

adv.java course (For java based web application/web site developemt)

- 1. JDBC + misc (pre-requisite:: Core Java oops) | (11 am slot)
- 2. Servlet + misc (pre-requisite:: Core Java oops) | (4pm slot)
- 3. Jsp +misc (pre-requisite:: servlet)
- 4. Mini Projects (pre-requisite:: servlet,jsp,jdbc)

course
Option1:: Learn entire in 11 am slot (100+ sessions) Link: zoom.us/j/93664073459
option2 :: attend both 11 am and 4 pm slots (70+ sessions)

Course fee :: 3500 /-

admin details :: Mr.Rambabu / Mr.Srikanth
9866545966 6302968665

natarazjavaarena@gmail.com
natarazjavaarena ->FB Group
<https://www.facebook.com/groups/388095825162910>

- 2 Hiberante f/w (pre-requisites :: 30% Jdbc) (In 3rd week of nov) (60+ sessions)
- 1 Spring with spring boot f/w (pre-requisites :: Strong Core oops + anything little in adv.java) (nov 5th :: 6pm) (120 + sessions) (rs: 3500)
- 3 WebServices & MicroServices (pre-requisites :: servlet +jsp) (In month of december) (50 sessions)

=>Servlet ,jsp are technologies of JEE module which can be used to
develope web applications/websites

Need of web applications/Websites

=>state is also called as Data

=>logic is also called as code in java methods

=>The data and logics of stanalone Apps are specific one computer ,where the application
is executing.. and only one user can operate the application at time..

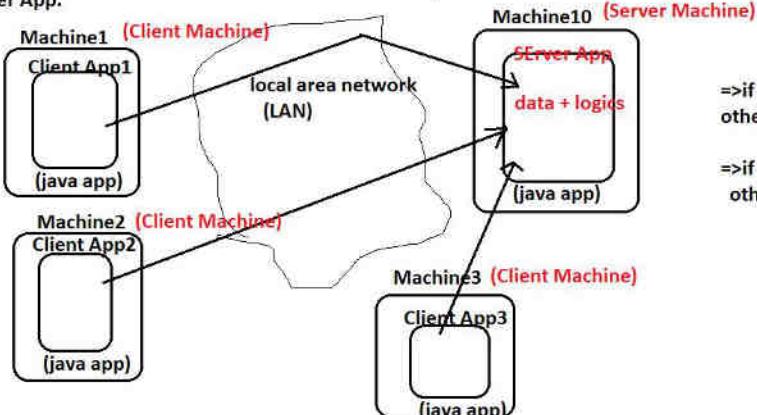
eg:: core Java Apps (class with main(-) method)

eg:: calculator Apps

eg:: AntiVirus software



=> In Client - Server Apps, the data ,logics of server App are visible and accessible
in multiple client Apps that are taking to server App. Here the data and logics of server
App are specific to a LAN (LocalAreaNetwok) using which the client Apps are talking to
Server App.



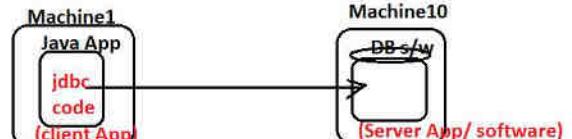
=>if App/machine is ready gives services to
others then it is called ServerApp /Machine

=>if the App/machine is taking services from
other App /machine is callied ClientApp/Machine machine

DB s/w :: Server s/w or App

Browser :: client App/ s/w

note:Using netwoking programming of java (java.net pkg)
we can develop both Client Apps and Server Apps as java Apps.

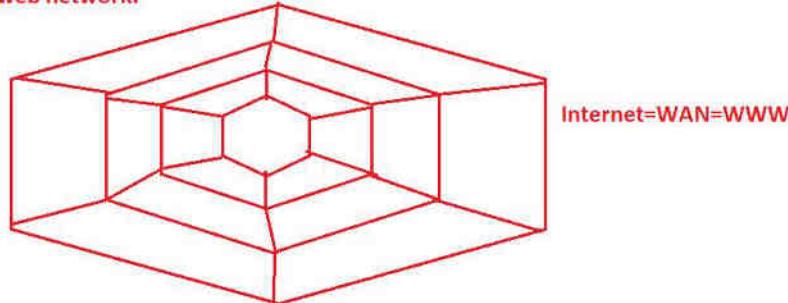


=>To provide more global visibility and accessibility for the data and logics we need to use
bigger network called WAN nothing but internet(network of networks).

=>we can not execute the regular standalone Apps, client-server Apps in the internet setup..
i.e we need to develop specifial Apps called websistes/web applications to execute in the
internet setup.

eg:gmail.com, nareshit.com, fb.com and etc...

=>internet network is also called as web network.. becoz internet network looks like spider's web network.



=> Currently 5 dominant technologies are used to develop web applications/websites

(a) Using PHP (Personal home page for hypertext pre-processing)

- => useful for small scale website.. development
- => useful for websites which need not be for long time
- eg:: movie promotional websites , car model promotional websites)

(b) Using asp.net technology (asp -> active serverpages)

- => useful for medium scale web site development
- eg:: college websites , university websites

(c) Using Java servlet,Jsp technologies (jsp : java server pages)

- => Useful for medium scale and large scale websites.. where security and complexity is involved
- eg: flipkart.com, amazon.com , icici.com, citibank.com

(d) Using NodeJS,ExpressJs

- => Grabbed PHP, asp.net market of developing small scale, medium scale web Apps Development..

(e) Django (Extension python)

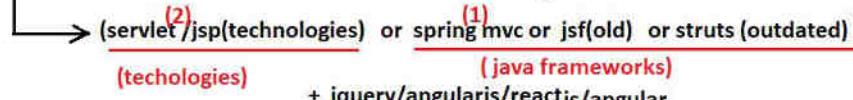
- => Came alternate to everything.. but failed becoz of security and performance issues..

Current trending technologies to develop the websites

jsf: java server faces..

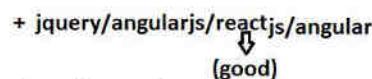
a) Java (For all Large scale websites)

Even for 40% small and medium scale web sites)



b) NodeJS +expressJs

=>To develop 60% small and medium scale websites



I heard Facebook.com is developed in PHP. Is it true?

Ans) Yes (But in the background lots of java webservices are used)

=>Initially java was used only to develop large scale complex website to take the advantage of java's performance and security..

=> Originally java is not so strong in User-Interface Development.. i.e we do not have ready-made advanced GUI components to provide rich GUI..

=> With the arrival of jquery/angularjs/reactjs and etc.. UI development is simplified.. very much and these can be integrated in any environment of website development like php,java,.net,django, node.js and etc.. So industry using java with jquery/angularjs/reactjs /.. to even develop small scale and medium scale apps

Java learning is all about

- 1.java language --> Core Java --> like raw material
- 2. java technologies --> adv.java --> like semi-finished products (jdbc, jndi, servlet, jsp and etc..)
- 3. Java frameworks --> spring, hibernate, web services --> fully finished products

what web application?

=>It is collection of web comps having capability to generate the webpages..

note:: any reusable code ^{of} that is in the form class or file is called comp

eg:: html file , servlet comp, jsp comp,php comp, asp.net comp and etc..

Two type web pages

a) static web pages /passive webpages

->The content of web page will remain same for all requests...

eg:: About Us page , ContactUs page , terms and condition page and etc..

b) Dynamic web pages /Active webpages

=>The content of web page will change based on the time of request generation or based in the input values of the request..

eg:: gmail In box page , Live game score page , wheather report page and etc..

note:: WebApplication or website contains both static and dynamic web pages..

