request due to this Servlet is not thread safe by default.

Assume that Request ①, ② are suspended after Executing `read tb` statement.



→ In order to make java object as thread safe object allow only one Thread at a time on the object to manipulate object

we can work with following 4 technique.

① By using local variables of Service method.

② By using synchronized service method.

③ By using Synchronized blocks in the Service method.

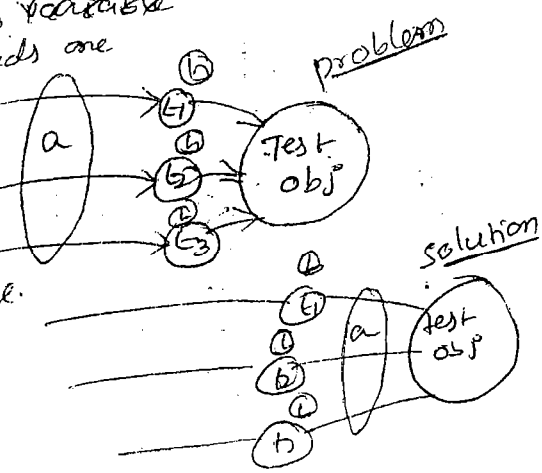
④ By implementing javax.servlet.SingleThreadModel Interface on Servlet class.

* → Local variables of a java method placed in a class (or) thread safe variables because every thread started on the object representing method will get one copy of local variable so local variable declared in Service method of Servlet (or) thread safe variables.

→ Every request comming to servlet will create on thread on Servlet class object pointing to Service method

Problem

Three t1, t2, t3 threads are same variable b. in 3 Threads one Thread will change automatically a will change.

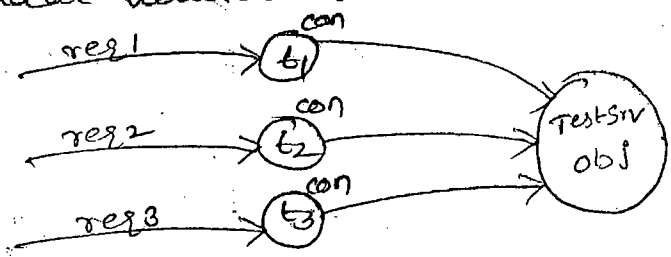



```
class Test
{
  int a — instace variable
  public void m()
  {
    int b — local variable
    ---
    ---
    ---
  }
}
```

→ instance variables of a java class are not Thread safe variables because all the threads started on that object will use single copy of instance variables to manipulate the data. So, instance variables declared in the class of Servlet component or not ~~class of Servlet~~

local variables of service() method :-

→ local variable of a service method is thread safe variable by default because every thread i.e started on the servlet obj. representing req will get it's own copy of local variable of service method.

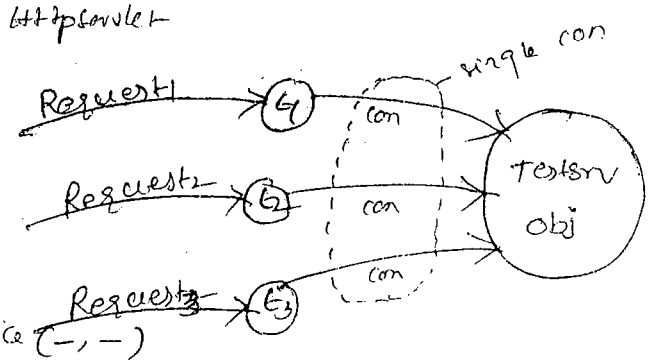



```
public class TestSrv extends
                              HttpServlet
{
  public void service(--,--)
  {
    connection con;
    class.forName(---);
  con = DriverManager.getConnection
        ---                  (---);
  }
}
```

→ taking service method by utilizing only it's local variable and no instance variables makes Servlet as thread safe servlet. but it is always not posible to work with only local variables of service method.

② By using Synchronized Service method :-

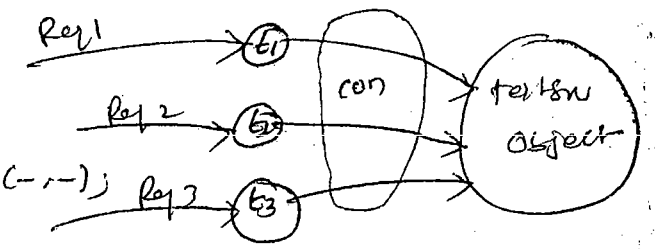
```
public class TestSrv extends HttpServlet
{
    connection con;
    public void init() {
    class.forName(---);
    con = DriverManager(---);
  }
public Synchronized void service(--,--)
 {
    use connection object here
    ---
    ---
    ---
 }
```



class if you make the service method as synchronized method only one thread can manipulate data of the instance variables at a time by executing logic of service method that means we are preventing concurrent data manipulation on instance variable data in service method by making service method as synchronized method.

③ By using synchronized blocks on special obj in Service()

```
public class TestSrv extends HttpServlet
{
   connection con;
   public void init() {
   class.forName(--,--);
  con=DriverManager.getConnection(--,--);
   }
 public void service()
   {  ---  }//Other statements.
      Synchronized(con) {
        ---  }// statement using con obj
   }  ---  }// otherStatements.
```



→ Instead of making hole service method as synchronized method it is recomanded to take those objects on which synchronization is required and manipulate them through synchronized blocks placed in service method. this technique is most popular meth and technique to achive thread safty in servlets

→ In the above code the statement placed in synchronized block allow only one thread at a time to manipulate the data of connection object even though the connection object is instance variable this makes servlet as thread safe servlet.

→ since ServletContext object, HttpSession are golobal objects of web resources belonging to a webapplication it is always recomended to manipulate data and attribute values of these two objects by keeping these objects in synchronized blocks.

```
Service(—,—)
{
  ==== //other statements
  synchronized(sc object)
  {
    == // ServletContext : attribute manipulation
  }
  synchronized(sc obj)
  {
    == // Service attribute manipulation
  }
}
```

Marker Interface means Empty Interface

the implementation

Q → what is the Thread safety technique utilized in your project?

— (A) we prefer working with synchronized blocks in Service method or doxxx methods.

④ Implementing javax.servlet.SingleThreadModel Interface on Servlet class :—

*** → when interface implimentation done on the class it is providing special identification for behaviour for the object then that interface is called "marker interface". most of the marker interfaces are empty interfaces (no methods declaration and also must be the predefined interfaces)

Ex — java.lang.Cloneable, java.io.Serializable

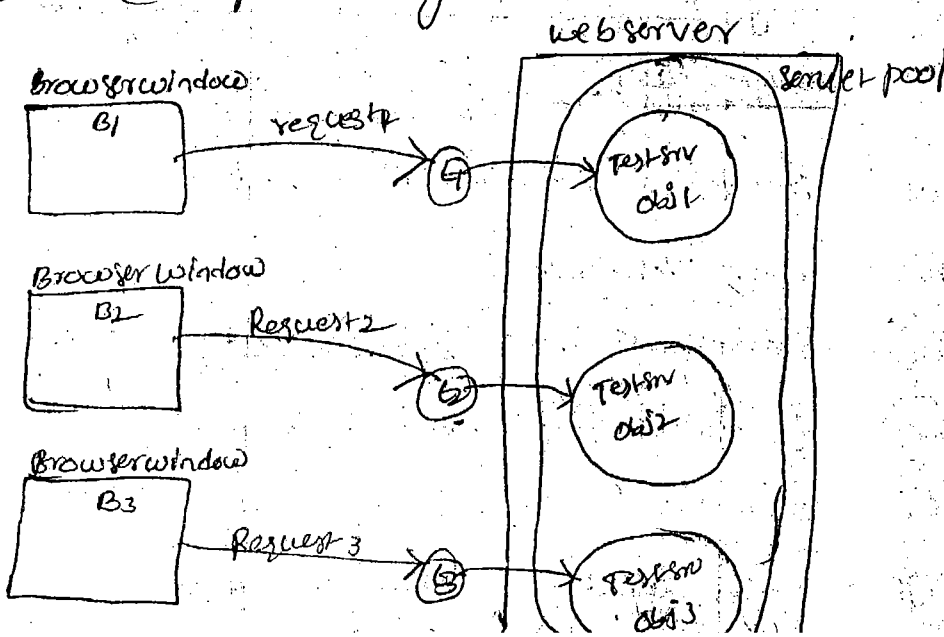
java.lang.Runnable

java.rmi.Remote and e.t.c

we can't develop user defined empty interfaces as marker interfaces.

```
public class TestSrv extends GenericServlet implements
{                                  javax.servlet.SingleThreadModel.
    public void service(-,-)
    {
        ---
        ---
        ---
        ---
    }
}
```

→ when this marker interface implemented as the underlying server uses its own choice of technique to make servlet as thread safe servlet. Some servers create multiple objects of our servlet class for multiple request on one per request basis by violating servlet specification principle when this interface is implementing.



principle that says a Servlet component must be single instance multiple threads component.

→ Since all the servers are not implementing this Single thread model interface in a common way this interface is deprecated ~~from~~ Servlet-API 2.3 on words So this is not at all good practice to achive thread safty in Servlets.

Note :- this always recommented to word with third technique to achive thread safty.

Note: All JSPs are thread safe JSP's by default because the page compilar generated JSP equalent Servlet will always come having threadSafety by default.

Q → How to make our HttpServlet as flexible Servlet againest HttpRequest methods ⊛ Get, Post.?

→ there are three approches to make our HttpServlet as flexible Servlet againest all the request.

approach:1 :- override service method and place request processing / business logic

```
public class TestSrv extends HttpServlet
{
  p void service(---, --,) throws ---, ---
  {
    - - - - -
    - - - - -  } b.logic / request processing logic
    - - - - -
  }
}
```

approach:2 :- override both doGet(), doPost(), keep request Processing logic in both methods. doGet() and call the method from doPost()

```
p class TestSrv extends HttpServlet
{
  p. v  doGet(--, --) throws ---, ---
```

```
    }
    p v doPost(--, --) throws --, --
    {
        ---
        --- doGet(--, --);
        ---
    }
}
```

approach:3 :- place Request processing logic, or business logic in user defined method of our HttpServlet and call that method from doGet(), doPost() methods of our HttpServlet.

```
public class TestSrv extends HttpServlet
{
    p v doGet(--, --) throws --, --
    {
        xyz(req, res);
    }
    p v doPost(--, --) throws --, --
    {
        xyz(req, res);
    }
    p v xyz(--, --) throws --, --
    {
        --- /- B.logic
    }
}
```

→* what happens if the source code of web page is modified by using View → source option of browser window.

(A) :- If that web page is generated a web resource of a web application that is deployed in web server modification will not be effected.

→ to the source code of web pages modified whose .html file is saved in local file system (not in webserver) then modification done in source code will be effected.

(11) → what is the diff b/w HttpRequest methods Get & Post

GET	POST
(1) Design to get the data from webserver but can send limited amount of data to the server (max (256 KB))	(1) Design to send unlimited amount of data to the server but can also to use to get the data from the server.
(2) Form page generated query string will apper in the address bar.	(2) Form page generated query string will not apper in the address bar.
(3) Dose not provide data secracy.	(3) provides data secracy.
(4) dose't support encryption techniques..	(4) Support encryption techni-ques.
(5) not suitable for file uploading operation.	(5) suitable for file uploading operations.
(6) can't send multi part form data (diff mime type based form data).	(6) can send multi part form data.
(7) use service() (or) doGet() in our httpservlet to procces the request.	(7) use Service() (or) doPost() to process the request. in our HttpServlet.
(8) GET is Idompotent	(8) post is non-Idompotent

Note :- Before complition of existing request processing given by a client if another request processing given by the same client is started then it is called Idompo-tent behaviour. This is bad behaviour

note:

(7) GET is Idompotent and POST is non-Idompotent so it is always recomanded to place sencitive business logic in doPost() of our Servlet.

/* <welcome-file-list> web.xml

<welcome-file> input.html </welcome-file>

</welcome-file-list> */

→ the first page of web application which comes automatically the moment web application is requested is called welcome page (or) Home page of the web application. if no welcome page is configured explicitly the webserver looks to take index.jsp (or) index.html file as default welcome page. we can configure multiple jsp or html ~~as~~ welcome pages as welcome pages but only one ~~will~~ welcome page will taken in the priority order of their availibility.

→ we need to write following code explicitly in web.xml to configure welcome page for web application

```
<welcome-file-list>
    <welcome-file> input.html </welcome-file>
    <welcome-file> input.jsp </ " >
    <  "  > input.html </ " >
</welcome-file-list>
</web-app>
```

→ If all the welcome files that are configured explicitly are not available then the web server does not look to take index.html (or) index.jsp (or) as default welcome page in this situation the web application runs without welcome

→ we can't configure servlet web resources as welcome files of web application.

03/12

→ Every object in this world contains it's life cycle methods. at different phases of object life cycle different events will be rise on the object to handle this events and to execute certain logics based on this events we need to work with event handle methods (or) lifecycle methods (or) container call back methods.

the methods of a resource which will be executed automatically by underlaying container s/w based on the event that are raised on the object are called life cycle methods (or) container call back methods.

A Servlet object state in 4 states in it's lifecycle

① init doesn't exist state.

② just create that state.

③ Request processing state.

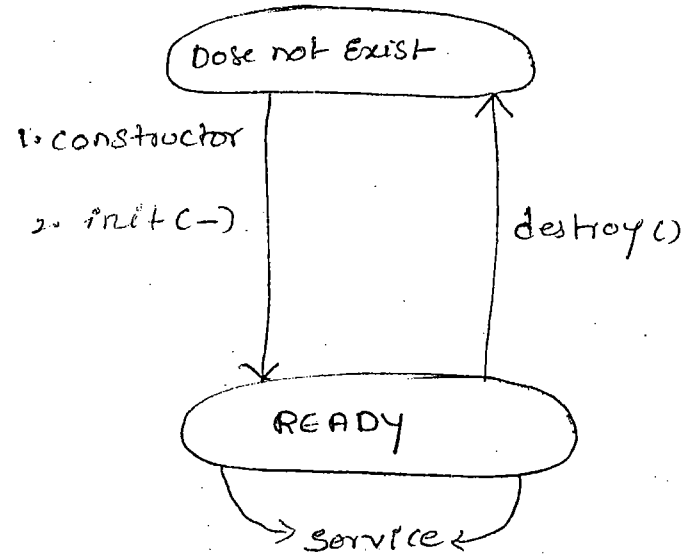
④ about to destroy state.

→ To handle various events raised on Servlet object there are 3 life cycle() in a servlets they are:-

① puplic void init(ServletConfig cg) throws ServletException.

② public void service(ServletRequest req, ServletResponce res) throws ServletException, IOException

③ public void destroy()

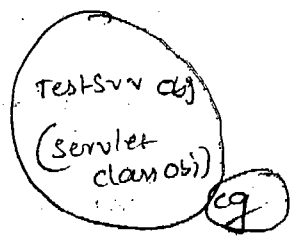


init(-) = init Single parameter.

init() :- is not a lifecycle() method it is a Helper method for lifecycle method called ~~init~~ para init(-) parameter.

→ protected service() method seven doXxx() methods are not lifecycle() methods they are direct (or) indirect helper methods for lifecycle() method public service.

cg → ServletConfig



→ ServletConfig object is the right-hand object of Servlet object. & ServletConfig object is one per Servlet object this ServletConfig object will be created and destroy based on Servlet object.

→ TO pass additional information to servlet and to gather details from servlet, we need ServletConfig object which is a righthand object of this Servlet.

→ init(-) :- init(-) is a one time execution lifecycle() method of a servlet this method executes automatically when Servlet object is created. this method generally contains initialization logic like creating JDBC connection.

→ service() :- this method is a repeatedly executing lifecycle() method of a servlet. this method execute for every request comes to servlet so we generally placed business logic & request processing logic in this method.

→ destroy() :- this method is one time execution lifecycle() method of a servlet : this method will be executed by container when servlet object is about to destroy so we generally place un initialization logic in this method. like closing JDBC connection.

Note :- No lifecycle() method of a servlet will be called by the programmer explicitly. all the lifecycle() methods will be executed automatically by web container at different phases of servlet object lifecycle.

flow of execution indicating order

→ when servlet component gets it's first request from Browser :

ⓐ webcontainer/webserver loads servlet class from WEB-INF/classes folder and creates object for that class.

Ex :- class.forName("OATESRV").newInstance();

ⓑ zero-argument constructor of servlet class will be execute.

ⓒ ServletConfig object will be created for servlet object.

ⓓ web container calls lifecycle method init(-)

ⓔ web container calls service(-) and performs request processing.

ⓕ Responce goes to browser.

→* webcontainer verifies whether request to Servlet component object is available (or) not.

if already available

→ web container calls Service(-,-) on that object
→ Request processing takes place and responce goes to browser

If not available

→ webcontainer performs all the operations of first request

**

→ S.O.P statement relate output messages kept in Servlet will come on console monitor of underlaying webserver/application server.

tomcat 5.5 → bin → start tomcat

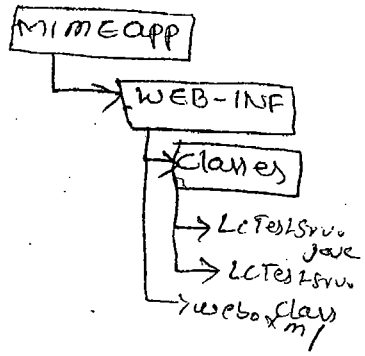
**

→ pw.println(); statements related output kept in Servlet will come on browser window using which the servlet is requested.

Servlet Component that demonstrate lifecycle of Servlet:-

```
//LcTestSrv.java
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;

public class LcTestSrv extends HttpServlet
{
    LcTestSrv() { S.O.P("Zero-arg constructor of LcTestSrv"); }
    public void init(ServletConfig cg) throws ServletException
    {
        S.O.P("init(-) method of LcTestSrv servlet");
    }
    public void service(ServletRequest req, ServletResponce res)
                        throws ServletException, IOException
    {
        res.setContentType("text/html");
        PrintWriter pw = res.getWriter();
```

```
pw.println("<b>Date and time is</b>"+new java.util.Date().
                                        toString());
sop("service(-,-) of LCTestSrv");
pw.close();
} // service(-,-)
public void destroy();
{
  s.o.p("destroy() of LCTestSrv");
}
} // class
```



Note:- Save the above servlet in the MIMEApp webapplications WEB-INF/classes folder and configar the servlet in web.xml file having /Lcurl as the url pattern.

04/12

→ When the web container creates object of the servlet webresource?

if servlet is not enabled with load-on-startup.

ⓐ when servlet gets first request from Browser.

ⓑ when servlet gets it's first request after reloading the first application.

ⓒ when the servlet gets it's first request after restarting server.

Note:- when servlet is enabled with load-on-startup the web container creates object of servlet either during server startup or during the web application deployment.

→ When web container destroy servlet class object?

- object

ⓐ when web container performs Garbage collection on unused servlet

ⓑ when the web application is stopped.

ⓒ when the web application is reloaded

ⓓ when the web application is undeployed.

ⓔ when undeploying the webservice is stopped or crashed

enabled servlet class object.

Q) what is need of enabling load-on-startup on servlet?

A) :- when load-on-startup is not enabled on servlet the responce time of the first request comming to servlet will have extra responce time when compare to the responce time of other then first request so to equalize this responce time create object of servlet before first request for this enable load-on-startup on servlet which creates object of servlet either during server startup (or) during the deployment of web application.

Note :- It is recomanded to enable load-on-startup on those servlets which will be utilized immidiatly after the deployment of web application like the servlets that genarates home page, main menues and -- etc.

→ write following code on web.xml during servlet configuration to enable load on startup on servlet.

```
<web-app>
  - - - -
  = - - -
    <servlet>
      <servlet-name>lc</servlet-name>
      <servlet-class>LCTest</servlet-class> </servlet>
      <load-on-startup> 1 </load-on-startup> </servlet>
                          └ priority value
    <servlet-mapping>
      <servlet-name>lc</servlet-name>
      <url-pattern>/lcurl</url-pattern> </servlet-mapping>
</web-app>
```

→ when multiple servlets of web application are enable with load-on-startup all these servlet objects will be created either during Servlet startup (or) during deployment of web application. In which order these Servlet objects will be created is decided based on their load on startup priority values. high value indicates low priority. low value indicates high priority, zero indicates least priority negative value ignores load-on-startup. enabling process

note :- there is no default load-on-startup priority value for servlet.

→ Enabling load-on-startup on servlet by having negative

Four Servlets of web application.

4 Servlets	<1-0-S>	<1-0-S>	<1-0-S>	<1-0-S>
SV_1	3 — (II)	3 — (III)	3	3 — (I)
SV_2	4 — (III)	2 — (II)	3	0 — (III)
SV_3	1 — (I)	0 — (IV)	3	1 — (I)
SV_4	15 — (IV)	1 — (I)	3	-3 ← ignored
			Servlet decided the order	

So servlet object will be created when it gets first request.

Note :- It is not mandatory to enable on all servlets of web application.

⊘ → what happens mandatory to enable on all servlets of web application?

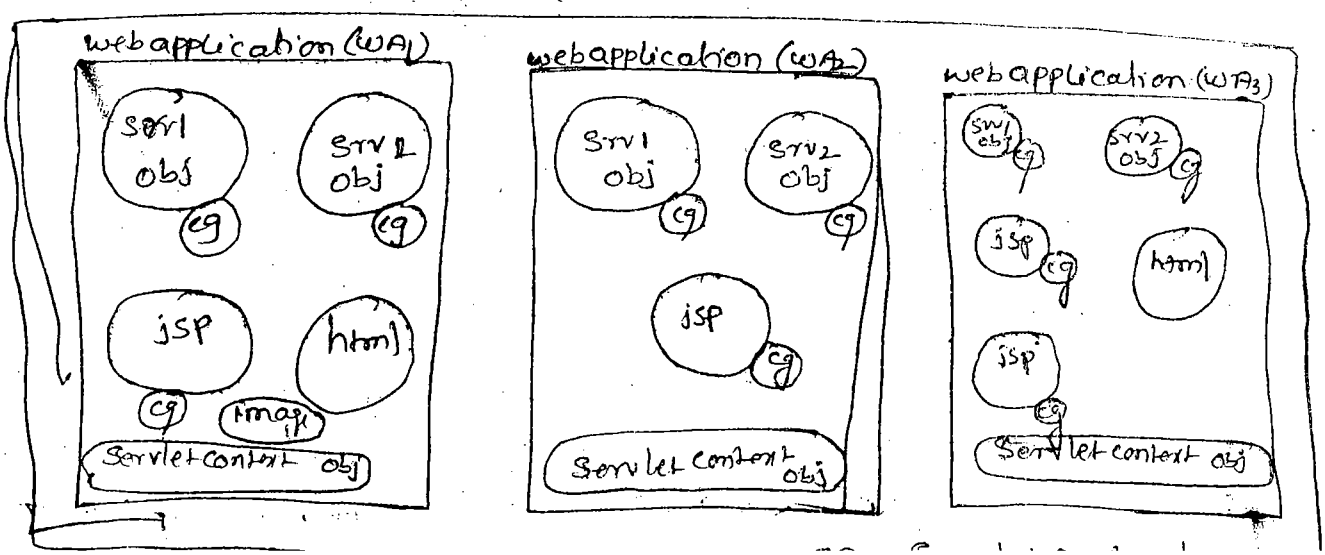
(A) :- servlet object will not be destroyed but logic of destroy() method will be executed once for every request given to servlet.

note :- when event is raised on servlet object the container automatically calls life cycle method (or) event handling method, but the container never raises the event on servlet object when life cycle method are called ~~manually~~ manually.

⊘ → what happens init() method is called explicitly from service()?

(A) The logic of init() method executes once for every request given to servlet.

⊘ → what is the difference b/w servlet config object and Servlet context object.



right hand object of Servlet.

② Each java web resource servlet or jsp will contain one servlet config object.

③ web container creates config object when it's servlet object is created and destroy servlet config object when it's servlet object is destroyed.

④ to pass additional information to servlet and to gather details about servlet we need servlet config object.

⑤ to pass init parameter values to servlet from out side the servlet we need servlet config object.

⑥ Servlet config object means it is the object of a class that implements javax.servlet.ServletConfig interface (this class will be supplied by web server)

ServletContext object :- ① It is one per web application so it is also called as global memory of web application.

② ServletContext object is visible and accessable in all the java web Browsers of web application.

③ web container creates servlet context object when web application is deployed and destroy servletContext object when the web application is stopped (or) reloaded (or) undeployed.

④ using ServletContext object we can pass global init parameter values to servlet from outside the Servlet.

⑤ ServletContext object means it is a object of a class that implements javax.servlet.ServletContext interface (underlaying server supply this class)

⑥ using ServletContext object we can gather details about current web application and the underlaying server like all servlet names, of web application, underlaying server name, version and the Servlet api version supported by server.

Ø → There are 20 web applications deployed in a server, 12 web applications are in running mode & 8 web applications are in stopped mode, can you tell me total how many no: of Servlet contexts are currently available?

Ⓐ :- 12

NOTE:- ServletContext obj will be destroyed when the web application is stopped (or) reloaded (or) undeployed.

Ø → There is a web application deployed in server in this web application 3 servlets are enabled with load on startup & other 3 servlets are requested by browsers (clients) can you tell me total how many servlet config objects are available in the web application?, when load or startup enabled on the servlet the servlet object will be created during server startup or web application is deployed. if load on start up is not enable on servlet the servlet object will be created when servlet gets its first request. according to this there will be 6 no: of Servlet config object in the above web application.

Ⓐ :- → object of our servlet class, ServletRequest obj, ServletResponce obj, HttpServletRequest obj, HttpServletResponce obj, ServletConfig obj, ServletContext obj will not be created by the programmers manually the underlying web server

To get access to our servlet class object from the same servlet use 'this' keyword. → To get access to ServletRequest obj, ServletResponce obj use parameters of public void service(-,-) method.

→ To get access to HttpServletRequest obj, HttpServletResponce obj use parameters are protected void service(-,-) method (or) doXxx(-,-) methods.

→ different ways of getting access to servlet config obj

① public class TestSrv extends HttpServlet
{
  public void init(ServletConfig cg)
  {
    - - - - -
    - - - - -
    - - - - -
  } use cg obj here
  - - - - -
  - ———
  - ———
}

② public class TestSrv extends HttpServlet
{
  public void
  ② ServletConfig cg;
  public void init(ServletConfig cg)
  {
    this.cg=cg;
  }
  public void Service(-,-)
  {
    use cg obj here
    - - - - - - -
    - - - - - - -
  }
}

③ public class TestSrv extends GenericServlet

```
{
  public void service(-,-)
  {
    ServletConfig cg = getServletConfig(); // public method available at Generic servlet
    ---------
    ---------
    ---------
  }
}
```

* To call selfclass methods and to call superclass public (or) protected methods in subclass methods there is no need of object.

ex getServletConfig is the public method available in javax.servlet.GenericServlet class so we can call that method in our servlet without any object.

Different ways of Getting access to ServletContext Object :-

① public class TestSrv extends HttpServlet

```
{
  public void service(-,-)
  {
    ServletConfig cg = getServletConfig();
    ServletContext sc = cg.getServletContext();
    ---------
    ---------  use sc obj here
  }
}
```

② public class TestSrv extends HttpServlet

```
{
  public void service(-,-)
  {
    ServletContext sc = getServletContext();
    ---------
```

→ In order to get access to the ServletContext obj we need to use ServletConfig object directly (or) indirectly

Ex

```
public class TestSrv extends HttpServlet
{
  public void service(HttpServletRequest req,
HttpServletResponse res) throws ServletException, IOException
  {
    PrintWriter
      String s1 = "raja";
      String s2 = "ryd";
    PrintWriter pw = res.getWriter();
    res.setContent("text/html");
    pw.println(s1+" "+s2);
  }
}
```

→ The above servlet is not flexible servlet because s_1, s_2 variable values are hard coded in order to pass new values to that servlet we need to modify source code of the servlet. to make the above servlet as flexible servlet we need to pass values of s_1 & s_2 variables from outside the servlet either in the form of request parameters are in the form of inites init parameters. (initialization parameters)

(1) Request Parameters :- It is given by end user through browser by appending query stream to the request r url of web resource. so only nontechnical values can be passed as input values to servlet as ~~init~~ RequestParameter

```
public class TestSrv extends GenericServlet
{
    public void service(—, —) throws —, —
    {
        String s1 = req.getParameter("sname");
        String s2 = req.getParameter("sadd");
        PrintWriter pw = res.getWriter();
        res.setContent("text/Html");
        pw.println(s1 + "  " + s2);
    }
}
```

TestSrv Parameters

browser window

http://localhost:4040/WB1/TestSrv(url)?sname="raja"&sadd="hyd"

address bar

→ Note Since request parameters are coming from client side the client can't pass technical input values to servlet as request parameter values to solve this problem use init parameters.

② init parameters :-

```
<web-app>
    <servlet>
        <servlet-name> xxx </servlet-name>
        <servlet-class> TestSrv </servlet-class>
        <init-param>                    ①
            <param-name> sname </param-name>
            <param-value> raja </param-value>
        </init-param>
        <init-param>                    ②
            <param-name> sadd </param-name>
            <param-value> hyd </param-value>
        </init-param>
```

```
    <servlet-name>xxx</servlet-name>
    <url-pattern>/testsrv</url-pattern>
  </servlet-mapping>
</web-app>
```

- init parameter values of servlet will be passed to Servlet from web.xml during the configuration Servlet in web.xml file.

→ init parameters are the server side input values passed to the servlet from outside the servlet. Since web.xml file is managed by programmers we can pass technical input values to servlet like JDBC Driver ve details from out side the servlet as init parameter value.

→ Servlet utes ServletConfig obj to read init parameter value of a servlet from web.xml file

Ex

```
public class TestSrv extends HttpServlet
{
  public void service(—,—) throws —,—
  {
    ServletConfig cg=getServletConfig();          ①
    String s1=req.getParameter("sname");
    String s2=req.getparameter("sadd");           ②
    PrintWriter pw=res.getWriter();
    res.setContext("text/html");
    pw.println(s1+"  "+s2);
  }
}
```

→ init parameters are specific to a Servlet for which they are configured.

8/12

→ If same name is given for multiple init parameters only last value will be taken as init parameter value.

Diff ways of Gathering init parameter values being from Servlet By using ServletConfig object :-

approach1 :-
```
ServletConfig cg=getServletConfig();
String s1=cg.getInitparameter("stno");
String s2=cg.getInitParameter("stname");
```
Here we must know the name in order to get the value.

approach2 :- Enumeration
```
ServletConfig cg=getServletConfig();
Enumeration e=getInitParameterNames();
while(e.hasMoreElements())
{
   String name=(String)e.nextElement();
   String val=cg.getInitParameter(name);
}
```
// This gives all names & values of init parameters.

→ init parameters are specific to a servlet for which they are configured.

→ If multiple servlets of a web application are looking to use same init parameters instead of configuring them as init parameters are every servlet it is recommended to configure them only once as context parameters.

inorder to read context parameters of web application from web.xml file we need to use ServletContext object

interact with DB s/w by using JDBC code then it is recomanded to make JDBC code as flexible code by passing JDBC Driver classname, url, DB user name, & password details as context parameter values.

Coding web.xml :-

```
<web-app>
    <context-param>
        <param-name>st-no</param-name>
        <param-value>1001</param-value>
    </context-param>

    <c-p>
        <p-n>stname</p-n>
        <p-v>Raja</p-v>
    </c-p>
</web-app>

<servlet>
    <s-n>ABC</s-n>
    <s-c>com.nit.s1</s-c>
    <load-on-startup>5</load-on-startup>
</s>

<s-m>
    <s-n>ABC</s-n>
    <u-p></u-p>
</s-m>

</web-app>
```

Code in Any Servlet of the web application:

```
ServletConfig cg = getServletConfig();
ServletContext sc = cg.getServletContext();
String s1 = sc.getInitParameter("stno");
String s2 = sc.getInitParameter("stname");
pw.println(s1 + " " + s2);
```

parameters are specified to a servlet, but context parameters are common for all the servlets of the web application. So context parameters are called global init parameters

Different approaches of Reading Context parameter Values from web.xml :-

Approach 1 :- ServletConfig cg = getServletConfig();
ServletContext sc = cg.getServletContext();
String s1 = sc.getInitParameter("stno");
String s2 = sc.getInitParameter("stname");

here we must know the name of context parameter in order to get the value

Approach 2 :- ServletConfig cg = getServletConfig();
ServletContext sc = cg.getServletContext();
Enumaration e = sc.getInitParameterNames();
while(e.hasmoreElements())
{
String name = (String) e.nextElement();
String val = sc.getInitParameter(name);
}

// this gives all names & values of Context Parameters.

* → Can i give same name for init parameter & context parameter?

A :- yes it is posible, we can read both values from servlet to read init parameters use ServletConfig object. Use to read context parameter value use ServletContext object.

parameter and context parameters values placed in web.xml file to read technical input values of a Servlet from outside the Servlet.

Q → wt is the diff b/w code kept in the constructor & init() method of a Servlet.

(A) :- ServletConfig obj will be created when Servlet obj is created after executing constructor and before executing init method. So that ServletConfig obj is not visible in the constructor and the ServletConfig obj will be visible in the init(-) method. due to this code kept in the constructor can't read init parameter of Servlet & context Parameters of a web application.

where as the code kept in the init(-) method can workwith init parameter values, context parameter values.

Q → wt is the diff b/w absolute url & relative url ?

(A) :-

input.html

```
<form action="testurl" method='get'>
        ____ relative url
        ____
</form>
```

input.html

```
<form action="http://localhost:4040/servletapp1/testurl" method='get'>
        ____ absolute url
        ____
</form>
```

If identity of destination web resource is given with respect to the source web resource identity then it is called relative url

If the identity of destination web resource is given without dependency upon source web resource identity then it is called absolute url.

n] web application development. In order to make the web applications
→ WORA web applications (write once deploy anywhere)

9/12

Weblogic
- → Type: Application server s/w.
- → vendor: BEA systems (oracle corporation)
- → version: 8.x (compatable with J2SDK 1.4)
  : 9.x (compatable with J2SDK 1.5)
- → default port no: 7001
- → commercial s/w.
- → downloade s/w: www.commiers.BEA.com
- → to get Help: www.edocs.bea.com
- → Give lots of built in tools: like weblogic builder (JDE) : weblogic workshop (DB s/w): point base and e.t.c
- → allows create domain: default domain is examples server

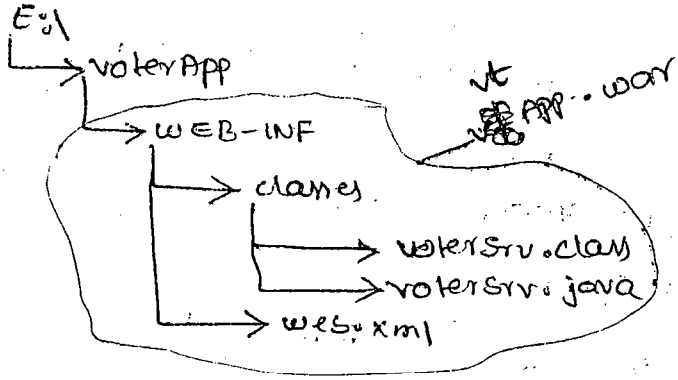
→ Each domain created in weblogic is called one application server, if company is having multiple projects and these projects are using weblogic s/w then the web-logic s/w will be installed only once. in a common or integrated mechine but multiple domains will be created for multiple projects on one per project bases.

→ the page where weblogic s/w is installed is called bea - BEA. home directory

→ pro

Procedure to deploy webapplication in example domain Server weblogic (console deployment):

step1 :- prepare deployment directory structure of web-application & generate WAR file on it



E:\voterApp> jar cf vtapp.war . ↵

c ⟶ create jar/war files

f ⟶ use specified input file.

⟶ Each war file represents one webapplication war file means webapplication archieve.

⟶ jar is built-in tool command J2SDK s/w i.e given to prepare the jar files (or) war files archiever files means zip files.

step2 :- start example server domain of weblogic

Start
→ programms
→ BEA products
→ examples
→ weblogic server
→ start examples server

(3) -open administration console of examples server placed in weblogic

(2) ⟶ open browser window
→ http://localhost:7001/console

→ username: weblogic

password: weblogic

→ login

④ → Deploy the web application.

open administration console

→ lock and edit

→ deployments

→ install

→ uplode your files

→ Brows & Select : vtApp.war file

→ next

→ next

→ Select vtApp.war file

→ next

→ next

→ next

→ finish

→ save

→ Activate changes

→ deployments → Select deployed app (vtApp) → start → serving all the request

⑤ Test the web application open browser window

http://localhost:7001/vtApp/input.html

→ There are 3 ways to deploy the webapplications in web application server tools

① console deployment

Deployment by using administration console (or) manager windows (war file is require)

② Hard deployment/~~not~~ Hot deployment.

copying the deployment directory structure of web application

Note :- like copying webapplication directory to tomcat home/webapps folder.

— ③ Tool based Deployment

note :- deployment by using tools like ~~ant, maven~~ ANT, MOVEN and etc.

Procedure to deploy webapplication in tomcat server through console deployment :-

Step1 :- prepare war file representing the webapplication

Same as above war file

② open tomcat webapplication manager window.

Start tomcat
↳ open home page (http://localhost:4040)
↳ tomcat manager
↳ submit username & password
↳ this gives tomcat webapplication manager window

③ Deploy the webapplication :- Goto tomcat webapplication manager window
↓ select war file to upload
↓ Browse & select the war file.
↓ Deploy

④ Test the webapplication.

open browser window
↳ http://localhost:4040/vtApp/input.html (type url)

note :- when war file is deploy in tomcat, weblogic servers the file name of war file will act as webapplication name (or) context path.

→ to hard deploy web application in tomcat server copy war file or the deployment directory structure of webapplication to tomcat Home/ webapps folder.

→ to hard deploy web application in example server domain of weblogic copy war file or deployment directory structure of web application.

D:\ bea/ weblogic 9.0/ samples/ domains/ wl-server/ autodeploy

MyApp.war

*

→ procedure to undeploy to the webapplication from admin consol of weblogic

Admin console
↳ deployments
↳ select lock and edit
↳ select the deploy webapplication
↳ delete

→ we can't undeploy web applications by using above procedure if they are deployed through hard deployment process

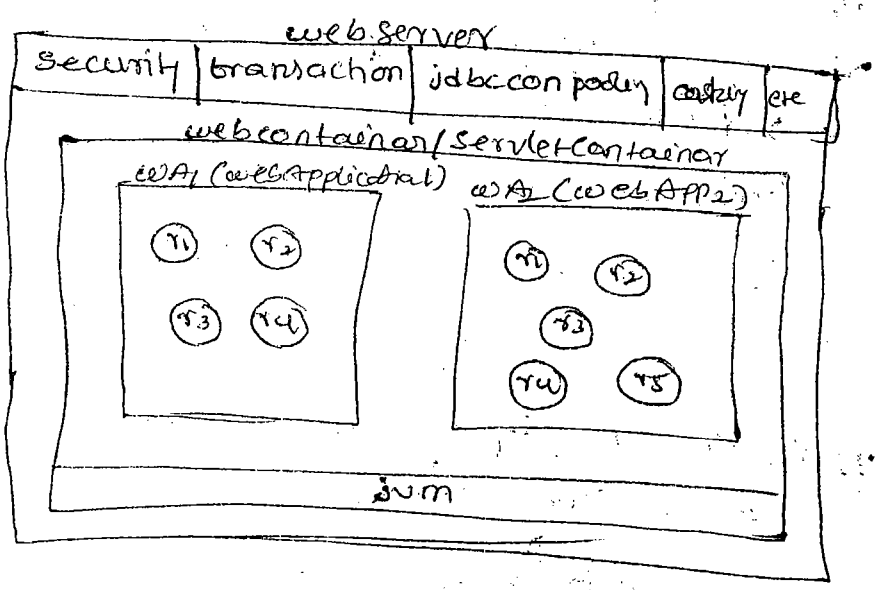
=

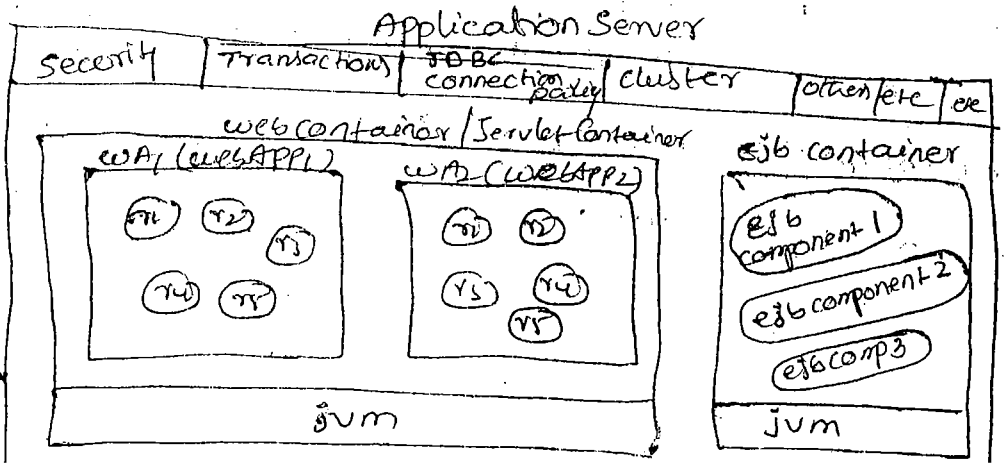
→ when webapplication is deployed in tomcat server if the modifications done in source code of servlet will be reflected after recompilation of source code & the reloading of the webapplication. where as weblogic server just recompilation is enought this feature is called automatic redeployment.

Application Server s/w :- 10/12

Start
→ programs
└ BEA products
└ examples tools
└ weblogic configuration wizard

* → what is the diff b/w web server & Application server :





* Application Server = web container + EJB container + middle ware services.

→ the additional services that are configurable on ower applications are called middle ware services

Ex security, transaction & etc.

Web Server

Application server

war file means webapplication archive, that represent web Application

jar file means java archieve that represent ejb component

ear file means enterprise application archieve that represent an enterprise application.

ear = jar + war.

rar file means resource adopter application archieve which represents a J2EE application that is interacting with ERP, CRM s/w's (customer relation manage)

SAP is a erp s/w (enterprise resource plan) web server

EBEL is a CRM s/w; C Application server

web server	Application Server
→ webserver is given to deploy, manage & to execute web Application (war files)	→ Application server is given to deploy, manage & execute web applications (war file), Ejb components (jar files), enterprise application (ear file) & resource adopter applications
→ web server gives only web container	→ application server gives both web, ejb container.

amount of middleware services

→ webserver is given by only Servlet, jsp api's of J2EE.

→ webServer allows only Http clients (Browsers)

→ recognizes only Http, Https protocals

→ use web server to develop small scale and mediam scale web application.

Ex:- Tomcat, Resin, JWS, ZT and etc

more number of middleware services

→ Application Server is given by based on all the API's of J2EE.

→ Application Server allows Http client (Browsers) and all kinds of java clients application.

→ Recognizes Http, Https, T3, SMTP, POP3 and etc protocalls.

→ use application server to develop large scale web applications enterprise application, ejb component & etc.

Ex:- weblogic, websper, JBoss, JRun, glanfish, oracle log as, etc

---

→ Reading form Data from Different types of form controllers.

text box

```
<input type="text" name="t1">
```

code in servlet

```
String s = req.getParameter("t1");
```

password blocks

```
<input type="password" name="t1">
```

code in servlet

```
String s = req.getParamer("t1");
```

text area name=

<textarea cols=10 rows=10>

</textarea>

code in servlet

String s=req.getParameter("ta1");

checkbox

ex1 <input type="checkbox" value="reading books" name="ch1">

code in servlet

String s1=req.getParameter("ch1");

s1 holds null when ch1 checkbox is not selected

s1 holds reading books as value when ch1 checkbox is selected.

ex2:

<input type="checkbox" value="reading books" name="ch1">

<input type="checkbox" value="playing cricket" name="ch1">

<input type="checkbox" value="stamp collection" name="ch1">

code in servlet

String s[]=req.getParameterValue("ch1");

radio button

<input type="radio" name="r1" value="male" />

<input type="radio" name="r1" value="female" />

when multiple radio buttons are having same name they will be grouped in to single unit so that we can select only one radio button at a time.

code in servlet

String s1=req.getParameter("r1"); //male or female.

→ when radio button & checkbox is selected the value available in value attribute of respective input tags that are used to prepare radio button, check boxes will be given as request parameters.

request parameter comes to server.

single selection list box

```
<select name="job">
    <option value="clerk">CLERKS</option>
    <option value="manager">MANAGERS</option>
    <option value="saleman">SALESMEN</option>
</select>
```

code in servlet

```
String s = req.getParameter("job");
```

if MANAGERS item is selected then s holds manager.

→ when an item of select box is selected the selected item will not goto the server as request parameter. The value available in the value attribute of option tag i.e representing selected item will go to server as request parameter.

multiSelection list box

```
<select name="job">
    <option value="clerk">CLERKS</option>
    <option value="manager">MANAGERS</option>
    <option value="salesman">SALSEMENS</option>
</select>
```

code in servlet

```
String s[] = req.getParameterValues("job");
```

when MANAGERS, CLERKS item are selected then the server gets clerk, manager as request parameter values.

NETBEANS (IDE) :-

type : IDE s/w to develop java/J2EE Application

version : 6.x (J2SDK 1.5), 5.x (J2SDK 64/1.5)

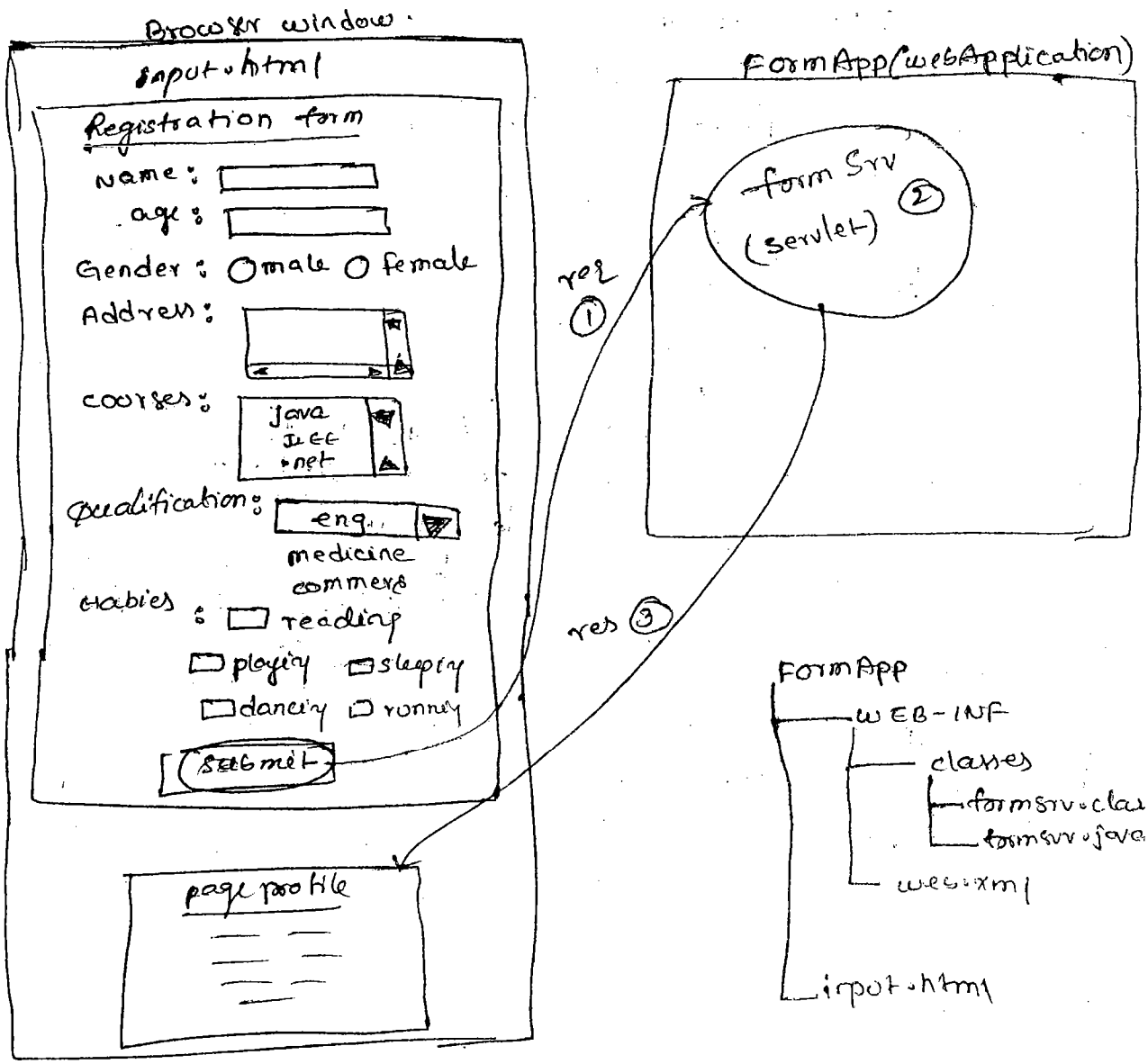
vendor : sun ms (oracle corporation)

open source s/w.

for downloading s/w : www.netbeans.org

to get help : www.netbeans.org

Built-in servers : tomcat (for netbeans 5.x)
glassfish (for netbeans 6.x)



Then it is recomanded to design html form as Html table Rows, & colums due to this true alignment of the form becomes easyly

Procedure to develop above diagram based web Application By using netBeans 6.7 IDE :-

Step1: lanch netbeans IDE

start
→ programms
→ netBeans
→ netBeans IDE 6.7.1

Step2:- create web project in netBeans IDE

Filemenu
→ newproject
→ java web
→ webApplication
→ next
→ myweb proj : project name
→ server = glashfish
→ next
→ finish
↳ delete index.jsp

Step3:- Add input.html file to the project.

Expand project:
→ rightclick on webpages
→ new
→ Html
→ Html filename : input
→ finish

input.html

```
<b> <center> Registration form </center> <br>
<form action = "formurl"
 <table border = "1" align = "right">
  <tr>
    <td> name </td>
    <td> <input type = "text" name = "name" size = "20" /> </td>
  </tr>
  <tr>
    <td> password </td>
    <td> <input type = "password" name = "pwd" size = "20" /> </td>
  </tr>
  <tr>
    <td> email </td>
    <td> <input type = "text" name = "email" size = "20" /> </td>
  </tr>
  <tr>
    <td> mobile </td>
    <td> <input type = "text" name = "mobile" size = "20" /> </td>
  </tr>
  <tr>
    <td> <input type = "submit" value = "submit" /> </td>
    <td> <input type = "reset" value = "reset" /> </td>
  </tr>
 </table>
</form>
</b>
```

```
<tr>
  <td>Age</td>
  <td><input type="password" name="page" value=" "/></td>
</tr>
<tr>
  <td>Gender</td>
  <td><input type="radio" name="g1" value="male"/> Female
    <input type="radio" name="g1" value="female"/>
  </td>
</tr>
<tr>
  <td>Address</td>
  <td><textarea name="paddr" rows="4" cols="20">
  </textarea></td>
</tr>
<tr>
  <td>courses</td>
  <td><select name="pcourse" multiple="multiple">
    <option value="java">java</option>
    <option value="dotnet">.Net</option>
    <option value="J2EE">J2EE</option>
  </select name></td>
</tr>
<tr>
  <td>Qualification</td>
  <td><select name="pqlfy">
    <option value="engg">BE/B.Tech</option>
    <option value="medical">medical</option>
    <option value="science">science</option>
  </select name></td>
</tr>
<tr>
  <td>Hobbies</td>
  <td><input type="checkbox" name="pch1" value="Reading"/>Reading books
  <input type="checkbox" name="pch1" value="cricket"/>playing
  <input type="checkbox" name="pch1" value="stamp"/>Stamp collection
  </td>
</tr>
<tr>
  <td><input type="submit" value="submit"/></td>
</tr>
</table>
</form>
```

(4) → Add Servlet to the project

right click on project
→ new
→ servlet
→ class name: formsrv
→ next
→ url pattern : formurl
→ finish
→ Add following logic in process request() method of servlet.

note :- process request is the user defined method generated by IDE and this method is called in doGet() & doPost() methods

```
public class formSrv extends HttpServlet
{
  protected void procesRequest(HSR req, HSR res) throws SE, IOE
  {
    responce.getContentType("text/html");
    PrintWriter pw = responce.getWriter();
    // Read form data
      String s1 = request.getParameter("pname");
      "      s2 = "      ("page");
      "      s3 = "      ("paddrem");
      "      s4 = "      ("g1");
      "      s5 = "      ("pcourse");
      "      s6 = "      ("pqlfy");
      "      s7 = "      ("pch1");
    pw.println(s1+"person profile");
              ("<br> age is "+s2);
              ("<br> Address is "+s3);
```

```
pw.println("<br> courses are");
for(int i=0; i<ss.length; i++)
{
    pw.println(ss[i]+" ");
    pw.println("<br> Qualification is "+sq);
    pw.println("<br> Hobbies are ");
    for(i=0; i<sz.length; i++)
        pw.println(sz[i]+" ");
    pw.close();
}
```

(5) Run the project

Right-click on project
→ Run

Note :- when the Run project option is clicked the IDE s/w will take care of every thing i.e related to web application deployment.

⊘ *** How Servlet is executing without main(-)?

(A) :- If a running java application wants to create object wants to call methods of another class then there is no need of placing main method in another class

Ex
```
public Test {
    P.S.V.M(-)
    {
        ABC ab = new ABC();
        ab.xyz();
    }
}
```
(Running java Application)

```
class ABC
{
    P.V. xyz()
    {
    }
}
```
(Another class)

→ the java application i.e directly executing by comand prompt needs main() to start the application execution.

is the web resource program given as java class to web. Server/ Application Server s/w. Since this web server, Application Server s/w is already executing deamon process based java application there is no need of main(-) method in the Servlet class for web server/Application Server to create Servlet class obj and to call lifecycle methods of the Servlet

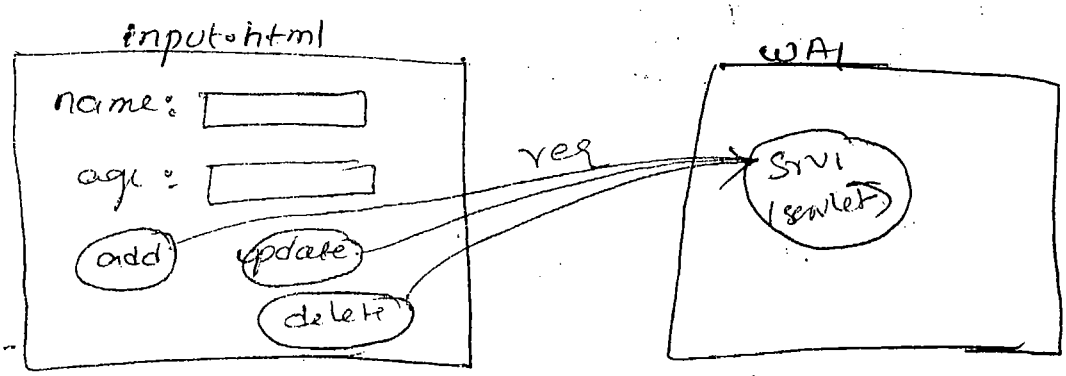
note:- when webserver/Application Server is started it internally uses main(-) method to get the Server startup.

note:- Servlet program will not be directly executed by jvm. It will be executed by another s/w application called webserver/Application Server.

Q→what happens if i keep main(-) method in the Servlet?

(A):- web server calls only lifecycle(-) methods of servlet class science main(-) method is not the lifecycle(-) method of servlet it will not be executed by ~~servlet~~ server. you may use main(-) method as helper method to other lifecycle(-) methods

→ How did you handle the situation to diff | 12 | 12 | erentiate the logic to servlet when the multiple submit button points to Srv1(Servlet)



code:-

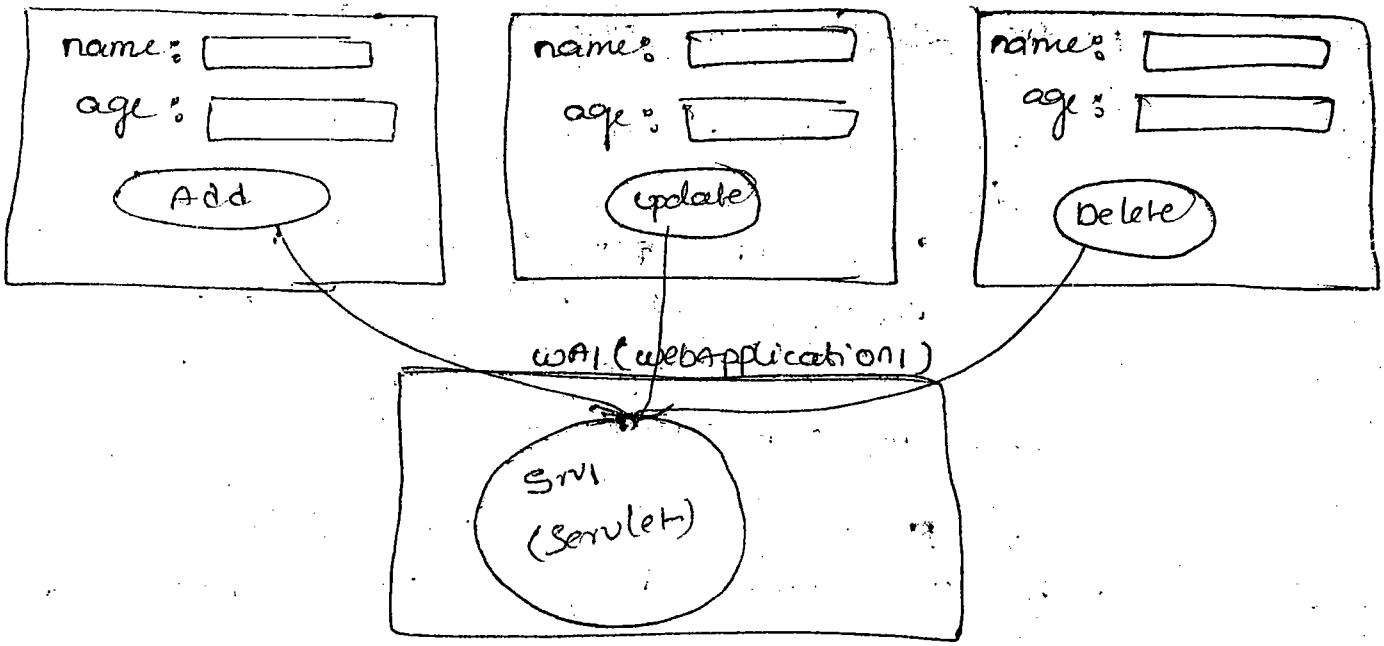
```
<form action="Srurl">
name <input type="text" name="pname"/>
age <input type="text" name="page" />
    <input type="submit" name="Sr" value="Add" />
    <input type="submit" name="Sr" value="Update" />
    <input type="submit" name="Sr" ...
```

```
public class Srv1 extends HttpServlet
{
    public void service(--,--) throws SE,IOE
    {
        String cap = req.getParameter("s1");
        if(cap equals("Add")) {
            ========= logic of add.
        }
        else if(cap equals("update")) {
            ========= logic of update
        }
        else {
            {
            ========= code of delete.
            }
        }
    }
```

→ to handle the above diagram take multiple submit button in the form page having same name and difference captions. and receive captions of the submit button in servlet as request parameter value based on that value differenciate the logics of multiple submit buttons

note!:- when submit button is design without main no request parameter will go related to submit button when form is submited. when submit button is design having name the name of the submit button & the caption of the submit button will goto Server as Request parameter name & value. when the form is submited

Q → How to handle the situation in Servlet to differenciate logic when submit buttons of different forms are generating request to a single servlet?



input1.html

```
<form action="Srurl">
    name :
    age :
    <input type="submit" name="s1" value="Add" />
</form>
```

input2.html

```
<form action="Srurl">
    name :
    age :
    <input type="submit" name="s1" value="update" />
</form>
```

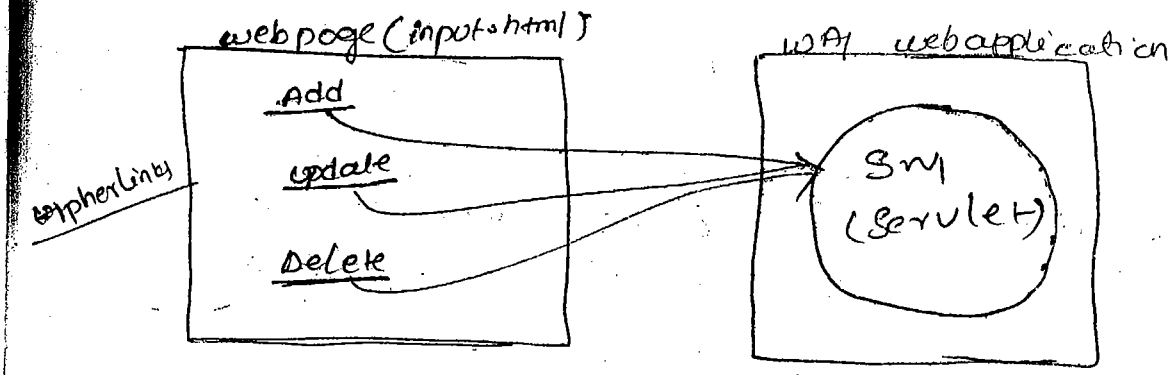
input3.html

```
<form action="Srurl">
    name :
    age :
    <input type="submit" name="s1" value="Delete" />
</form>
```

code in servlet

Same as above.

→ How to handle ... when multiple hyperlinks of a webpage are pointing to single Servlet web resource..?



code in input.html

```
<a href="$url?P=val1">Add</a>
<a href="$url?P1=val2">Update</a>
<a href="$url?P1=val3">Delete</a>
```

Srv1 Servlet code

```
public class Srv1 extends HttpServlet
{
    public void service(-,-) throws SE, IOE
    {
        String s=req.getParameter("P1");
        if(s.equals("val1")) {
          } ==== code for (Add) Hyperlink
        else if(s.equals("val2")) {
          } ==== code for (Update) Hyperlink
        else {
          } ==== code for (Delete) Hyperlink
    }
}
```

- To send input value along with the request genarated by Hyperlink it is always recomended to work with additional query string appended to the url kept in href attribute value belonging to <a> tag
  - anchor tag

type : Application Server s/w

version : 2.x

vender : sm (oracle corporation)

open source s/w

default port numbers : 4848 for admin console
8081 for web application.

to download s/w : www.java.sun.com

to get help : www.java.sun.com

Allows to create domains default domain is : domain 1

→ when netBeans 6.x s/w is installed it gives built in glassfish s/w this s/w/ server we can operate from IDE & outside the IDE.

glashfish home/ AppServer/ lib/ javaee.jar file represent hole J2EE API's

App Server

(1) procedure to deploy webApp in glashfish server

→ prepare war file on the staging directory of the web application (or better vtapp.war file prepared on 9/12 date)

(2) start defaut domain server of glassfish

Start
→ programms
→ sun micro system
→ Application Server
→ start default Server

(3) open administration console of glassfish

user name : admin

password : adminadmin

→ login

④ Deploy the web application:

Administration console
→ Expand Applications
→ web applications
→ Deploy
→ Browse & select war file (vtapp.war)
→ ok Application name: some name (vtapp)
context Route: vtapp
→ OK

⑤ Test the web Application:

→ Open Browser window
→ "http://localhost:8081/vtapp"

→ http://localhost:8081/vtapp/input.html

12/12

→ verifying the format of the form data is called form validation the logic used for this purpose is called form validation logic.
→ it is always recomanded to perform validations on form data before it is used in the business logic as input values
→ check in whether required fileds are typed or not, checking E-mail-id is having @, . symbol or not and extra other logics are called form validation logics.
→ It is always recomanded not to use form data directly in business logic of Server side web resource like Servlet, JSP because the enduser may supply wrong format value to form page. So perform form data validation by using form validation logic before using form data in business logic.

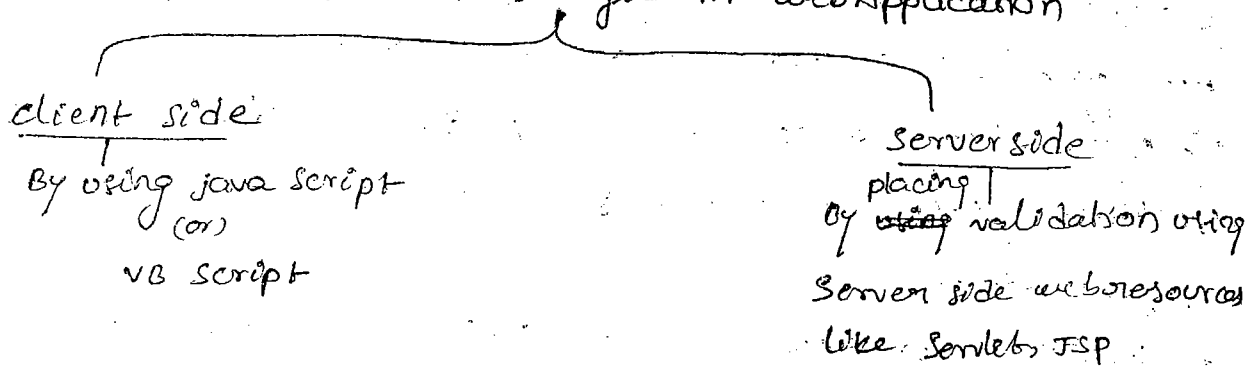
logic?

(A):- form validation logic verifies the format of the form data and it is not responcible to genarate results by using form data, when form page is giving student information checking whether all details are typed or not, checking m_1, m_2, m_3 values are given as number values or not comes under form validation logic.

Busines logic is main logic of the application it uses form data as input values to perform colucalation and analars to genarate result.

Genarating total & avg values based on student details comes under busines logic (or) Request Processing logic

=



→ Decide whether form validation logic is client side or Server side based on the page where it execute not based on the place where it reside Science, java Script, VB Script technologies are client side technology so their validation logic is client side form validation logic, Science Servlet, JSP's are Server side technologies so the form validation logic kept in them is called Server side form validation logic.

→ it is always recomanded to perform client side form validation, it reduces unnecessary n/w round trip b/w client & server (web server), because client side form validation blocks the request in the browser it self until current format of form data is supplied.

→ java script is the most recomanded technology to write client side form validation logic.

* → the form validation logic done in server increases n/w round trip b/w client (browser) & server (webserver).

Ex code of server side form validation logic

→ write following code in the doGet() of voterSrv Servlet representing form validation logic & business logic.

note:- this voterSrv servlet is given in voterApp web application developed on 28/11

```
String s1=req.getParameter("pname");
String s2=req.getParameter("page");

// write responce
    PrintWriter pw=res.getWriter();
    res.setContentType("text/html");

    int age=0;
    String errs="";

// form validation logic (Server side)
    if(s1==null || s1.equals("") || s1.length()==0)
    {
        errs="name is Required.<br>";
    }
```

```
{
    errs = errs + "Age is Required.<br>";
}
else
{
    try
    {
        age = Integer.parseInt(s2.trim());
    }
    catch(Exception e)
    {
        errs = errs + "Age must be number.<br>";
    }
}
if(!errs.equals(""))
{
    pw.println("<font color=red><center>"+errs+"</center></font>");
    return
}
```

// write b.logic

Same as 28/11

==

client side form validation (using java script).

→ we can't write java script base webresource directly because it is a scripting language so it has to be executed along with html file code. Science html & java script code comes to browser for execution only java script code based form validation form is called client side validation logic. to place form validation logic based on java script in voterApp web application develop input.html file as shown in bellow.

→ To get each form component value through java script code use form object.form component.value property.

→ isNAN is a predefined java script function that verifies whether given number/s value is numaric value or not.

code in input.html Belanging to voterApp web Application to perform java script based client side form validations :-

```
<!-- input.html -->

<html>
  <head>
    <script language="javascript">
    function validate()
    {
      var name = f.pname.value
      var age = f.page.value

      if(name=="")
      {
        alert("person name is Required");
        f.pname.focus();
        return false;
      }
      if(age=="")
      {
        alert("person age is Required");
        f.page.focus();
        return false;
      }
      else if(isNaN(age)) // number rule
      {
        alert("person age is number");
        f.page.focus();
```

```
<b> input.html </b>
<form action="vturl" method="post" name="f" onsubmit="
                                    return validate()">
<p> person name <b> <input type="text" name="pname"> <br>
<p> person age <b> <input type="text" name="page"> <br>
        <input type="submit" value="check vote eligibility">
</form>
```

→ In the above code onsubmit = "return validate()" kept in form tag calls validate java script function when submit button is clicked to perform form validations. The return statement placed there gives the return value of validate() method (true/false) to browser s/w. based on this value browser s/w will block or send the request to certain web resources of web application.

→ when browser s/w gets true value from validate function that indicates there are no form validation errors so request goes to server, when the browser s/w gets false value that indicates there are form validation errors so the request goes to will be blocked by the browser (return statement is mandatory in the above java script function call)

```
class Test
{
  public int x() (b)
  {
    y();
  }
  public void y(){
  }
}
```

```
class Demo extends test
{
  public void y() { (c)
  }
  Demo d=new Demo();
    d.x(); (a)
}
```

→ init() is not the lifecycle method it is helper method to the life cycle method init(ServletConfig cg). Scenario-1

```
public abstract class GenericServlet implements Servlet Interface
{
    ServletConfig cg;
    public void init(ServletConfig cg)
    {
        this.cg=cg;
        init();
    }
    public void init(){
    }                                   } null method
    ---
    ===
    ===                                 } other methods of class
}

public class Test Srv extends GS/HS
{
    public void init(){
    ---
    ===
    }
    public void service(S req, S res)
    {
    ---
    ===
    }
```

→ the init(ServletConfig cg) method of predefined GenericServlet class initializes container supplied ServletConfig object with our servlet object by having some logic and also calls init() method

from the browser → web container creates our servlet class object → zero argument constructor of our servlet class executes.

② web container creates servlet config object for servlet object

③ web container calls the life cycle method init ServletConfig cg method by passing ServletConfig object as argument value. since that method is not there in our servlet class

④ the superclass init ServletConfig cg method executes

⑤ the init(ServletConfig cg) of predefined GenericServlet class internally calls init no parameter method since our method is available in our servlet class and that will be executed servlet initialization will be completed.

note: init(ServletConfig cg) method of pre defined generic servlet class initializes the web container created ServletConfig obj with our servlet class object.

⑤ web container calls next life cycle method called public service method science that method is available in our class that method will be executed and responce goes to browser.

Senario ②

public abstract class GenericServlet implements ServletInterface {

```
ServletConfig cg;
init(ServletConfig cg)
{ this.cg = cg;
  init();
}
```

```
public void init()
{
}
```

```
public class TestSrv extends GS/HS
{
    public void init(ServletConfig cg)
    {
        ---- (3)
    }
    public void service(S req, S res)
    {
        ---- (7)
    }
}
```

Scenario (3)

```
public class extends GenericServlet implements ServletInterface
{
    ServletConfig cg;
    init(ServletConfig cg)
(4) {
        this.cg = cg;
        init()
    }
    public void init(){
        ---
(5) }
    ---
}
```

```
(1)                    (2)
public class TestSrv extends GS/HS
{
    public void init(ServletConfig cg)
    {
(8)     super.init(cg);
    }
    public void service(S req, S res) {
        ---
        ---  (6)
    }
}
```

following scenario ①
& ②, because they are giving chance to execute init(ServletConfig cg) method of predefined GenericServlet class so that ServletConfig initialization takes place properly. and programmer need not to write that logic manually and these two scenarios the service method of our servlet can get access to ServletConfig object of our servlet by calling getServletConfig() method. but this is not posible in scenario ② because control is not passing to predefined init(ServletConfig cg) method of predefined GenericServlet class.

→ this relate information ~~main~~ init()., & init(ServletConfig cg)

refer page no: 23, 24

Ω

by

16

ver

ariab

by

1

1)

→ It is a subclass of Hashtable. using Hashtable we can maintain the data in the form of key, value. but the data of Hashtable is not permanent. and it is unable to read the data from properties file (property file is one which is organizing the data in the form of key, value and it is given bellow)

abc.prop / abf

```
sno = 101
sname = xyz
marks: 99.5
cname = vidya
```

using properties class in java we can develop flexible applications

API :- constructor

① property s();

Instance methods()

② public abstract Object getproperties(String)

③ " Object get(String)

④ public void load(InputStream obj)

→ constructor ① is used for creating an object of properties class

Properties p = new Properties(); // p → no data avalible.

→ methods ②, ③ are used for obtaining value of value by passing key value.

→ method ④ used for loading the content of properties file in to properties class object which is residing in main memory

FileInputStream fis = new FileInputStream("abc.prop");

fis → abc.prop

p.load(fis);

Object obj = p.get("sno");

Object obj = p.get("sname");

Object obj = p.get("cname");

→ write a java program which illustrate the concept of properties class:

create the properties file first as a text file with an extension .properties (or) .prop

└ resource bursing file

write java file to read data from properties file.

```
import java.io.*;
import java.util.*;
public class propp
{
    p s v m (String KLT)
    {  try{
       properties p=new properties();
       FileInputStream fis=new FileInputStream("student.prop");
       p.load(fis);
       Object obj"= p.get("sno");
       object obj= p.get("name");
       object obj= p.get("marks");
       object obj= p.get("cname");
       S.O.P("student number" +sno);
       S.O.P("student name" +name);
       S.O.P("student marks" + marks);
       S.O.P("student collegename" +cname);
    }
    catch (FileNotFoundException fi@)
    {
       S.O.P("prop file dos not exist");
    }
    catch(IOException e)
    {
       S.O.P("file corrupted");
    }
}
}
```

create table

student.prop

```
sno = 101
sname = xyz
marks = 99.9
cname = vidya
```

write command prompt

emp.vbb

```
sno=1
sname = abc
sal = 10000
cname = PSV
```

c:\> copy con emp.vbb

```
sno=1
sname = abc
sal = 10000
cname = PSV
```

ctrl + z (save)

display

type emp.vbb

→ write a java program to copy the content of one file into another file.

```
import java.io.*;
class xcopy
{
  p s v m (String k[])
  {
    if (k.length != 2)
    {
      S.o.p("Invalid arguments");
    }
    else { try {
    FileInputStream fis=new FileInputStream(k[0]);
    FileOutputStream fos=new FileOutputStream(k[1]);
    int i;
    while (i= fis.read()) != -1)
    {
      char ch=new (char) i;
      fis.write(ch);
    }
    fis.close();
    fos.close(); } catch (FNFE fe)
    { s.o.p("source file does not exist");
    }
    catch
    { (IOException e)
      s.o.p("problem in opening file in write mode");
    }
    }
  }
}
```

c:\>copy emp.abc satya

→ java xcopy prop.java satya.java

(prog name: xcopy = arg[0], prop.java = arg[0], satya.java = arg[1])

open in read mode → source

open in write mode → destination.

→ wrt a java prog which will display the content of the file (or) implement type command of dos.

Sometimes we need to process request given by a browser by using multiple servlets are chain → the process of getting request from a browser and processing that request with the support of multiple servlets kept in a chain is called servlet chaining →

Q

→ what is the need of init(), when init(ServletConfig cg) is given as lifecycle method.? ~~init~~

init() is given as convinience method as given as programmer to avoid the process of overriding init(—) method in our servlet class.

For related information refer Last two paragraphs for page no:- (22) & first two paragraphs of page no:- (23)

→ <u>understanding two service(—,—) & 7 doxxx(—,—) of predefined HttpServlet class</u> :-

```
public abstract class HttpServlet extends GenericServlet
{
    public void service (S req, S res)
    {
        (3) typecasting
        Sreq ——→ HSreq
        Sres ——→ HSres;
        service(HSreq, HSres)
    }
    protected void service (HSreq, HSres)
    {
        (4) read http request method (Get/post) ——;
            call an appropriate doxxx(HSreq, HSres)
            based Http req method
            doGet(HSreq, HSres) and doPost(HSreq, HSres) & ——
    }
    protected void doGet (HSreq, HSres)
    {
        return 405 error response;
    }
    protected void doPost (HSreq, HSres)
    {
        return 405 error response;
    }
    protected void doxxx (HSreq, HSres)
    {
        return 405 error response;
    }
    ——— other methods
}
```

```
Sreq = ServletRequest
Sres = " Responce
HSreq = HttpServletReq
HSres = HttpServletRespons
```

public void testSrv extends HttpServlet {

public void doGet(HSReq, HSRes) {

response goes to Browser

} —— (5)

}

→ In Generally says public Service() method of predefined HttpServlet class as Service() (1) & protected Service(-) of same class as Service(-) (2).

→ protected Service(-) (Service(-) (2)) & 7 doXxx(-) of predefined HttpServlet class are not lifecycle methods they are the helper methods for lifecycle method called public Service(-) (Service(-) (1)).

→ when Servlet is ready for requesting process a webcontainer ~~class~~ calls public Service(-) as lifecycle method

with Respect to code.

(1) → Browser give request testSrv Servlet having HttpRequest(-) Get.

(2) → web Container locates testSrv object then request is arrived.

note :- Here we are anuming the requested Servlet object is already available (testSrv object)

(3) → web container looks to call life cycle method public Service on our Servlet object. Since that are not available in our servlet the public service method of super class (predefined HttpServlet class) will execute.

(4) → public Service(-) of predefined HServlet typecast HSreq, Sres. Objects to HServlet req, HSres object & calls protected Service(-)

our only these objects since protected service method is not available in our servlet class the protected service method of superclass will execute.

(5) → the protected service(—) of predefined service(—) of calls doGet(—) method based on the HttpRequest-Method Get since doGet(—) is there in our Servlet class that method will execute & the responce goes to browser.

→ For Related Example Senario to understand flow of execution related to HttpServlet refer all the six Senarios given in page no: 23 & 24

→ dont ~~to~~ give chance to executes doxxx methods of predefined HS (HttpServlet) class because they send error response to browser having http status code 405 which indicate that our Servlet is not-fully capable of processing HttpRequest.

→ the abstract class in java can contain 17/12
only concrete method, only abstract methods & mix of both. In order to stop object creation for a java class the java class must be taken as abstract class.

→ In order to use logics of methods available in java class only in (or) only through it's subclasses the class must be taken as abstract class.

Q → when predefined HttpServlet class is having all methods as concrete methods why that class itself is given as abstract class

+. doXXX methods

doGet(-,-)
doPost(-,-)
doHead(-,-)
doTrace(-,-)
doPut(-,-)
doDelete(-,-)
doOptions(-,-)

Related doXXX method

7. HttpRequest methods

put
Head
Trace
Delete
Get
post
options

} generally web app.. these two are enaph

→ The webapplication developer generally deals with HttpRequest method "Get" & "Post" so that he deals with doGet, doPost methods of HttpServlet class and not bother about remaining 5 HttpRequest method & 5 doXXX methods.

→ A servlet can have 3 types of URL patterns.

① Exact match.
② Directory match.
③ Extension match.

① Exact match :- In exact match URL pattern multiple words can be there saparated with / (slash) symbol but must not be any * (star) symbol.

Ex :-

```
<servlet>
    <servlet-name>abc</servlet-name>
    <servlet-class>TestSrv</servlet-class>
</servlet>
<servlet-mapping>
    <servlet-name>abc</servlet-name>
    <url-pattern>/test1</url-pattern>
</servlet-mapping>
```

:-

http://localhost:4040/wA1/test1 (valid)

/test2
/test3
/test4
/abc
/xyz1
} Invalid.

other exact match patterns :-

<url-pattern>/test1/test2</url-pattern>

<url-pattern>/x/y/z</url-pattern>

note :- the URL pattern that we have used so for exact comes under exact match URL pattern.