3 types of web comps

=>A web application can have 3 types of web comps

a)static web comps

=>these are given to develop static web pages

eg:: html files

b) dyanic web comps

->These are given to develop dynamic webpages

eg: Servlet comp (class developed by using servlet technology)

jsp comp (file created using jsp technology/tags)

asp.net comp

php comp

and etc.

servlet,jsp, asp.net,php and etc..

web technologies

their programs or code or

files are called web comps

c) Helper web comps

=>These web comps do not generate any web pages directly , but they help other comps in the generarration of webpages

servlets (x) -->wrong
servlet (valid)

oop (valid)

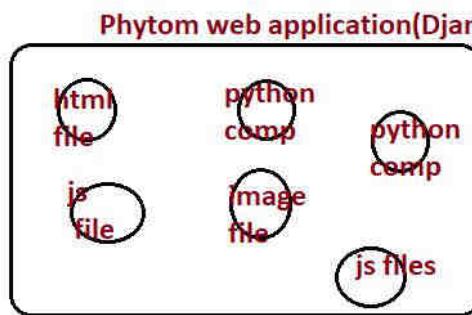
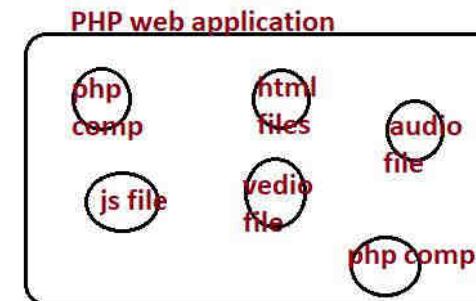
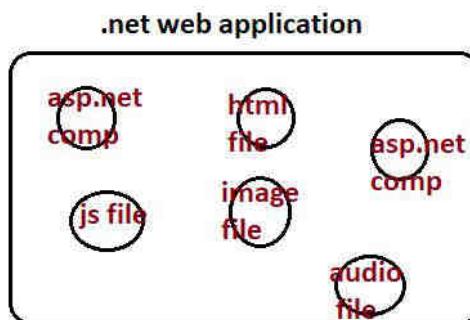
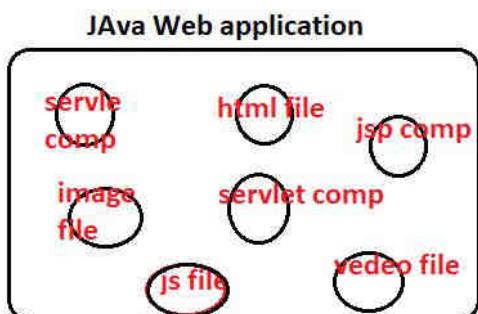
oops (invalid)

eg:: image files, audio files, video files , js files , css files , flex files and etc..
(.swf files)

In All env.. of web application development, we use the same helper web comps and static web comps but the dynamic webcomps will change env... to env.. i.e (java to .net to php to phyton and etc..)

- =>In java web application dynamic web comps are servlet,jsp comps
- =>In php web application dynamic web comps are php comps
- =>In asp.net web application dynamic web comps are asp.net comps
- =>In nodeJS/ExpressJS web application dynamic web comps are js files
- =>In phyton /Django web applicatiin , the dyamic webcomps are .py files (django)

[In modern days we can use js files as hepler web comps(form validations) and also as dynamic web comps(In nodeJs env..)]



static and hepler comps are same in all env.. (technologies) based web application development.. but the dynamic web comps will change env.. to env.. or technology to technology as shown above..

=>since the standalone App is specific to one computer and will be operated by one user at a time we prefer manual execution of standalone App..

=> Since web applications or websites get requests from multiple clients/browsers/users simultaenously from different locations and having different timings or same timings (24/7), it is better to execute Web application in a automated env.. like by keeping them web server s/w (like tomcat, IIS, weblogic and etc)... becoz manually execution of web application and its web comps practically impossible.

Standalone App --> specific to 1 computer + allows 1 user at time
(Manually execution app is sufficient)

Web application/website --> requested by multiple clients/users
at a time from diff locations that to 24/7

[Here the manual execution of web comps when ever they
are requested not going to work out, So automation through
as special s/w called webServer s/w]

eg: Tomcat (java), IIS (.net), AWS (php), nodejs (js) and etc..

IIS --> Internet Information server /service

AWS --> Apache WebServer

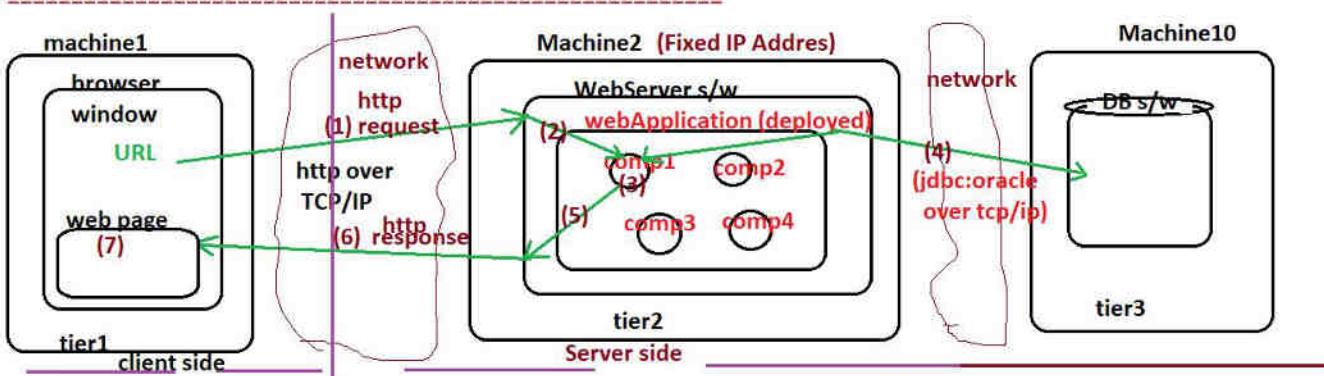
What is WebServer?

=> WebServer is a piece of software that automates web application and its web comps execution.. i.e it listens to clients requests continuously, takes them , passes them to appropriate web comps , executes them dynamically , gathers the results from them and sends the results as responses to browser..

note:: We need not to develop webServer s/w .. we have lots readily available webserver s/w.

But developing web application having web comps and keeping that web application in webServer is our responsibility.

Understanding Web application setup that is required for execution



=> protocol http runs over protocol TCP/IP i.e machines where browser and webServer reside will interact with each other using protocol TCP/IP whereas browser and webServer will use protocol http for interaction so it is http over TCP/IP

http :: hyper text transfer protocol

https :: http over SSL (Secured socket Layer)

=> we can develop web application either as two-tier App (with out Db s/w) or as 3 -tier Application (with Db s/w)

=> In One WEbServer we can deploy multiple web applications...

=> The Process of keeping developed web application in web server is called deployment and reverse is called undeployment

=> Protocol is bunch of rules followed by two parties who wants to participate in communication

Two types of protocols

(a) Network protocol

--> Gives rules to get interaction b/w

W. r. to diagram

- (1) Enduser types URL in the browser address and generates the request
- (2) WebServer s/w that listens to requests continuously takes the requests maps to appropriate web comp of appropriate deployed web application
- (3) WEB comp process the request
- (4) If necessary, the web comp interacts with Db s/w
- (5) The output/results generated by the web comp goes to webServer
- (6) webServer sends that output/results to browser as response
- (7) Browser receives the response and displays to web page..

Physical computer

eg:: TCP/IP

(b) application protocol

-> Gives rules to get interaction b/w two softwares or software application
eg:: jdbc:oracle , http, https , smtp and etc..

note:: Application protocols run over network protocols

List of browser softwares /user-agents /web clients

Internet explorer ---> from MS (2)
Chrome ---> from Google (1)
FireFox ---> from Mozilla (3)
Opera ---> from Opera soft
Safari ---> from Apple (4)
Edge ---> from MicroSoft
hotJava ---> from RedHat
and etc...

Browser provides platform to enduser/visitor
to send requests and to receive responses..

=> After requesting web comp, if its code is coming to browser from web Server for execution then it is called client side web comp.

eg: html file(html interpreter is available in the browser)
js code (js engine is available in the browser)

=> After requesting web comp, if its code is executing in web server itself then it is called Server Side web comp..

eg:: Servlet Comp ,jsp comp, php comp, asp.net comp and etc..

List of client side web technologies

=> given to client side web comps..
html ---> from w3c
css ---> from w3c
java script ---> from sun Ms+NetScape
vb script (old)--->from Ms
jquery ---> from Jhon Resig
angular/angularJS ---> from Google
ajax ---> from Google
reactJs ---> from FB
and etc..

note:: Do not decide whether web comp is client side or server side based on the place where it resides.. decide it based on the place where its code executes...

Scripting language?

=====

The language whose can not be executed directly or independently and must be embedded with other technology/language code is called Scripting language..

=> JS (java script) can not be executed directly.. it must be embedded with html tags/code for execution.. So it is called scripting language

note:: Ajax, jquery, angular, angularjs, bootstrap, typescript, reactjs and etc .. java script extensions.

=> JS is basically client side language/technology becoz its code executes in browser by taking the support java engine..