② Directory match :-

:- <servlet>
<servlet-name> abc </servlet-name>
<servlet-class> testSrv </servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>abc</servlet-name>
<url-pattern> /x/y/* </url-pattern>
</servlet-mapping>

-> the Directory match URL pattern must contain * symbol as the last word.

* says zero or more characters.

Example URL's given from browser :-

http://localhost:4040/wA1/x/y/abc (valid)
wA1/x/y/abc.c (valid) 'c' language output.
wA1/x/y (valid)

/WA1/y/x/abc (invalid)

/WA1/a1b/x1y (invalid)

/WA1/x/y/x1y (valid)

/WA1/x (invalid)

→ In Directory match URL pattern the words you keep before * symbol are called directorys they can be one (or) more directorys.

③ Extension match :-

:- <servlet>

<servlet-name>abc</servlet-name>

<servlet-class>testSrv</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>abc</servlet-name>

<url-pattern>*.abc</url-pattern>

</servlet-mapping>

→ the extension match URL pattern starts with * symbol and contains extension word.(.abc)

Example URL gives from browser

http://localhost:4040/WA1/a1b1d.abc (valid)

/WA1/.abc (u)

/WA1/a.abc (u)

/WA1/xyz.abc (u)

/WA1/xyz.abc/.abc (u)

/WA1/a.abc/a.xyz (invalid)

/a1/n1a/xyz.c (invalid)

→ we can't prepare URL pattern for Servlet by mixing up multiple styles. It is against Servlet Specification when these kind of URL patterns are used our web applications will stuck up.

Ex :- <url-pattern>/x/y *.abc </url-pattern> (Invalid URL pattern preparation)

note :- Difference style of URL pattern are useful to provide security for web resources and to hide to technology name of web resource from end users.

18/12

Servlet to DB communication :-

→ DB s/w is a perminent memory store to manage the data for long time.

→ In order to gather input values required for Servlet from DB s/w and in order to store result generated by servlet in DB s/w, the servlet needs to interact with DB s/w's.

→ For Servlet to DB s/w communication place jdbc code in the Servlet by following one of these three approches.

→ In every jdbc code logic there will be 3 important operations to perform.

1. Create JDBC connection.
2. Use JDBC connection to manipulate table data.
3. Release JDBC connection.

→ instance variables declared in our servlet class are not thread safe variables by default.

→ local ~~Instance~~ variables declared in service(-,-) method of Servlet class

① create JDBC connection in init(-) method.

② use JDBC connection in service(-,-) method.

③ close JDB connection in destroy() method.

→ dis advantage:- Here JDBC connection object must be taken as instance variable so that it is not thread safe variable, to make connection object was thread safe object the multithreading issues related synchronization must be taken care by the programmer explicitly.

→ Advantage:- since JDBC connection object is created in the init(-) method, all the request comming to servlet will use single JDBC connection this improves perbormence of the application.

→ Approach:2:-

① create JDBC connection in service(-,-) method

② use JDBC connection in service(-,-) method.

③ close JDBC connection in Service(-,-) method.

- Advantage:- Here JDBC connection object is local variable in the service(-,-) method, so that it is thread safe object by default. So programmer is not bother about taking care of multithreading issues.

- disAdvantages :- for every request comming to servlet one new JDBC connection will be created blw servlet & DB s/w, this degrade the perbomence of the application.

① get JDBC connection in service(-,-) from JDBC connection pool.

② use JDBC connection in service(-,-)

③ Release JDBC connection back to JDBC connection pool being from service(-,-).

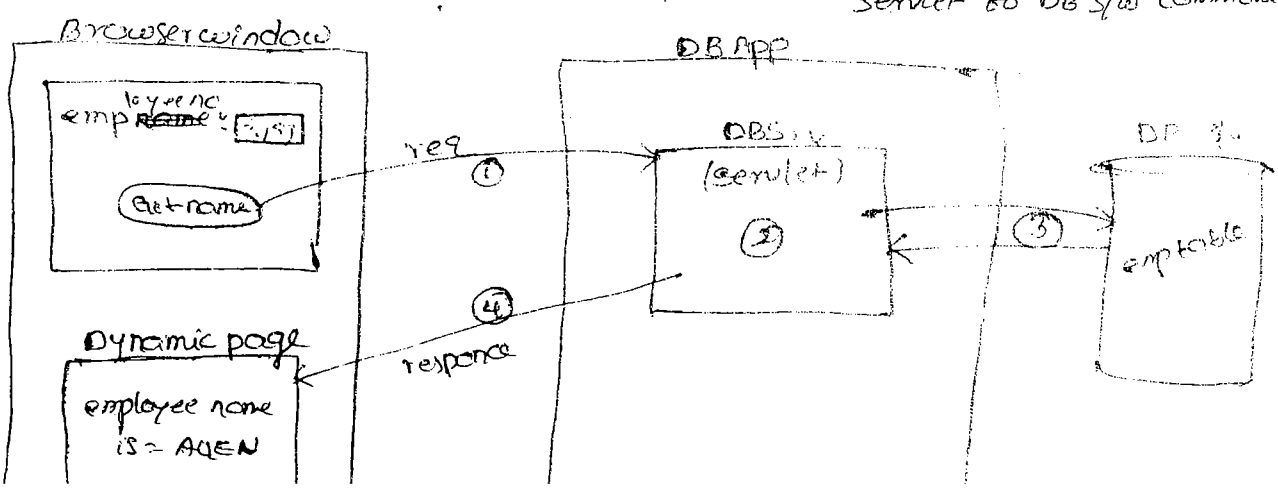
→ JDBC connection pool is a factory that contains readily avalable JDBC connection object

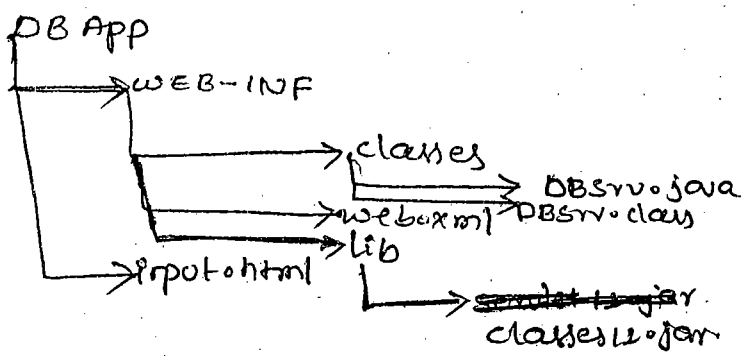
Advantages:- Here JDBC connection object is local variable of service(-,-), so it is thread safe object by default. since JDBC connection pool allows to work with less amount of JDBC connection object for more request given by more client performence will be quite good.

→ programmer need not to create manage & destroy JDBC connection object manually.

note: most recomanded approch for servlet to DB communicatio is approch no:3

→ To implement following Diagram. Cosing approch ② for servlet to DB s/w communicat





→ If Servlet webresource of webApplication uses third-party API (other than J2SDK api, servlet api) the api related jar files must be add in the class path and also in WEB-INF/lib folder of webapplications staging directory.

→ jar files added in the classpath will be used by java compiler to recognise 3rd party api during the compilation of servlet api

→ jar files added in the WEB-INF/lib folder will be used by webserver to recognise and use 3rd party api during the ~~compilation~~ execution of servlet web resource.

→ A servlet webresource compilation happens from command prompt and execution takes place from the web-server.

→ If a servlet web resource of web application uses oracle thin driver the thin driver related classes12.jar file (or) classes111.jar (or) ojdbc14.jar file must be added in the class path and and also in WEB-INF/lib folder of web application.

```
<html>
    <b> input.html </b>
    <form action="dburl" method="get">
    <b> Employee Number </b> <input type="text" name="teno" /> <br>
      <input type="submit" value="Get name" />
    </form>
</html>
```

DBSrv.java

```
// DBSrv.java
// package P1;
import java.io.*;
import java.sql.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class DBSrv extends HttpServlet
{
    connection con=null;
    public void init();
    {
        try
        {
            class.forName("oracle.jdbc.driver.OracleDriver");
            con=DriverManager.getConnection("jdbc:oracle:thin:
                @localhost:1521:satya","scott","tiger");
        }
        catch(Exception e)
        {
            e.printStackTrace();
    } //init()
```

```
{ try
  {
    // read from data
      int no = Integer.parseInt(req.getParameter("eno"));

    // write JDBC code
      Statement st = con.createStatement();
      ResultSet rs = st.executeQuery("select ENAME from
                    emp where empno = "+no);
      String name = null;
      if(rs.next())
      {
        name = rs.getString(1); // name = rs.getString("ename");
      }

    // write result to Browser as response
      PrintWriter pw = res.getWriter();
      res.setContentType("text/html");
      pw.println("<b> employee name is "+name+"</b>");
      pw.close();
  }
  catch(Exception e)
  {
    e.printStackTrace();
  }
} // service
public void destroy()
{ try
  {
    con.close();
  }
  catch(Exception e)
  {
    e.printStackTrace();
  }
} // destroy
```

```
<web-app>                    url pattern.
    <servlet>
        <servlet-name> abc </servlet-name>
        <servlet-class> DBSrv </servlet-name>
    <servlet>
    <servlet-mapping>
        <servlet-name>abc </servlet-name>
        <url-pattern>/dburl </url-pattern>
    </servlet-mapping>
</web-app>
```

=

→ ~~when servlet source code is~~

→ How to configure a servlet when it is placed in a package?

(A) :- step1 compile the servlet source file by using javac (space) -d option (syntax -d) ex: javac -d . *.java

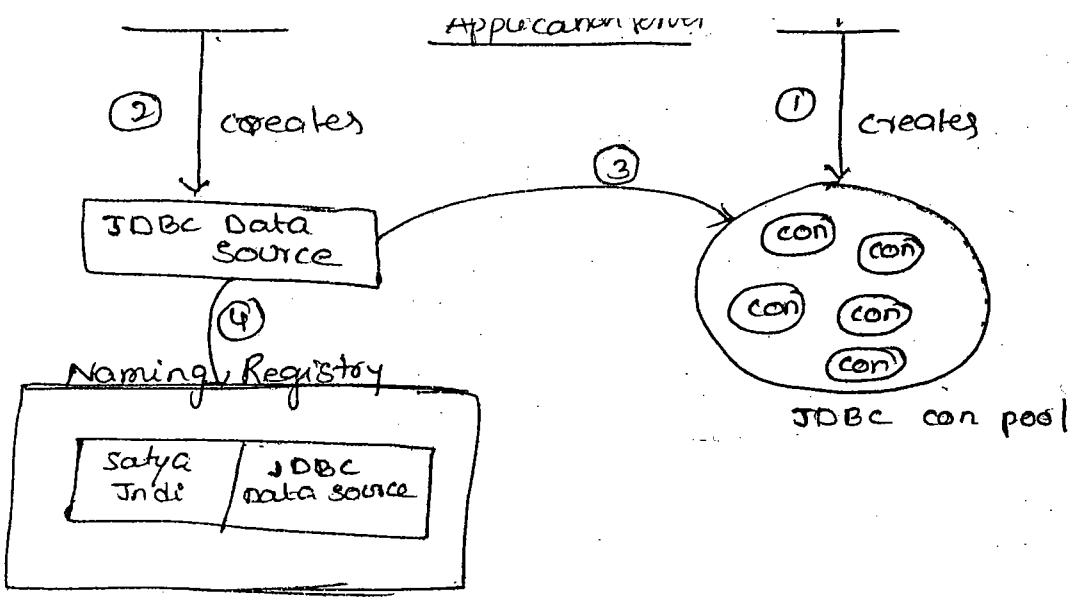
ex:- >javac -d . ~~DBSrv~~ DBSrv.java

→ configure servlet in web.xml along with the package name

```
<servlet>
    <servlet-name> abc </servlet-name>
    <servlet-class> p1.DBSrv </servlet-class>
</servlet>                    ↑ package name
```

19/12

→ Every jdbc connection pool will be represented jdbc datasource object, to get global visibility for jdbc data source object that will be registered with naming registry service, every java application should use jdbc datasource obj to get access to jdbc connection obj available in the jdbc connection pool.



→ when the Jdbc data source obj is registered with naming registry, identified w/r nickname / Alies name of Jndi name (satya Jndi in the above diagram)

- Procedure to create JDBC Data source Representing JDBC connection pool of oracle in weblogic 9.x Example Server domain!

Step no:1 :- start ~~the~~ Example weblogic server domain of weblogic 9.x and open the admin console

start
→ programs
→ bea products
→ Examples
→ weblogic Server
→ start examples server
→ open browser window

type the url : http://localhost:7001/console
↳ user name: weblogic
password : weblogic

pool for oracle

select * globle-name

Admin console
→ lock & edit
→ services
→ JDBC
→ data sources
→ new

name: (logical name) = mydb

jndi name = satya-jndi

data base type = oracle

DB Driver = Oracle's thin Driver

→ next
→ next

DB name: satya — SID name

Host name: localhost

port-num: 1521

username: scott

password: tiger

conform password: tiger

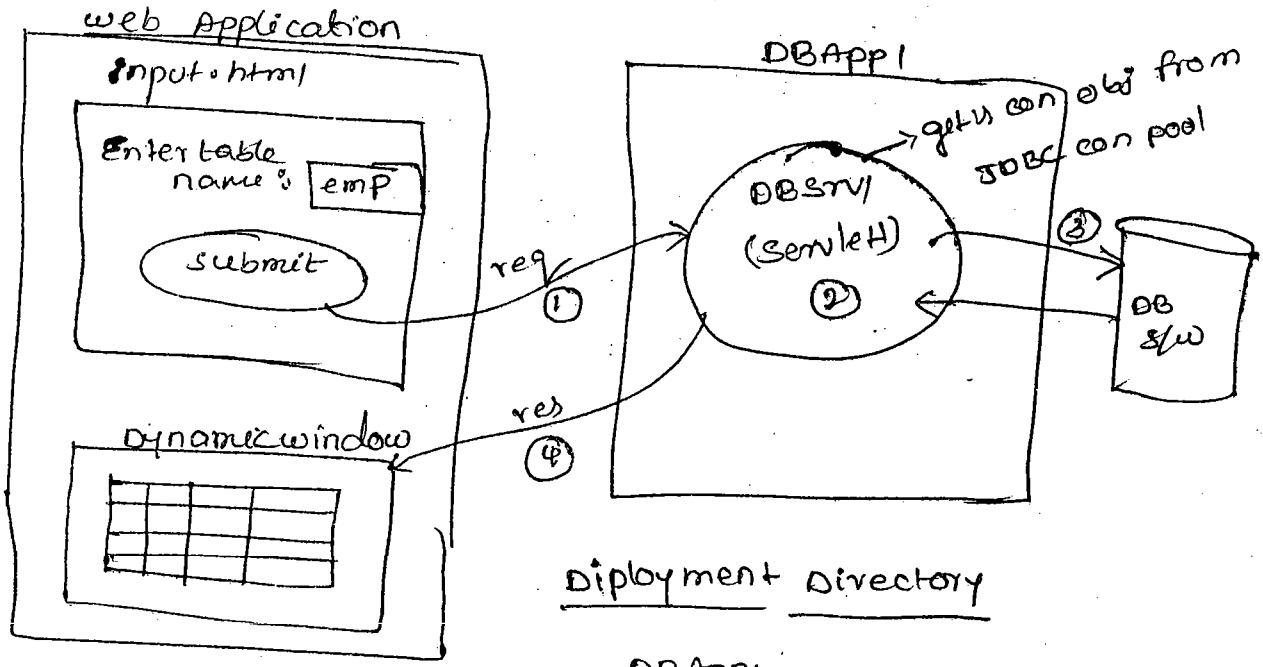
→ next
→ next
↓
select examples server
→ finish
→ Activate changes

note :- ① when the Activate changes button is clicked the JDBC data source object will be registered with naming registry service of weblogic having jndi name satya jndi.

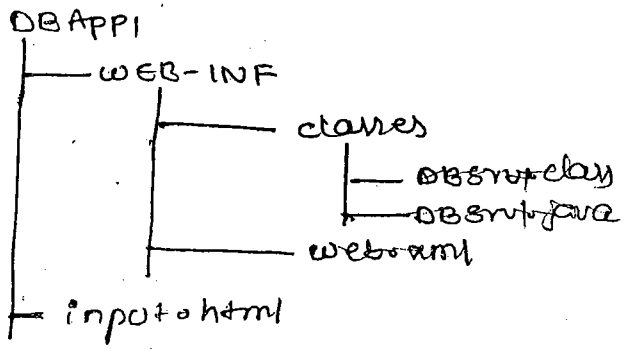
→ the Initial context obj of jndi programming represents connectivity with naming registry s/w and allows free programm

it's nickname from naming registry s/w.

→



Diployment Directory



→ The above application displays Given DB table records in the form of html table.

Since above application is using JDBC connection object of JDBC connection pool created & managed by under laying web logic server therefore no need of keeping in WEB-INF/lib folder of web application.

→ The above application should be run created above in web logic server, so this application must be deployed in weblogic server.

input.html

```
<b>input.html</b>
Enter table name
<form action="dbsrv1" method="get">
Enter table name <input type="text" name="table" /><br>
<input type="submit" value="submit" />
</form>
```

configar DBSrv1 servlet having url pattern ~~DB~~dburl

```
<web-app>
    <s>
      <s-n> abc </s-n>
      <s-c>DBSrv1 </s-c>
    </s>
    <s-m>
      <s-n> abc </s-n>
      <url-pattern>/dburl </url-pattern>
    </s-m>
</web-app>
```

d DBSrv1.java

```
import java.io.*;
import java.sql.*;
import javax.servle.*;
import javax.servlet.http.*;
import javax.naming.*; // for jndi pool
import javax.sql.*;

    // following approch:3 for Servlet to DB s/w communicati

public class DBSrv1 extends HttpServlet
{
    public void service(HttpServletRequest req, HttpServletResponse
                        res) throws ServletException, IOException
    { // PrintWriter pw=null;
      try
      { PrintWriter pw = res.getWriter()
        res.setContentType("text/html");

        // read from data
        String tabname = req.getParameter("table");

        // create initial context object
        InitialContext ic=new InitialContext();
```

```
DataSource ds = (DataSource) ic.lookup("satya-jndi");

// write JDBC code.

Connection con = ds.getConnection(); // use one connection obj from con pool
Statement st = con.createStatement();
ResultSet rs = executeQuery("select * from "+tabname);

ResultSetMetaData rsmd = rs.getMetaData();

// Display result set obj data is html table data

int colcnt = rsmd.getColumnCount();

pw.println("<table border='1'><tr>");

for(int i=1; i<=colcnt; ++i)
    pw.println("<th>"+rsmd.getColumnLabel(i)+"</th>");
    pw.println("</tr>");

while(rs.next())
{
    pw.println("<tr>");

    for(int k=1; k<=colcnt; ++k){

        pw.println("<td>"+rs.getString(k)+"</td>");
    } // for
    pw.print("</tr>");
} // while

pw.println("</table>");

// need con obj back to jdbc con pool
con.close();
catch(Exception e)
{
    pw.println("<center>"+table doesnot exist+"</center>");
} // catch

} // service class
```

Domain of web logic copy that DBAppl folder to

- beaHome/weblogic90/samples/domains/wl_s.
server/auto Deploy folder.

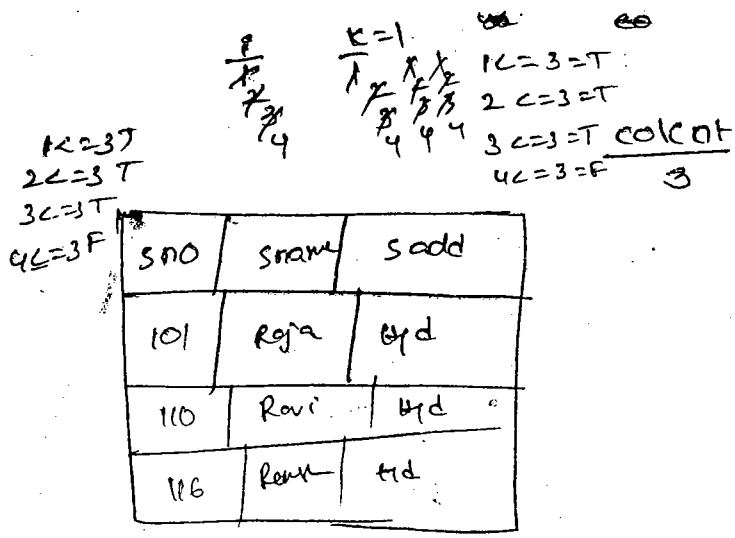
BeaHome/weblogic90/samples/domains/WL-Server/autoDeploy folder.

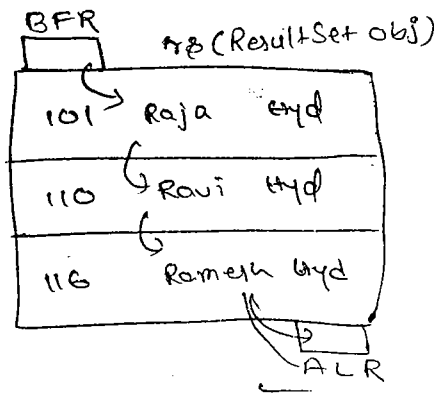
note:- make sure that weblogic server is in running mode while giving request to the above web Application

url for testing Application: enter to db table name: [redacted]

↓

http://localhost:7001/DBAppl/input.html (Get records)





20/12

→ In approach:1, approach:2 of Servlet to DB communication we manually place code to create JDBC connection object, to make this code as flexible code it is recomanded to collect driverclassname, url, DB username, DB password, Details of that code from out side the Servlet as initial parameters of the Servlet.

→ To make JDBC code of DBSrv servlet as flexible code use init-cs parameters as shown bellow.

ore technical input values for servlet we need to pass them as servlet init() parameter values (or) context parameter values of web application.

code in web.xml

```
<web-app>
    <s>
     <s-n> abc </s-n>
     <s-c> P1.DBSrv1 </s-c> // <s-c> P1.DBSrv1 </s-c>
    </s>
     <init-param>
        <param-name> driver </param-name>
        <param-value> sun.jdbc.odbc.JdbcOdbc
                      Driver </param-value>
     </init-param>

     <i-p>
        <p-n> url </p-n>
        <p-v> jdbc:odbc:oradsn </p-v>
     </i-p>
     <i-p>
        <p-n> dbuser </p-n>
        <p-v> scott </p-v>
     </i-p>
     <i-p>
        <p-n> dbpwd </p-n>
        <p-v> tiger </p-v>
     </i-p>
    </s>
    <s-m>
        <s-n> abc </s-n>
        <url-pattern> /dburl </url-pattern>
    </s-m>
</web-app>
```

```
public void init()
{ try
  {
    ServletConfig cg = getServletConfig();
    // read init parameter values
      String s1 = cg.getInitParameter("driver");
      String s2 = cg.getInitParameter("url");
      String = s3 = cg.getInitParameter("db user");
      String s4 = cg.getInitParameter("dbpwd");

      class.forName(s1);
      con = DriverManager.getConnection(s2, s3, s4);
}
catch(Exception e)
```

same as

note:- Inorder to pass technical input values to servlet from outside the servlet take the support of servlet initParameter (or) context parameter of web application. inorder to pass non-technical input values to servlet from out side the servlet take the support of request ~~of~~ parameters.

→ If multiple servlets of a web application want to make ~~JDBC~~ their JDBC code as flexible code instead of writing same init parameter in every servlet configuration it is recomanded to configar JDBC Driver details as context parameters only for one time and use them in multiple servlets of a web-application.

```
<web-app>
    <context-param> driver </context-param>
        <p-n> driver </p-n>
        <p-v> sun.jdbc.odbc.JdbcOdbcDriver </p-v>
    </context-param>

    <c-p>
        <p-n> url </p-n>
        <p-v> jdbc:odbc:oradsn </p-v>
    </c-p>
    <c-p>
        <p-n> dbuser </p-n>
        <p-v> scott </p-v>
    </c-p>
    <c-p>
        <p-n> dbpwd </p-n>
        <p-v> tiger </p-v>
    </c-p>

    <s>
        <s-n> abc </s-n>
        <s-c> DBSrv </s-c>
    </s>
    <s-m>
        <s-n> abc </s-n>
        <url-pattern> /dburl </url-pattern>
    </s-m>
</web-app>
```

→ In any servlet of webapplication that is configured in web.xml write following code based on contextparameter to create jdbc connection object.

```
public void init()
{
  try
  {
    ServletConfig cg = getServletConfig();
    ServletContext sc = cg.getServletContext();
   // Read init parameter.
    String s1 = sc.getInitParameter("driver");
    String s2 = sc.getInitParameter("url");
    String s3 = sc.getInitParameter("dbuser");
    String s4 = sc.getInitParameter("dbpwd");

    class.forName(s1);
    con=DriverManager.getConnection(s2, s3, s4);
  }
  catch(Exception e)
```

save as

→ If multiple web application deployed in the tomcat server are using same jdbc Driver to interact with DB s/w instead of placing the driver related jar file in WEB-INF/lib folder of every web application it is recommended to place that jar file to:

tomcat-home/common/lib folder (or)
tomcat-home/server/lib folder.

→ Ex:- if multiple webapplication diployed in tomcat uses Server* oracle thin driver to interact with DB s/w instead of placing the classes12.jar (or) classes11.jar (or) ojdbc14.jar fil

tomcat-home/ common/ lib folder

(or)

tomcat-home/ server/ lib folder for global visibility

→ when web resource of web application uses 3rd party API (other than J2SDK, servlet, JSP - API's) then during the execution of web application/web resources the server searches for new API classes & interfaces in following places having a sequels.

① In classes kept in WEB-INF/ classes folder of web application

② In jar files kept in WEB-INF/ lib folder of web application

③ In jar files kept in tomcat home/ common/ lib folder of ~~web application~~ tomcat Server

④ In jar files kept in tomcat-home/ server/ lib folder of tomcat server.

→ when new class or 3rd party API class is available in one of the above 4 places server uses that class and doesn't search in further phases.

JBOS

- type: Application Server
- version: 4.x (J2SDK 1.4, 1.5) 5.x (J2SDK 1.5)
- vendor: apache foundation
- Default port no: 8080
- open source s/w
- for help & Download s/w: www.apache.org.
- to install s/w: s/w common as zip file so extract the zip file to folder.

→ JBOS application server contains tomcat server as built in server as web container.

→ to change port no: of JBOS server depend upon server.xml file (same as tomcat server).

→ To start JBOS Server use

JBOS Home/bin/run.bat file

→ In JBOS Server

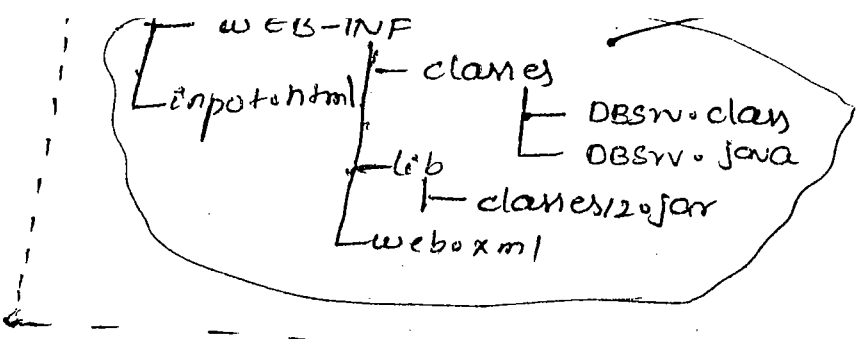
JBOS Home/client/JBOS-J2EE.jar file represents J2EE-API

→ In JBOS server console deployment is not posible but only war file base hoard deployment is posible. (hoard deployment without prepoaring war file is not posible)

→ Procedure to deploy a Java webApp in JBOS 4.x AppServer

step1:- Generate war file on staging directory of webApplicati

DocApp/WEB-INF/classes



E:\DBApp> jar cf DBApp.war *

(2)

→ Start the JBOS Server.

JBoss Home/bin/run.bat file

: Delete 82 to 84 line no: content of run.bat file

3 → Deploy webApplication in JBoss Server

copy the above Genarated DBApp.war file to JBossHome/server/default/deploy folder

4 → Test the web application

open browser window : http://localhost:6060/DBApp/input.html

=

Note :- when webApp Deployed in Jboss Server in the form of war file the file name of war file will act as context path of webApplication.

→ weblogic is the popular Server in commercial Server s/w

→ Jboss is the popular Server in free Server s/w

note :* the technologies to utilize in project development will be desided the project manager(or) project Architech

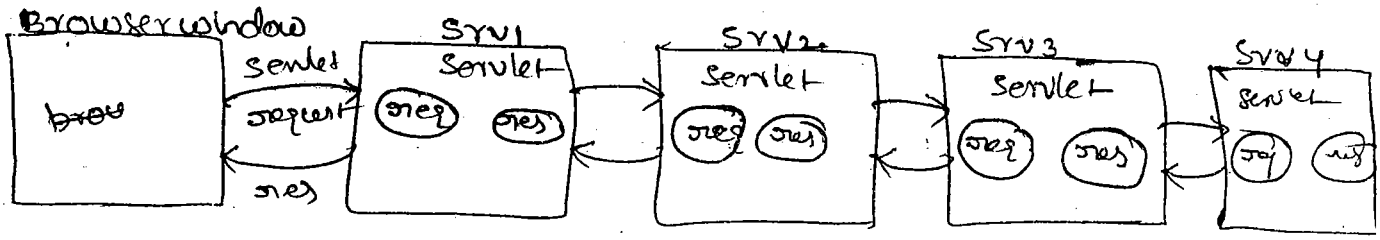
=

→ In all previous Applications we process the request given by a browser by using single servlet, some times we need process request given by a browser by using multiple servlet as a chain.

→ Process of getting request from a browser and Processing that Request with the support of multiple Servlets kept in a chain is called Servlet chaining.

→ Servletchaining Demands Communication b/w servlet

→ Servlet-chaining



→ In the above diagram request given by a browser is processed by using multiple servlet that are there in the chaining.

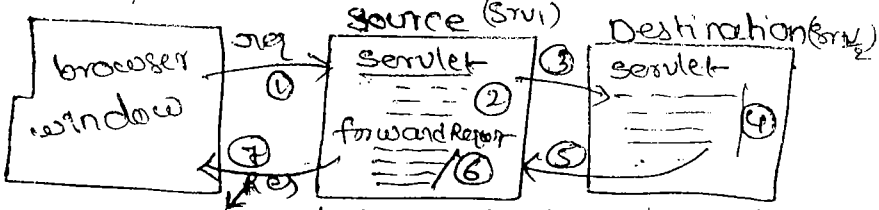
→ All the servlet that participate in servlet chaining will use same request & response objects.

Servlet chaining

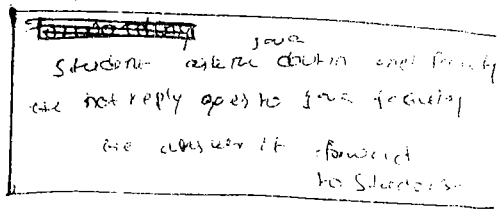
forwarding Request mode of Servlet chaining

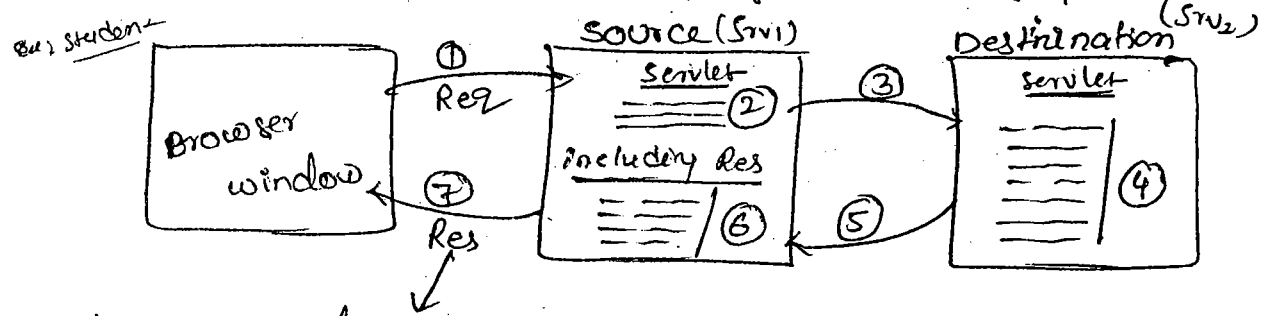
Including Responce Request mode of servlet chaining.

forwarding Req



The Html output of source servlet will be discarded





Html output of both Srv_1, Srv_2 Servlets goes to browser as Response

→ In forward req of Servlet chaining the Html output generated by source Servlet will be discarded only the html output of destination servlet goes to browser as response.

→ In including Responce mode of Servlet chaining the Html output of source & destination servlet to gathes goes to browser as response.

→ Any no: of servlet can participate in any mode of Servlet chaining & all these servlets will use same Req & Res objects.

→ Request Dispature object Means it is the object of a class that implement javax.servlet.RequestDispature Interface. the source Servlet create RequestDispature servletobj obj pointing to destination & and becomes ready to perform either forwarding Request mode of Servlet chaining (or) including Res. Response mode of Servlet chaining.

in Source Servlet of Servlet Chaining (Srv1) :-

approach:1 :- (source servlet)

```
RequestDispatchve rd = req.getRequestDispatcher("/s2url");
rd.forward(req,res);
   (or)
rd.include(req,res);
```

("/" symbol is optional)

s2url pattern of Destination Servlet (Srv2)

approach:2 :- (source servlet)

```
ServletContext sc = getServletContext();
RequestDispatchee rd = sc.getRequestDispatcher("/s2url");
rd.forward(req,res);
   (or)
rd.include(req,res);
```

"/" symbol is mandatory

s2url pattern of Destination Servlet (Srv2)

approach:3 :- (source servlet)

```
ServletContext sc = getServletContext();
RequestDispatcher rd = sc.getNamedDispatcher("s2");
```

logical name of Destination Servlet (Srv2), given in web.xml file as <servlet-name> body_value.

Q) What is the diff b/w creating RequestDispatcher obj by using Request obj & by using ServletContext obj ?

A) :- The ServletContext obj based RequestDispatcher obj can be used to perform Servlet Chaining b/w two Servlets that are residing either in the same web Application or in two diff webApp of same Servlet.

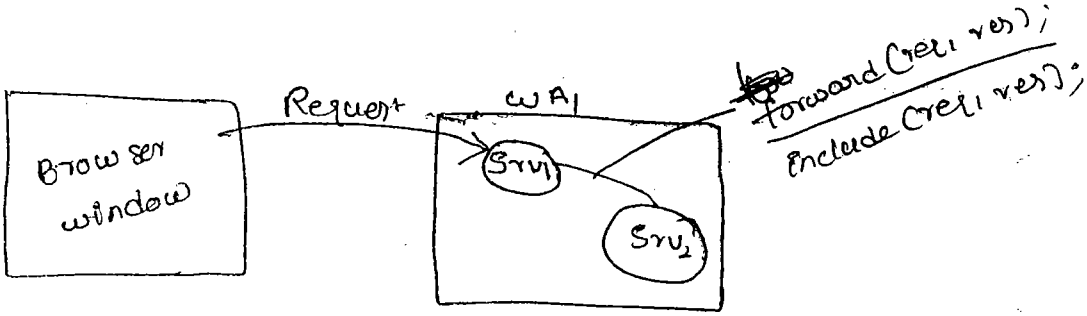
obj is use-full only to perform Servlet chaining b/w two Servlets of same webApplication.

Q→ wt is the diff b/w ~~req~~ req.getRequestDispatcher() method & sc.getRequestDispatcher() method?

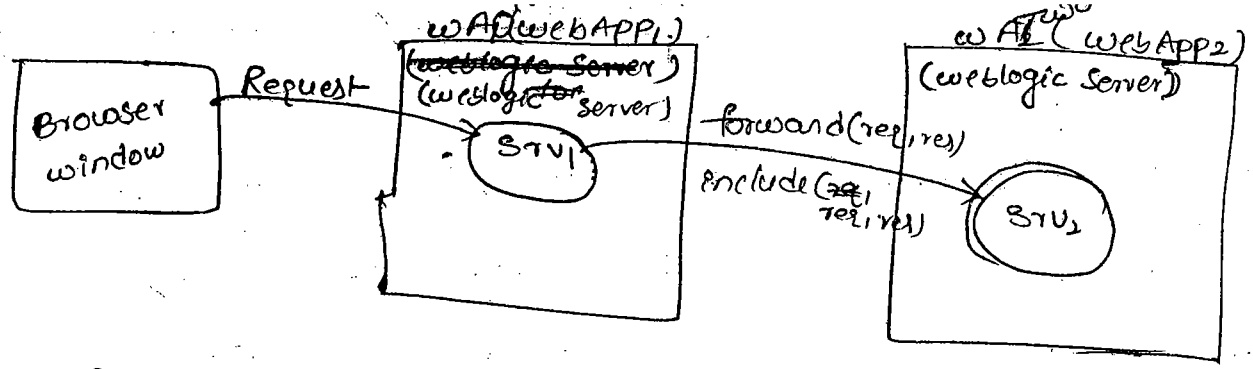
(A):- the ~~req~~ RequestDispatcher obj given by req.getRequestDispatcher method can be used only to perform Servlet chaining b/w two Servlets of same webApplication where as the RequestDispatcher obj given by sc.getRequestDispatcher method can be used to perform Servlet chaining b/w two Servlets of same (or) different webApp available in Server.

while calling req.getRequestDispatcher method using '/' symbol is optional when passing destination Servlet url pattern as argument value, where as passing '/' symbol is mandatory for same situation while working with sc.getRequestDispatcher() method.

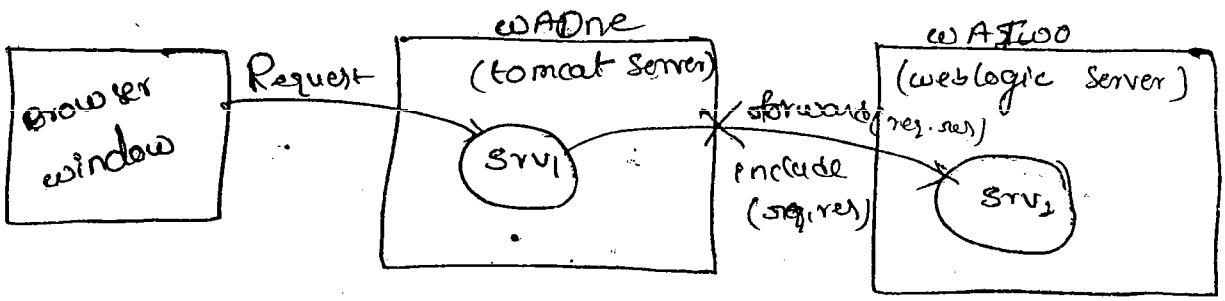
=



Srv1 Servlet can use either ServletContext request object based RequestDispatcher obj because both source & destination Servlets are there in the same webApplications.



Srv1 ~~Request~~ Servlet must use ServletContext obj based RequestDispatcher obj because, source & destination Servlets are there in two different webApplication of ~~source Servlet~~. (both that means Same server)



wrong

X ⟶ Servlet chaining b/w two servlets which reside in two different ~~Servlets~~ servers is not possible with RequestDispatcher object. (Here using two different Servers it is not possible to use Servlets)

note:- when you are using RequestDispatcher obj for ServletChaining the Source & destination Servlet must reside in the same Server belonging to either same webApp (or) two different webApp.

⟶ when rd.forward() is called in Source Servlet the source servlet performs forwarding Request mode of ServletChaining with destination Servlet.