Servlet videos

=>https://www.youtube.com/watch?v=2oTIHv-GwII&list=PLVIQHNRLfIP88b4uv8X_EtHAPHpjkw1qc

List of server side web technologies

servlet ----> from sun Ms/Oracle corp (java)

jsp ----> from sun Ms/Oracle corp (java)

asp(old) --> from Ms (micro soft) .net platform

asp.net ----> from Ms

PHP ----> from Apache (php based)

SSJS ----> from netscape (ssjs -server side java script)

nodeJs, ExpressJS --> from ryan dahi (google)

cold fusion --> from adobe

and etc..

**note:: Use java technologies like Servlet,jsp to develop
java based small scale web applications**

eg:: univesity websites, collge websites

**note:: Use java frameworks like spring mvc to develop
java based large scale /complex web applications**

eg: e-commerce website, banking websites and etc..

Servlet videos

=>https://www.youtube.com/watch?v=2oTlHv-GwII&list=PLVIQHNRLfIP88b4uv8X_EtHAPHpjkw1qc

=>in the market we have frameworks
to develop server side web comps and
web application very easily by interally using
technologies

struts(old) --> from apache (java)

spring MVC --> from interface21 /Pivotal team

Spring boot MVC --> from interface21 /Pivotal team

Webwork --> from openSymphony

ADF --> from oracle corp

asp.net mvc--> from micro soft (.net platform)

dhrupal --> from apache (php platform)

Django --> PSF (from python platform)
| -->python software foundation

MVC :: Model View Controller

ADF --> Application development
framework

List of webServer s/ws

=> To provide automated env.. to execute web application and its server side web comps

tomcat ----> from apache (java) **(good for java env..)**

jws -----> from sun Ms (jws: java webServer) (java)

pws -----> from Ms (ms platform)

(personal webSErver)

IIS ----> from Ms (.net platform) **(good for .net env..)**

(InternetInformationSErvice,)

AWS ----> from apache (php platform) **(good form php env..)**

(apache web server)

NodeJs --> from chrome (provides JS run time env..) (Server Side Js platform) **(for for js env..)**

NFTS --> from netscappe (for server side js)

(Netscape fast track sevrer)

good

(Netscape fast track sever)

Django server --> PSF (python platform) (good for python)

Resin Server --> from Resin soft (java platform)

jetty -----> from eclipse (java platform)

and etc...

note:: Use java webServer like Tomcat to automate the execution of java based small scale web applications

eg:: univesity websites, college websites

note:: Use java Application Server like wildfly to automate the execution of java based large scale /complex web applications

eg: e-commerce website, banking websites and etc..

AppServer = webServer ++ (extension webServer)

weblogic --> from Oracle corp (3)

websphere --> from IBM

GlassFish --> from Sun Ms /Oracle corp (2)

WildFly ----> from RedHat (Jboss) (1)

Jrun -----> from adobe

Oracle10gAs --> from oracle corp

and etc..

java based

List of DB s/w's

=====

oracle --> from oracle corp

mysql --> from DevX/ oracle corp

postgresql --> from enterprise DB

DB/2 --> from IBM

mariadb --> from mariadb and etc..

MongoDB -- MongoDB Inc

and etc..

RDBMS-SQL
(for structured Data)

NoSQL DB
(for unstructured Data)

To develop and execute website

=====

(a) Choose any browser s/w

(b) choose any DB s/w

(c) choose one or more client web technologies to develop client side
web comps

(d) Choose any webServer /App server to automate the execution

(e) Choose one or more Server Webtechnologies to
develop server side web comps

These are common for
all domains/envs.. of web application development
like java, .net, php, JS, python and etc..

[We choose webSErver/App server and
Server side webtechnologies/ frameworks
having compatibility]

Java Full stack Developer (3+ /4+ /.. experience)

=====

=> Java language (corejava), java technologies (Adv.java), Java frameworks (Spring, hibernate,
webServices/micro services), design patterns, tools and etc..

+

=> DB s/w (SQL, PL/SQL --> oracle)

+

=> UI Technologies (html,css,java script, BootStrap, jquery/angular/Angular Js/ReactJs and etc..)

+

=> DevOps and Aws (cloud)

Mean Stack
developers

Java Fresher (Java developer)

=====

core java +adv java + oracle + spring +CRT
(with boot)

Java 1+ Experience d

=====

Java Fresher topics + hibernate + design +tools
(no CRT)

java 2+ experienced

=====

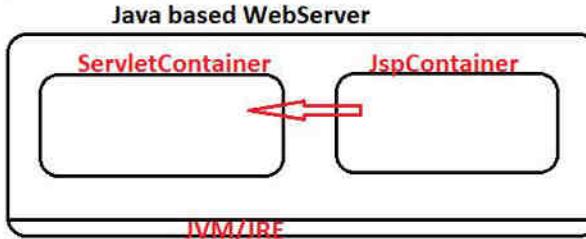
java 1+ experienced topics + WebSERveices/Microservices

java 3+/4+/5+....

=====

full stack developer refer above..

=>To execute stanalone App we need the JVM
 =>To execute html code we need html interpreter (browser)
 =>To execute java script code we need Java script engine (browser)
 => To execute Applet comp we need Appletviewer (runs on the top of JVM)
 Similary To execute Servlet comp , we need Servletcontainer/SErvletEngine (runs on top of JVM)
 To execute jsp comp , we need JspContainer/Jsp Engine (runs on the top of ServletContainer)
 note:: every java webServer like tomcat gives one set of built-in ServletContainer,JspContainer..



=>Servlet comps java code based comps.. where
 jsp comps are tags based comps which are internally
 servlet comps.. now industry is using both servlet,jsp
 technologies togather in java web applicaiton development

**webContainer = Servlet Container +
 JspContainer**

html ,java script comps code goes to browser for
 execution from WebService becoz they are client side
 web comps

Servlet ,jsp comps are serverside web comps , So
 they execute in WebServer itself by taking the support
 of SErvetcontainer,jspcontainer respectively (webContainer)

What is Container

=====

=>Container is a software program that runs continuously and manages the whole life life cycle given comp/prg from

birth to death (object creation to object destruction) .

=> Servletcontainer manages the life cycle of servlet comp i.e loading servlet comp class, creating object, calling methods, managing and destroying object will be taken care servlet container birth to death.

=>Jspcontainer and Servletcotnainer togather manages the wholle life cycle of Jsp comps..

note: ServletContainer,Jsp cointainers (webcontainer) internally runs on the top of JVM/JRE

=>WebServer and its webContainer are continuosly runing processes then
 they can take requests , they process requests by executing web comps
 and they can deliver responses 24/7. (Internally they run as daemon process)

cmd>java App1 (process started)

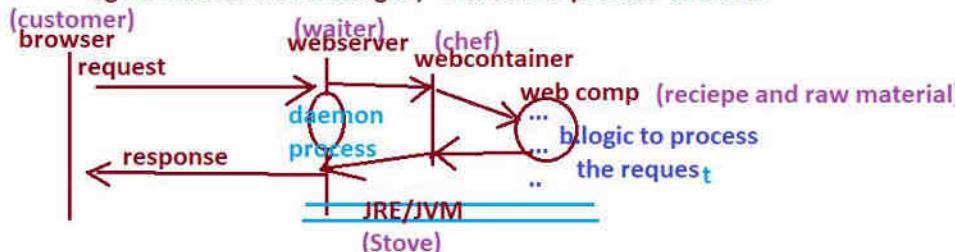
....

process end..

cmd>

note:: the Process that runs continuosly once started and will be stopped only when
 it is stopped manually is called daemon process.

eg:: Windows task manager , webServer process and etc..



=>Container is like an aquirium managing the whole life cycle of fishes called comps
 =>Servletcontainer is aquirium for the fishes called Servlet comps
 =>Jsp container is aquirium for the fishes called jsp comps..

Adv.java course

|---> jdbc , servlet, jsp , mini Projects

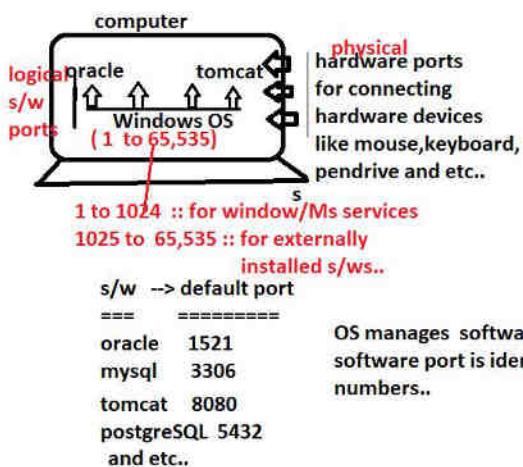
11am

4pm slot

Tomcat

=====

type :: Java based webServer (upto version 6)
java based Application server (from version 7)
latest version:: tomcat 9.x (comptible with java8+/jdk 1.8+)
vendor :: Apache 10 (comptible with jdk1.9+)
default port no :: 8080 (changeable)
Gives two built-in Containers : ServletContainer,jsp container
ServletContainer name :: CATALINA
Jsp container name :: JASPER
To download s/w :: <https://tomcat.apache.org/download-10.cgi> (download as WindowsService Installer)



Socket number :: <ip address of computer> + <software port no>

Machine1--> oracle Db s/w

oracle port number:: 1521

oracle socket number :: machine1+1521 (hostname +port no) 180.56.77.44+1521 (ip address+ port no)

note:: One externally installed will give multiple s/w services running on multiple software ports..

eg: Oracle Db s/w installation gives multiple s/w services

- =>DB service :: 1521 (port no)
- =>sqlplus :: 1234
- => web based sqlplus :: 8080 and etc..

C:\WINDOWS\system32>netstat -a (open cmd prompt as administrator)

To list out all port numbers of running services

To refer a computer being from that computer we can use "localhost" or 127.0.0.1

=>Change Tomcat server startup type manual after installing tomcat

=> open services.msc (window search box) ---> goto apache tomcat ---> right click --->properties ---> startup type :: manual --->....

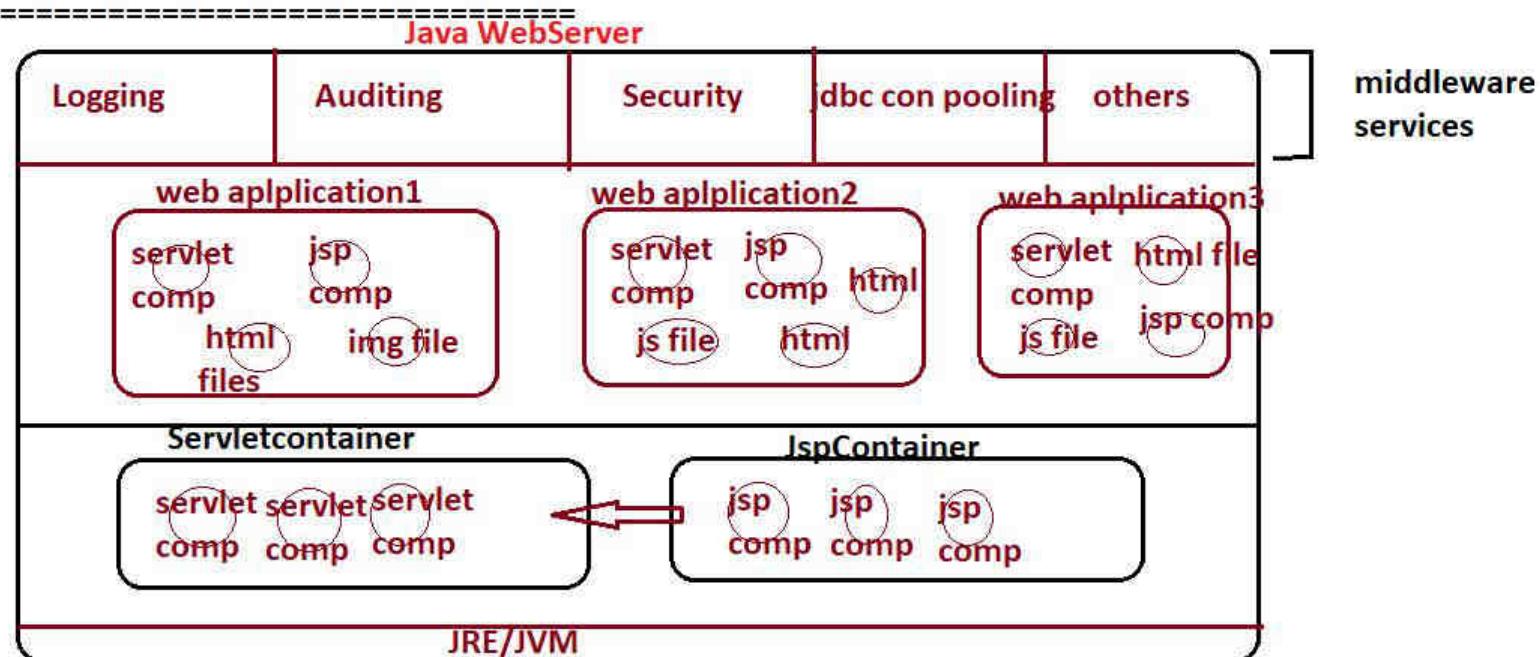
Q) Tell me whether Tomcat is WebServer or WebContainer ?

Ans) WebServer upto version 6

Application server from version 7

=>It is webServer/Application server having two containers :: 1.Servlet container 2.jsp container

High level architecture of Java webServer



=>When servlet comp is requested from browser ,it will go to servletcontainer for execution

=>When jsp comp is requested from browser ,it will go to jps contianer + servlet container for execution

=>when html files, js file comps are requested , their code comes to browser for execution..

=> **Middleware services are not minimum logics of App devleopment.. they are addtional optional and configurable (ability to enable or disable) logics of the applicaitons..**

=> when we develop web application by applying middlwareservices it becomes more and perfect and accurate.

eg ::Security :: To protect the application (Authentication +Authorization)

Auditing :: keeping track of user activities

Logging:: keeping track of code flow

Con pooling :: keeping bunch of readily available jdbc con objects

|-->Checking the access permissions of a user

on resources/comps

-->Checking the identity of a user.

note::we can apply these middleware services on one or more deployed web applicaitons of webServer..

Responsibilities of webServer

- =>Listening to client requests continuously by having daemon process
- =>Taking requests and handovering them to Webcontainer
- => Providing middleware services
- => Providing WebContainer (Servlet container+ jsp container)
- => Providing env.. to deploy or undeploy the web application and other activities of on web application like stoping , reloading(stop +start) web applicaiton and etc..
- => Provides JRE/JVM
- =>taking the results/outputs from Webcontainer and sending them browser as responses and etc..

Responsibilities of webContainer

- =>Taking requests from webServer and mapping them to web comps
- => Managing the whole life cycle of server side web comps like servlet,jsp comps
- => Sending client side web comps(html,js files) code to browser for execution when requested
- => Executing Server side web comps code when requested
- => Provides special garbage collector for destroying objects
 - (ServletContainer will not use jvm's garbage collector.. it will use its own garbage collector)
- => takes care of inter communication b/w web comps..
- => gives the output/results for server side web comps to webServer
- => applying configured middlware services on web comps of web application s and etc...

for videos

https://www.youtube.com/watch?v=2oTIhv-GwII&list=PLVIQHNRLfIP88b4uv8X_EtHAPHpjkw1qc

for notes

<https://www.facebook.com/groups/388095825162910/files>

mail id :: natarazjavaarena@gmail.com

To view and change http port number of Tomcat server after installation

=> Go to <Tomcat_home>\conf\server.xml file and modify "port" attribute of first <Connector port="....." tag ---> restart tomcat server
|-->2020 to 3030

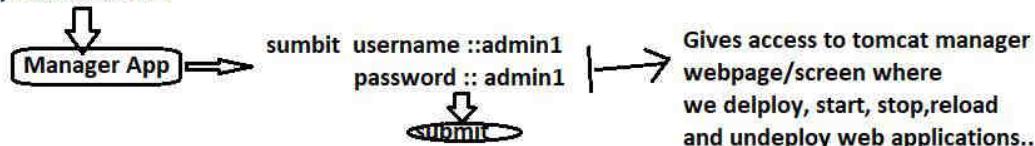
=> Display hom page
url in the browser address bar :: <http://localhost:3030>

To view and change admin username,password after Tomcat Server Installation

==> Go to <Tomcat_home>\conf\tomcat-users.xml file and modify username,password attribute values of <user> tag ---> restart tomcat server...