⟶ when rd.include() is called in the Source Servlet then the source servlet performs including response mode of servlet chaining with destination Servlet

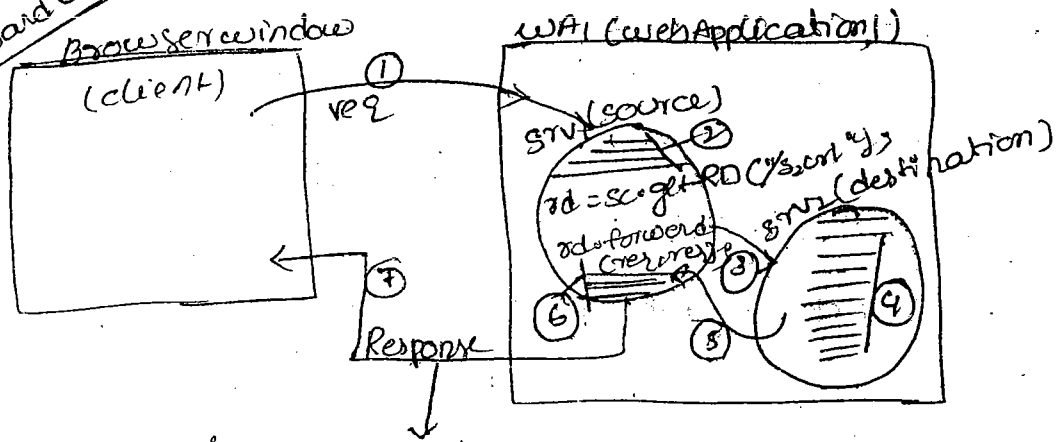
Q getNamedDispatcher() ?

(A):- getRequestDispatcher() is invoked on Request obj & ServletContext obj. this method expects url pattern of destination servlet as argument value.

- getNamedDispatcher() is ~~invokable~~ invokable only on ServletContext obj this method expects logical name of destination servlet given in web.xml file as argument file.

22/12

Understanding rd.forward(-,-)



Html output of Srv_1 Servlet will be discarded the only output Srv_2 web resource goes to browser as response

→ url pattern of ~~we~~ Srv_2 webresource is : /$s_2$url

→ Key points regarding rd.forward(-,-):-

① → Both Srv_1 & Srv_2 Servlet will use same request & response obj so the request data comming to Srv_1 Servlet is visible & accessable in Srv_2.

② → The html output of Srv_1 Servlet will be discarded only the html output of Srv_2 Servlet goes to browser as response.

③ → To pass Additional data b/w Srv_1 & Srv_2 resources use Request attributes,

file residing in the same or different web application of Srv, Servlet.

(5) → rd.forward(-,-) is very usefull while configuring error Servlet for webApplication and also usefull to make certain request processing logic of webApplication as reusable logic.

Q → what is an ErrorServlet?

(A):- The Servlet is execute when exception is raised in other main servlets of webApp is called error Servlet. error Servlet is very usefull to display exception related exception as non technical messages for endusers

Example code:

ErrSrv.java

```
public class ErrSrv extends HttpServlet
{
    public void Service(HSR req, HSR res) throws SE, IOE
    {
        PrintWriter pw = res.getWriter();
        res.setContentType("text/html");

pw.println("<font color=red> internal problem </font>");
pw.println("<a href='input.html'> home </a>");
    }  → url pattern of ErrSrv Servlet is: "/evol"
}
```

main Servlet of webApp:

```
public class 12/31--n extends HttpServlet
{
    public void service(-,-) throws SE, IOE
    {
```

```
try
{
    rd=req.getRequestDispatcher("/eorl");
        ------------------
        ------------------
        ------------------          B.logic of main Servlet
        ------------------
        ------------------(which may rised Exception)
}//try
catch (Exception e)
{
    try
    {   if(rd!=null)
        rd.forward(req, res);
    }
    catch (Exception e)
    {
        e.printStackTrace();
    }
}// catch
~~}//try~~
} // service
} // main
```

→ The output Genarated by source Servlet by using s.o.p statemets will not be discarded eventhough it is using rd.forward(-,-) to forward the request to destination webresource.

→ Inorder to make RequestDispatcher object pointing to html file or image file or text file pass filename as argument value in getRequestDispatcher(-,-) method.

RequestDispatcher rd = req.getRequestDispatcher("/abc.html");

" rd = req. " ("/flower.jpg");

" rd = req. " ("/abc.txt");

→ we can't use getNamedDispatcher(-,-) to create RequestDispatcher obj that points to html file, jsp file, image file, txt file etc (non-Servlets)

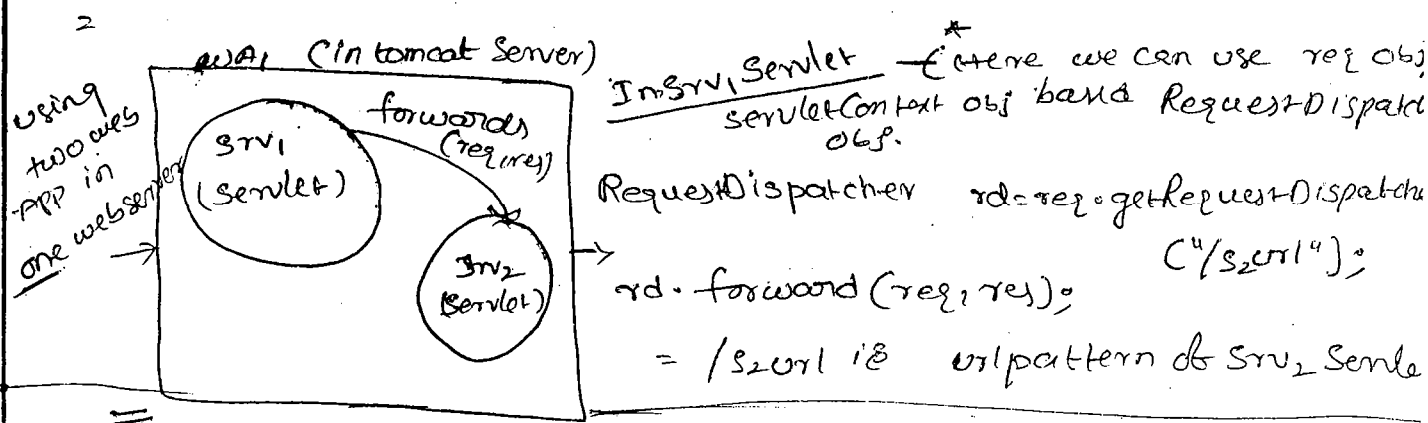
→ If Destination web resource commit the response by calling pw.close() when Source Servlet forward request to Destination Servlet by using 'rd.forward(-,-)', no Exception will be raised

23/12

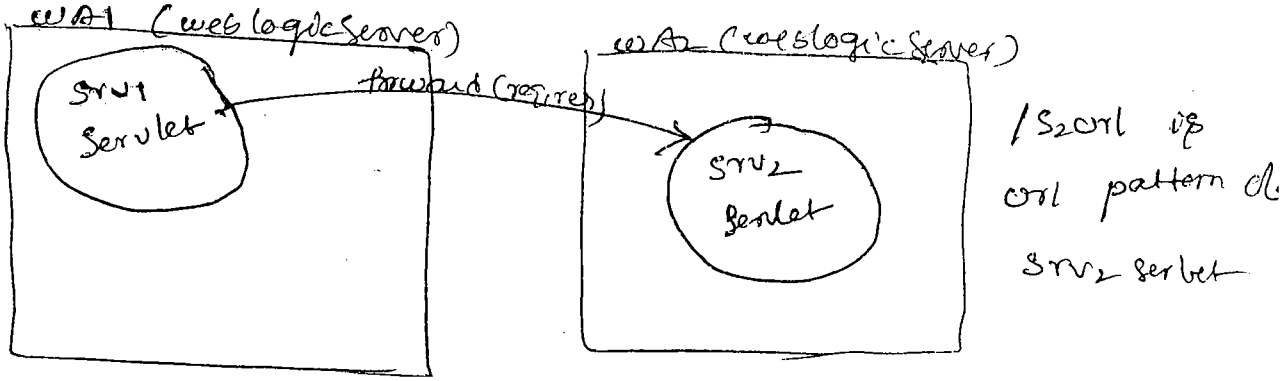
→ ErrorServlet Configuration done in the webApp helps the enduser to get exception related messages as non-technical Error messages.

→ Always develop webApp by keeping non technical, and layman endusers in mind.

2



2



Dispatcher obj;

in Srv1 Servlet of WA1 (webApplication)

→ ServletContext sc = getServletContext();

represents ServletContext obj of WA1

ServletContext sc1 = sc.getContext("/WA2");

represents ServletContext obj of WA2

RequestDispatcher rd2 sc1.getRequestDispatcher("/s2url");

rd.forward(req,res);

→ /s2url is url pattern of Srv2 Servlet

→ WA2 is pointing to WA1 Servlet in ServletContext obj(sc1)

*

→ note:- being from one webApplication to get Access to ServletContext obj of another webApplication we need to use getContext(-) by passing destination webApplicaion name as argument value as shown in the above code.

→ when Servlet is executed in the Server

ⓐ Genorated output will be stored in the buffer of servlet at server side.

ⓑ Output comes to that browser which has given request to the Servlet.

ⓒ Output will be store in the buffer of the browser

ⓓ Output will be displayed on the browser window.

Source & destination Servlet responce is flushed from the buffer by calling res.flushBuffer(-), before- calling rd.forward(-) (or) rd.include(-) in Source Servlet then these methods will raise java.lang.IllegalStateException

→ res.flushBuffer(-) forces the buffer of the Servlet sending it's output data to browser.

```
res.flushBuffer();
rd.forward(req,res);
```

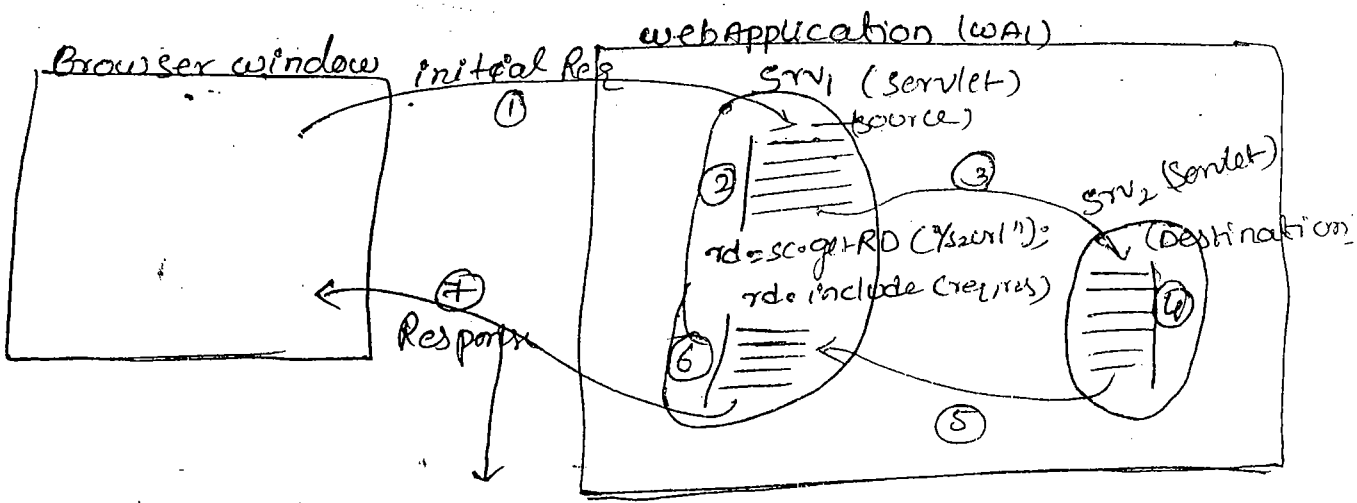
* → Servletchaining b/w two servlets belonging to two diff webApp of same server is not possible in tomcat, JBOS servers because these servers have not provided proper implementation for getContext(-,-) (or) ServletContext interface but the above type of Servletchaining is posible in weblogic, websphere & glassfish servers.

(Sri)

** → use of rd.forward(-,-);

→ configuring errorServlet in mainServlet of webApplication. we are using rd.forward(-,-)

→ Understanding rd.include(-,-) Method:-



html output given by both SRV1, SRV2 Servlets goes to browser

① → rd.include(-,-) is given to perform including response mode of Servlet Chaining.

② → Both source & destination servlets use same req & res obj so req data comming to Source Servlet is visible & accessable in destination Servlet.

③ → The Response generated by Srv_1 Servlet Contain output of both Srv_1, Srv_2 Servlets.

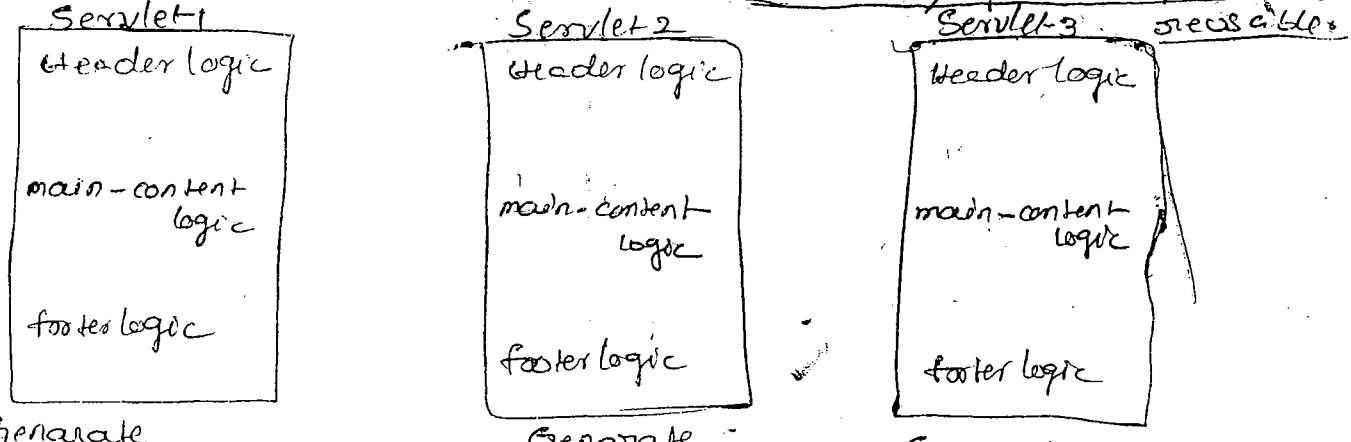
④ → Inorder to pass additional data b/w Srv_1 & Srv_2 Servlets use request attributes.

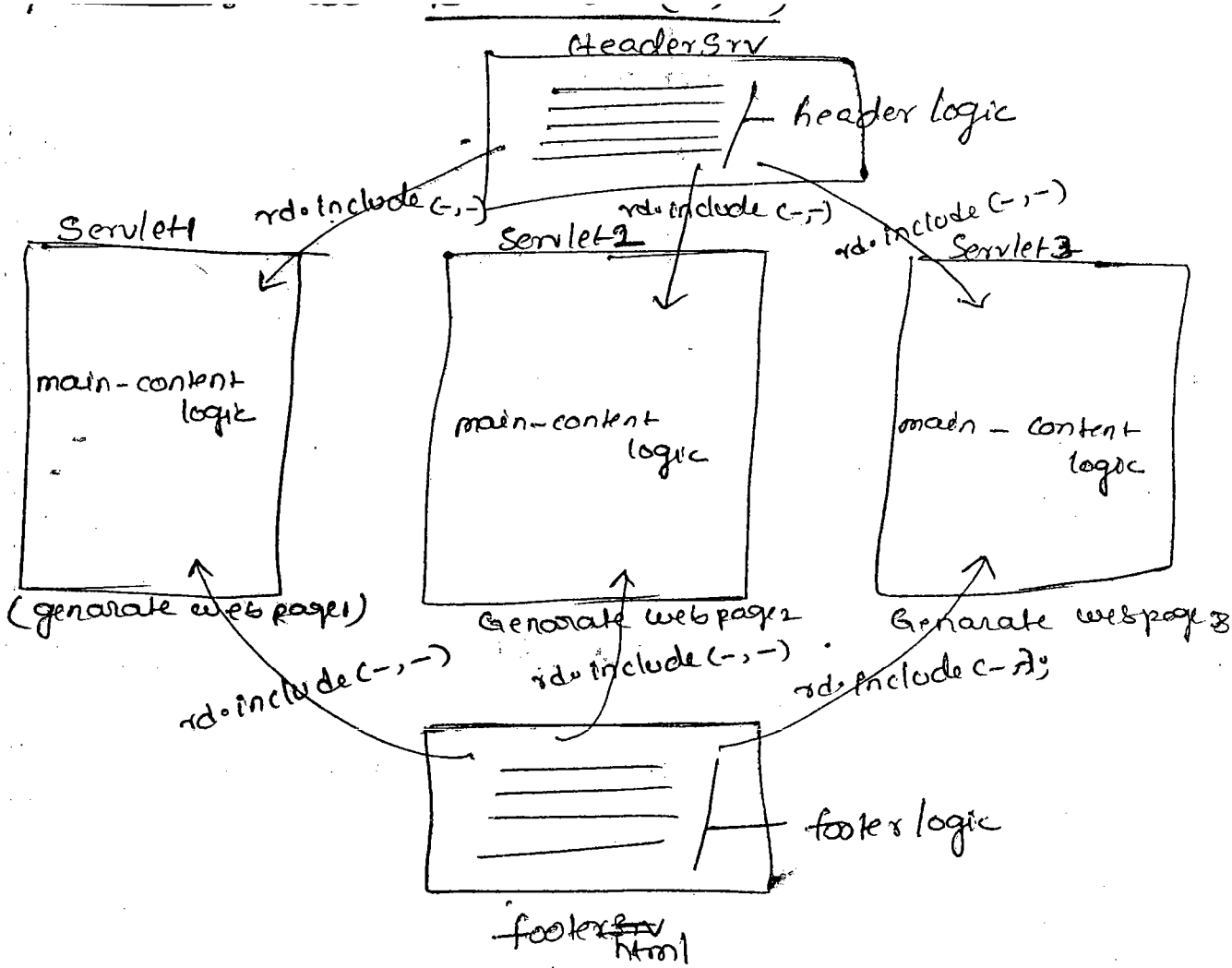
⑤ → Srv_2 can be Servlet or JSP, or html file residing in the same webApp or diff webApp of same Server where Srv_1 resides.

⑥ → The html output of Srv_2 will be included in the html output of Srv_1 in the place where rd.include(-,-) is called in Srv_1 Servlet.

⑦ → *** rd.include is very usefull to make common logics of multiple webresources like "header, footer logics has reusable logics in the webApplication".

Problem:- All the webpages of webApp contain same Header, Footer content. but "clear Header logic & Footer logic are not reusable.





→ HeaderSrv is Dynamic.

footerSrv is static.

25/12

Q → How to make one Servlet of webApp interacting with multiple Servlet/ web resources?

A:— In the source Servlet create multiple RequestDispatcher object pointing to multiple other web resources of webApp & call forward (or) include methods on these objects to communicate with other ~~web~~ multiple webresources.

Code implementing Solution Diagram :—

HeaderSrv.java (Servlet)

```
public class HeaderSrv extends GenericServlet
{
    public void service(-,-) throws SE, IOE
    {
```

```
PrintWriter pw=res.getwriter();
res.setContentType("text/html");
pw.println("<center><font color=red> sattya tech </font>
                                         </center>");
pw.println("<br><br> <br>");
pw.println("<hr>");
} // service
} class
```

<u>url pattern of HeaderSrv is : / ~~header~~ headurl</u>

<u>Footer.html</u>

```
<b> <center> &copy: copy right received 2007-2008
                                   </center></b>
```

<u>main servlet</u> :-

```
public class TestSrv 1/2/3 - - n extends HttpServlet
{
    public void service(-, -)
    {
        RequestDispatcher rd1=null; rd2=null;
        try
        {
            rd1 = req.getRequestDispatcher("/headurl");
            rd2 = req.getRequestDispatcher("/footer.html");
            rd1.include(req, res);
            - - - - - - - -
            - - ----------- b.logic that generates
                header logic
            - - - - - - - - main content of
            - - - - -------       webpage.

            rd2.include(req, res);
                - Footer logic
        }// try
        catch (Exception e)
        {
            e.printStackTrace();
        }
```

rd.forward(-), rd.include(-) method to communicate with two different web resources the effect of rd.include will not be there because rd.forward method performes forwarding request mode of Servlet Chaining and discards the entair output generated by SourceServlet.

→ when a Servlet looking to include responce or output of another webresource of webApp don't commit responce in another webresource by calling pw.close(); .. be

→ For Example Application on a webApp that uses both rd.forward(-), rd.include(-), concepts refer suplamentary hand our pages given on 25/12.

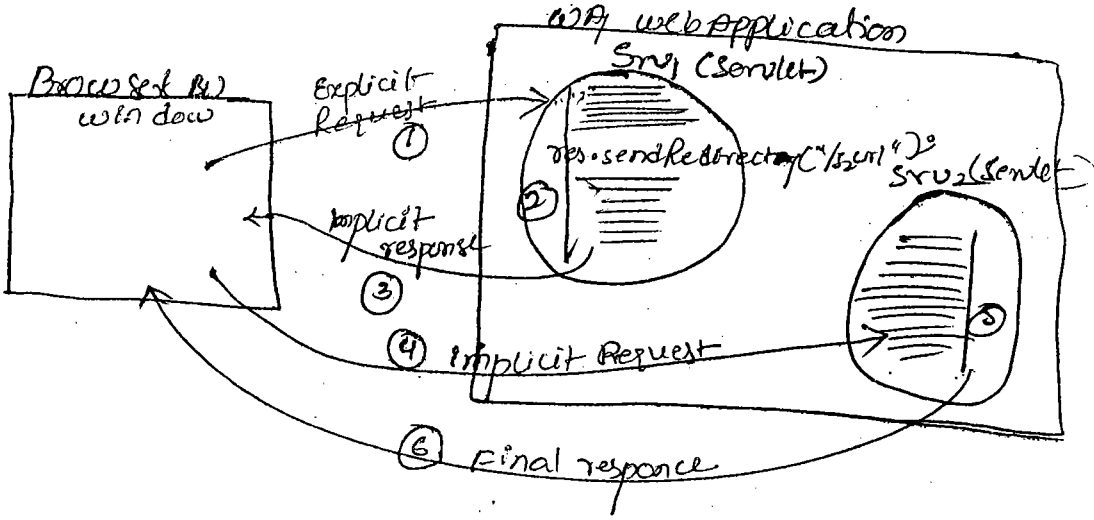
**

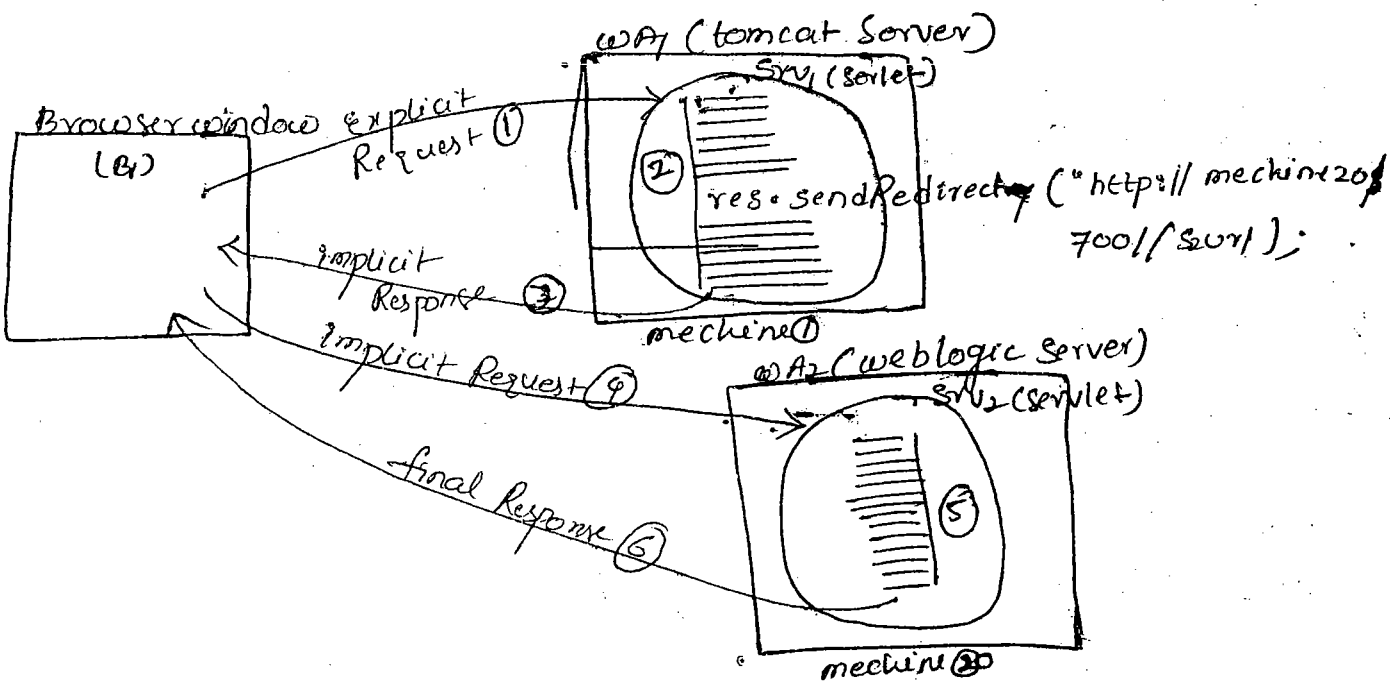
→ what is the diff blw rd.forward(-), rd.include(-) methods?

Ⓐ:- rd.forward(-) is given to perform forwarding request mode of servletchaining, rd.include(-) is given to perform including responce mode of Servlet chaining for related differences refer diagram/keypoints both rd.forward(), rd.include(-).

=

→ rd.forward(-), rd.include(-) methods can't show communi-cation blw Servlet & another web resource with reside in two different Servers, to solve this problem use send redirect by using ~~response.sendRedirect(-)~~ res.sendRedirect(-);





⑥ final response :- html output of Srv_1 will be discarded the only html output of Srv_2 Servlet goes to browser as Response.

→ with Respect to the Diagram no: ① → Browser window gives Explicit Request to Srv_1 Servlet

② Srv_1 Servlet execute all the statement including res.sendRedirect(-) method

③ because of res.sendRedirect(-) execution Srv_1 Servlet sends implicit responce to browser having url of Destination Srv_2 Servlet, the statuscode of this response will be in b/w 300 to 399

④ Browser user url of implicit responce and generate

(5) All the statements in Srv_2 Servlet executes ~~(6)~~ ,

(6) Response goes to browser from Srv_2 Servlet containing only html output of Srv_2 Servlet, that means the Html output generated by Srv_1 Servlet will be discarded. The status code of this response in b/w 200 to 299.

Key points :-

→ Srv_1 Servlet communicate with Srv_2 Servlet after having n/w round trip with browser.

→ Srv_1 & Srv_2 will not use same request & response obj- so the request data comming to Srv_1 Servlet can't be accessed from Srv_2.

→ Srv_2 can be any webresource developed by using any web technology like Servlet, JSP, ASP, ASP.net, html & etc.

→ Srv_1 & Srv_2 can reside in the same or different webApp of same or different Servers.

→ when Srv_2 is local to Srv_1 Servlet pass relative url in ~~response~~ res.sendRedirect(-) other wise pass absolute url.

→ we can't use request attributes to pass additional data b/w Srv_1 & Srv_2 webresources.

26/12

→ rd.forward is usefull to forward the control b/w two web resources which reside either in the same webApp or two different webApp of same Server.

→ rd.include is usefull to include output of one web resource in another webresource which reside in the same webApp or two diffs webApp of same Server.

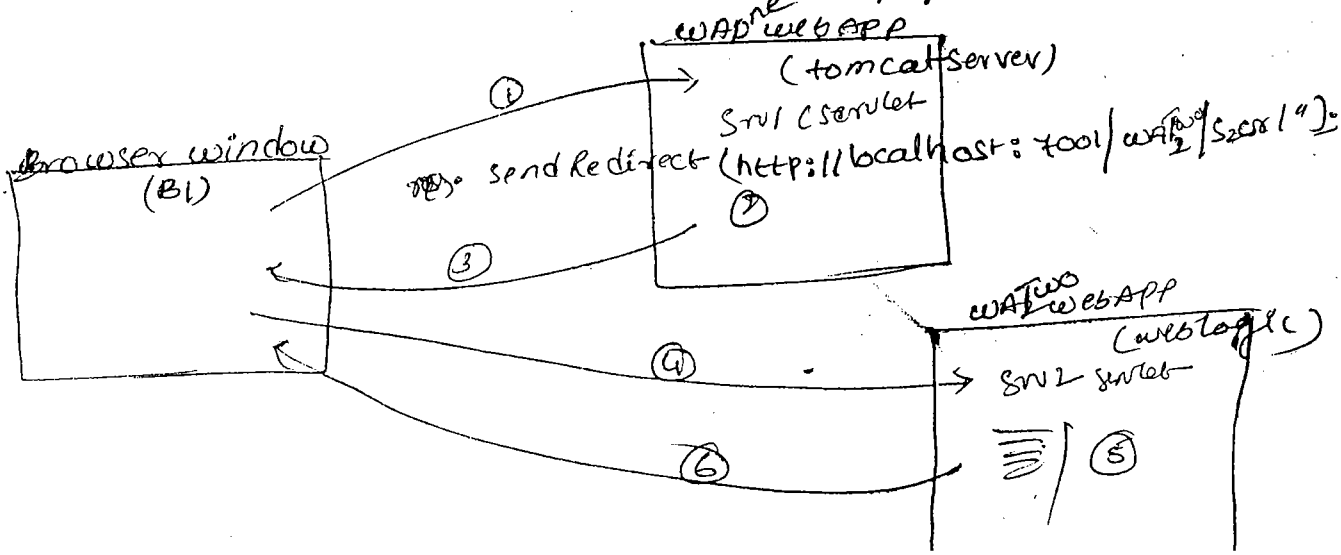
control b/w two web resources which reside in the same webApp or two diff webApp of same server (or) two different servers.

→ note: there is no provision given to include the output of remote webresource in another web resource.

** → when one company ~~owns~~ purchases another company the request comming to first company website should be redirected to certain webresources of second company website in these situation SendRedirection is very useful if the source webresource is java web resources then it uses res.sendRedirect(-) for this SendRedirection purpose.

Ex company IBM as purchased the company rational so the request comming to rational.com will be redirected to certain webpage of IBM.com

the company sunmicrosystem as purchased company "seebeyond" so the request comming to www.seebeyond.com will be redirected to certain webpage of www.sun.com.



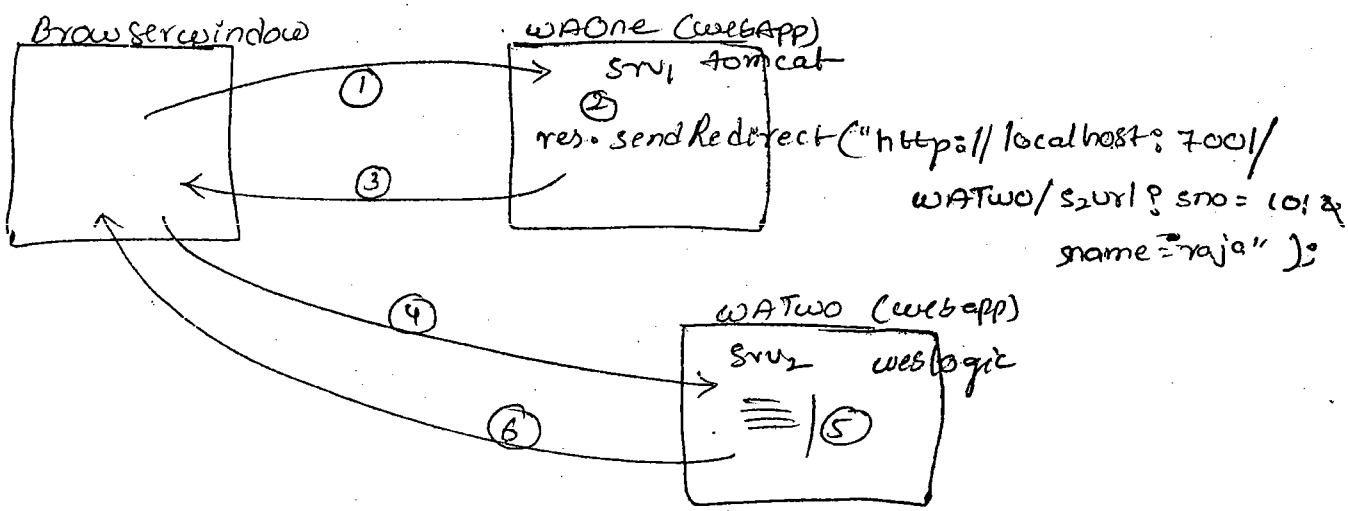
```
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*

public class srvone.extends HttpServlet
{
    public void service(HSR req, HSR res)throws SE,IOE
    {
        PrintWriter pw = res.getWriter
        res.setContentType("text/html");

        pw.println("<br> html output from srv1 (before res.sendRD (-) </br> ");
        system.out.println("from servletone(before sendRD(-) ");
        res.sendRedirect("http://localhost:7001/WATwo/s2url");
        pw.println("<br> html output of srv1 (after res.sendRD(-)</br>");
        system.out.println("from srv1 (after sendRD(-)");
    }
}
```

srvTwo.java

```
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;
public class srvTwo extends HS
{
    p v service(HSR req, HSR res)throws SE, IOE
    {   printwriter pw=res.getwriter(); res.setContentType("text/html");
        pw.println("<br> html output from srv2 </br> ");
        system.out.println("output from srv2");
        pw.close();
    }
}
```

Servlet (remote or local) ~~to~~ by using res.sendRedirect(-) can you tell me how to pass data b/w source & destination Servlets.

(A):- In the source servlet append the query string representing the data to the url that is passed in res.sendRedirect(-) method. as shown bellow.



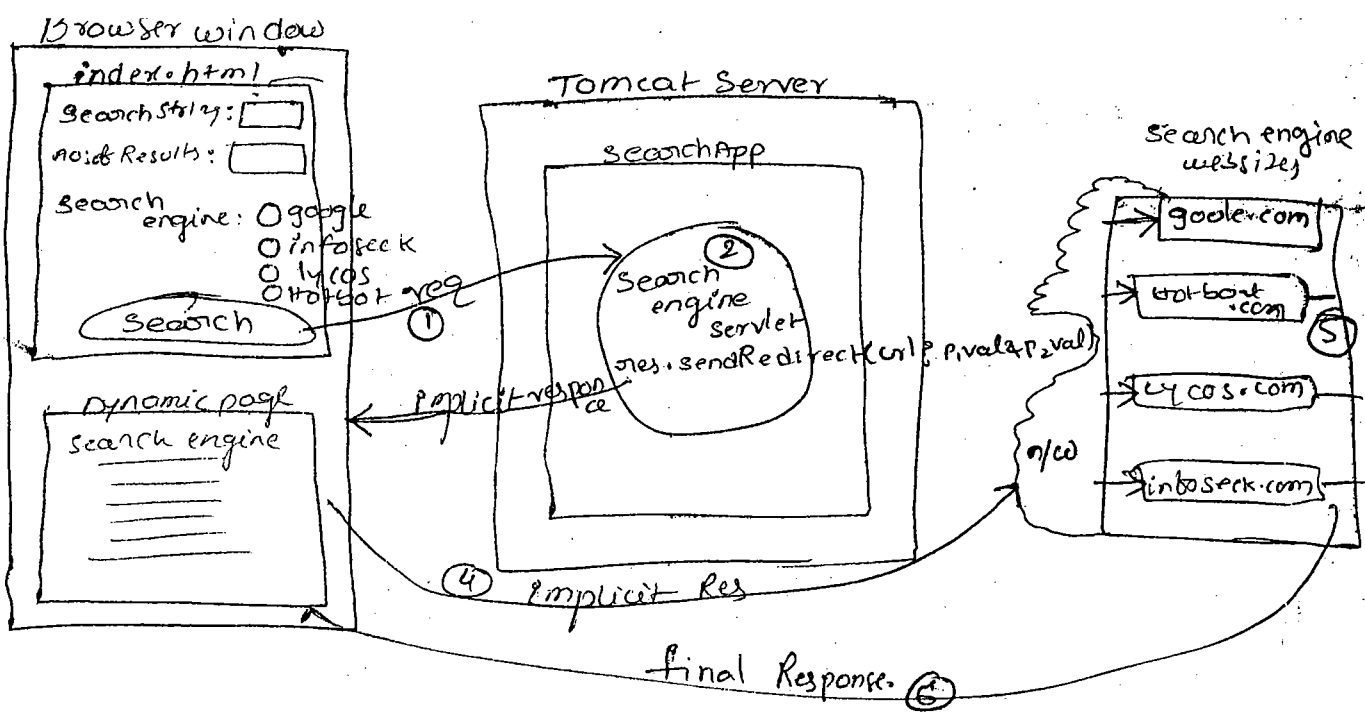
code in Srv_2 servlet to data send by Srv_1;

```
String s1 = req.getParameter("sno");
String s2 = req.getParameter("sname");
```

⟶ inorder to send input value comming to Srv_1 Servlet to the ~~remote~~ remote Srvlet Srv_2 write following code in Srv_1 servlet.

```
String s1 = req.getParameter("sno");
res.sendRedirect("http://localhost:7001/WATwo/s2url?sno="+s1);
```

rd.forward(-,-)	res.sendRedirect(-) method.
→ Given to perform forwarding request mode of servlet chaining	→ Given to perform sendRedirect based control transfer.
→ Transfer the control directly to destination web resource from source webresource.	→ Transfer the control to destination web resource from source webresource after having network round trip with browser.
→ Both source & destination webresources use same request & responce obj.	→ Dose not use same Request & response obj.
→ Both source & destination can be there in the same web App or in two diff webApp of same Server.	→ Both source & destination web resources can be there in the same webApp or in two diff webApp of same (or) diff servers.
→ the destination webresource must be java webresource (or) Html file.	→ the destination must be any technology based webresource like Servlet, JSP, ASP, PHP, Html & etc.
→ To pass adition data blw source & destination webresources use RequestAttributes	→ To pass aditional ~~use~~ datas append queryString to the url place in res.sendRedirect(-) method.
→ rd.forward is recomanded to use to transfer the control blw two local web resources	→ res.sendRedirect(-) is recomanded to use to transfer the control blw two remote web resources.
→ when rd.forward transfer the control from source to destination webresoure the url in the browser Address bar will not be changed	→ when url in the browser Address bar will be changed.



→ In the above Application the Servlet called search engine is redirecting the request to a search engine website passing additional data as query string to the url, the above application also performing server side form validation & generates form validation error having http statuscode by using res.sendError().

→ For the source code of above App refer App:5 pageno: 334, 34

Q → How many ways are there to pass data b/w Servlet webresource & another webresource?

(A):- when both webresources are there in the same webApp

1. RequestAttributes
2. Session Attributes
3. ServletContext Attributes

when webresources are there in two diff webApp of same (or) different Servers.

* res.sendRedirect(url?queryString);

→ Attribute is a String name which can hold any java obj as value. We can't store premitive values as attribute values. but we can store wrapper class obj, user defined java obj string obj as values for the attribute

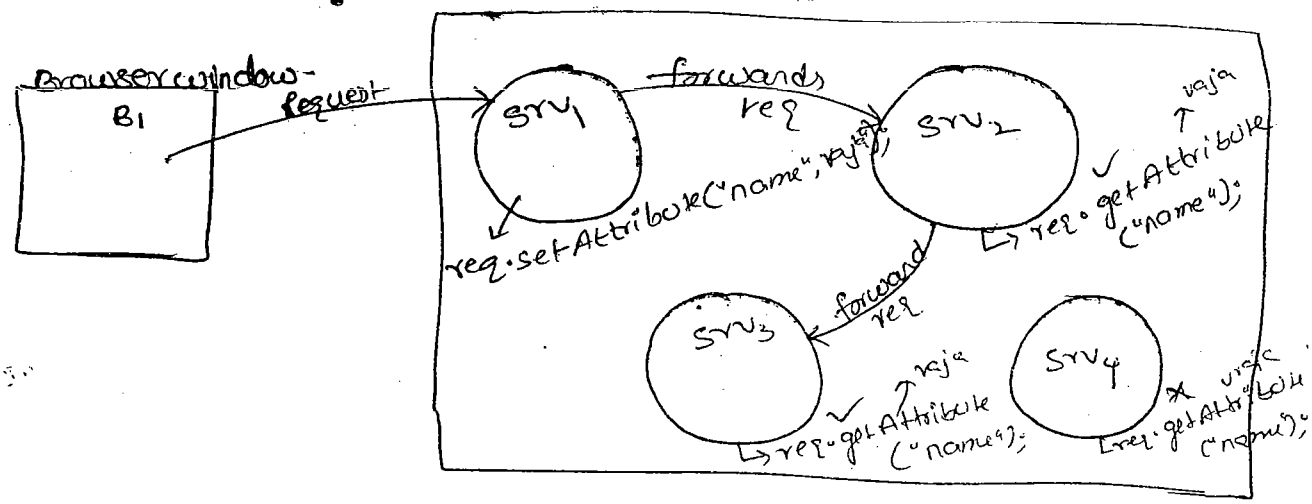
share the data b/w webresources of webApp.

→ Request Attributes will be stored in req object - Session attributes will be stored in httpsession object ServletContext ~~obj~~ attributes will be stored in ServletContext obj

note :- All these three types of attribute created in one resource are visible in other webresources of webApp based on their base obj (req (or) ServletContext (or) Httpsession) visibility.

28/12

1. RequestAttribute :-



→ RequestAttributes are visible in the webresources of webApp which participate in single request processing.

→ RequestAttributes are visible throughout a request cycle related webresources.

→ RequestAttributes are managed in Request object so the webresources who work with same request & responce object can share and use request attribute values.

--> In the above diagram SRV_1, SRV_2, SRV_3 are participating in a servlet chaining so all these three servlets will use same req & response objects. due to this the request attribute

servlets but not visible in srv_4 servlet because the srv_4 servlet is not using same request object of srv_1

to create the request Attribute

```
req.setAttribute("name","raja");
req.setAttribute("age",new integer(20));
```

Alt+Shift+i = input start in reverse way

To modify request Attribute value

```
req.setAttribute("name","raja1");
req.setAttribute("age",new integer(30));
```

→ req.setAttribute(-) method modifies the given attribute value; if Attribute is already available, other wise creates the new attribute that means we can use req.setAttribute() method, either to create new reqAttribute are to modify existing reqAttribute value.

TO read RequestAttribute value

```
object obj = req.getAttribute("name");
String s = (String)obj;

String s2 = (String)req.getAttribute("name");
```

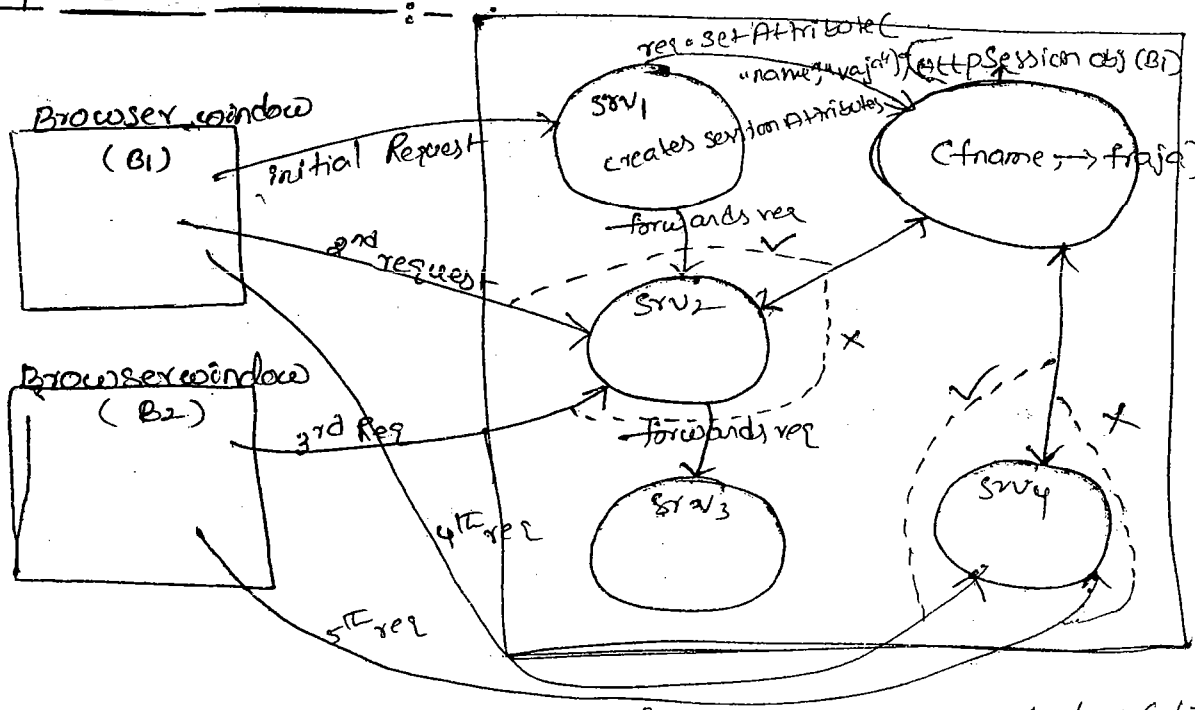
TO remove req Attribute

```
req.removeAttribute("name");
req.removeAttribute("age");
```

→ In the above diagram srv_2, srv_3 servlets are able to read requestAttribute("name") value only ~~then they~~ get request forwarded by srv_1, If srv_2 & srv_3 servlets get directRequest from browser we can't read request Attribute name value i.e created in srv. servlet.

webApp only when the webresource i.e creating requestAttribut and other webresources of same webApp are using same request obj that means the webresource i.e creating requestAttribut and other webresources of webApp should participate in a single servlet chaining.

2 → Http Session Attribute :-



→ HttpSession Attributes are specific to a Browser window (client because they will be stored in Http Session obj which will acted specific to a browser window.

→ HttpSessionAttributes are visible in all the web resources of web Application irrespective of their request object only when they get Request from that browser window for which HttpSession Attributes are created.

→ HttpSessionAttributes are client Specific (Browser window) global attributes for webresources of webApplication

✓ — when Srv_2, Srv_4 Servlet get request from Browser window (B1) it can read 'session Attribute value. req.getAttribute("fname");

request from Browser window (B_2) it can't read SessionAttribute value.

```
req.getAttribute("fname");
```
→ Raja

→ In the above diagram SessionAttribute fname is created by Srv1 Servlet in HttpSessionObj for Browser window (B_1) So, the SessionAttribute "fname" value is visible & accessable in all the Servlets of webApp only when they get request from the Browser window (B_1).

29/12

To create/local HttpSession obj :-

```
HttpSession ses=req.getSession();
```

→ req.getSession() creates new HttpSession obj on the Server for browser window if session obj is not already available for that particular browser window, if the session object already available browser window this method provides access to that session obj.

Session attributes will be managed & store in HttpSession obj :-

to create/modify session attribute :-

```
ses.setAttribute("fname","fraja");
ses.setAttribute("age", new Integer(20));
```

- Session.setAttribute() creates new session attribute if Session attribute is not already available in HttpSession obj other wise modifies the given session attribute value.

:-

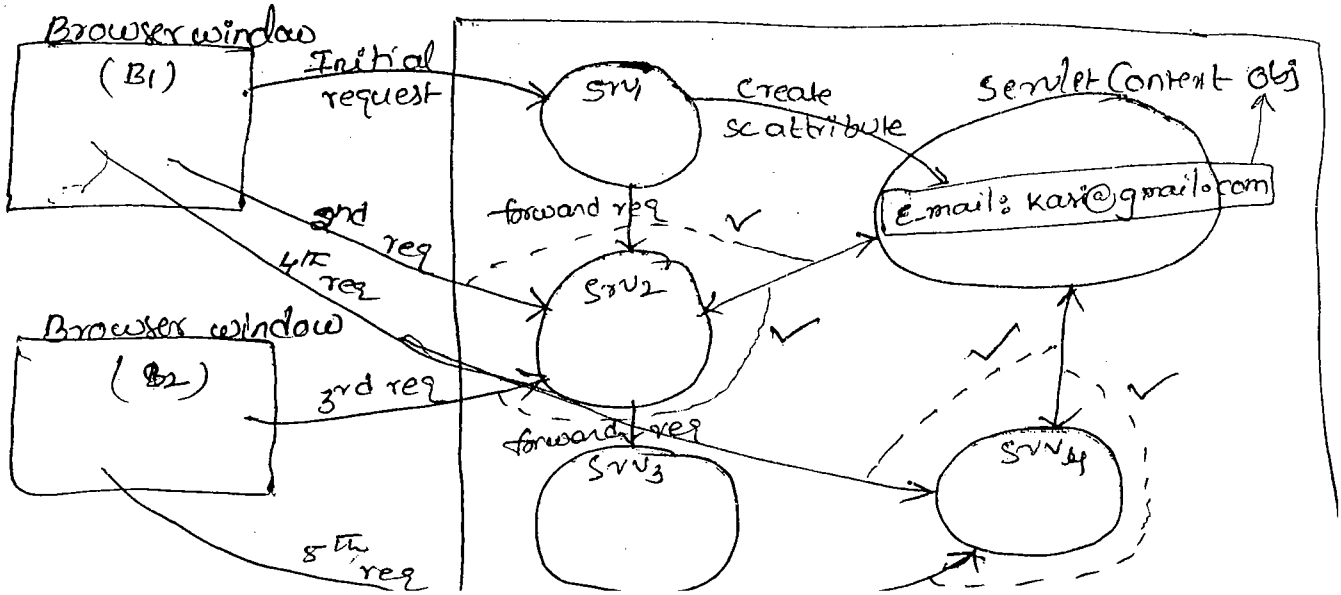
```
String s1 = (String) ses.getAttribute("fname");
String s2 = (String)
Integer i1 = (Integer) ses.getAttribute("age");
int value = i1.intValue();
```

To remove session attribute

```
ses.removeAttribute("fname");
ses.removeAttribute("age");
```

③ ServletContextAttribute :-

→ ServletContext obj is one per webApplication and it is global memory of webApp so that the attributes created in ServletContext obj which are servletContext attribute are globally visible attributes in all webresources of webApp irrespective of the request obj they use and irrespective of the browser window which they are getting the request.



→ ServletContext Attribute email is visible & accessable in all the webresources of webApplication irrespective of their request obj & irrespective of Browserwindow using which the web resources are requested.

→ programmer never creates ServletContext obj manually ServletContext obj will be created automatically the moment webApp is deployed in the Server.

TO GetAccess to ServletContext obj :-

```
ServletContext sc = getServletContext();
```

TOcreate/modify ServletContextAttribute :-

```
sc.setAttribute("email","kasi@gmail.com");
sc.setAttribute("age", new Integer(30));
```

TO read ServletContext attribute value :-

```
String s1 = (String) sc.getAttribute("email");
Integer i1 = (Integer) sc.getAttribute("age");
```

TO remove ServletContextAttribute :-

```
sc.removeAttribute("email");
sc.removeAttribute("age");
```

→ requestAttributes scope is request scope.

→ SessionAttribute scope is session scope (specific to a browser window)

→ ServletContextAttribute scope is webApp scope

global attributes, but Session attributes are specific to a browser window.

→ The Html form page i.e designed in the form of .html file is called static form (fixed components).

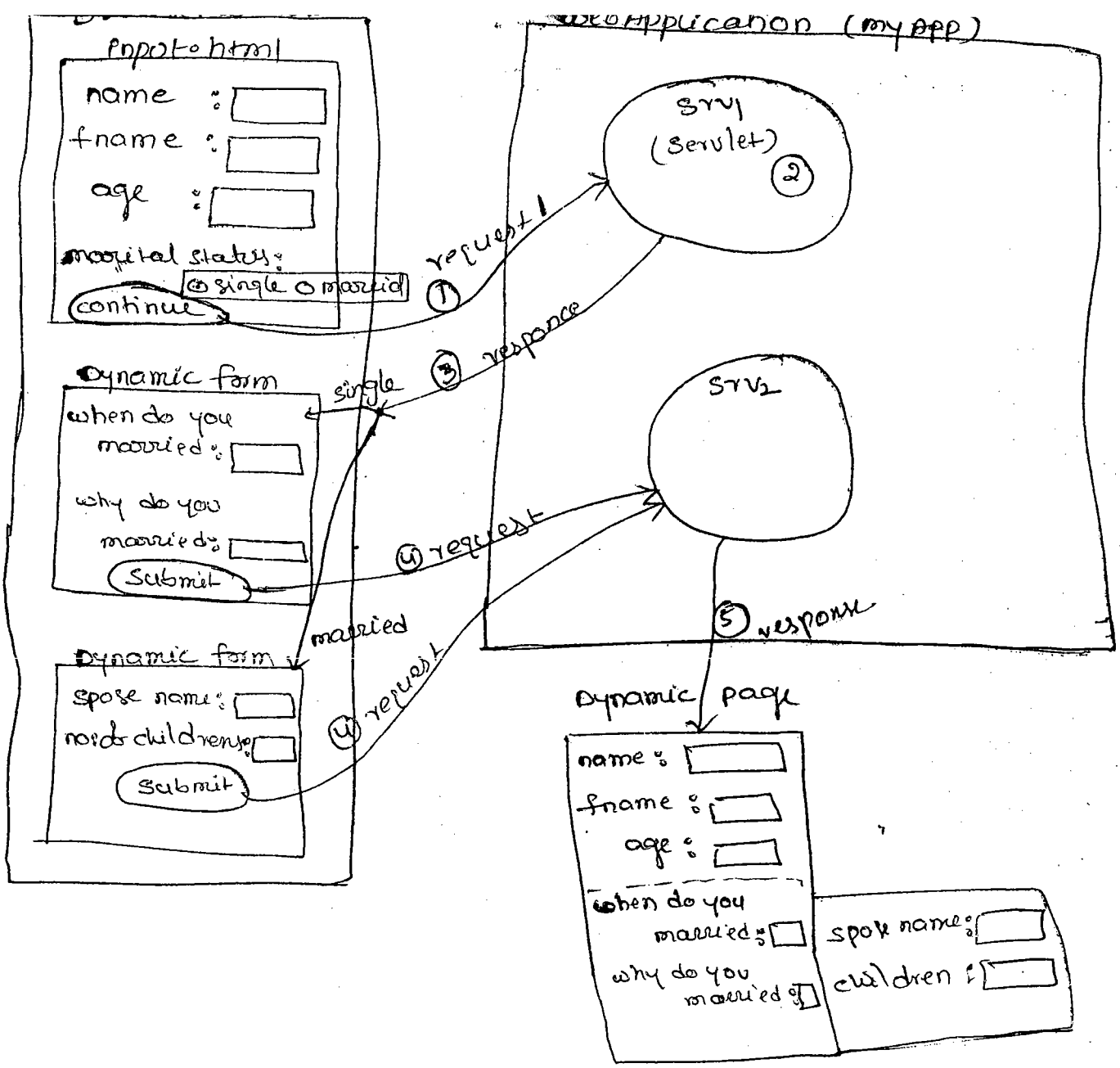
→ The formpage i.e generated as response of webresource program like Servlet JSP's based on the input values that are comming is called Dynamic form.

→ To Genarate Dynamic form from Servlet place form tag input tag, & other form component tags in pw.println statements.

→ Inorder to gather huge amount of details from end user instead of designing single long static form page it is recomanded to take multiple forms support having first form as static form & other forms as Dynamic forms.

Reallife example

→ The email-ID Application registration form of yahoo mail is design based on the Dynamic forms concept. similarly the job portal website naukri.com is also asking resume details based on Dynamic forms concept.



(IDE) (IDE) MYECLIPSE(IDE) 3/12

myeclipse = eclipse + Built-in plugins

The Application code that represents built-in functionality as pluggable functionality is called plugin. Through plugins the functionalities of or features of existing s/w can be improvised

type = ~~any~~ IDE s/w for developing java/J2EE Application.

version = 6.0

vendor = eclipse org

commersial s/w,

to download s/w = www.myeclipse.com

for help = "

No built in webApp servers, but all external servers can be configured with myeclipse IDE.

Q → wt is diff b/w eclipse IDE & myeclipse IDE?

A :→ eclipse IDE is the base s/w for myeclipse IDE s/w In eclipse IDE there will not be built in plug in to work with latest and high end technology we need to arrange them manually for eclipse IDE.

→ In myeclipse IDE all the plugins to work with latest technology or built in plugins, eclipse IDE is open source s/w myeclipse IDE is commersial s/w.

Q → How to add plugin's to the project of eclipse IDE

A :→ Create project in eclipse IDE → observe 'plugin' folder coming in project → download free plugin related jar file from internet to deal with heigh & technology like struts, JSF & etc → Add these jar files to plugin folder of

activate the plugin:

01/01/10

→ procedure to develop webApp shown in the diagram of 29/12 Diagram①:

Step1:- lanch myeclipse 6.0 IDE by choosing workspace folder. (the directory all project will be saved.)

note: the Directory where all projects of myeclipse IDE will be saved is called workspace folder.

Step2:- create web proj from myeclipse IDE

File menu
→ new
→ project
→ web project
→ next
→ myweb proj1
→ finish

Step3:- Add input.html form page to the project.

expand project
→ right Click on web root
→ new
→ html:
filename: input.html
→ finish.

 input.html

<form method="get" action="s1url" name="f1">

name : <input type="text" name="tname1">

father name : <input type="text" name="tname1">

age : <input type="text" name="tage" />

marital status : <input type="radio" cheked="cheeked"

value="single" name="r1" /> single

<input type="radio" checked="checked"

value="married" name="r1" />

<input type="submit" value="continue" />

</form>

=

① step

Add Srv1 servlet to the project

right-click on proj

└ new

└ servlet : name of the servlet : Srv1

options : create doGet create doPost

↳ next

Servlet mapping url : /s1url

|

└ finish

└ write following code in the doGet() method & call the doGet() method from doPost() method

```
import java.io.IOException;

public class srv extends HttpServlet
{
    public void doGet(HttpServletRequest req, HttpServletResponse res)
            throws ServletException, IOException.
    {
        PrintWriter pw2 = res.getWriter();
        res.setContentType("text/html");
        System.out.println(" doGet (--) of srv servlet");

        String s1 = req.getParameter("tname");
        String s2 = req.getParameter("tfname");
int age = Integer.parseInt(req.getParameter("tage");

        String s4 = req.getParameter("r1");
        if(s4.equals("single"))
        {
            // Generate Dynamic form1

pw.println("<form action="s2url" method="get" >");
pw.println("<b> when do u marri <input type="text" name="t2t1" /><br>
pw.println("<b> why do u marri <input type="text" name="t2t2" /><br>
pw.println("<b> <input type="submit" value="continue</b> ");
pw.println("</form>");
        }
        else    // dynamic form 2

pw.println("<form action="s2url" method="get" ><br>
pw.println("<b> spose name <input type="text" name="t2t1" /><br>
pw.println("<b> no: of childrens <input type="text" name="t2t2" /><br>
```

pw.println("</form>");

}

}

(5) STEP

→ develop Servlet Srv_2 having the url pattern /$s_2$url

res) rightclick on proj

→ new

→ servlet

→ name: Srv_2

options : create doGet create doPost

→ next

url : /$s_2$url

→ finish

- write following code in doGet(-,-) & call that doGet(-,-) method from doPost(-,-);

```
import java.io.IOException;
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Srv2 extends HttpServlet
{
    public void doGet(HttpServletRequest req, HttpServletResponse
            res) throws ServletException, IOException
    {
        PrintWriter pw = res.getWriter();
        res.setContentType("text/html");
```

```
String s2 = req.getParameter("fname");
String s3 = req.getParameter("age");
String s4 = req.getParameter("r1");
String s5 = req.getParameter("f2t1");
String s6 = req.getParameter("f2t2");

pw.println("(form data)
        PrintWriter pw = res.getWriter();
pw.println("name=" + s1 + "fname=" + s2 + "age=" + s3 "marital
        status" + s4);
pw.println("<br>second form data" + s5 + " " + s6);

    }

}

public void doPost(HSR req, HSR res) throws SE, IOE
{
        doGet(req, res);
}

}
```

window menu
→ preferences
→ myeclipse
→ Applies servers
→ tomcat
→ configure tomcat 5.x
→ Enable
→ tomcat home directory : C:\tomcat 5.5'
→ apply
→ ok

Step #1

Start tomcat Server from myeclipse IDE

Goto Servers icon in the tool bar
→ tomcat 5.x
→ start

Deploye project in tomcat server from myeclipse IDE

Goto Deploy icon in the tool bar
→ proj mywebproj1
→ add
→ tomcat 5.x
→ finish
→ ok

Open browser window (Glob simbol in the tool bar)

http://localhost:4040/mywebproj1/input.html

=

→ session is a series of continuous & related operation done by the user on the webApp

Ex: login to logout

→ start of advance java class to end of adv java classes on a particular day and etc

→ during a session if the webApp is remembering client data across the multiple request then it is called state full behaviour of webApp other wise it is called state less behaviour of webApp.

→ webApp's are stateless webApps by default because the HttpProtocal is stateless protocal according to this for every request given to a webApp from browser one connection will be created b/w browser & webServer. Once the request related resesponse goes to browser this connection will be closed.

→ stateless behaviour of webApp is nothing but not having the ebility of using first request data in second request processing & first, second request data in 3rd request processing and etc.

we need to develop our webApp by applying session tracking technique to make our webApp remambering client data across the multiple request that are given during a session.

02/01

Procedure to configar weblogic 9.x Server with myeclips

myeclipse IDE
→ window menu
→ prebierces
→ myeclipse
→ Application Servers
→ weblogic
→ configar weblogic 9.x
→ enable

Bea home directory : D:\bea

weblogic installation Directory: D:/bea/weblogic90

Administration name: weblogic

Administration pwd: weblogic

execution domain root: D:\bea\weblogic\samples\domain\
wl-server\ source execution (or) sourcename: Example Server

security policy file: D:/bea/weblogic90/server/lib/weblogic.poli

→ apply
→ expand weblogic 9.x
→ JDK
→ Add
→ Jre name: some logical name: xy21
→ browse & select J2SDK1.5 s/w i.e comming with weblogic s/w installation (D:/bea/jdk1.5
→ ok
→ apply → ok

window menu
└→ Prebarences
└→ myeclipse
└→ Application servers
└→ Jbos
└→ configar JBOS 4.0X
└→ Enable
└→ Jbos Home directory

Instalation folder of JBOS: E:/ Jbos s/w 4.2.2
└→ Apply
└→ Ok

Procedure to configar Glabish server with myeclipse IDE

windowmenu
└ prebarences
└ myeclipse
└ Application servers
└ Glashfish
└ configar glashfish 2.0X
└ enable
└ Homedirectory: D:/som/Appserver

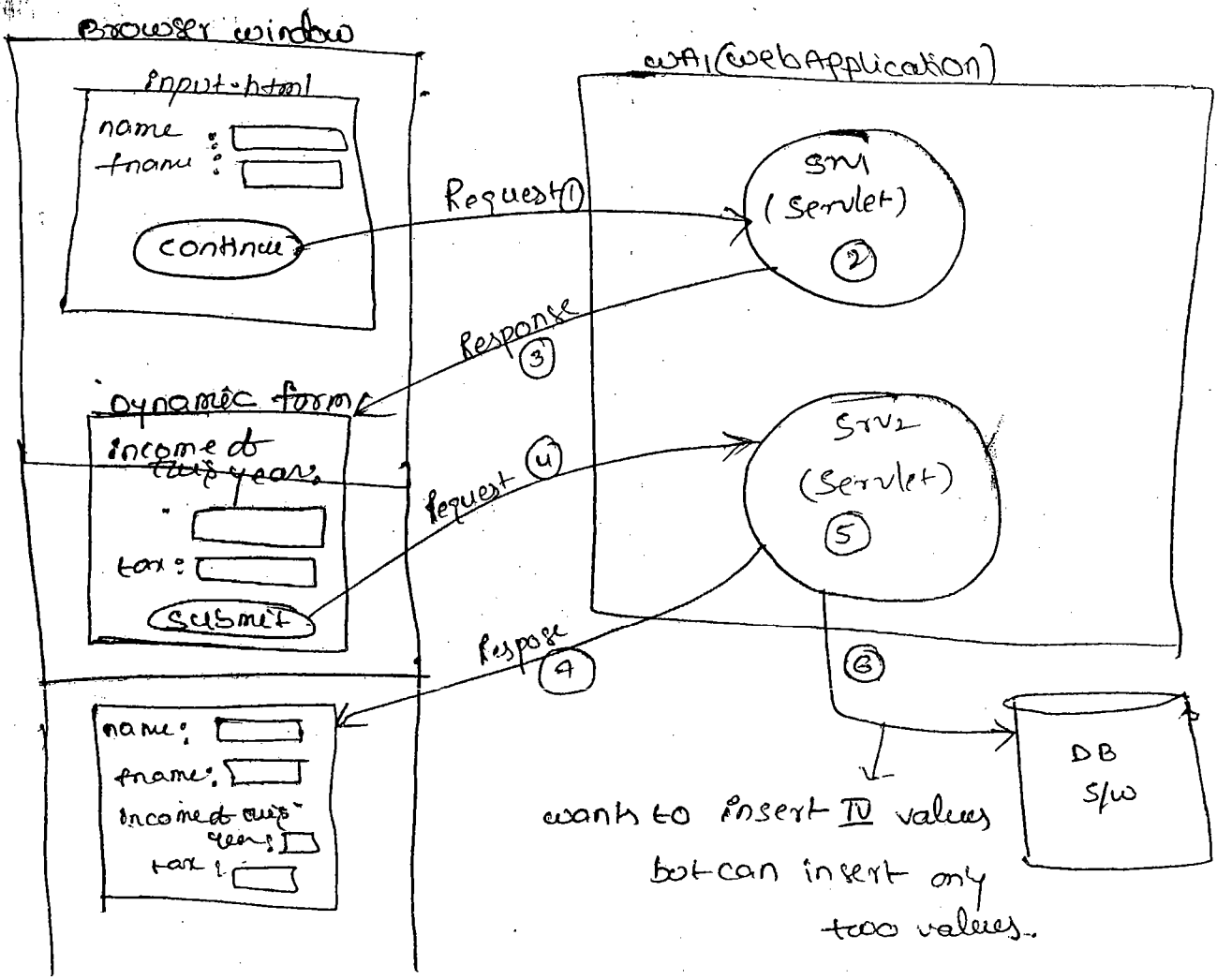
configar directory: D:/som/ Appserver/config

Server name : server

Domain name: domain1

Domain directory: D:/som/Appserver/domains/
Domain1

└ Apply
└ Ok



⟹ the above application is stateless webApp because the Srv2 servlet i.e getting second request from the browser window is not capable of using 1st request data

Q ⟹ why Http protocal is designed as stateless protocal?

A :- If Http protocal is state-full protocal the browser & web server will use single Http connection but are given - during a session due to this. multiple request that are given during a session due to this the Http connection may sit idel even though it is not utilized for long time, this process may make the server reaching to max number of connections when multiple clients engage the connection for long time

connection b/w browser & webserver will be closed immediately once the request related responses come to browser moreover for every request a new conn will be created b/w browser & server. due to this conn will not be sitting idel to get this advantage Http protocal is designed as stateless protocal.

→ TO make webApp as "statefull webApp" to remember the client data across the multiple request we take the support of session tracking technique.

1. Hidden form fields
2. cookies
3. HttpSession with cookies
4. HttpSession with URL Rewritting

(1) Hidden form fields :-

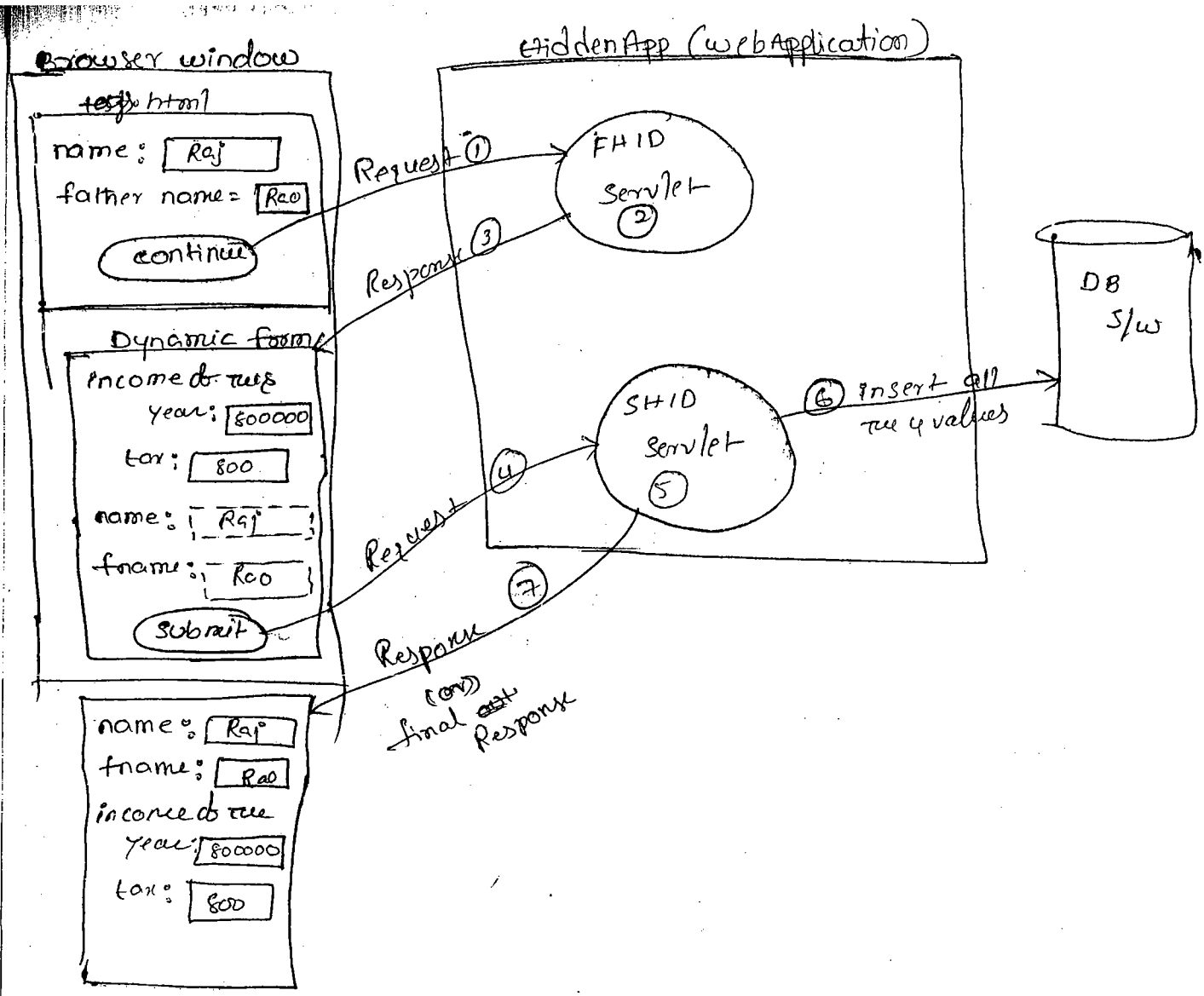
```
<Input type="text" name="h1" value="hide">
```

→ An invisible text box in the form page that contains a value is called Hidden box. when the form is submitted Hidden box name becomes request parameter name & value becomes request parameter value.

TO read hidden box value being form Servlet code

Ex

```
String s1 = req.getParameter("h1");
```



→ In the above diagram first form data is received by FHID Servlet & kept in Second form/Dynamic form as hidden box values, due to this the Second Servlet SHID which is receiving request from Second form is able to use first form/ first request data while performing request processing. this is called Hidden form fields technique based session tracking

client data across the multiple req that are given during a session is called "Session traking".

→ For example Application of Hidden form fields "Session traking i.e given in above diagram refer page no: 34 to 36 Application: 6

→ In email-ID Application forms & job portal web site Application form the end user information will be collected in multiple forms & initial forms data will be preserved until last form data & gather for this they take the support from Session traking ~~tecka~~ technique.

Advantages:-

1 → only Html knowledge is sufficient to work with this technique.

2 → this technique works with all web servers & Application servers.

disadvantages:-

1 → since the Hidden form files data can be viewed through ~~we~~ view source option in browser window, there will not be data security.

2 → Hidden form fields can store only string value but doesn't allow to store java objects as values

3 → Since the hidden form fields travel over the n/w along with response & request they improved n/w trafic blw client & server.

=

→ cookies are small textual information to carry client data across the multiple request by residing at the client side for session taking operations.

→ cookies will be created by webresources of webApp but they come to client at resides and client side.

→ cookies go to webresources along with request to make web resources using client data across the multiple request, every cookie contains cookie name & cookie value as String values.

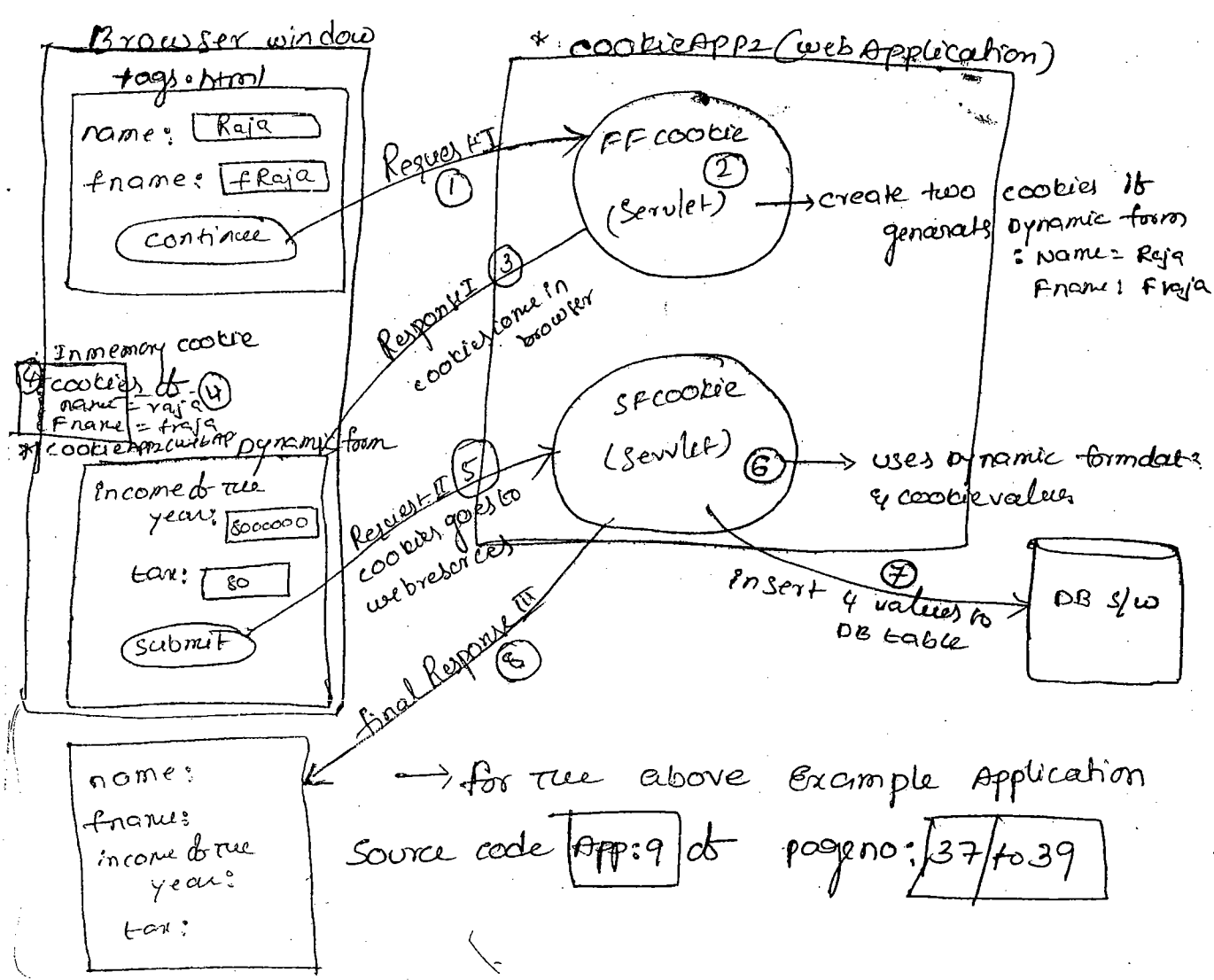
→ Every cookie remember the webApp name or domain name for which it is created.

→ There are two types of cookies

1. In memory cookies/ session persistance cookies
2. persistence cookies

(1) Inmemory cookie reside on browser window with out any expiry time so that Inmemory cookie will distroys when its browser window is closed.

(2) persistance cookie is reside in the files of client machine harddisk having expiry time so that they will not be distroyed once browser window is closed. They will distroy only when expiry time is reached.



→ for the above Example Application Source code App:9 of pageno: 37 to 39

→ cookies comes to browser / client mechine along with the ~~request~~ Response generation by webresorces as the values of HttpResponse Header ~~of~~ called Set_cookie.

→ cookies of webApp available in browser window or client mechine go back to webApp along with the request as "cookie" Http Request Header values when browser window gives request back to that webApp for which cookies are created

Ex
→ cookies of gmail.com website available in browser window/ client mechine will go back to gmail.com website along with request only when browser window gives request back to a web resorce of gmail.com website.

of other webresource of other than gmail.com website.

with Respect to Diagram.

① → Browser window submit the req from static form to FFcookie servlet.

② → FFcookie servlet receives form data and creates two cookies having form values. as inmemory cookies

③ → FFcookie servlet generates dynamic form as response req response also carry cookies to the browser.

④ → cookies to allowcate memory as browser window as inmemory cookie remembering webApp name cookieApp2.

⑤ → Dynamic form generates req to SFcookie servlet belonging to some cookieApp2 webApp, so cookies having first form data will goto SFcookie webresource along with the request.

⑥ → SFcookie servlet reads dynamic form data, cookie values and uses them in request processing, since cookies are representing first request data we can say SFcookie servlet we can using first request data while processing second request (session tracking)

⑦ → SFcookie servlet inserts all the 4 values to DB table.

⑧ → SFcookie servlet generates dynamic response having both form data (4 values).

→ Since the cookies are http protocal specific concepts inorder to do cookies based session taking/session management our servlet type must be httpServlet.

1. cookies will be created by webresources of webApp & they will be added to http response to send to client.

2. cookies must be read from httpRequest comming from the browser.

&

~~To create cookie~~

3. creating cookie is nothing but creating object for javax.servlet.http.Cookie class object (it contain name & value both are Strings).

To create cookie

```
Cookie ck1=new Cookie("uname","vajo");
                      |       |
                      |       value of the cookie
                      name of the cookie.
response.addCookie(ck1);
                   |
                   In memory cookie.

Cookie ck2=new Cookie("age","30");
       |
       persistent

ck2.setMaxAge(3600);  } — expiry time
              |
              seconds
response.addCookie(ck2);
                   |
                   persistent cookie (Expiry time is set)
```

To know max age period :-

```
int age=ck1.getMaxAge();
        |
        default max period of inmemory cookie
int age=ck2.getMaxAge();
```

is called inmemory cookie.

→ The cookie that contain positive number as maxAge period is called persistance cookie.