<user username="admin1" password="admin1" roles="admin-gui,manager-gui" />
change here changer

=> Display hom page
url in the browser address bar :: <http://localhost:3030>



Tomcat server Installation directory structure

```
G:\Tomcat 10.x
|---bin
|   |--->*.exe (tomcat10.exe)
|--->webapps
|   |--->root,exampels, docs, manager and etc.. directories
|   |---> *.war files
|---conf
|   |--->*.xml (server.xml , tomcat-users.xml, context.xml and etc...)
|---lib
|   |---> *.jar (servlet-api.jar, jsp-api.jar,catalina.jar ,jasper.jar and etc..)
|---work
|   |--->CATALINA
|---logs
|   |--->*.txt (everyday one log file will be generated)
|---temp
|
|--->other files
```

=> After web applications having web comps , we deploy them in tomcat web server by copy them to <Tomcat_home>\webapps folder either in the form of directories or in the form war file

war --->web application archive (a kind of jar file)

The <Tomcat_home>\lib folder gives multiple jar files
catalina.jar -->represents servlet container
jasper.jar -->represents jsp container
servlet-api.jar -->represents servlet technology api
jsp-api.jar -->represents jsp technology api

=> we can deploy web applications to web servers either as
directories or as war files..

=> In the development of core Java Apps , we use Jdk apis like java.lang, java.util and etc.. packages
=> In the development servlet comps (web comps) we use the support of servlet api

 servlet api old package names
 javax.servlet, javax.servlet.http, javax.servlet.annotation and etc..

java module names
JSE (java standard edition) -->jdk s/w
JEE (java/jakarta enterprise edition)
JME (Java Micro Edition)
and etc...

 servlet api new package names
 jakarta.servlet, jakarta.servlet.http, jakarta.servlet.annotation and etc..

JEE --> java enterprise edition(old)
JEE --> jakarta enterprise edition

 jsp api old package names are
 javax.servlet.jsp, javax.servlet.jsp.el, javax.servlet.jsp.tagext

all these are available in
servlet-api.jar

 jsp api new package names are
 jakarta.servlet.jsp, jakarta.servlet.jsp.el, jakarta.servlet.jsp.tagext

all these are available
in jsp-api.jar

note:: Every Servlet comp (web comp) is a java class that is developed by using servlet api , So before
compiling that java class we must add <Tomcat_home>\lib\servlet-api.jar file to CLASSPATH.

note:: if our java code any api that is not part jdk apis (core java apis) then that api related jar file
or directories should be added CLASSPATH env..

note:: Servlet api,jsp api and other JEE Technologies , frameworks api are part not of jdk apis.. they
are separate apis .. if u want to use them as part of your java code /app development then
add relevant jar files to CLASSPATH

jar file :: java level zip file

jar ->java archive file

Diff types of Containers based on the technology in which they are developed

- a) ServletContainer --> Developed based on ServletTechnology
- b) Jsp container --> Developed based on ServletTechnology + Jsp Technology
- c) EJB Container (old) --> Developed based on EJB Technology
- d) IOC /SpringContainer --> Developed in spring framework
and etc..

Diff types of containers based on the location of container with respect to server

a) Standalone Container

-> Here Servlet and container together comes as single s/w program

eg:: ServletRunner (very old)

Container cum Server



b) In Process Container

-> Here Container resides inside the server as separate unit which can be added/deleted based on the need..

eg:: Servlet container(catalina.jar) of Tomcat server
Jsp container (jasper.jar) of Tomcat server

Server



c) Out of Process Container

=>Here container resides outside the server as separate unit but linked with the Server..

eg: ServletContainer linked with IIS (.net server)
Jspcontainer linked with IIS (.net server)

Container Server



for FB group to get class notes

=====

<https://www.facebook.com/groups/388095825162910>

for vedios

=====

https://www.youtube.com/watch?v=2oTIHv-GwII&list=PLVIQHNRLfIP88b4uv8X_EtHAPHjkw1qc

email Id::

natarazjavaarena@gmail.com

Procedure to develop, deploy and test First Web application having static webpages (html files)

step1) keep the following the s/w setup ready

- a) editplus 2/3
- b) Tomcat server 9/10

step2) create Deployment Directory structure representing the web application..



=>WEB-INF is the fixed named folder that acts as partition in the web application to separate static, helper and dynamic web comps.

=> we generally place java classes based web comps like servlet comps inside WEB-INF folder like WEB-INF/classes folder (fixed names locations) and we place static , helper web comps like html files, images files, audio files, video files, js files and etc.. outside the WEB-INF folder.

note:: The deployment directory structure /staging directory structure is given Sun Ms as part of servlet specification and all servers requested to recognize it ..So we can say the deployment directory is common for all based servers.

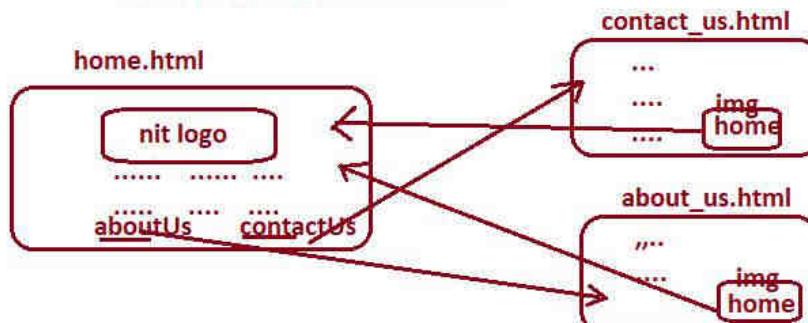
note:: web root folder name and location is not fixed.. we can take any thing but that folder name later becomes web application name in our case it is "FirstApp" but WEB-INF, classes, lib and etc are standard folder names i.e we can not change their names and locations..

We use WEB-INF/classes folder for placing java classes (both source code and compiled code) like servlet classes , helper classes and etc..

we use WEB-INF/lib folder for placing jar files

note:: In the current web application taking WEB-INF, classes folder is optional becoz we ^{not} do have any java classes as web comps or helper classes...

note:: Since Web Server and WebContainer takes the responsibility managing and executing web application ,So we must create deployment directory as expected by WebServer and webcontainer..



<h1> to <h6> :: To present content as headings

<p>/<pre> :: To present content as paragraph

**
 :: to get new line**

for html learning

<a> :: for hyperlink

<https://www.w3schools.com/>

** :: to display image**

<a> with :: for graphical hyperlink

<a> with text :: for textual hyperlink

** --> gives single space (entity in html)**

step3) develop web comps (html files) by arranging helper web comps (images)

=> refer git url :: <https://github.com/natarazworld/NTAJ414>

**ctrl+B :: to see preview
in edit plus browser...**

step1 to step3 :: complets the development of the web application

step4) start the Tomcat server...

use <Tomcat_home>\bin folder's tomcat10.exe file..

step5) Deploy the web application...

**copy G:\FirstApp folder to <Tomcat_home>\webapps folder
(E:\Tomcat 10.x)**

step6) Test the web application...

open browser and type this url in address bar

http://localhost:3030/FirstApp/home.html

**protocol the host name web app name/
and port number context path/
of Tomcat server context root**

web comp name

**note::: The web root folder name of deployment directory structure becomes
the name of the web application/context path/context root.**

-----home.html-----

```
<body bgcolor="cyan">
<h1 style="text-align:center;color:red"> Welcome to Naresh It </h1>

<h1>NNV Naresh, Founder and Managing Director. </h1>
<pre><b>A recipient of "International Achievers Award for Education Excellence", </b>
NNV Naresh is an entrepreneur armed with a noble vision to make a difference in the career aspirations
of the students. 20+ years of experience in the education sector, Naresh is the founder and the driving
force behind the victorious journey of NareshIT.
<h4>Mission Statement </h4>
To create and provide innovative, quality educational environments and opportunities via affordability
fostering the society to grow, thrive, and prosper. </pre>

<a href="contact_us.html">contactUs</a> &nbsp; &nbsp; &nbsp;&nbsp;&nbsp;<a
href="about_us.html">aboutUs</a>

</body>
```

-----contact_us.html-----

```
<body bgcolor="grey">
<h1 style="color:green;text-align:center"> Contact Us page </h1>
<pre>
<b>India – Hyderabad Office </b>
2nd Floor, Durga Bhavani Plaza, Ameerpet, Hyderabad Tel: +91 40 2374 6666 (IN – Hyderabad)
Tel: +91 40 2373 4842 (IN – Hyderabad)
+91 9000994007
+91 9000994008
for Projects : +91 9000994005
Email: info@nareshit.com

<b>India – Chennai Office</b>
2nd Floor Plot No.172 & 173, Above Axis Bank, Behind PTC Bus Stop, OMR, Thoraipakkam, Tamil Nadu, Chennai – 600097.
Mobile/Whats App: +91 9566042345
Email: chennai@nareshit.com

<b>USA Office </b>
5007 Arbor View Pkwy NW Acworth, GA, 30101
Ph: +1 404-232-9879, +1 248-522-6925
Email: sriram@nareshit.com

<b>For Online Training </b>
Mobile/Whats App:
+91 81 79 19 1999
+91 92 93 22 6789
Email: online@nareshit.com
</pre>

<br> <a href="home.html"></a>

</body>
```

about_us.html

=====

```
<body bgcolor="pink">
<h1 style="color:blue;text-align:center"> About Us Page </h1>

<pre>
Naresh i Technologies (Pronounced: NareshIT) is a leading software training institute providing Software Training, Project Guidance, IT Consulting and Technology Workshops.
```

Using our enhanced global software training delivery methodology, innovative software training approach and industry expertise, we provide high-value corporate training services that enable our clients to enhance business performance, accelerate time-to-market, increase productivity and improve customer service.

We serve Global 100 companies and the leading software vendors in Banking & Financial Services, Insurance, Telecommunications, Technology and Media, Information & Education industries. We design and mentor human resources for our clients who create competitive advantage.

Founded in 2004 and headquartered in Hyderabad, India, we have offices and training institutes throughout the India.

Headquarters – Durga Bhavani Plaza, Ameerpet, Hyderabad.

Offices – Hyderabad, Chennai.

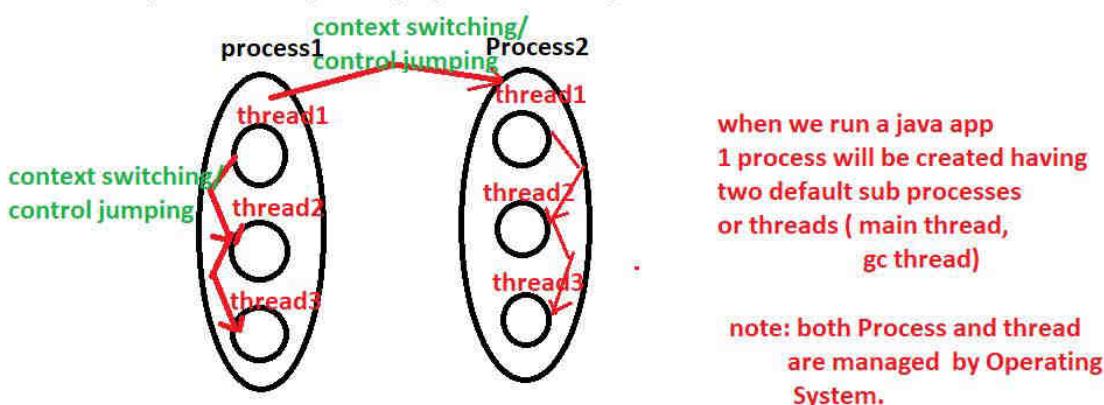
Training Institutes – Hyderabad, Chennai.

```
</pre>
<br> <a href="home.html"></a>
```

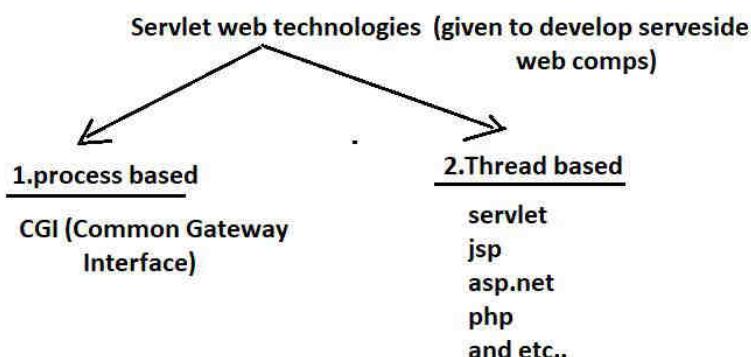
```
</body>
```

What is the difference between process and thread?

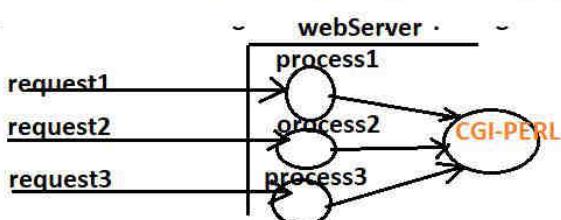
Ans) A thread is light weight process or sub process.



=>context switching /control jumping between two threads of a process always takes less time where compare to context switching/control jumping between two processes.



=>In process web technologies for every request given to web comp one separate process will be created



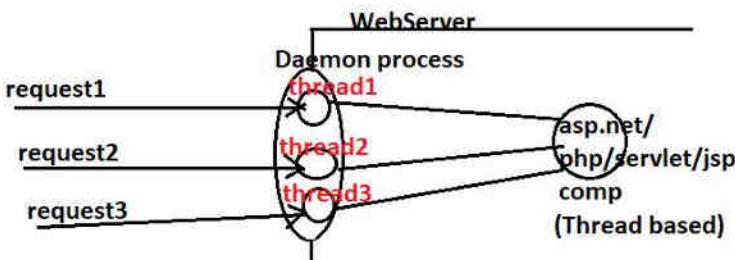
CGI Programs can be written
in diff languages like VB, C++,
PERL (best)

VB: Visual Basic
PERL :: practical extraction reporting
language

note:: Since the context switching b/w processes take more time the performance of web application becomes very poor when no.of requests are increased.. This indicates process based web comps like CGI web comps do not allow to develop scalable web applications..

note:: if the web application is giving good performance for less no.of requests, but giving bad performance for more no.of requests then it is called non-scalable web application.. CGI web applications are non-scalable.. (out dated)

=>if u give request to thread based web based web comp like servlet /jsp/asp.net/php comp.. one thread will be created for every request..



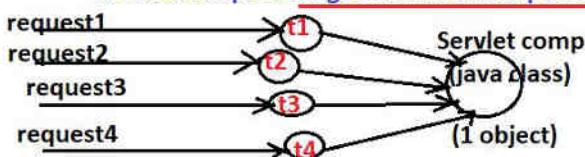
note:: Since the context switching b/w two threads of process takes less time the thread based web comps gives good performance irrespective of no.of requests increased or decreased.. This allows to develop scalable web applications..

What is Servlet? Definition of Servlet?

Ans1) Servlet is a java based web technology that allows us to develop dynamic web comps having ability to generate dynamic web pages..

Ans2) Servlet is JEE module Server side web technology that can be used to enhance the functionalities of web Server/Http Server .

Ans3) Servlet is a java based Server side web technology that allows to develop its web comps as single instance multiple threads based web comps.



=>Servlet comp is java class that is developed using servlet api
=>ServletContainer manages servlet comp life cycle i.e creating , managing and destroying of servlet class obj will be taken care by servlet container...

note:: if we give 10 requests to a servlet comp..then the ServletContainer creates only 1 object for servlet comp class and starts 10 threads on that object representing 10 requests. This make Servlet comp as "single instance -multiple threads" comp.

Ans4) Servlet is a JEE module server side web technology that looks like rules and guidelines for vendor companies to develop ServletContainers and same thing looks like api for programmers to develop server side web comps in java web application.

for videos

=====

https://www.youtube.com/watch?v=2oTlHv-GwII&list=PLVIQHNRLfIP88b4uv8X_EtHAPHjkw1qc