To modify cookie value

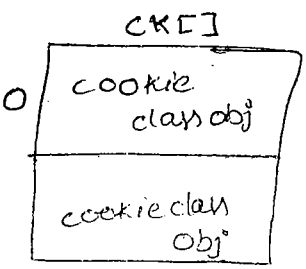
```
ck1.setValue("raja");
ck2.setValue("31");
```

To know Domain/webApp of cookie.

```
String s1 = ck1.getDomain();
String s2 = ck2.getDomain();
```

To read cookie value:

```
Cookie ck[] = req.getCookies();
if(ck!=null)
{
    for(i=0; i<ck.length; ++i)
    {
        s.o.p(ck[i].getName() + " " + ck[i].getValue());
    }
```



def:

→ A cookie is a small amount of information sent by a Servlet to a web browser, saved by the browser, and later sent back to the server to support session tracking".

→ For Example behaviour of cookie to under stand their basic behaviour Refer App:7 given in pageno: 36 & 37 :

arrived from browser window internetexplorer will be saved in the following directory: C:/ Documents & Settings/ welcome/ cookies &

windows login username.

the file name will be: windows login username@webApp name[n].txt:

n:- the no: of cookies that are stored in a file.

Ex: admin@cookieApp[2].txt — no: of cookies that are available.

* → once the content of persistant cookies file is modified/changed the persistent cookies & their file will be distroyed Automatically

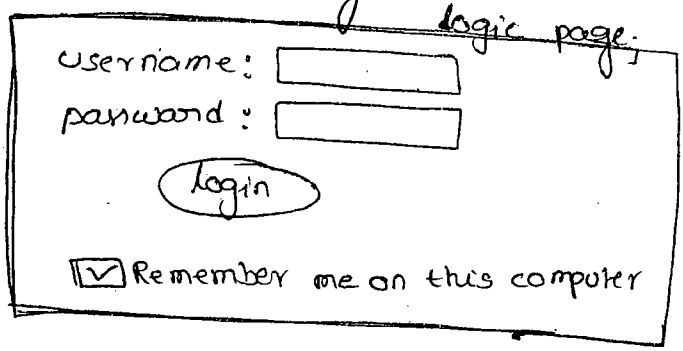
→ Eventhough cookies are residing at the client side their values can't be altered but they can be distroyed.

→ webresources of webApp can't distroy the cookies because cookies allocate memory at client side.

→ NetScape Navigator manages both persistance & inmemory cookies in browser window it self but the persistent window contain expiry time so they will not be distroyed once browser window is closed.

* → Cookies are not specific to browser windows but specific to browser s/w's due to this for cookies based session taking the webresources of webApp should get request from single or multiple browser windows of same browser s/w.

→ In every email system website like yahoo mail, gmail & etc the login page contains following content



→ when Remember me on this computer check box is selected the webApp of email system uses persistence cookies to remember user details across the multiple visits given to login page of website from that computer.

Advantages :-

1 → since the cookies allocate memory at client side the server will not have any burden.

2 → All webservers & Application server gives support to work will cookie.

→ All server side technology lik servlet, JSP, Asp.net & etc give support to work with cookies.

disAdvantages.

→ cookies can store only string values so we can't store java objects as values of cookies.

→ since cookies travel over the n/w across the net along with the request & response there is a posibility of n/w trafic.

→ persistence cookie may bring virous to the computer

→ since cookies data can be viewed & cookies can be distroyed there is no data security

→ there is a limitation of on the no: of cookie, per browser window, per webApp maximum of 20 cookies all together maximum of 300 cookies per browser window.

→ cookies can be restricted comming to browser window using various options of browser setting

→ In internet Explorer

tools menu.
↳ internet options.
↳ privacy
↳ select high privacy that will block all cookies.

→ In Netscape Navigator:

tools menu
↳ cookie manager
↳ block cookies from this site.

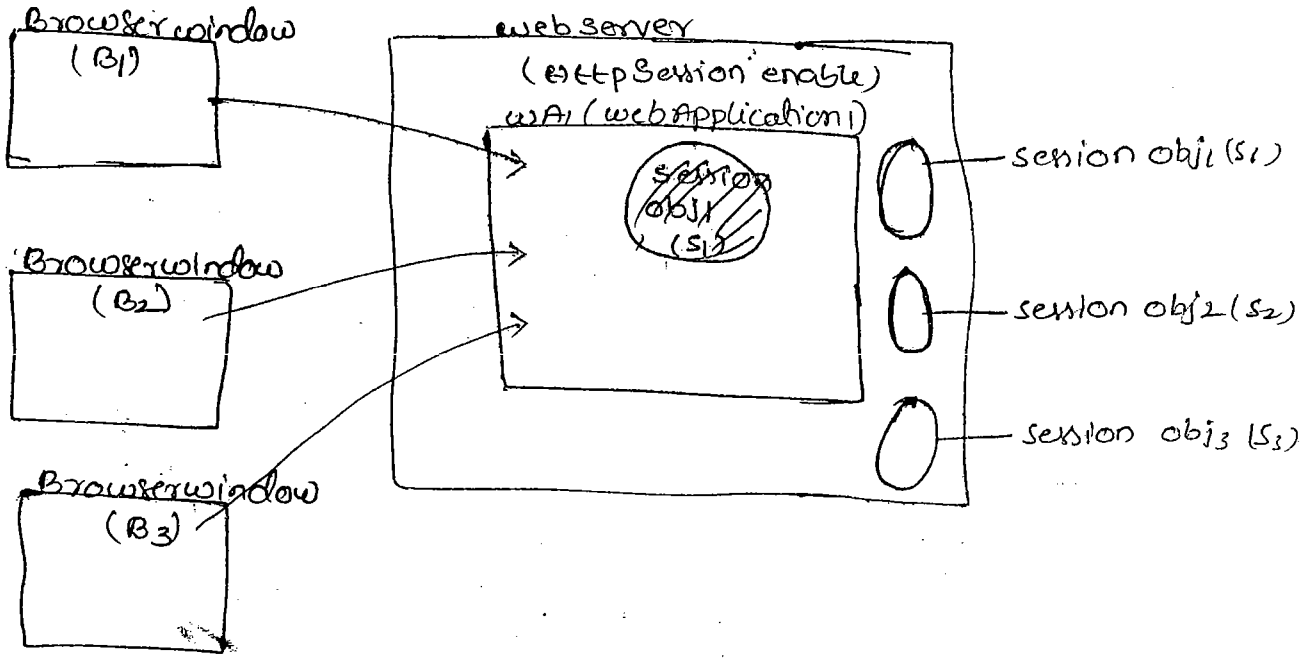
(3) HttpSession with cookie based session traking:

→ HttpSession object is client specific object that resides on the server.

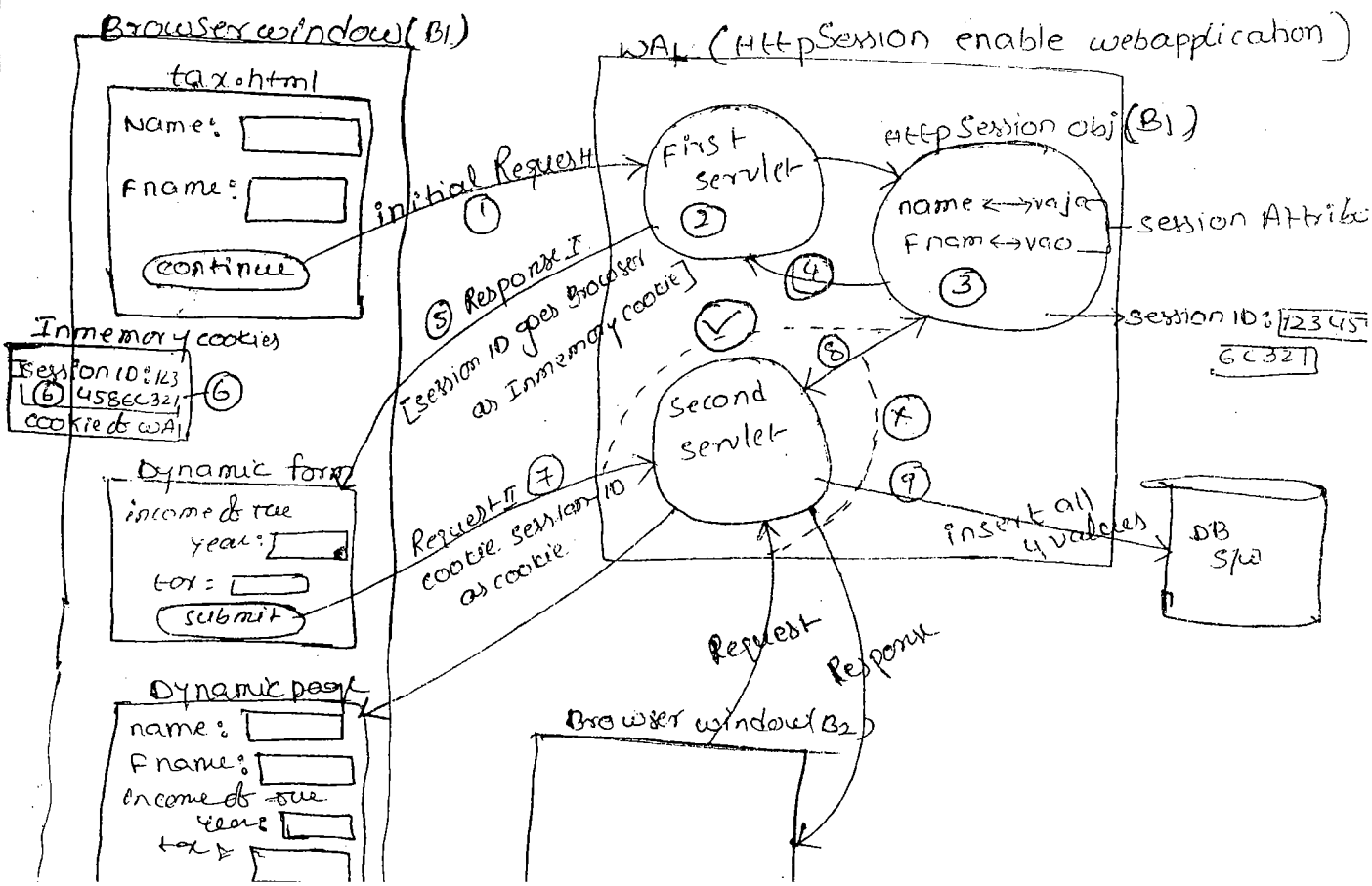
→ HttpSession object preserve client data across the multiple request in the form of session attribute value.

→ HttpSession object & it's session attribute are visible and accessable in all the webresources of webApp across the multiple request only when they get request from that particular browser window for which HttpSession obj & session attributes are created.

app. in which HttpSession obj based session taking is enable then multiple HttpSession object will be created in the server for multiple browser windows on one per browser window bases.



Application



of browser window

on the server, the session ID of session object will be given to browser window response based in memory cookie value. and this browser window send session ID along with every request i.e given back to same webApp as request based cookie value to recognise browser window across the multiple request.

→ Session attributes of session obj can store client specific data in the form of java obj values. Session attribute can't store premitive values.

→ Session ID for session obj will be automatically generated by server, & it will be automatically kept in response based cookie having cookie name "JSESSIONID".

With Respect to the Diagram

① static form tax.html gives initial request to first Servlet.

② First Servlet reads formdata and creates httpSession obj for browser window(B1).

③ First Servlet keeps form data (first form data / first req. data) in httpSession obj session attribute values.

④ Session ID of HttpSession obj generated by the server comes to browser comes to servlet (1st servlet).

⑤ 1st servlet generate dynamic form as response that response carries session ID as inmemory cookie value.

⑥ Inmemory cookie having session ID allocates memory on browser window remembering

(7) Dynamic form generate req to Second Servlet of WA1 webApp so the sessionID cookie belonging to WA1 webApp goes to webApp carring session ID.

(8) Second Servlet receives req from Dynamic form & uses session ID of req based cookie to get access to HttpSessi obj of browser window B1 & to read session attribute values. that are representing 1st form / 1st req data.

note :- Since 2nd Servlet the able to use 1st req / 1st form data while processing 2nd req given by browser window we can say session traking is performed here.

(9) Second servlet inserts both forms data to the DB table

(10) Second Servlet generate Dynamic page having the values of both forms.

→ Since this technique is using cookies to send session ID to browser to bring session ID back to the webApp from browser this technique is called HttpSession with cookies technique.

→ Session API :- (working with javax.servlet.http.HttpSession Interface)

• HttpSession object means it is the object of a class that implements javax.servlet.http.HttpSession (interface)

To create/ to Get access to HttpSession obj :-

ⓐ. `HttpSession ses = req.getSession();` (create/locates)

· it provide to already there in browser window, it is provide access to servlet.

* · the above method creates HttpSession obj on the server for browser window if the HttpSession obj is not already available for browser window other wise, this method provides access to HttpSession object i.e already available for browser window.

Note :- when this method is called in the first servlet of above diagram it creates new HttpSession object on the server for browser window (B1). ~~when~~ in the second servlet of above diagram when it is getting request from browser window B1 this method provide access to already available HttpSession obj for browser window B1

ⓑ. `HttpSession ses = req.getSession(true);`

(same as ⓐ) (creates/locates)

To Get access to HttpSession obj

ⓒ. `HttpSession ses = req.getSession(false);` // only locate not create.

→ the above method gives access to already available HttpSession obj ~~of~~ for browser window. If session obj is not available for browser window it will not create new HttpSession obj more over it returns null.

note :- Inorder to see servlet of webApp to start new session/ to work with existing session use ⓐ, & ⓑ options. Inorder to see servlet webresources only to participate in the existing session and not to start new session then use ⓒ option.

String id = ses.getId();

To know Session obj creation date & time:

e. long ms = ses.getCreationTime();

Date d = new Date(ms);

} Here java.util.Date obj will be created to hold the date

System.out.println("session obj creation time is:" + d.toString());

→ Session obj is created means one browser window is starting session with webApplication.

• Session obj is invalidated means one browser window is completing session with webApp.

* The above code the variable ms represents no: of milliseconds that are allapsed from 1970 jan 1st 00:00 hours (epoch standard) to the Session obj creation date & time.

Last Accessed time of session obj

f. long ms = ses.getLastAccessedTime();

Date d = new Date(ms);

System.out.println("Session obj is lastly Accessed at" + d.toString);

To Get access to ServletContext obj

g. ServletContext sc = ses.getServletContext();

To Invalidate Session obj:

note:- Invalidated session obj is nothing but Inactive session obj and browser window can't perform session traking operations with webApp by using this session obj

→ Inactive session obj is about to destroy HttpSession obj.

session obj is closed.

note:- when browser window is closed the inmemory cookie that contains sessionid will be distroyed so that the browser window related session obj will be invalidated

(ii) By calling session.invalidate(); method.

(iii) when maxInactive interval period/session idle time period is completed.

→ If session obj is continuesly idel/workless for inactive interval period (or) idel timeout period the session obj will be ~~invalid~~ invalidated automatically.

ⓐ programatic Approch to specify Inactive interval period (java code)

```
ses.setMaxInactiveInterval(1800); (secons)
```

ⓑ Declarative approch (xml code)

web.xml

```
<web-app>
    - - - - -
    - - - - -
    - - - - -
    - - - - -
    <session-config>
        <session-timeout> 20 </session-timeout>
    </session-config>          └ minits.
</web-app>
```

note:- the default session idel time out period/ inactive interval period for session obj is 30 minits.

Q→* what happens if maximum inactive interval period/session idel time is kept(or) configard by using both poogramatic & declorative approcher having two different values.?

(A) since the java code kept in servlet executes after web.xml entries so the session timeout period specified in programatic approch will be ...

int max = ses.getMaxInactiveInterval();

(in) seconds

To know whether Session obj is new or not :-

boolean b = ses.isNew();

→ then this is method is called before SessionId & Session obj goes to browser this this method returns true otherwise false.

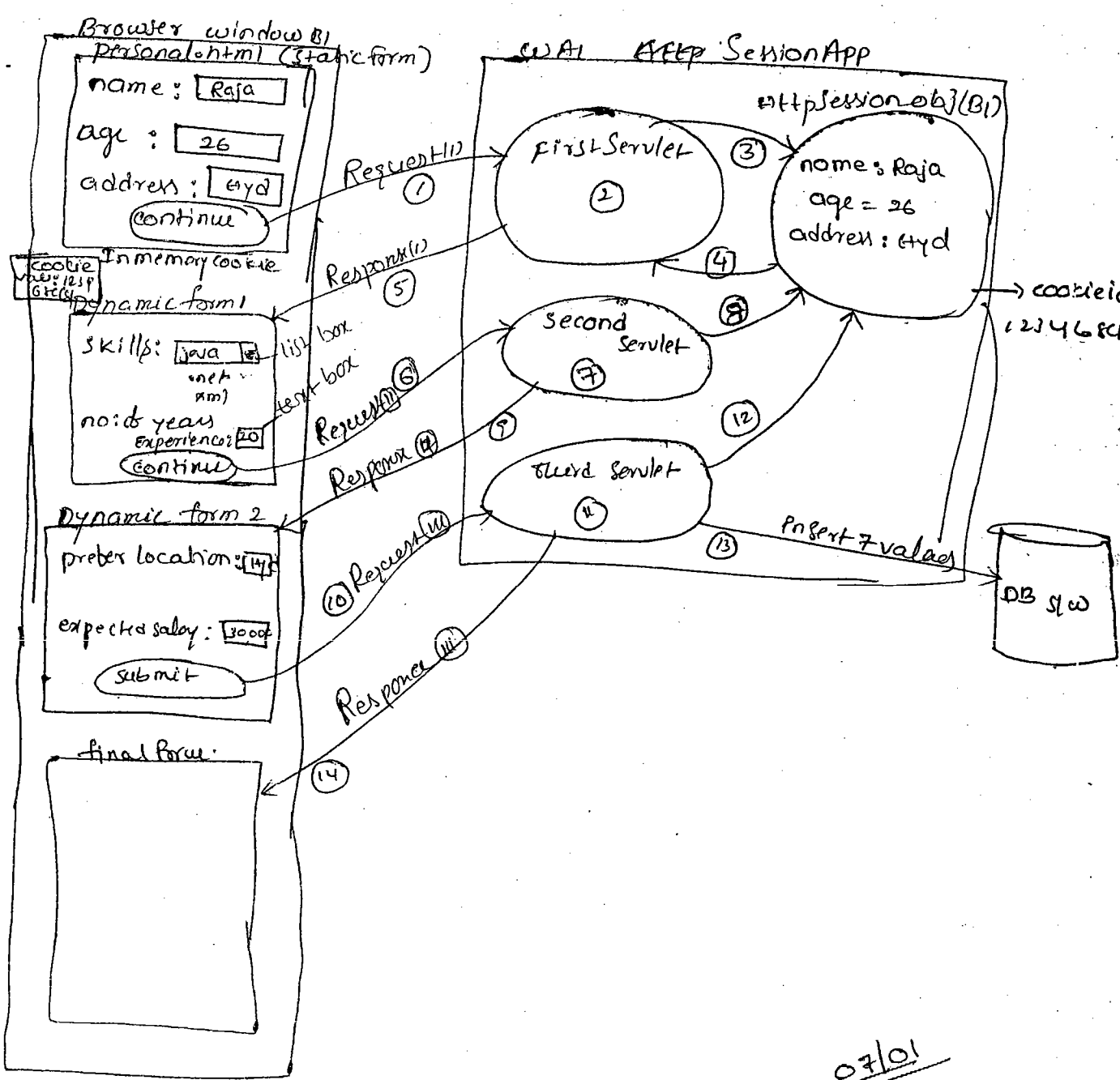
→ by using this method we can notice that the session obj that is accessed by web resource is just created new session obj or already available old session obj, when this method is called in first servlet of above diagram before 5th step. Then this method returns true. Then this method is called any diagram of first servlet after 5th step then this method returns false.

→ client data in HttpSession obj will be stored as session attribute, Session attribute name is String value must be java object.

Operation on Session Attribute :-

To create/to modify

Operation on session Attribute	method	deprecated method.
To create/to modify	ses.setAttribute(-,-);	ses.putValue(-,-);
To read	ses.getAttribute(-,-);	ses.getValue(-);
To remove	ses.removeAttribute(-,-);	ses.removeValue(-);



07/01

→ In the above Application first two forms data is preserved in the Session obj as Session attribute value until third form data comes to Servlet so that the third Servlet can use first, second form data with processing this third request this is called Session tracking

→ for the source code of this Application Refer App:10 available in page no: 39 to 42

→ what happens to server if restarted in the middle of session taking operation while working with Http Session with cookies technique based webApp.

(A) :- Session will not be invalidated that means session will continued after restarting the server.

→ In the middle of the session taking operations if the browser window is closed while working with HttpSession with cookies technique based webApp the Session will be invalidated.

Advantages (HttpSession with cookies technique) :-

→ since the HttpSession obj and it's Session attributes allocate memory on the server the client data will not be travel across the multiple request & over the nlw this gives data securacy.

→ HttpSession objects session attributes are able to store java objects as values including strings.

→ All webservers & All server side technologies support this technique.

DisAdvantages :-

→ HttpSession objects allocate memory on the server so they improve burden on the server.

→ when cookies are Restricted comming to browser this technique fails, because this technique sends sessionid to browser in the form of Inmemory cookie value.

Q :-

→ the disadvantage of 3rd technique is when ~~sess~~ cookies are restricted coming to browse that technique will fail because sessionid will not come to browser and will not go back to server from browser.

→ To overcome this problem don't depend upon cookie to sessionid to browser along with response and - - - - - - - - - - to bring back sessinid to server from browser along with the request, as alternate you appen session-id to the URL that goes to browser along with response and that comes to server from browser along with the Request this work is called URL Rewriting.

→ genarally the action url of dynamic form comes to browser along with the response and goes back to server from browser along with request. So Append SessionID this url to perform HttpSession with url Rewriting technique based session trating in the web App.

→ To Append SessionID to the given url you can use

res.encodeURL("/s2url"); // res.encodeURL("/s2url");

url

output

/s2url;sessionID:4381 8A BDEA31

→ Genarally Dynamic forms will design as shown bellow to support HttpSession with URL Rewriting technique-

pw.println("<form action="+res.encodeURL("/s2url")+" method=

GET>");

pw.println("</form>");

→ For Example Application on HttpSession with URL Rewriting

in Reffer : 11 page no: 42 to 45

Advantages & DisAdvantages in HttpSession with URL Rewriting

→ Same as third technique but this technique perform Session traking eventhow the cookies are restricted comming to browser window.

08/01

conslusion on Session traking :-

* → If your webApp is dealing large scale commersial activites having huge customer base then use cookies based session traking technique.

* → while developing small scale (or) mediam scale webApp having limited no: of customer base like domestic webApp (or) organization based webApp's then use HttpSession with URL Rewriting Session traking technique.

→ A webApp can use multiple techniques of session traking for session management operations.

File uploading & File downloading

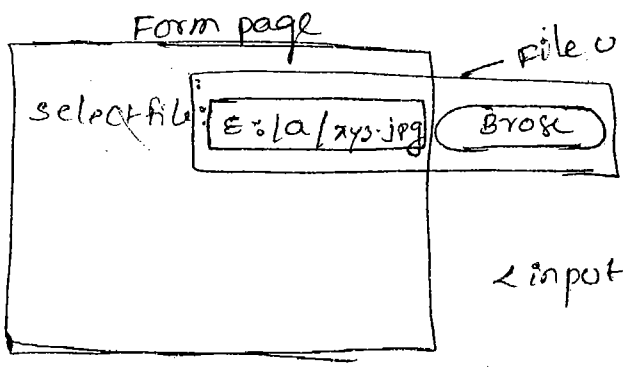
*

→ The process of moving a file of client mechine file system to server mechine file system through web is called file uploading & Reverse is called file downloading.

→ while developing matrimoni website, job porter website these file uploaded, downloaded operations are quite important.

→ To select the file from the client mechine file system through form page. we need to take file upload control on the form page

Ex:-



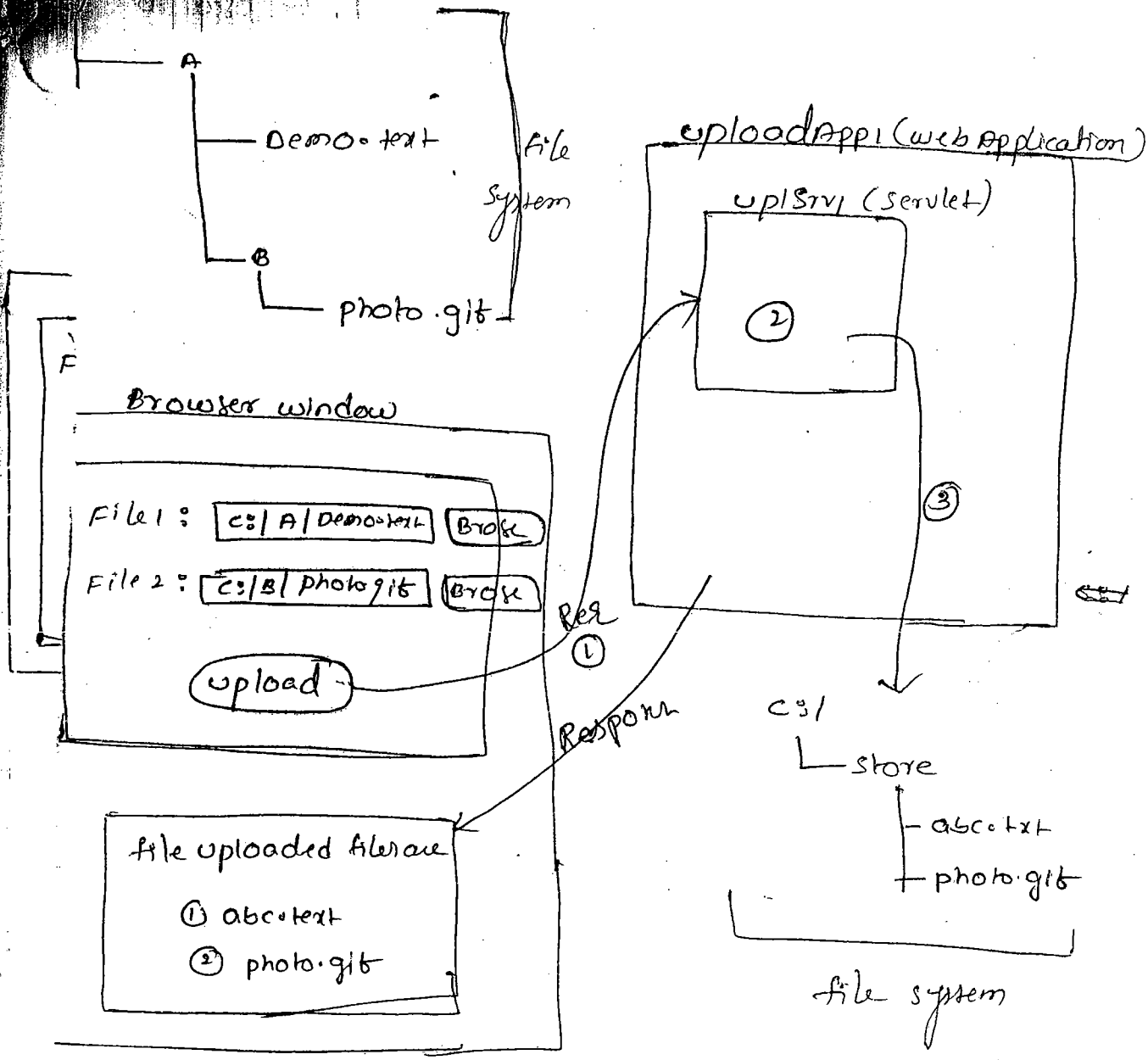
Html code

```
<input type="file" name="f1" />
```

→ Most of the developers prefer working with 3rd party APIs like java zoom-API to perform file upload operations in our servlet programming.

→ This API Related main & dependent jar files we can down load from www.javazoom.net website.

→ Set of files, directorys available in a computer is called file system of that computer.



⇒ For Example Application on file uploading Rebber

application : 14 page no: 49 to 50

**

→ when java webApp uses 3rd party API the third party API Related main jar file should be add in the class path & the third party API Related main & dependent jar files should be added in the ~~c:\~~.WEB-INF/lib folder of webApp. Here the jar files kept in the class path will be used by java compiler during the

jar files kept in web-INF/lib folder will be used by
* → web server to recognize & execute 3rd party API
fi during the execution of Servlet.

ie → If the classes of a.jar files uses classes, interface, meth
d of b.jar file then b.jar file is called dependent jar file of
— a.jar file.

th → If one jar file classes use the classes of another set of
in jar files then the another set of jar file or dependet jar
— file are the main jar file.

f → The classes of upload bean.jar file are internally using
the classes & interfaces of cos.jar, struch.files, so that
fi these two jar files are dependent jar files for uploadbean.j

* → In order to make response of any webresource as
downloadable add the following two lines of code that
deals with response content type & Response hedder
called contend/disposition.

res.addHeader("Content-Disposition","attachment:filename
=abc.doc");

res.setContentType("application/msword");

→ An Event is a action i.e done on the component/object.

→ Event handling is the process of catching event and performing certain task by executing logic we always perform Event handling through event listeners. In java programming each event comes as an object and listener to handle that event comes as an interface.

→ we generally see events & event handling operations in awt, swing programming.

→ Servlet listners are introduced from Servlet API 2.4 to perform event handling on Servlet context, HttpSession, ServletRequest objects.

→ By performing event Handling on ServletContext object we can keep track of when Servlet context object is created and distroy based on this we can notice when the webApp is deployed and undiployed or relo'aded or stoped.

→ By performing event Handling on ServletRequest object we can know the ServletRequest obj is created & distoyed for each request based on this we can keep track of the request processing time of each request.

→ By performing EventHandling on HttpSession obj, we can keeptrack of when the HttpSession obj is created & invalidated Based on this we can notice the amount of time he spent by the user during a session.

... on any object the following details are required.

1. Source obj
2. Event class
3. Listener Interface
4. Event handling Methods

Ex:- To perform event handling on awt button.

1. Source obj is button.
2. Event class is ActionEvent
3. Listener is ActionListener
4. Event handling method is public void actionPerformed

→ The tomcat webserver generates one logfile in tomcat-Home/logs folder on one logfile per day bases.

→ Inorder to write messages to this log file use log() method of ServletContext interface.

sc.log("Date & time is" + new Date().toString());

Event handling details for Servlet programming

Event Handling details in Servlet Programming

Source obj	Event class	listener Interface	Event handling methods.
ServletContext obj	ServletContextEvent	ServletContextListener	public void contextInitialize (ServletCont public void contextDistroyed (ServletCon
ervletRequest obj	ServletRequestEvent	ServletRequestListener	public void requestDistroyed (ServletRequ public void requestInitialised (ServletReq
ttpSession	HttpSessionEvent	HttpSessionListener	public void sessionCreated (HttpSessior public void SessionDistroyed (HttpSessio
ServletContext obj: Attributes)	ServletContextAttribute Event	ServletContextAttribute Listener	public void attributeAdded (SCAE s public void attributeRemoved (public void attributeReleased (
ServletRequest obj Attributes)	ServletRequestAttribute Event	ServletRequestAttribute. Listener	public void attributAdded (SRAE public void attributeRemoved (public void attributeReplaced (

HttpSessionAttribute HttpSessionBindingEvent HttpSessionAttributeListener : public void attributeAdded(HS

public void attributeRemoved(HSB

public void attributeReleas(HSBE

HSBE=(HttpSessionBindingEvent se)

Procedure to develop Listener classes to configar with webApplication

Step:1 :- develop Listener classes by impleming Aeroriate xxxListener interface and save this classes in WEB-INF/classes folder.

Step:2 .

```
import java.io.*;
import java.util.*;
import javax.servlet.*;

public class MyScListener implements ServletContextListener
{
    long sttime, endtime;
    public void contextInitialized(ServletContextEvent sce)
    {
        sttime=System.currentTimeMillis();
        ServletContext sc=sce.getServletContext();
        sc.log("webApp is deployed at"+new Date().toString());
    }
    public void contextDistroyed(ServletContextEvent sce)
    {
        endtime=System.currentTimeMillis();
        ServletContext sc=sce.getServletContext();
        sc.log("webApp is undiployed/stopped is"+new Date().toString);
        long diff=endtime-sttime;
        sc.log("webApp is executed for "+(diff/1000)+" secs");
    }
}

import java.io.*;
import java.util.*;
import javax.servlet.http.*;
import javax.servlet.http.*;

public class mySesListener implements HttpSessionListener
{
    log sttime, endtime;
    public void sessionCreated(HttpSessionEvent se)
    {
```

```
HttpSession ses = sce.getSession();
ServletContext sc = ses.getServletContext();
sc.log("Session is Started " + new Date().toString());
}
public void sessionDistroyed(HttpSessionEvent sce)
{
    endtime = System.currentTimeMillis();
    HttpSession ses = sce.getSession();
    ServletContext sc = ses.getServletContext();
    sc.log("session is invalidated by user " + new Date().toString());
    long diff = endtime - sttime();
    sc.log("User is There is session" + (diff/1000) + "sec");
}
}
```

stepno:2 :- Configar listeners classes with webApp.

```
<web-app>
    <servlet>
    ---------
    ---------
    </servlet>
    <servlet-mapping>
    ---------
    ---------
    </servlet-listener mapping>
    <listener>
        <listener-class>MyScListener</listener-class>
    </listener>
    <listener>
        <listener-class>MySesListener</listener-class>
    </listener>
</web-app>
```

→ A webApp is collection of webresources containing both static & dynamic webresources to develop static webresource we use Html, to develop Dynamic webresources we use server side technology like Servlet, JSP, ASP, ASP.Net, PHP & etc

→ Servlet JSP's are Java based server side technology to develop Dynamic webresources of java based webApp.

Servlet — sunms
Jsp — sunms } — small, middle, large & high scale App.
Asp — microsoft
ASP.Net — microsoft. } — Develop small, mediam scale App
PHP — Apache

Ø → when sunms as already given a technology called Servlet why it is giving another server side technology called JSP?

(A) :- Servlet technology is given to market after the arrival asp. In asp full tag based programming is posible servlets is having good features when compare to asp but in the initial days of servlets programmers are not shown intrest to come & work with servlets because of it's traditional java programming so non java programmers have not prefered servlets as alternate for asp even though servlets is having goodfeatures compare to asp to Solve this problem sunms as design a tags based technology called JSP having all the benifits of servlets.

→ JSP is given based on page compilation process. The process of converting JSP into equalent Servlet by using a page compilar is called page compilation. ASP, JSP, ASP.Net, PHP are the page compilation based technology.

→ Drabacks of Servlet :-

1 → Strong Java knowledge is required so it is not suitable for non-java programmers.

2 → placing Html code in servlets is quite complex.

3 → In Servlets the presentation logic (Html code) will be mixed up with java code (Business logic). So any modification in one logic effects other logic.

4 → There are no implicit object in Servlet programming.

5 → Modification done in source code of the Servlet forces the programmer to Recompile the Servlet & to Reload the webApplication. 6 → Exception handling in servlet must be done explicitly.

→ Features of JSP :-

1 → Given as a tag based server side technology.

2 → Strong java knowledge is not required so it is suitable for both java & non_java programmers.

3 → Given based on page compilation technique so every JSP generates an Equalent Servlet all the benifits of Servlets are there in JSP.

4 → JSP can be used along with other technologys in web resorces development.

5 → modification done in source code of JSP will be effected automatically without recompilation of JSP & reloading of web app.

separate presentation logic (HTML) from business logic (Java code).

7 → JSP takescare of exception handling automatically & need a allows to perform explicit exception handling in declarative mode (xml & JSP tags based).

8 → Allows to work with predefined third party supply & user defined JSP tags.

9 → Easy to learn & easy to apply

10 → Gives set of implicit objects or built-in objects.

=

Latest version of servlet API → 2.5 (J2SDK 1.4 & 1.5)

Latest version of JSP API → 2.1 (J2SDK 1.4 & 1.5)

→ To execute JSP's there is no need of saparate JSP containar, executing JSP is nothing but executing JSP equalent servlet, so that every webserver uses jsp page compiler to get equalent servlet from JSP and executes that equalent servlet by using servlet containar/ web containar.

Tomcat —page compilar→ Jasper

weblogic —page compilar→ JSPC

→ JSP API is nothing but working with classes & Interface of javax.servlet.jsp package, javax.servlet.jsp.el package javax.servlet.jsp.tagext package.

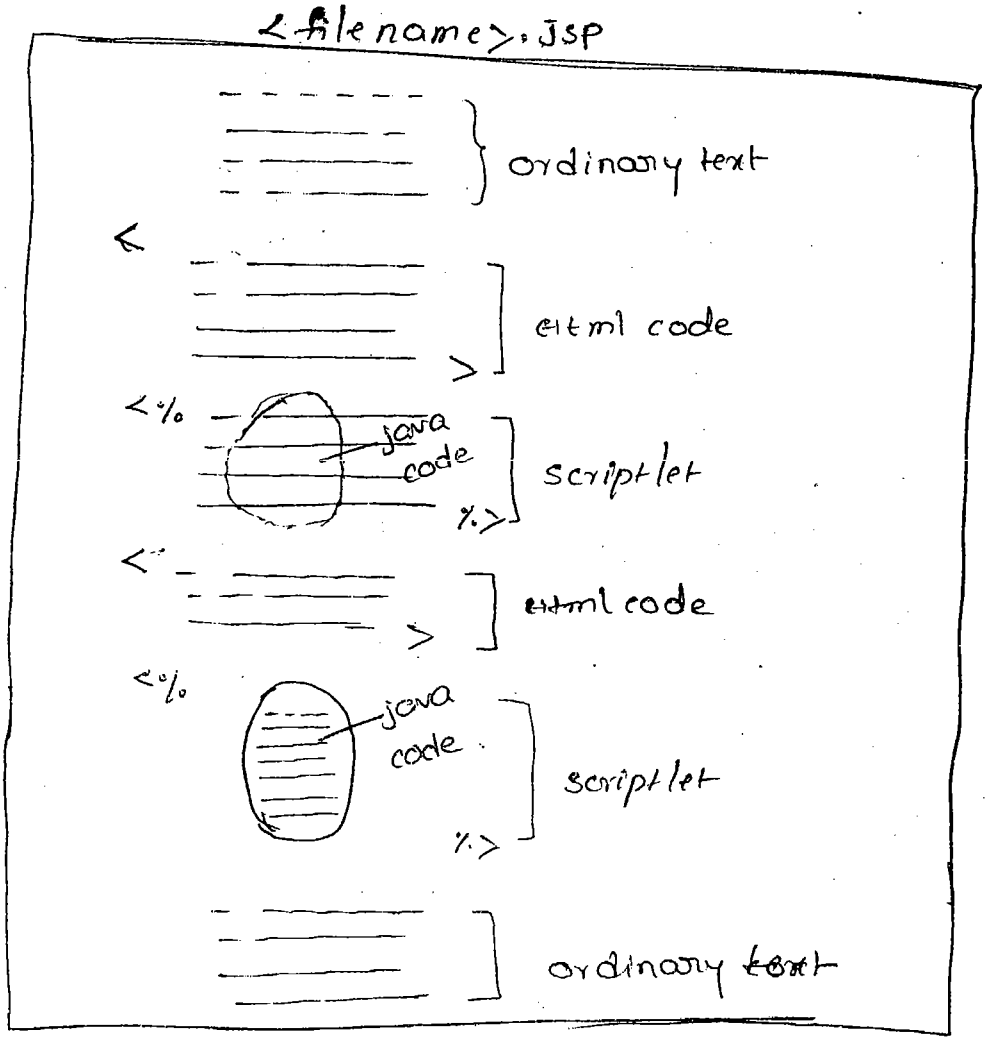
el = expression language

tagext = tag extension.

- file to represent JSP API.