for classnotes

=====

<https://www.facebook.com/groups/388095825162910/files>

understanding servlet api

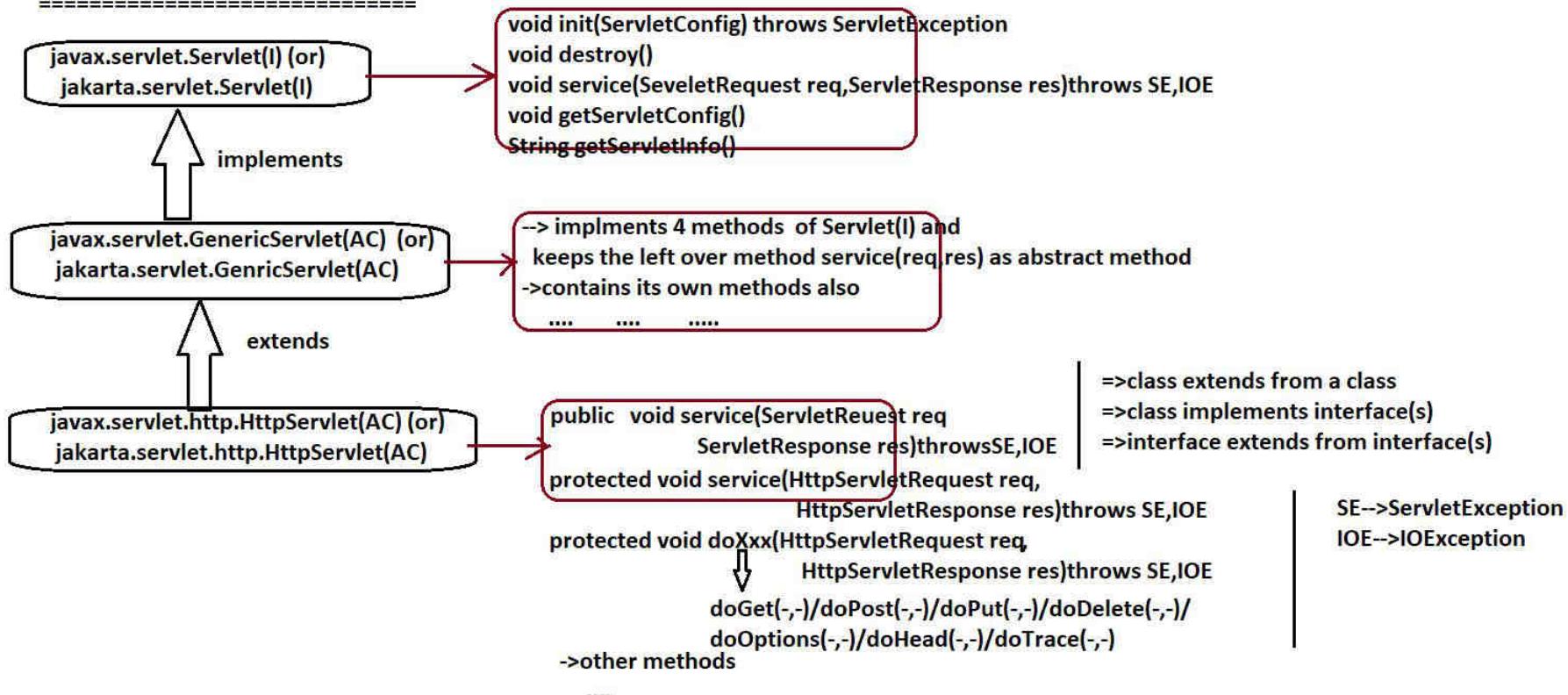
=====

it is having 4 packages

old days :: javax.servlet, javax.servlet.http,
javax.servlet.annotation , javax.servlet.descriptor

from tomcat10 :: jakarta.servlet, jakarta.servlet.http,
jakarta.servlet.annotation , jakarta.servlet.descriptor

3 important resources of servlet api



=>**javax.servlet.Servlet(I) or jakarta.servlet.Servlet(I)** is having 5 methods declaration.

=> **javax.servlet.GenericServlet(AC)** is implementing the above **Servlet(I)** providing implementation for 4 methods out of 5 methods. does not implement **service(req,res)** method ,So it is maintained abstract methods by giving **GenericServlet** itself as abstract class. This class also having its own direct methods..

=>**javax.servlet.http.HttpServlet(AC) (or) jakarta.servlet.http.HttpServlet(AC)** is an abstract class containing no abstract methods.. It is having 2 **service(-,-)** methods (One inherited from **GenericServlet** and another one is direct method) .It is also having 7 **doXxx(-,-)** methods..lots of other methods..

=>In java programming we can take abstract class only having abstract methods or only having concrete method or mix of both

=> In SErvlet comp development , there will be no main(-) method becoz Servlet comp is not a stanalone App to begin the execution with main(-) method.. It is a web comp whose life cycle is managed by SErvletContainer birth to death(object creation to obj destruction). This Servletcontainer calls service(-,-) method on Servlet comp for request ,So we place request processing logic in service(-,-) method servlet comp.

=>SErvlet comp is a java class that is developed by using servlet api .Since this class is reusable class .. we can call it servlet comp/servlet comp class.

3 Approaches of development SErvlet comp

Approach1 Take a java class implementing javax.servlet.Servlet() / jakarta.servlet.Servlet() and provide implementation for all the 5 methods .. having request processing logic in service(-,-) method.

//sample class

```
public class TestServlet implements javax.servlet.SERVLET/jakarta.servlet.Servlet{  
    ....  
    .... // all the 5 methods implementation including service(-,-)  
    ....  
    public void service(SERVLETRequest req,SERVLETResponse res) throws SE,IOE{  
        .... //req proceing logic  
        ....  
    }  
}
```

Limitations of Approach1

a) generally we need to place only service(-,-) having request processing logic.. but in this approach programmer forced to implment all 5 methods..through remaining 4 methods are unnecessary.

b) We can not work with protocol http features like "auto refresh" and etc..

c) It is not industry standard..

note:: Servlet comp is a java class that implements javax.servlet.Servlet() /jakarta.servlet.Servlet() directly or indirectly.

note:: ServletContainer takes only those java classes as servlet comp classes who are implementing javax.servlet.Servlet() /jakarta.servlet.Servlet() directly or indirectly..

Approach2 Take a java class extending from javax.servlet.GenericServlet() / jakarta.servlet.GenericServlet() and provide implemtation only for service(-,-) method having request processing logic..

//sampe code

```
public class TestServlet extends GenericServlet{  
    public void service(SERVLETRequest req, SERVLETResponse res) throws SE,IOE{  
        .... //request processing logic  
        ....  
    }  
}
```

=>Only in standalone Apps development , we can see main(-) method becoz jvm directly executes the App

=>In webApps,webServices development ,there is no need of main(-) method. becoz here comps are executed by Containers.. not the jvm directly

while iheriting from abstract class we can override concre methods (but not mandatory) and we must implement abstract methods..

Advantages

=> We need to implement only required service(-,-) method.. i.e no need of implementing unnecessary methods

Limitations

=> we can not work with protocol "http" features..
=> It is not industry standard approach..

Approach3 Take a java class extending from javax.servlet.http.HttpServlet(Ac)/
===== (Best) jakarta.servlet.http.HttpServlet(AC) and override
one of two service(-,-) methods or one of 7 doXxx(-,-) method having request
processing logic..

//sample code

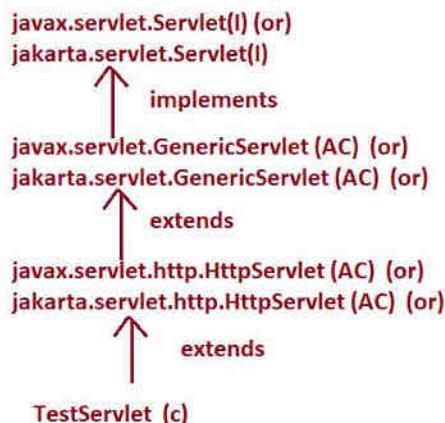
```
public class TestServlet extends HttpServlet{  
  
protected void service(HttpServletRequest req, HttpServletResponse res) throws SE,IOE{  
    .... //request processing logic  
    ...  
}  
}
```

advantages

=>We can override our choice service(-,-) or doXxx(-,-) methods having
request processing logic
=>Allows to work with protocol "http" features
=> It is industry standard approach to develop servlet comp class.

GenericServlet (AC) :: does not allow to use
protocol "http" features..

HttpServlet (AC) :: allows to work
protocol "http" features..