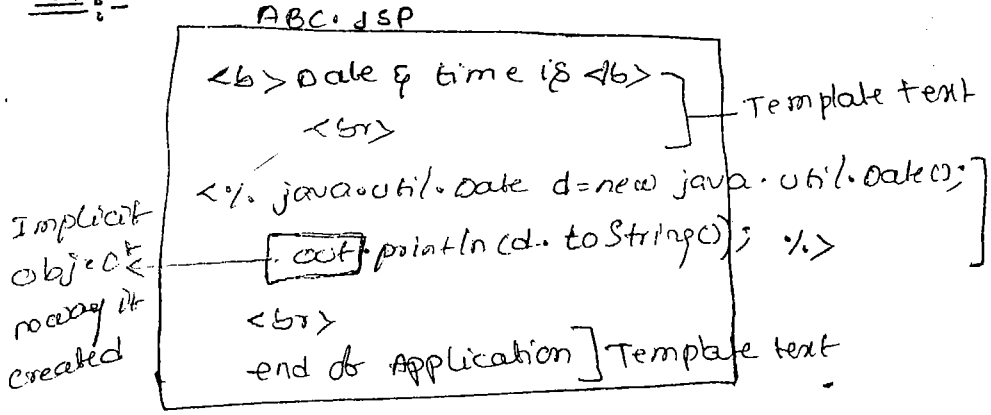
→ In weblogic server the weblogic.jar file represents JSP API and other J2ee API's.

structure of JSP



Template text = ordinary text + Html code + xml text of jsp.

Eg:-

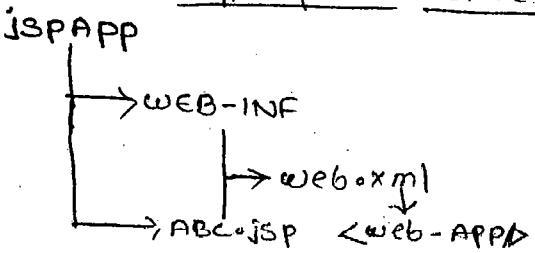


→ the scriptlet of JSP contains java code and this java code generates dynamic content as response.

is static content so the response of Jsp is Dynamic responce but also contains static content given by Template text.

→ Note: In the above code the object out is the implicit obj of Jsp. which comes automatically programmer never creates implicit objects of Jsp manually.

Deployment Directory



→ configuration of Html files & Jsp files in web.xml file of web Application is optional but configuration of servlet in web.xml file is mandatory. Any webresource i.e kept under subdirectorys of private directory called WEB-INF like Servlet must be configard in web.xml file.

→ the webresources that are kept out side the WEB-INF folder (or) parlall to WEB-INF folder like (JSP & Html) are optional to configar.

→ WEB-INF foldder is called private directory of WEB-App because that directory & sub Directory are visible only to the webserver where webApp is deployed but not visible to the Applications & users of which reside outside the webserver.

```
<b><center> current Date and Time is. </center></b>
<br><br>
<%. java.util.Date d=new java.util.Date();
    out.println(d.toString()); %>
<br><br>
    End of the Application
```

web.xml

```
<web.xml/>
```

→ To Deploy above JSPApp webApplication move JSPApp folder to tomcat/home/webApps Folder.

→ to request JSPApp folloing URL: http://localhost:4040/JSPAPP/ABC.JSP;

→ when jsp is requested the page compilar avelible in underlaying webserver converts jsp into an equalent Servlet by veriftying the syntax of jsp code.

Q → How many types of objects can be there in jsp.

A :→ In JSP we can see two types of object

1. Implicit / Built in objects

These objects will be created automatically in JSP equalent servlet so that our jsp code can use these objects directly with out any code of creating them they are.

① out ② request ③ responce ④ config

⑤ Application ⑥ Exception ⑦ session ⑧ page ⑨

→ the objects that are explicitly created by the programmer in the java code placed in jsp are called explicit object.

Eg: java.util.Date d=new java.util.Date();
(d: explicit object)

note:- All implicit obj of JSP are automatically created in JSP equalent servlet by page compilar.

Name	Type	scope
out	javax.servlet.jsp.JspWriter(C)	page (specific to one JSP page)
request	javax.servlet.http.HttpServletRequest(I)	request scope
response	javax.servlet.http.HttpServletResponse(I)	response
config	javax.servlet.ServletConfig (I)	page (a) current
application	javax.servlet.ServletContext(I)	application
exception	java.lang.Throwable (C)	page
session	javax.servlet.http.HttpSession (C)	session
page	"this" keyword	page.
pageContext	javax.servlet.jsp.pageContext(C)	page

P. → page scope means object is specific to each jsp page.

→ request scope means obj is visible in multiple web reserces of webApp which are there to process single request through chaining.

→ Session scope means visible in the all web resources of webApp but specific to a client (browser window)

→ Application scope means visible in all web resources of webApp but not specific to a browser window

Note :→ while creating explicit obj / user defined obj in jsp the built in obj names must not be used.

→ the page compiler generated jsp equalent servlet in all the server is always HttpServlet type servlet.

18/01

Life cycle methods of jsp :-

① jspInit()

② _jspService() (HttpServletRequest req, HttpServletResponse res)

③ jspDestroy()

→ jspInit() lifecycle method executes only when jsp equalent servlet class object is created.

→ _jspService() executes for every request i.e comming to jsp,

→ jspDistroy() executes when jsp equalent servlet class object is about to distroy

out.println("\r\n") = carrage return - (Enter key)

→ In tomcat server the jsp equalent servlet will be ~~saved~~ available in tomcat-Home/work/catalina/localhost/jspApp/org/apache/jsp folder.

webApp name

↳ package name of jsp equalent servlet.

java code kept in ... in the ... of jsp ...
same as it is in _jspService() of jsp equalent servlet

→ the jsp equalent servlet will generate/contain only _jspService() as default lifecycle() methods, Inorder to work with other two lifecycle methods they must be added manually.

→ when we diploy & execute jsp in multiple web, Application Servers the output of jsp will be same but the source code generated by of jsp equalent servlet will change based on the underlaying webServer or Application server.

→ the tomcat server takes x_jsp as jsp equalent Servlet class name when the jsp file name is x.jsp.

→ the weblogic server gives jsp equalent servlet classname as __x when the jsp file name is x.jsp,

→ when jsp is executed from examples server domain of weblogic the jsp equalent servlet will be generated in the following folder:-

bea home/ weblogic 90/ samples /domains/ wl_server/ servers/ exampleservers/ tmp/ _wl_user/ _appsdir_jspApp_dir/ orgens/ jsp_servlet

→ the page compilar of weblogic whose name is Jspc will automatically distroy source code file of jsp equalent Servlet the moment compiled code is generated. (that mean distroy .java of jsp equalent Servlet the movent .class is generated)

url pattern for jsp is always optional.

→ to provide url pattern for jsp write following code in web.xml file

```
<web-App>
    <servlet>
        <servlet-name>a1</servlet-name>
        <jsp-file>/ABC.jsp</jsp-file>
    </servlet>
    <servlet-mapping>
        <servlet-name>a1</servlet-name>
        <url-pattern>/#$url</url-pattern>
    </servlet-mapping>
</web-app>
```

(mandatory)

→ url pattern for jsp is required to avoid technology name and resource name from the end users of webApp.

→ There is posibility providing/configuring multiple url pattern for a single jsp or Servlet.

Q* → what happens if the jsp equalent Servlet source code will directly modify by the programmer,

(A) :- the modification related output will be reflected on the browser window if only when the following two situations are satisfy other wise the modification related changes will not be effected.

① Recompile the jsp equalent manually & reload the webApp.

② make sure that source code in jsp file is not going to be modify after step 1 tasks.

jsp equalent servlet manually we need to add the following two jar files.

① tomcat Home/common/lib/jsp_api.jar

② tomcat Home/common/lib/jasper_runtime.jar.

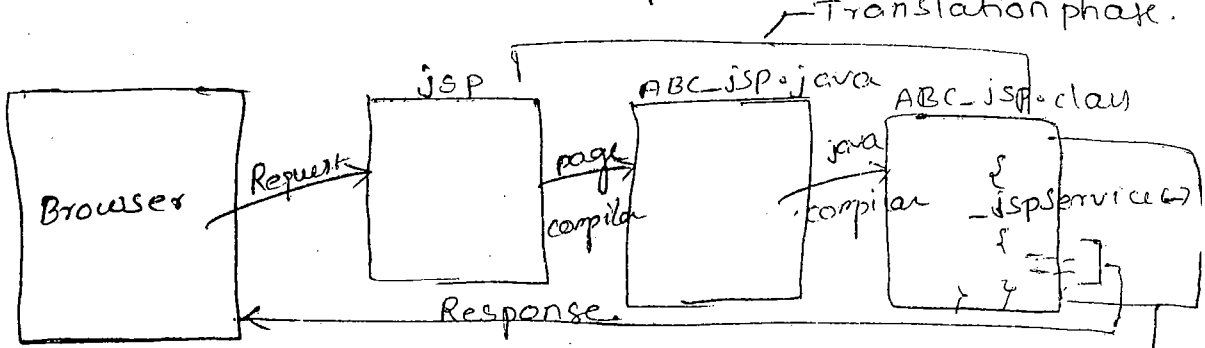
→ the moment ~~jsp~~ e source code of jsp is modified the jsp equalent servlet of that jsp will be regenarated.

A|01

→ Every jsp page participate in two phases during it's execution.

1. Translation phase :- In this phase JSP will be converted in to an equalent servlet source code and compiled code.

2. Request processing phase :- In this phase jsp equalent servlet will be executed and the genarated output comes to browser as responce.



execution of _jspService() is Request processing phase..

→ In Request processing phase the _jspService(-,-) of jsp equalent servlet will be executed and the genarated output goes to browser as response.

→ The jsp will participate directly in request processing phase when compiled code of jsp equalent servlet is

when compare to previous request. In all other situations the request comming to JSP will participate in both request- Translation phase, Request procesing phase.

Q → can we Genarate JSP equalent servlet outside the server?

A → New webservers & Application servers are providing tools to perform this operations the advantage of this process is we can genar know mistakes of JSP outside the Server before actually requesting the JSP.

→ the weblogic server supplying weblogic.jspc is command prompt level tool to genarate weblogic specific jsp equalent servlet from outside the server to work with this tool bea Home/weblogic9/Server/lib/weblogic.jar file must be added in the classpath.

→ In order to work with weblogic.jspc tool from the command prompt make sure that following 3 jar files are added to the class path.

(i):- weblogic.jar

(ii):- rt.jar (java Home/jre/lib folder)

(iii):- tools.jar (java Home/lib folder)

~~E:\temp~~:> java weblogic.jspc

E:\
└ temp
└ ABC.JSP

E:\temp> java weblogic.jspc -compiler javac ABC.JSP

If give -compiler

→ the above servlet gives jsp equalent servlet compiled code in the form of ~~ABC.class~~ but destroyes source code of JSP equalent servlet.

preserves .java file of JSP equalent Servlet after generating compiled code.

→ Script code means small amount of code Scriptlet in JSP is nothing but java code placed in JSP.

→ java is the only language i.e allow to place script code of JSP the JSP tags which allow java code are called scripting tags or scripting elements.

→ There are 4 catogiries of JSP tags.

1. Scripting elements
   - (i) scriptlet
   - (ii) Declaration
   - (iii) Expression
2. comment Elements
   - (i) JSP comment
3. Directive elements
   - (i) page Directive
   - ii) Include directive
   - iii) Tag lib directive.
4. standard action Elements
   - (i) <jsp:forward>
   - (ii) <jsp:include>
   - iii) <jsp:usebean>
   - iv) <jsp:setproperty>
   - v) <jsp:getProperty>
   - vi) <jsp:param>
   - vii) <jsp:plugin>
   - viii) <jsp:fallback>

- - -
= - - -

etc

(A):→ jsp tags are caseSencitive but the html tags are not caseSencitive.

→ Html tags can genorate only static content as responce where as jsp tags can genorate both static & Dynamic content as responce.

→ Html tags dose not allow java code where as jsp tag allows java code as script code.

→ we can't develop user defined Html tags but there is a posibility of developing user defined jsp tag.

→ Html tags will be executed at client side (browser). jsp tags will be executed at server side (webserver or application server).

→ Html tags will be recognized by the html interpriter available in the browser s/w.

→ jsp tags will be recognized by page compilar available in the webservers & application servers.

note:- Both jsp & Html tags can be there in a jsp program.

Q→ what is the diff blw printwriter & jsp writer?

(A):→ Every webresources of webApplication maintains it's own butter at server side. when webresorces is using printwriter obj the output of web resources will not be stored in butter & directly goes to browser as responce.

→ when webresorce uses jsp writer the output of web resource will be stored in the butter before it goes to browser as responce. that means jsp writer perform buttering operations & print writer dosn't perform buttering operations.

→ Every jsp contain 8 kb butter by default & jsp writer is the classname of the implicit object "out" when jsp ~~enabled~~ butter is enabled w.r.t out

→ JSP output to browser as response.

→ printWriter class is available in java.io package

→ jspwriter class is available javax.servlet.jsp package.

22/01 per

23/01

prog

→ Develop a JSP by using all the three Scripting element tags & template text:

first.jsp

```
<%! public long multiply(int x, int y)
    {
        return x*y;
    } %> <br><br>

<b> current Date & time is: </b> <%=new java.util.Date() %>

<br> <% int x=10;
        int y=20; %> <br><br>

The x value is <%= x %> The y value is <%= y %><br><br>

The multiplication of <%= x %> and <%= y %> is <%= multiply(x,y) %>
<br><br>:
```

Same program to develop xml syntax :-

```
<jsp:declaration>
    public long multiply(int x, int y)
    {
        return x*y;
    }
</jsp:declaration>

<b> current Date and Time is: </b> <jsp:expression> new javaoutil.
                Date() </jsp:expression> <br>

<jsp:scriptlet> int x=10;
        int y=20; </jsp:scriptlet>
```

<jsp:expression> y </jsp:expression>

the multiplication of <jsp:expression> x </jsp:expression> and <jsp:expression> y </jsp:expression> is <jsp:expression> multiply (x,y) </jsp:expression>

=

→ while working with xml syntax of scripting elements we can't write java code in those tags by using (<) lessthan symbol as relational operator because there the lessthan symbol will be taken as begining of the subtag inside the scripting element & ~~will~~ it will not be treated as relational operator.

problem eg <jsp:declaration>

```
public long multiply(int x, int y)
{
  if (x<0 || y<0)
  {
     return x+y;
  }
  else
  return x*y;
}
```

</jsp:declaration>

 Results <jsp:expression> multiply(10,20) </jsp: expression>

the problem in the above code is the lessthan "(<)" in ~~the~~ x<0 statement will not be treated as relational operator it will be treated as the begining of subtag. due to this page compilation fails & JSP execution will also fails.

To solve this problem write entair body content of <JSP:declaration> as <![CDATA[content of xml.

<![CDATA[

```
public long multiply (int x, int y)
{
   if (x<0 ; y<0)
   {
   } return x+y;
     else
     return x*y;
}
```

]]>

</jsp:declaration>

 Result <jsp:expression> multiply (10,20) </jsp:expression>

→ while working with expression tag having xml syntax we can't evaluate the expression i.e having '<' symbol. By using given expression as CDATA Based expression we can't solve this problem. Eg

```
<jsp:scriptlet>
  <![CDATA[
      boolean res = 10<5      ✓
    out.println (res);
    ]]>
</jsp:scriptlet>
```

Eg 1

```
<jsp:expression> 10<5 </jsp:expression>      ×
```

Eg 2

```
<jsp:expression>
   <![CDATA[
      10<5          ×
   ]]>
</jsp:expression>
```

<%=10<5 %> ✓

~~<jsp: expression>~~ └ standard syntax

Ex4

```
<jsp:scriptlet>
    boolean res = 10<5;
    out.println(res);
</jsp:scriptlet>
```

×

Ex5:

```
<% boolean res = 10<5;
   out.println(res); %>
```

✓

→ It is always recomanded to work with standard syntax based jsp tags while developing the logics in JSPs of webApp because there is no necessity of dealing with CDATA based java code when tags are having logics based on "<" symbol.

procedure to develop JSP's of webApp by using Netbeans IDE

→ NetBeans 6.x versions ~~come~~ ~~with~~ gives glash-fish 2.x server as built in Server.

→ NetBeans 5.x gives tomcat 5.x as built in server.

Step no1:- lanch NetBeans IDE & create webproject.

File menu
└→ New project
  └→ web
    └→ webApp
      └→ next
        └→ project name: myprof
          └→ next
            └→ finish.

Rightclick on webpages folder of netbeans project
↳ new
↳ JSP : jsp filename : test
options ⊙ jsp file standard syntax
○ jsp document
↳ Finish
↓
→ write the code of your choice

③ Test the JSP.

Rightclick on project
↳ Run

24/01

→ Every jsp page can have 3 types of commenting symbols.

① jsp comment (hidden comments)
↳ To comment jsp tags of jsp page

(<%-- ... --%>)

② Html comments (Template Text)
↳ To comment HTML code ordinary text ~~to gather~~ (template text) of jsp page

(<!-- ... -->)

③ Java comments (script comment)
↳ To comment java code of jsp

single line : //

multi line : /* ... */

where as Html comment will be recognized by ~~Html~~ interpretor and java comments will be recognized by java compilar

NOTE :→ when compilar or interpretor recognizes the comment the commented area of the code will be replaced with white spaces due to this the result of commented code will not apper in the output.

Type of comments	comment symbols visibility: In Source code of equalent Servlet	In the compiled code of equlant Servlet	In the Html code that goes to browser	output that com on the Browse
Jsp comment (<%-- == --%>)	×	×	×	×
Java comments (// (or) /* == */)	✓	×	×	×
Html comments (<!-- --- == --->)	✓	✓	✓	×

→ Since JSP comments are not visible in various phases of JSP execution they are called as Hidden comments.

→ we can't use JSP comments to comment java code of scripting element but we can use JSP comments to comment JSP tags and template text.

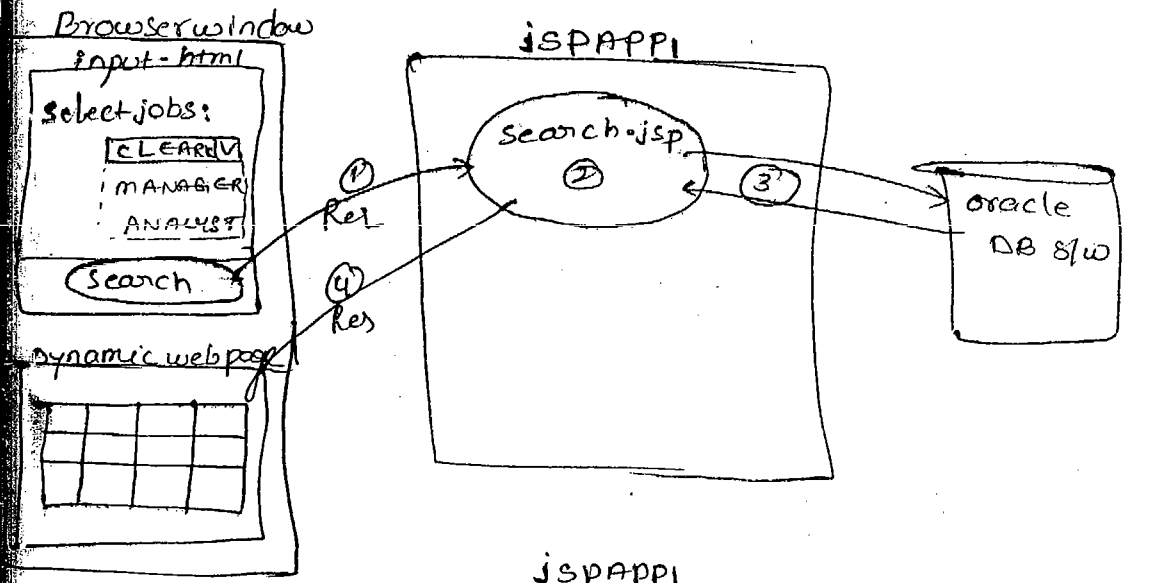
→ we can't use Html comments to comment java code of JSP but we can use them to comment JSP tags template text of JSP.

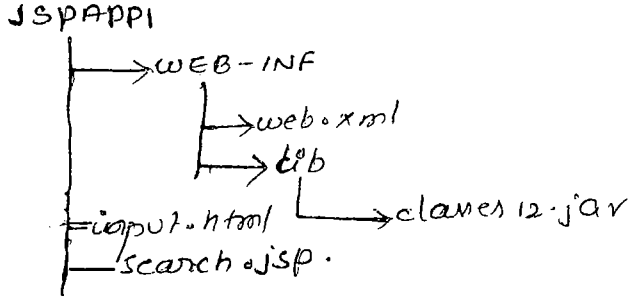
→ we can't use java comment to comment template text & JSP tags of JSP.

pattern of jsp either as action url of form page (or) href url of hyper link.

→ For jsp to DB s/w communication place jdbc code in the scripting elements of jsp, but there is need of performing exception handling because the jsp equalent servlet will take care of exception handling automatically.

→ Application





input.html

```
<form action="search.jsp" method="get">
<b> select job </b> <select name="desg">
        <option value="CLARK"> clark </option>
        <option value="MANAGER"> managers </option>
        <option value="ANALYST"> analysts </option>
        <option value="SALEMEN"> salemen </option>
</select>
<input type="submit" value="search">
```

1. *.jsp page

```
<%@ page import="java.sql.*;" %>
<% Class.forName("oracle.jdbc.driver.OracleDriver");
   Connection con = DriverManager.getConnection("jdbc:oracle:
                    thin:@localhost:1521:satya","scott
                    ","tiger"); %>
<% String designation = req.getParameter("desg");
PreparedStatement ps = con.prepareStatement("select * from
                       emp where job=?");
   ps.setString(1,designation);
   ResultSet rs = ps.executeQuery();
ResultSetMetaData rsmd = rs.getMetaData();
   int colcnt = rsmd.getColumnCount();
out.println("<table border=1>");
out.println("<tr>");
for(int i=1; i<=colcnt; ++i)
   out.println("<td>"+rsmd.getColumnLabel(i)+"</td>");
   out.println("<tr>");
while(rs.next())
{
   out.println("<tr>");
   for(int i=1; i<=colcnt; ++i)
   out.println("<td>"+rs.getString(i)+"</td>");
   out.println("</tr>");
}
   out.println("</table>"); %>
<% rs.close();
   ps.close();
   con.close(); %>
```

// logic to print data of the result set obj in the form of Html table.

web.xml

```
<web-app/>
```

page :→ this tag is given to provide global information to jsp page like specifing buffer size enabling (or) disabling session, enabling (or) deseabling autoflush of buffer configuring errorpage for JSP & e.t.c operations.

racle:
→ The tag names & attributes names of JSP are case sensitive.

:cobt"

Standard syntax

~~<%-page~~

<%@ page attributes %>

rom

xml syntax

<jsp:directive.page attributes%>

attributes

language="java" default → java is default & only one langue that we can pass here as a value

import="java.util.*,java.net.*" --- // no default value

extends = "class name" // not recomonded to use, no default value

contentType = "text/html" // response content type // no default

buffer = "8kb" or "none" // default = "8kb"

autoFlush = "true" (false) // default = true

session = "true" (false) // default = true

errorPage = "url" // no default

isErrorPage = "false" (true) // default → false

isThreadSafe = "true" (false) // default → true

info = "string" // the short desc about jsp page // no default

isELIgnored = "true" (false) // default → true.

Ex:- <%@ page isThreadSafe = "true" %>

makes jsp equalent servlet as thread safe servlet by implementing javax.servlet.SingleThreadModel().

Note:- By default every jsp is threadSafe jsp because is

<%@ page buffer = "10kb" %>

Ex:3 → <%@ page buffer="none" autoFlush="true" %>

The above statement is wrong statement. when there is no buffer, there is no posibility of Flushing the buffer.

→ In order to desable the buffer for jsp write the following statement <%@page buffer = "none" %>

Q → when jsp buffer size is lesser than response content size generated by Jsp can you tell me what happens when that jsp page is requested.

A → ~~total~~ Total output jsp page goes to browser port by port having delay to display the webpage on the browser window.

Ex:- <%@page session = "false" %>

implicit obj is session will not be created.

→ the implesit obj session is not the regularly used obj of jsp programming so that ~~troger~~ provision is given to stop getting session as the implesit obj

→ points to remember while working with page directive tags.

① page directive tag name & attribute names are casesensitive.

② Except import attribute the attributes of page directive tag dose not allow multiple values as a comma sapa rated list.

③ Except import attribute the attributes of page directive tag should not be repeated in a single ~~jsp~~ page directive tag (or) in multiple page directive tags of jsp having different values.

④

session="true" %> (invalid)

except import attribute, other attribute of page directive should not be allowed where session will be repeated.

④ invalid attribute names will not be recognised by JSP.

Ex: <%@ page info="firstojsp" message="hello" %>

the above code generates page compilar error. because "message" is an invalid.

→ for related information on Attributes of page directive tags refer page no: 15 & 16;

⑤ → what is the diff b/w implicit object page & pageContext

Ⓐ → Implicit object page holds "this" keyword as the value. Representing reference of the jsp equalent Servlet class obj. this obj is very useful to differenciate instance variable of jsp from local variable.

the implicit obj ~~page context~~ pageContext is always constructed by using multiple details of jsp page like jsp equalent servlet obj, req, res, buffersize, status of session obj, autoFlush, errorPageName details. most of the implicit objects of the jsp are created based on page context obj, ~~etc~~

In jsp equalent Servlet both these objects are created as shown bellow.

- object page = this.
- pageContext = jspxFactory.getpageContext(this, request, response, ~~err~~
  session, autoflush
  "abc.jsp", true, 7168, true).
  errorpage, buffer size

→ Exception Handling in JSP

In JSP there is no need of handling exceptions manually because the JSP equalent servlet automatically takes care of exception handling but this exception handling displays

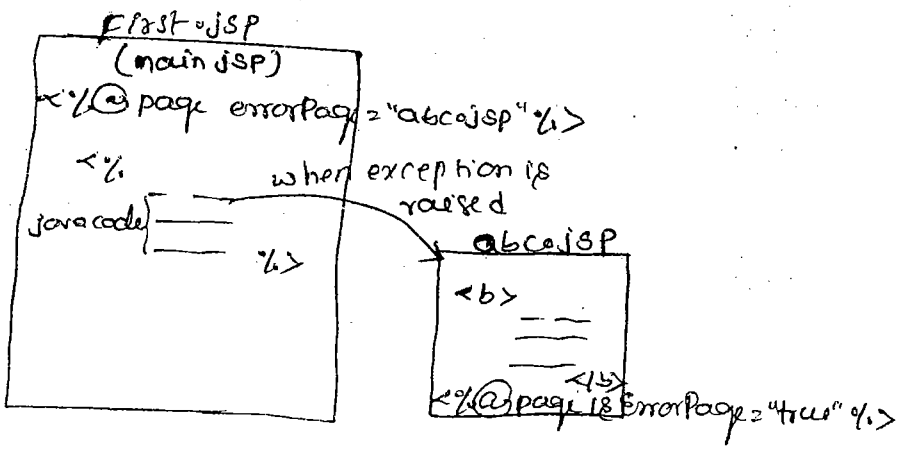
can't follow them.

→ Inorder to help the enduser it is recomand to handle the exceptions in jsp page manually by configaring error pages for jsp. The jsp or html page that executes when exceptions are raised in other jsp page of webApp is called errorpage.

Exception handling in JSP

local exception handling
→ it is specific to one jsp page
[use errorPage, isErrorPage attributes of <%@ page %> tag]

global exception handling
→ common for all the JSP's of a webApp
[use <error-page> of web.xml file]



→ The Implicit obj exception is visible only in those JSP pages that are acting as errorPages inorder to make jsp page as errorPage use isErrorPage="true" attribute of pageDirective tag in that jsp page.

→ In the above diagram exception is raised in the java code of mainpage the control automatically goes to errorpage called abc.jsp & output of fist.jsp will be discarded, only the output of abc.jsp goes to browser as response.

file or jsp file as errorpage. when Html file is taken as error page we can't work with implicit obj exception. while working with local exception handling we can use one error page as errorpage of multiple jsp's.

→ Global exception was not supporting to tomcat server and we will take another server (weblogic 9.0 and e.t.c).

:- local exception

```
<!-- first.jsp (main jsp) -->
<%@page errorPage="abc.jsp" %>
<b> First.jsp </b>
<% int x=Integer.parseInt("130"); %>
```

=

errorpage

```
<!-- errorpage.jsp -->
<%page isErrorPage="true" %>
<b> abc.jsp </b>
<br>
<b> Exception is raised in First.jsp </b>
<b> Exception is <%= exception.toString() %>
```

=

code for global exception handling in JSP's

web.xml

```
<web-app>
    <error-page>
        <exception-type> java.lang.Exception
            </exception-type>
        <location> /abc.jsp </location>
    </error-page>
</web-app>
```

exception is raised in any jsp page of webApp.

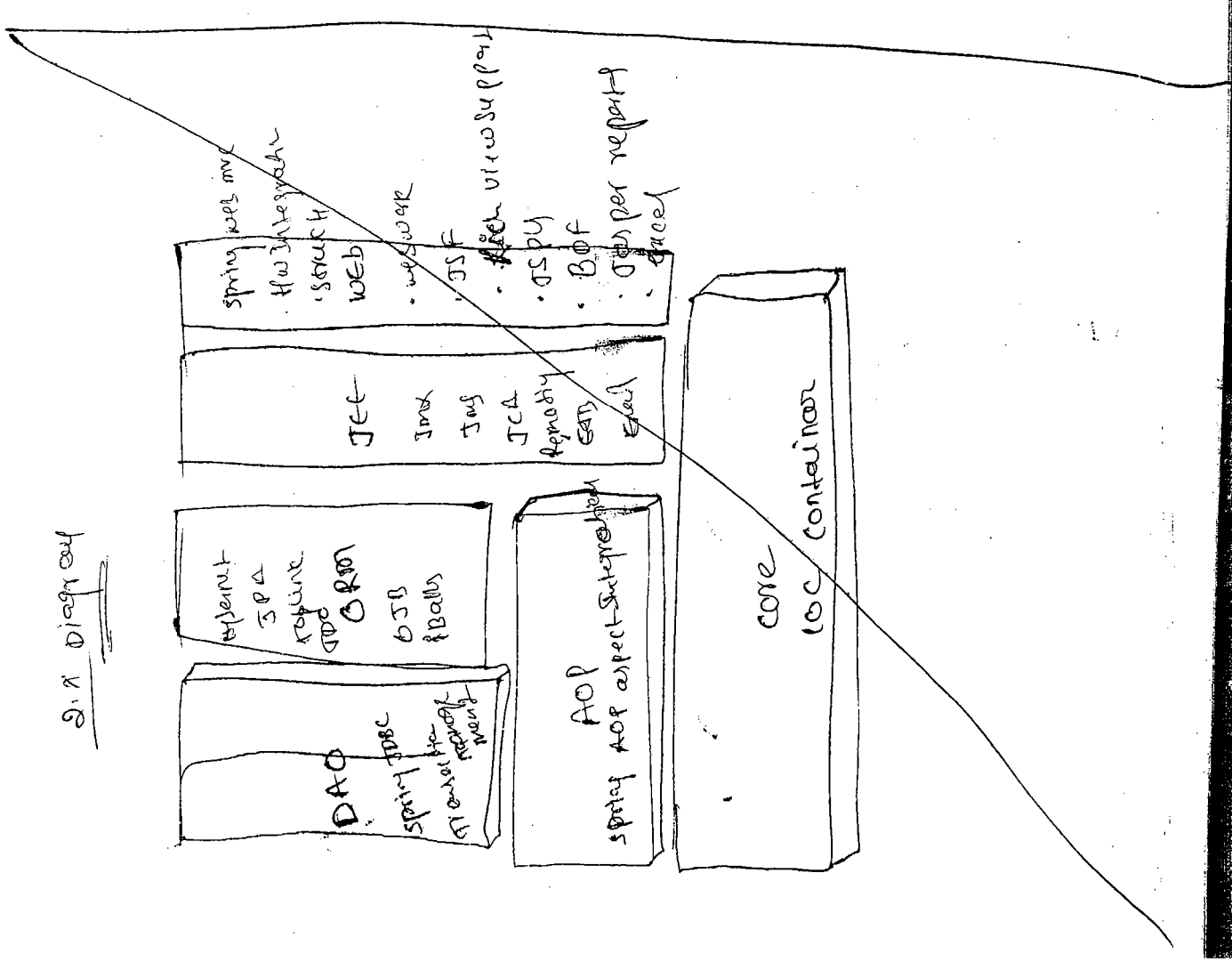
→ Global exception handling also allows to take Html file as errorPage.

⇒ if Global exception handling and local exception handling is configured for certain jsps of webApp having two diff pages as errorPages which errorPage will be executed when the exception is raised in the jsps.

(A) → the errorPage configured through local exception handling will be attached when exception is raised in jsps.

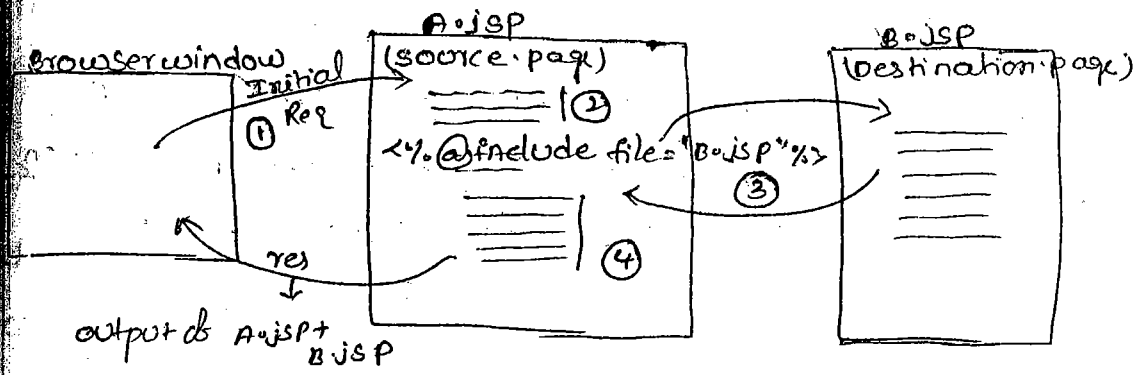
→ the Global exception Handling code kept in web.xml file will not handle the exceptions that are raised Servlets of the webApp.

→ the errorPage i-e configured in localexception Handling can work with the Implicit obj exception -



...t workwith implicit obj exception.

include Directive tag:



③ → the code of B.JSP will be included in the source code of jsp equalent servlet belonging to A.JSP at translation phase

→ Directive include tag is given to include of destination web resorce in the source code of source jsp page equalent servlet at translation phase due to this saparate servlet will not be generate for destination jsp page

Syntax: <%@include attributes%> ← Directive include. standard syntax.

@ A.JSP

```
<!-- A.JSP -->
<b>Begining of A.JSP</b>
<br>
<%@ include file="B.JSP"%>
<br>
<b> end of A.JSP</b>
```

B.JSP

```
<!-- B.JSP -->
<b>in B.JSP</b>
```

web.xml

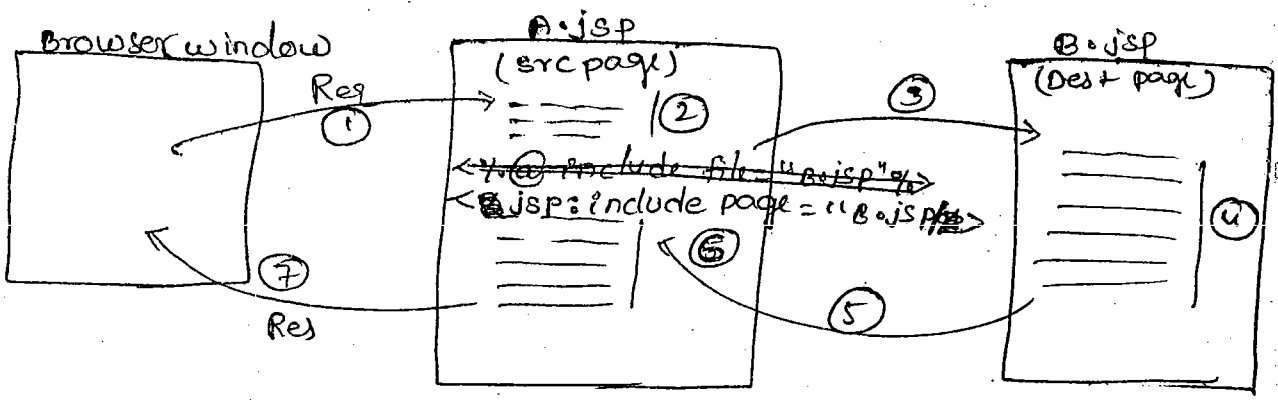
```
<web-app>
```

<%@ jsp:directive.include attributes %>

<jsp:include> (action include tag)

Syntax <jsp:include attributes/> .

Note :- All action tag of jsp only will have xml syntax. <jsp:include> tag is given to include the output of Destination webresources in the output of source jsp page Dynamically at request processing phase.



③ ④ ⑤ ⑥ B.jsp executes saparately and is output will be including in the output of source jsp called A.jsp.

Ex A.jsp

```
<!-- A.jsp -->
<b> Begining A.jsp </b>
<br>
<jsp:include page="B.jsp"/>
<br>
<b> end of A.jsp </b>
```

B.jsp

```
<!-- B.jsp -->
<b> In B.jsp </b>
```

web.xml

```
<web-app>
```

programming.

Q → what is diff b/w Directive include tag and action include tag of jsp

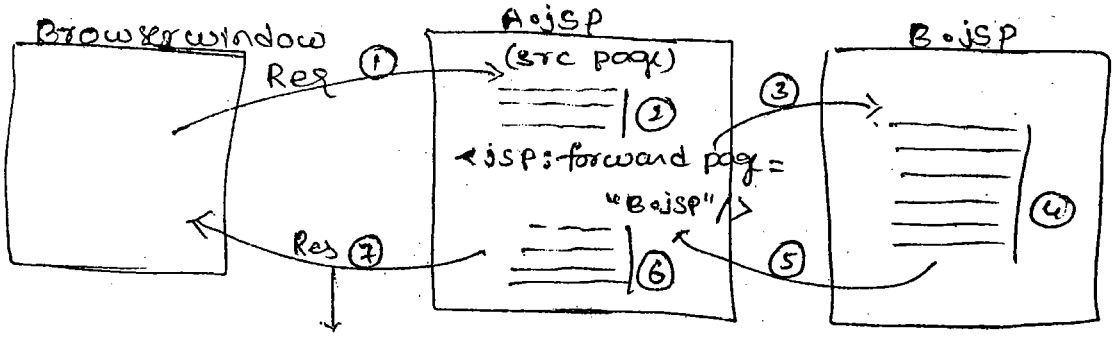
Directive include	Action include
→ Design to include code of destination webresource in the code of jsp equalent Servlet genarated source jsp page.	→ Design to include output of Destination webresource in the output of source jsp page.
→ code inclusion takes place at translation phase so this operation is called static binding or compile time binding.	→ output inclusion takes place at request processing phase of jsp so this operation is called runtime binding or Dynamic binding.
→ use directive include when destination webresource is static webresource like html file	→ use action include when the destination webresource is Dynamic webresource like Servlet, JSP.
→ It is not alternate for rd.include() of Servlet programming.	→ It is alternate for rd.include() of Servlet programming.
→ Directive include tag gives both standard, xml syntax	→ Action include tag gives only xml syntax.
→ attribute name directive include tag is file	→ Attribute tag in action include tag is page

Note :- most of the jsp programmer prefer working with action includetag.

27/01

→ In weblogic 9.x the JSP webresources based webApplications must not be deploy through hard deployment process always use the war file based console deployment.

Syntax : <jsp:forward attributes/>



output of A.jsp will discarded. and only output of B.jsp goes to browser.

→ jsp:forward tag is given to forward the request to destination web resource from the source jsp page.

→ <jsp:forward> tag is given as alternate tag for rd.forward() of Servlet Programming.

→ <jsp:forward> tag forwards the request to destination web-resource dinamically at runtime, so it the destination webresorces is jsp we can see one saparate jsp equalent saparate for that.

Q → why there is no directive forward tag.?

A → Directive tags perform their operations at translation phase by including code of destination webresource in the source jsp page equalent Servlet source code, because of this behaviour there is no posibility of discarding the output of source jsp page. but forward tag demands to discard the output of source jsp page due to this reason directive forward tag is not given.

Example application on <jsp:forward>

A.JSP

```
<b>Begining of A.JSP</b>
<jsp:forward page="B.jsp"/>
<br>
<b> end of A.JSP</b>
```

B.JSP

```
<b> In B.JSP</b>
<br>
<b> B.JSP ended</b>
```

web.xml

```
<web-app/>
```

windows webresources of webApp, attribute name must be string and value must be an object.

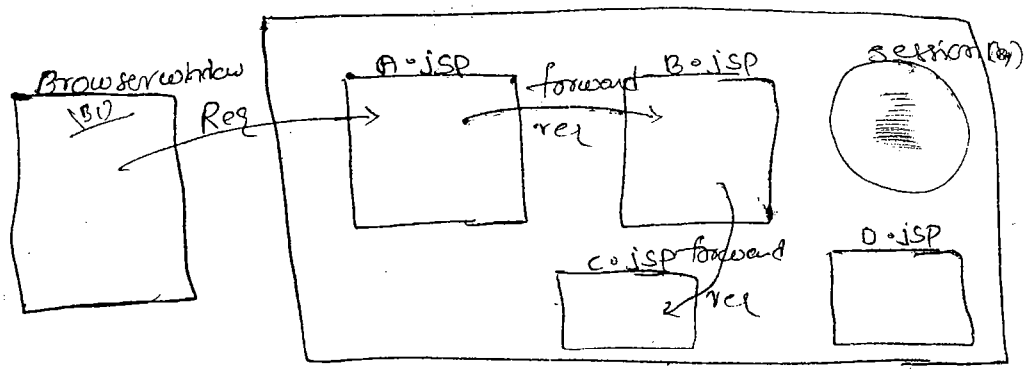
→ In servlet programming we can work with 3 types of attributes

1. Request attributes
2. Session "
3. ServletContext "

→ In jsp programming we can work with 4 types of attributes.

1. Request Attribute.
2. Session "
3. ServletContext / application attribute.
4. pageContext attribute.

Note :→ using pageContext attribute we can create and work with request attribute, session attribute, ServletContext attribute & page scoped attribute.



→ Request attributes are visible through out the Req cycle. Req attributes are visible in those web resources of webApp who share same req & res obj in the above diagram. request attribute created in A.jsp is accessable in B.jsp & C.jsp. but not in D.jsp

→ Session attributes are visible in all the webresources of webApp irrespective of their req obj but they are specific to a browser window.

In the above diagram session attribute created in A.jsp is

→ Application attributes / ServletContext attributes are global attributes so they are visible in all the web resources of webApp irrespective of their req obj & irrespective of browser window's from which they are getting the request.

→ In the above diagram application attribute created in A.jsp is visible in all jsp's of webApp irrespective of all conditions.

→ use setAttribute() method to create & modify attribute.

→ use getAttribute() method to read attribute value.

→ use removeAttribute() method to remove the attribute value.

→ Ex

A.JSP (request, session, & application scope attributes) (creating)

```
<%request.setAttribute("name1","raja1");
  session.setAttribute("name2","raja2");
  application.setAttribute("name3","raja3"); %>
<jsp:forward page="B.jsp"/>
```

B.JSP

```
<b>B.JSP</b> reading attribute
<br>
<%=request.getAttribute("name1")%><br>
<%=session.getAttribute("name2")%><br>
<%=application.getAttribute("name3")%> <br>
<jsp:forward page="c.jsp"/>
```

B.JSP

```
<b>D.JSP</b>
<%=request.getAttribute("name1")%>
<br>
<%=session.getAttribute("name2")%>
<br>
<%=application.getAttribute("name3")%>
```

c.JSP

```
<b>c.JSP</b>
<%=request.getAttribute("name1")%> <br>
<%=session.getAttribute("name2")%> <br>
<%=application.getAttribute("name3")%>
```

→ If a jsp is interacting with another jsp by using <jsp:forward> tag or <jsp:include> tag then both jsps will use same request & response objs.

→ Request attributes scope is better than request scope we can't use req obj to create & manage other scope attributes.

Session attribute scope is better than session scope & we can't use session obj to create & manage other scope attributes.

application obj to create other scope attributes.

page context attribute to use one of these 4 scopes based on this

- pagescope (default),
- requestScope,
- sessionscope.
- applicationscope.

28/01

pageContext attribute :-

to create

```
pageContext.setAttribute("uname","raja");
// to create pageContext attribute uname is page scope

pageContext.setAttribute("age",new Integer(30),pageContext.REQUEST_
// create pageContext attribute age is request scope.   SCOPE);
```

other scopes are

```
pageContext.SESSION_SCOPE
pageContext.APPLICATION_SCOPE
pageContext.PAGE_SCOPE
```

to modify

```
pageContext.setAttribute("uname","raja1");
// modify uname attribute of page scope

pageContext.setAttribute("age",new Integer(50),pageContext.
// modifies the age attribute of request scope.   REQUEST_SCOPE);
```

to remove :-

```
pageContext.removeAttribute("uname");
// remove uname attribute from page scope

pageContext.removeAttribute("age",pageContext.REQUEST_SCOPE);
// removes age attribute from request scope.
```

to read

```
String s1=(String)pageContext.getAttribute("uname");
// reads uname attribute value from page scope

Integer i1=(Integer)pageContext.getAttribute("age",pageContext.
                                    REQUEST_SCOPE);
```

Integer i1 = (Integer) pageContext.findAttribute("age");
// Searches and reads the attribute value from multiple scope like page Scope → Request Scope → session scope → application scope

Q → what is the diff b/w calling getAttribute() on pageContext object & calling findAttribute() on pageContext object?

A → getAttribute() method can search and read attribute value only from the specified scope where as the findAttribute() method can search and read attribute values from all the scopes.

→ the attribute create by using the regular session, request, application obj can be read by taking pageContext object.

→ Ily the attribute created using pageContext obj can be read through session, request, application obj based on the attribute scope.

→ Inorder to pass additional data b/w source jsp page and destination web resource as additional request parameter values we need to use <jsp:param> for this the source & destination web resources should communicate each other by using either <jsp:forward> or <jsp:include> tags

→ <jsp:param> tag must be used only as the sub tag are <jsp:forward> or <jsp:includ> tag

→ Ex

one.jsp

```
<b> one.jsp </b>
<jsp:forward page="two.jsp">
    <jsp:param name="bname" value="CRJ">
    <jsp:param name="author" value="yesman">
</jsp:forward>
```

two.jsp

```
<% int a=10;
   int b=20;
   int c=a+b; out.println("Result is" +c);
   request.setAttribute("result", new Integer(c));
%>
```

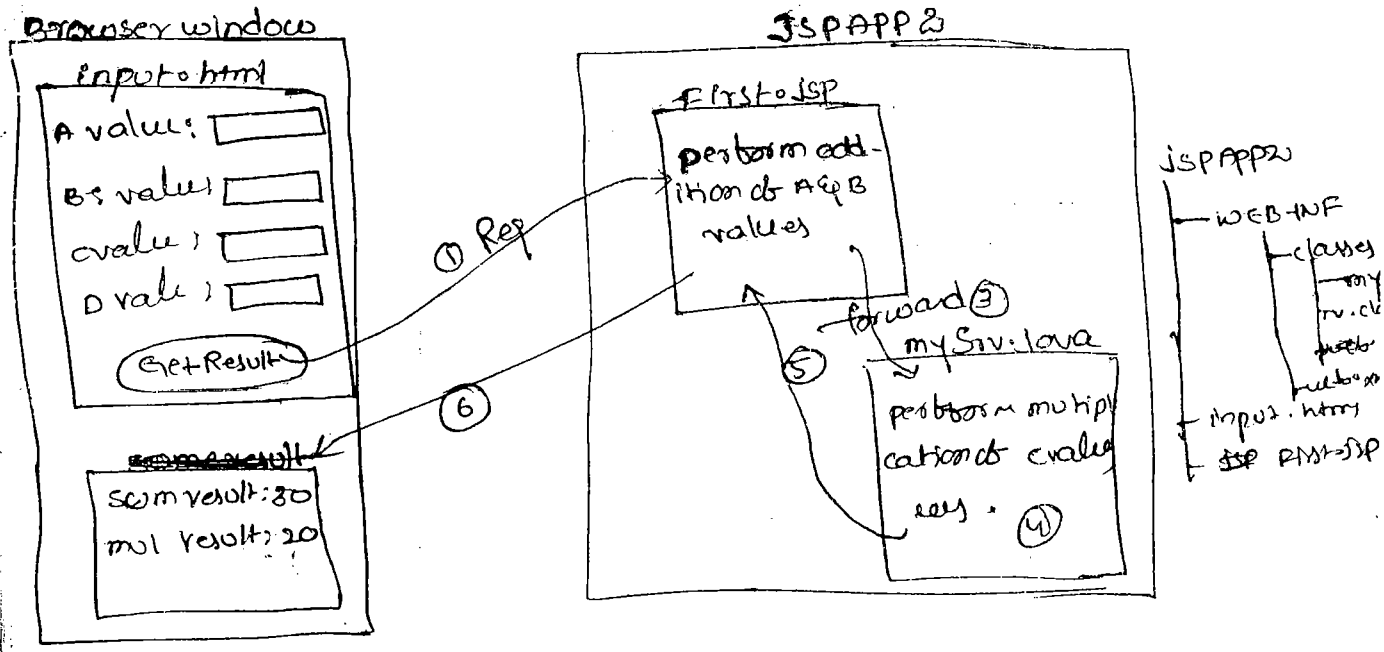
```
<br>
<b> additional request parameters are </b>
<%= request.getParameter("bname") %>
<%= request.getParameter("author") %>
<br>
<b> the Result is: </b> <%= request.getAttribute("result") %>
```

Q) what is need of working with addition request parameter various types of attributes in jsp programming?

A) when source jsp:page forwards the request to destination web resource the output of sorce jsp page will be discarded due to this the result genarated by logics of source jsp page will be discarded in order to preserve this result and in order to display this result as responce on the browser window we need to send the result to destination web resorce either in the form of additional request parameter or requestAttributes.

JSP to Servlet Communication

→ Inorder to make request comming to jsp going to Servlet to perform certain operations of request processing the jsp to servlet communication is required for this purpose the JSP can use <jsp:forward> or <jsp:include> tag.



```
<form action="first.jsp">
  A value <input type="text" name="t1"><br>
  B value <input type="text" name="t2"><br>
  C value <input type="text" name="t3"><br>
  D value <input type="text" name="t4"><br>
<input type="submit" value="getresult">
</form>
```

first.jsp

```
<%. request.getParameter
<% int a=Integer.parseInt(request.getParameter("t1"));
   int b=Integer.parseInt(request.getParameter("t2"));
   int sum=a+b;
   request.setAttribute("add", new Integer(sum)); %>
// forwarding request to servlet
<jsp:forward page="myurl"/>
```

mySrv.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class mySrv extends HttpServlet
{
   public void service(HSR req, HSR res) throws SE,IOE
   int c=Integer.parseInt(req.getParameter("t3"));
   int d=Integer.parseInt(req.getParameter("t4"));
   int mul=c+d;

// read request Attribute value.
   int sum=(CInteger) req.getAttribute("add").intValue();
   PrintWriter pw=res.getWriter();
   res.SetContentType("text/html");
```

```
pw.println("The multiplication is "+mul);
}
}
```

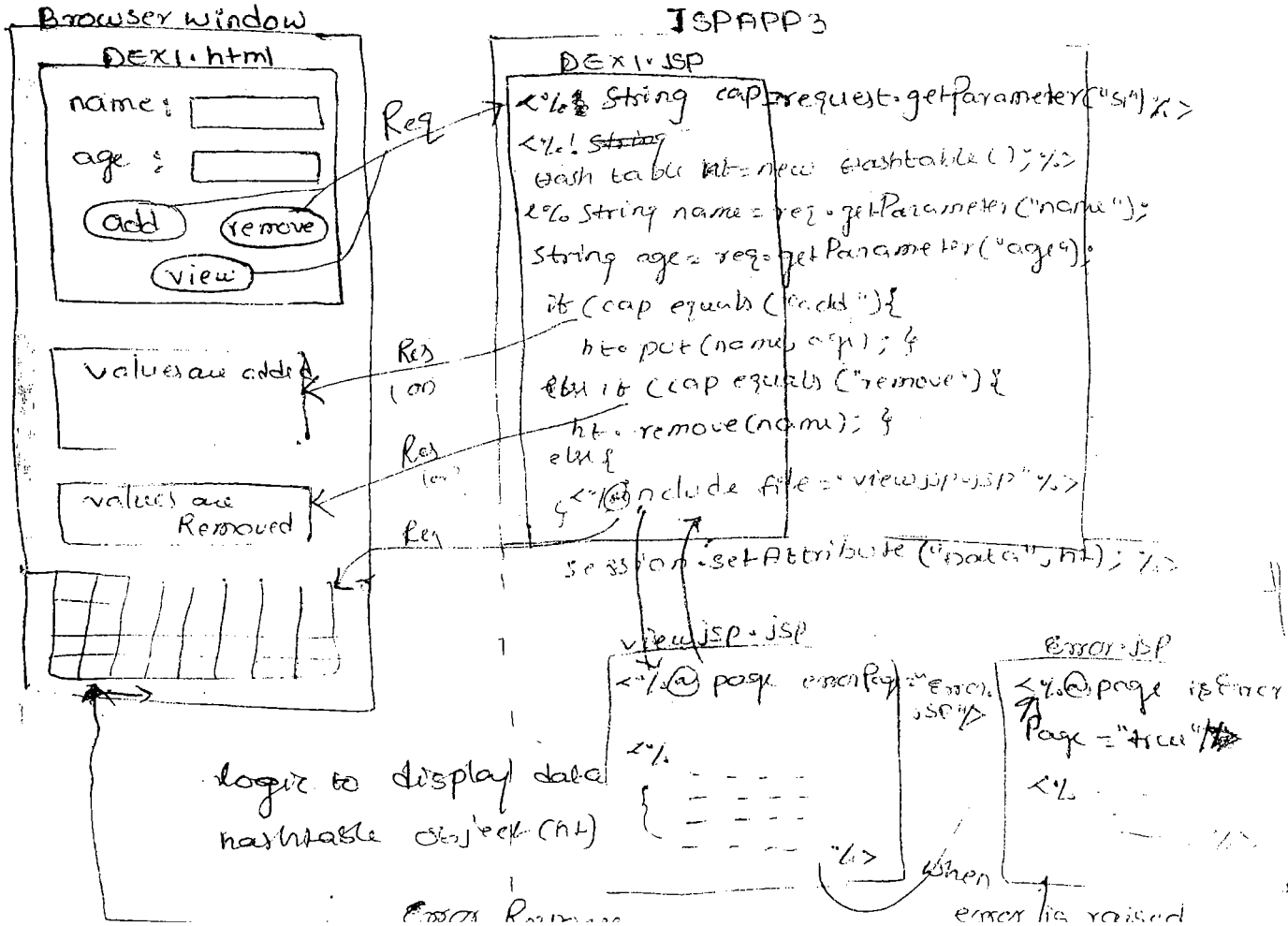
web.xml

```
<web-app>
    <servlet>
        <servlet-name>abc</servlet-name>
        <servlet-class>myStr</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>abc</servlet-name>
        <url-pattern>/myurl</url-pattern>
    </servlet-mapping>
</web-app>
```

29/01

=

→ Application



→ multiple submit buttons are taken in the form page pointing to one destination web resource. In the destination web resources submit buttons are identified based on their capsions.

2 → exceptions are handled through errorPage configuration.

→ form page Data is prisened acrok the multiple request by storing the data as session attributes.

→ For the source code of this Application refer (APP 3: given in pages: 9 & 10)

→ when multiple browser window giving req. to JSP of webapp for every browser window one new session obj will be created in the webApp.

→ when multiple browser windows are giving request to a jsp all these browser windows will use same instance variables declard in the JSP.

30/01

Q → How to store form page Data for multiple no of times as attribute value in the jsp?

A when formpage is having more than two values store them in userdefined java class object & add these objects in arraylist then make this arraylist as attribute value like sessionattribute value or ApplicationAttribute value.

<jsp:plugin>:

→ <jsp:plugin> tag is given to display applets on the browser window by internally using <embed> (or) <applet> (or) <object> tags.

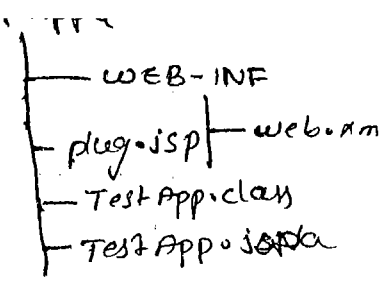
→ this tag is having subtag called <jsp:fallback> which executes when the browser doesn't support applets. so we use message of this tag to display error message on browser window when the browser doesn't support applet.

Syntax :- <jsp:plugin attributes>

<jsp:fallback> msg </jsp:fallback>

</jsp:plugin>

```
import java.applet.*;
import java.awt.*;

public class TestApp extends Applet
{
    public void paint(Graphics g)
    {
      setBackGround(Color.red);
      g.drawString("This is from", 50,50);
    }
}
```



plug.jsp

```
<html>
  <body bgcolor="pink">
    <jsp:plugin type="applet" code="TestApp.class" (1)
                codebase="/jspApp4" (2) width="300" height="200">
    <jsp:fallback>browser doesnot support applet</jsp:fallback>
    </jsp:plugin>
  </body>
</html>
```

① that class that is acting as applet

② the directory where applet class is avalible

→ javaBean is a java class that contains member variable and contain getter & setter methods for those member variables a java class that is developed by following standard set of rules is called javabean class.

→ the member variables declared in javabean class are called bean properties. getter(-) methods are usefull to set the data to bean property getter(-) methods are usefull to read the data from bean properties.

methods are mandatory.

→ Setter(-) methods of java bean programming is given to make all java programmers developing java classes standard role because of this exchanging of classes among the programmers becomes easy.

Q→ what is the diff b/w javabean & EJB ?

A→ except naming similarity nothing is comparable b/w these two concepts because javabean is a ordinary java class that contains getter & setter methods. & EJB is a distributed technology to develop distributed App, so better not to compare a class with technology.

Ex java bean class

```
public class StudentBean
{
    // bean properties
    private int no;
    private String name;
    private float avg;

public void setNO(int no)
{
    this.no=no;
}

public int getNO()
{
    return no;
}

public void setName(String name)
{
    this.name=name;
}
public String getName()
public void setAvg(float avg)
{
    this.avg=avg;
}
public float getAvg()
{
    return avg;
}
}
```

public class

→ the java bean class can have 3 types of bean properties

① simple properties

② Boolean properties

③ indexed properties.

of request, session & application attributes we need to combine formpage values in to

① user defined java class obj & add that obj as attribute value. it is recomanded to take class of this user defined obj as java bean class.

① simple properties :→ can take one simple value at a time,

```
int age;
public void setAge(int age)
{
    this.age=age;
}
public int getAge()
{
    return age;
}
```

② boolean properties :→ can take one boolean value at a time.

```
boolean flag;
public void setFlag(boolean flag)
{
    this.flag=flag;
}
public boolean getFlag()        // public boolean isFlag()
{                                {
    return flag;                     return flag();
}                                }
```

③ indexed properties :→ can store one or more values.

```
String colors[];
public void setColors(String cl[])
{ colors=cl;
}
public void setColors(int index, String value)
```

```
{   return colors;
}
public String getColors(int index)
{
    return colors[index];
}
```

→ we can create obj for java bean class in a normal way like an ordinary java class obj creation.

→ If Servlet wants to communicate with java bean the Servlet has to create object of java bean manually & it should call methods of java bean manually.

→ when jsp wants to interact with java bean we need to use following 3 tags.

① <jsp:useBean>
② <jsp:setProperty>
③ <jsp:getproperty>

} these are independent tags.

→ A java bean can be executed inside or outside of the webApp, when java bean is taken in the webApp resides in WEB-INF/ classes folder of webApplication.

31/01

→ In real world web App will be developed in two Approches

webApp Development models

1. Approch1 (Model I)
2. Approch2 (Model II) (MVC architecture)

→ In model I approch webApp will be developed by using only Servlets (or) only jsps

multiple technology like jsp, servlet, javabean & e.t.c.

→ Generally every webapp contains the following logics

1. request Data Gathering logic
2. form Data validation logic
3. B. logic
4. persistance logic
5. presentation logic
6. session Management logic
7. middle ware Services.

→ the logic i.e there to gather form data from form page is called requestDatagathering logic. (logic return through req.getParameter());

→ the logic i.e given to verify the format of the form data is called form validation logic. (checking the symbol of e-mail id, checking whether required fields are typed or not and e.t.c)

→ the main logic of the Application that performs calculations and analysis is called business logic

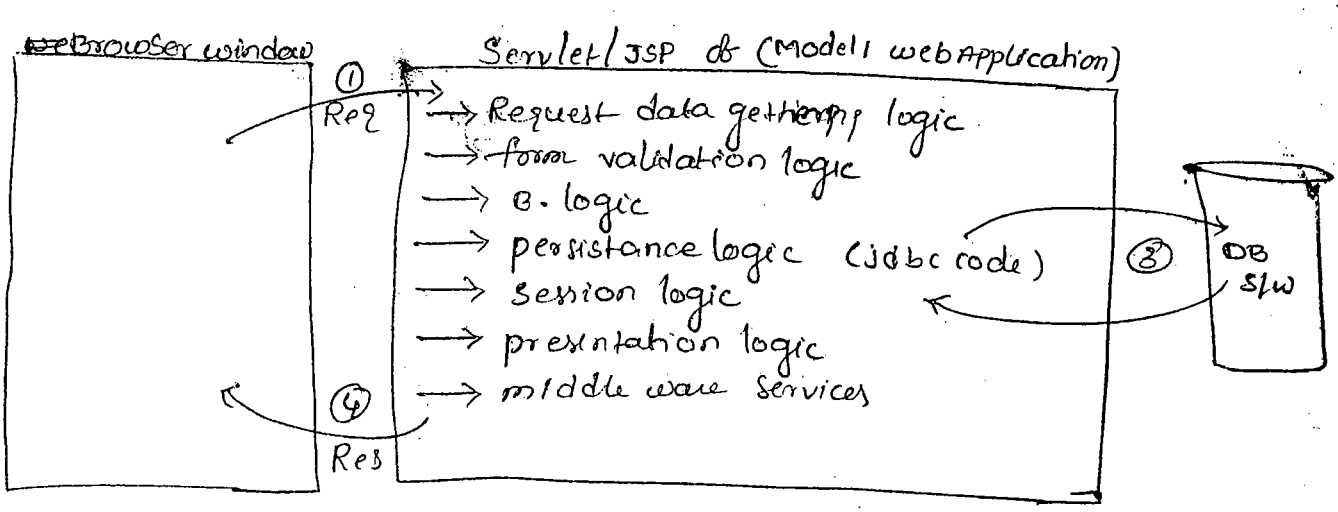
→ the logic that interact with DB s/w, and manipulates table Data is called persistence logic

→ the logic i.e given to remember the client data across the multiple request is called session managent logic.

→ the logic that formats the results and sends the results to browser window is called presentation logic. It is also responcible to prepare the user interface.

→ the additional services that are configarable on our webApp to make them perfect & accurate or called middle ware Services.

Eg:- Transaction, Security, Jdbc connection pooling & etc



→ In Model I Architecture based webApp development in every webresources of webApp multiple logics will be mixed up.

→ In model I Architecture based webApp multiple Servlets/JSP's can be there but if servlets are avalible in the webApp JSP's must not be places and viseversa.

drwabacks:

→ since multiple logics are mixed up in every webresources of webApp there is no clean saparation of logics.

→ modification done in one logic effect other logics this provide complex in maintenence and enhancement of the project

→ parlall development is not posible so the productivity is very poor.

→ Note:- During more work in less time having perfection is called productivity.

→ middle ware services must be implemented manually this improves borden on programmer.

→ advantage:→ knowledge is only servlet & only JSP is enough to develop model I architecture based webApp.

MVC (Model View Controler)

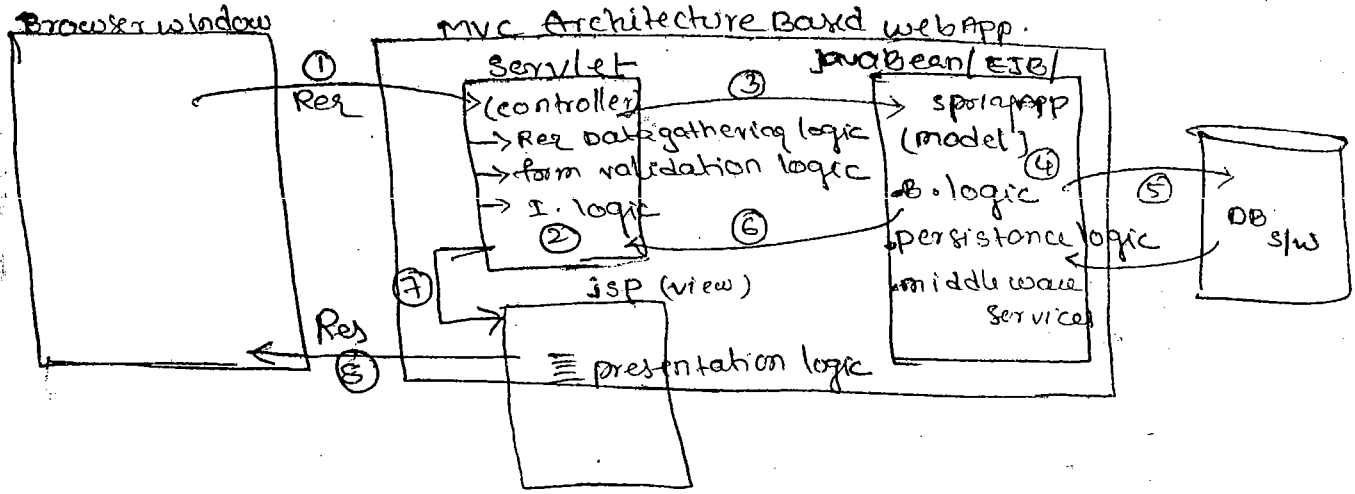
M → model → B.logic + persistence logic → java, ejb, spring, spring with hibernate) (Account office)

V → View → presntation logic → Html, JSP → with (Button)

C → Controller

components and model layer components this logic controlls & monitors all the logics available in webApp's. this logic keep-s track of all the operations that are happning in the webApp.

→ Taking request from browser, passing that request to appropri ate model layer component, taking, result from model component, & passing that result to appropriate view layer component ~~comes~~ is the responsibility of intigration logic of controller layer.



→ Developing webApp based on MVC Architecture is most popular approch in the realworld - with respect to diagram

① → Browser gives request to MVC architecture based webApp.

② → the controller of webApp traps & takes the request.

③ → controller uses intigration logic & passes the Request to an appropriate model layer component

④, ⑤ → the B.logic of the Application generates the result for the request by executing B.logic in this process model layer component interact with DB s/w by using persistance logic if necessary.

⑥ → the Result Generated by model layer component comes to the controller Servlet.

layer

(8) → presentation logic of the view layer executes to format the result & sends the formated result to browser as responce.

Advantages & disAdvantages of MVC Architecture webApp:

Advantage

→ since multiple layers are there in webApp there will be clean saparation of logic.

→ since multiple layers are there modification done in the logic of one layer dosenot effect logics of other layer.

→ maintenence & enhancemet of mvc based applications is quik easy

→ parlall development is posible so productivity is good

→ EJB & spring technology give built in middle ware services this redoses burden on the programmer.

→ disadvantages

→ For parallel development multiple programers are requir knowledge on multiple technology is requir

→ How parallel development is posible in

(A) → project leader devide the team in to two parts in mvc based application.

part1 web authors

→ these people uses jsp, html technology to develop presentage logic.

part2 J2ee Developers

→ these people uses all J2ee technology like servlet, ejb & etc to develop business logic & intigration logic of application

since these two parties can work parlally there is a possibility of parlall development

~~Five principles~~ :-

(2) → By using multiple technology in mvc architecture based App development we need to follow set of roles as mvc principles.

① Every layer is given to hold certain kind of logics place only those logics in that layer.

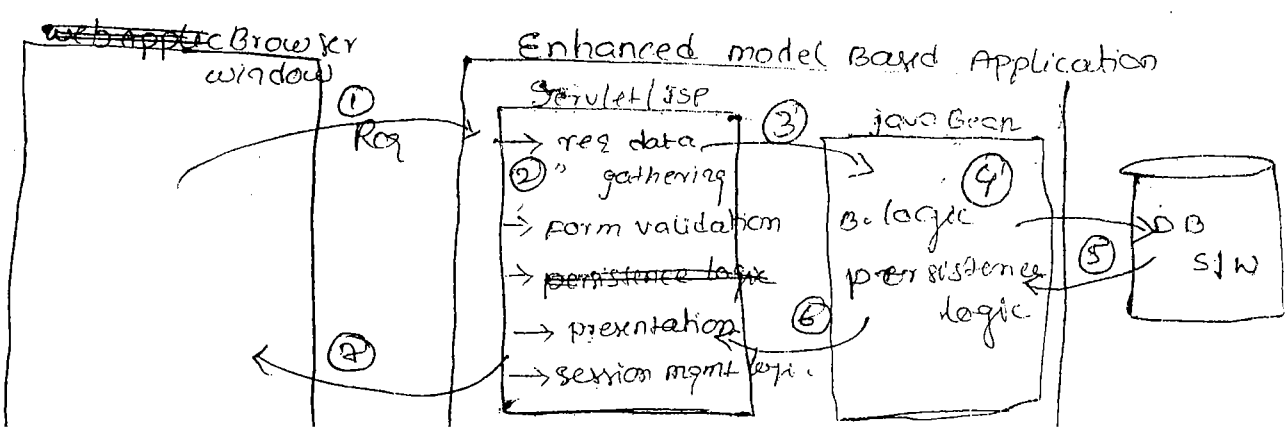
② Every operation i.e taking place in webapp execution must be happend under the control of controller Servlet.

③ They can be multiple jsp's in view layer & multiple Business components are applications in model layer but there must be only one servlet in controller layer acting as controller.

④ Two jsp's of view layer should not interact with each other directly, they must interact with each other through the controller servlet.

⑤ View components must not talk with model components directly and viceversa, this communication must take place through controller servlet.

Enhanced model1 :-> It the webresources of model 1 Application saparates B.logic and persistence logic by keeping them in java been then it is called Enhanced Model 1.



it can use <jsp:useBean> tag, <jsp:setProperty>, <jsp:getProperty>.

→ The javabean class of webApp can be used to store form data of the form page or to store maintain b.logic & persistence logic as shown in enhanced model Diagram.

syntax ① : <jsp:useBean>

↳ <jsp:useBean attributes> // used for directing/locating an obj of javaBean class

attributes

A. id = " " // Instance name (object name)

B. class = " " // a fully qualified class name (javaBean.class)

C. scope = " " // page/session/request/application "(default scope is page)"

page (default) ↳ bean class obj is avalible with in the decl page

request : → bean class obj avalible through the request

session : → " " " with in a session

application : → Bean class obj will be avalible allpages with in the context (webApp)

D. beanName = "logic name"

E. type = "reference class type"

Ex :- <jsp:useBean id="st" class="P1.StudentBean" scope="session" />

// the above stmt create P1.StudentBean class obj by name of "st" and placed in session scope.

// jsp equalent Servlet code is

```
if(pageContext.getAttribute("st", pageContext.SESSION_SCOPE) == null)
{
    P1.StudentBean st = new P1.StudentBean();
    pageContext.setAttribute("st", pageContext.SESSION_SCOPE);
}
else
{
    P1.StudentBean st = (P1.StudentBean) pageContext.getAttributes
                                ("st", pageContext.SESSION_SCOPE);
}
```

Eg:

Test t = new Test(); // "t" reference type = Test & obj type = Test

ABC abc = new Test(); // ABC is the super class of Test

Ex:- <jsp:useBean id="st" class="p1.StudentBean" scop="request">

// jsp equalent Servlet creates Bean class obj shown bellow

ABC st = new p1.StudentBean();

"st" refference type is "ABC" class. "st" object type is "p1.StudentBean" class

here p1.StudentBean class extends from ABC class

② setProperty :→ call setXXX methods of javabean class to set data to Bean class object

syntax :→ <jsp:setProperty attribute>

attributes :→

property = " " // method name // XXX part of setXXX() // Bean property name

name = " " // id attribute values give in the <jsp:useBean

value = " " // arg // value to be set.

param = " " // input (request) parameter name.

Note :→

value or param → Any one attribute has to be used

Ex1 :→ <jsp:setProperty name="st" property="sno" value="567"/>

// this internally calls setSno() method to assign value 567 to Bean property sno

Ex2 :- <jsp:setProperty name="st" property="sname" value="raja" />

// this internally calls setSname() method to assign value "raja" to property sname.

③ getProperty :→ calls getXXX method to read data from bean property to bean class object.

syntax : <jsp:getProperty attribute>

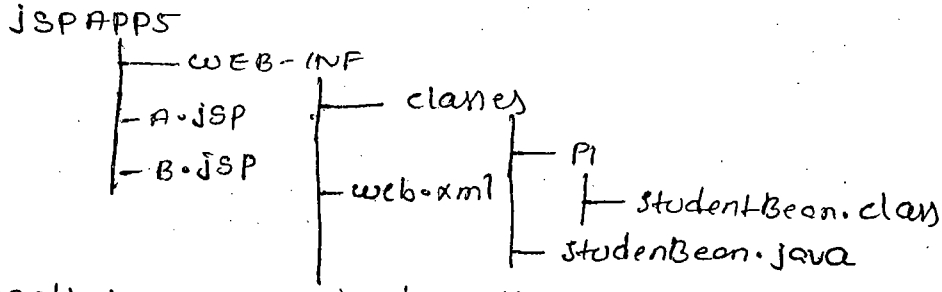
attributes :- name = " " // id given by <jsp:useBean>

property = "variable name / bean property name" // the XXX part in getXXX method

Ex :- <jsp:getProperty name="st" property="sno" />

gives sno variable value by executing getSno() method on bean class object

→ Example Application (to understand the basics of above 3 tags)



* → JSP wants to communicate with JavaBean class of webapp the javaBean class must be there in a package of WEB-INF/classes folder.

* → <JSP:useBean>, <JSP:setProperty>, <JSP:getProperty> tags are 3 indipendent tags.

```
// studentBean.java
package P1;
public class StudentBean
{   // Bean properties are private
    private int sno;
    private String sname;
    private String sadd;
  // setXXX()
  public void setSno(int no)
  {
     this.sno=no;
  }
  public void setSname(String name)
  {
     this.sname=name;
  }
  public void setSadd(String add)
  {
     this.sadd=add;
  }
// getXXX()
public int getSno()
{
   return sno;
}
public String getSname()
{
   return sname;
}
public String getSadd()
{
   return sadd;
}
}
```

web.xml

<web-app>

→ JavaBeans of webApp need not to be configured in webApp.

// A.JSP // it will create

```
<JSP:useBean id="st" class="P1.StudentBean" scop="session"/>
    <!-- set data to bean class obj -->

<JSP:setProperty name="st" property="sno" value="345"/>
<JSP:setProperty name="st" property="sname" value="raja"/>
<JSP:setProperty name="st" property="sadd" value="hyd"/>
```

// B.JSP // it is locating (reading)

```
<JSP:useBean id="st" class="P1.StudentBean" scop="session"/>

<JSP:getProperty name="st" property="sno"/>
<JSP:getProperty name="st" property="sname"/>
<JSP:getProperty name="st" property="sadd"/>
```

note → To test the above webApp first request A.JSP then request B.JSP using the same browser window.

02/02

→ If you want to set Request parameter values (form page data) as the values of javabean properties use param attribute of <JSP:setProperty> as shown here.

A.JSP

```
<JSP:useBean id="st" class="P1.StudentBean" scop="session"/>
                                              (or) Application

<JSP:setProperty name="st" property="sno" param="stno"/>
<JSP:setProperty name="st" property="sname" param="stname"/>
<JSP:setProperty name="st" property="sadd" param="stadd"/>
```

→ request parameter names coming from browser window having

```
<JSP:useBean id="st" class="P1.studentBean" scope="session (or) application"/>
    <JSP:getProperty name="st" property="sno"/>
    <JSP:getProperty name="st" property="sname"/>
    <JSP:getProperty name="st" property="sadd"/>
```

→ note use the following url to give the request to A.JSP:

http://localhost:2010/JSPApp6/A.JSP?sno=101&sname=vv&sadd=xx

→ If all the request parameter names are matching with the names of beanProperty names of javaBean class then use '*' symbol in the as the value of property tag to set requestParameter values to bean properties values.

A.JSP

```
<JSP:useBean id="st" class="P1.studentBean" scope="session (or) application"/>
    <jsp:setProperty name="st" property="*"/>
```

B.JSP

same as above.

→ give request to A.jsp having following request url:

http://localhost:2010/JSPApp6/A.JSP?sno=106&sname=xxx&sadd=444

→ the Application i.e design to display advertisements in the webpages is called advertisement rotator application. this application should have following behaviour

1. Advertisements must be changed automatically at regular intervals.
2. Each advertisement must be populated as graphical hyper link.
3. Advertisement should be picked up randomly from the list of advertisements.

<%-- datacollector.jsp --%>

```
<html>
    <jsp:useBean id="rotator" scope="session" class="myapp.Rotator" />
    <%
       response.setHeader("refresh","2");
       rotator.nextAdvertisement();
    %>
    <body color="red" scroll=on leftmargin=0 topmargin=0>
    <a href="<jsp:getProperty name="rotator" property="link" />">
     <img border=0 width=300 height=100 src="<jsp:getProperty
        name="rotator" property="image" />">
     </a>
     <br>
      <b> it is rest of the page</b>
    </body> </html>
```

→ code to refresh the page automatically after every 2 seconds.

① → rotator.java

```
package myapp
import java.util.*;

public class Rotator
{
    private String images[] = {"1.jpg", "2.jpg", "3.jpg", "4.jpg",
                               "5.jpg"};
    private String links[] = {"http://www.yahoo.co.in", "http://
    www.scores.sify.com", "http://www.cricinfo.com", "http://www.
    gmail.com", "http://www.redif.com"};
    private int randval;

        public String getImage()
        {
            return image[randval];
        }
        public String getLink()
        {
            return links[randval];
        }
```

```
public void nextMovement()
{
    Random rad = new Random();
    randval = rad.nextInt(5);
}
```

} B. logic to Generate random number.

}

Mini Project DISCUSSION

→ Inorder to send the java obj over the n/w the object must be Serializable obj java object becomes Serializable obj when the class on that object implements java.io.Serializable Interface.

→ ResultSet obj is not Serializable obj by default so we can't send result set object over the n/w directly. to solve this problem there are two solutions.

① stop working with ResultSet use RowSets.

note: very few jdbc Drivers are supporting RowSet.

② copy data of the ResultSet obj to any other collectionframe work data structure and send that obj over the n/w.

note:- all collectionframe work DataStructures are Serializable obj by default.

note:- most of the developers prefer working with solution ② to solve the above problem.

→ II

problem → we can't move one record of ResultSet obj to one element of array list because in record of ResultSet obj multiple objects will be there but on an element of array list we can store only one obj at a time.

To solve this problem 1. helper javaBean class and store data of the record collected from ResultSet obj in to that obj and add that obj to Array list as element value.

Ex:- public class Student

```
{   int no;
    String name;
    String add;
    getXXX() {
        ---------- ③ methods
    }
    setXXX() {
        ---------- ③ methods.
```