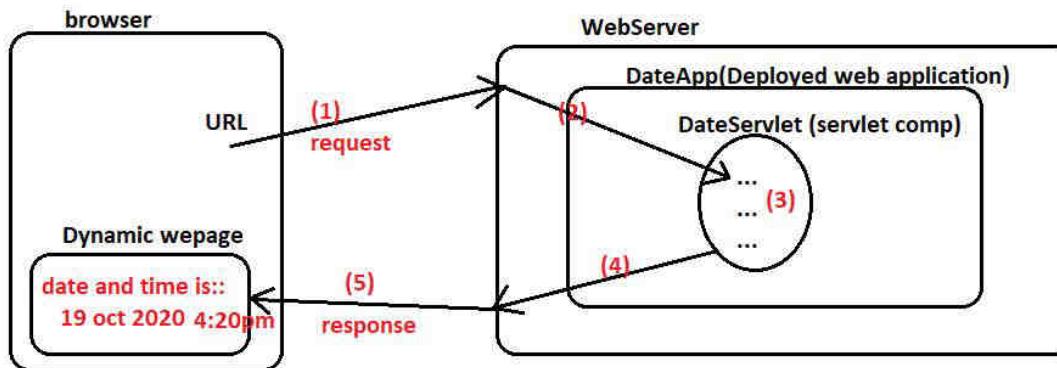
note:: we must take our servlet comp class as public class
to make it visible outside its package , So servlet container
can load the class and can manage the life cycle..

for videos

===== https://www.youtube.com/watch?v=2oTIhv-GwII&list=PLVIQHNRLfIP88b4uv8X_EtHAPHpjkw1qc

for class notes::

<https://www.facebook.com/groups/388095825162910/files>



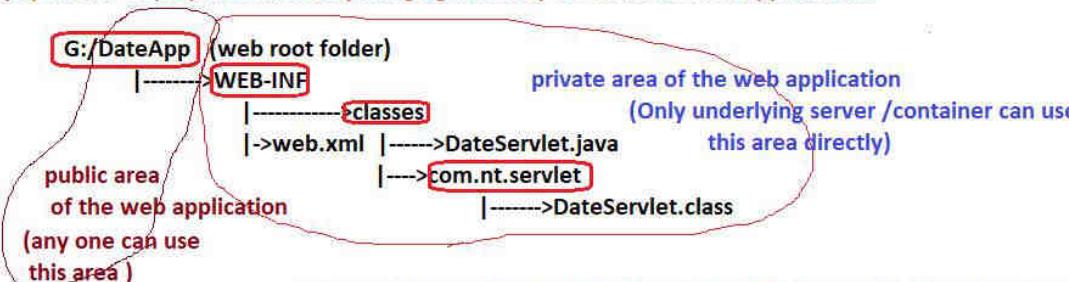
step1) make sure that the following s/w setup is ready

Tomcat 9.x/10.x

jdk 1.8+

editplus 2+ /notepad++

step2) create Deployment directory /staging directory structure for web application..



=>request coming to public area web comps like html files does not have any Server/container permission to execute the web comps whereas the request coming to private area web comp like servlet comps will execute web comps only with permission of Server/container and we need to get permission through web.xml file..entries

=>Java classes and java classes web comps are asked to place in private area (WEB-INF and its sub folder) to provide protection for source code(.java) and byte code(.class).

=> we place html files (static web comps) in public area (outside of WEB-INF folder and in the Webroot folder) because the html files code goes to browser for execution , So there is no possibility of protected source code of the html files.. whereas as the servlet comps code executes in the server and does not go to browser in any angle.. So the source protection is possible for servlet comps by keeping them private area.

=>web.xml , WEB-INF,classes folder,file names and locations are fixed .

=> webroot folder name and location is developer choice.. but later it becomes the name of web application...

note:: The WebContainer generally looks for java classes and servlet comps in private area like WEB-INF/classes folder and it looks for static web comps (like html files) and helper comps (like images, audio files ,video files and etc...) in public area of the web application...

step3) Develop the Servlet comp in WEB-INF/classes folder

```
//DateServlet.java  
=====  
import javax.servlet.*; import java.io.*; import.java.util.*;  
jakarta  
public class DateServlet extends GenericServlet{ //incomplete  
  
    public void service(ServletRequest req,ServletResponse res) throws SE,IOE{  
  
        //get PrintWriter stream obj from res obj  
        PrintWriter pw = res.getWriter();  
  
        //write b.logic /request processing logic  
        Date d = new Date();  
  
        //write output message to res obj using PrintWriter obj  
        pw.println("<b> date and time:::" + d.toString() + "</b>");  
  
    } //service(-,-)  
}  
//class
```

=> We use PrintWriter or other streams in service(-,-) method to write generated output messages to response obj.. In this process there is a possibility of getting IOException.

=> If WebContainer is having any problem towards calling and executing service(-,-) method then it throws ServletException...

=> System.out stream is pointing to console/monitor so System.out.println() writes the messages to console..
=> res.getWriter() given PrintWriter stream is pointing to response obj , So pw.println() writes the messages to response obj.

=> For every request ServletContainer creates 1 set of req,res objs and also calls service(-,-) method on servlet comp having that req ,res objs as the arguments..

=> Programmer uses the req, res objs in service(-,-) method while writing the request processing logic.. "req" obj is very useful together inputs that are coming along with the request (like form data) where programmer places generated output in "res" object in order to send to browser as response ..

=> The output/results written to response obj goes to browser through web Container and webServer. At the end browser displays that output/response as web page..

service(-,-) generates output -->res obj --> web container--> webServer --> browser as response --> displays as webpage.

=> A Stream is continuous flow of data which can be used to read data from destination and also to write data to destination

=> Reader/InputStreams are capable of reading data from destination

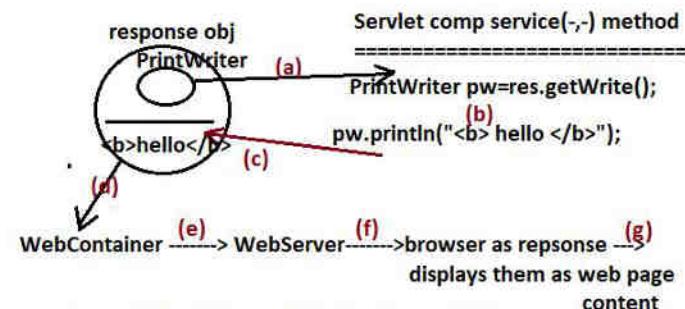
=> Writer/OutputStreams are capable of writing data to destination.

=> A stream can have file, console, keyboard, object, array and etc.. as the destinations.

=> System.in stream destination is keyboard

=> System.out stream destination is console

=> FileInputStream/FileOutputStream takes the file as the destination



=> response obj is having one built-in stream called PrintWriter having response obj it self as the destination ..By calling res.getWriter() method, we get that stream to service(-,-) method and we use it to write output to response obj by calling pw.println(-,-) method..

`res.setContentType("text/html");` gives instruction to browser through `res` obj, WebContainer, webServer to treat the received output as html tags based text content.. so the web page will be displayed on browser by recognizing the html tags of the output text.

note:: Based on the content type instruction the browser s/w becomes ready to display web page in different formats on the browser.

content types/ MIME types	<p><u>text/html</u> ---> displays the web page as html tags based text content</p> <p><u>text/xml</u> ---> displays the web page as xml tags based text content</p> <p><u>application/msword</u> --> displays the web page as msword application document</p> <p><u>application/ms-excel</u> --> displays the web page as msexcel application document</p> <p><u>text/plain</u> ---> displays the web page as plain text content.. no html tags, xml tags will be recognized</p>
------------------------------	---

MIME:: Multipurpose Internet Mail Extension

(E:\Tomcat\lib\servlet-api.jar)
step4) add <Tomcat_home>\lib\servlet-api.jar file to CLASSPATH... becoz our servlet comp class is servlet api that is available in servlet-api.jar file.

This PC ---> properties ---> advanced system settings ---> env.. variables --->
system variables ---> new -->

variable name :: CLASSPATH

value :: E:\Tomcat 10.0\lib\servlet-api.jar;.

-->ok -->ok -->ok

"." --> means current directory

System variables	
Variable	Value
CLASSPATH	E:\Tomcat 10.0\lib\servlet-api.jar;.

step5) compile servlet comp class

G:/DateApp/WEB-INF/classes>javac -d . DateServlet.java
(make sure that the active java version is java8/9)

To create .class file in a package that is specified (com.nt.servlet)
(com.nt.servlet)
new package will be created in the current directory

step6) Develop web.xml file to configure Servlet comp and to map Servlet comp with url.

=> Servlet configuration means giving the class name and package name of Servlet comp to ServletContainer to recognize given class as Servlet comp

=> SErvlet mapping means linking certain request url with Servlet comp asking ServletContainer to allow request to private area SErvletComp for the specified request url

=>Xml is case-sensitive
=>Xml strictly typed

```
web.xml
=====
<web-app>
    <!--Servlet configuration-->
    <servlet>      logical name
        <servlet-name>date</servlet-name>
        <servlet-class>com.nt.servlet.DateServlet</servlet-class>
    </servlet>      fully qualified servlet class name

    <!--SErvlet mapping-->
    <servlet-mapping> logical name
        <servlet-name>date</servlet-name>
        <url-pattern>/first</url-pattern>
    </servlet-mapping>  request url
</web-app>
```

The logical name of <servlet> tag and the logical name of <servlet-mapping> must match.. to link/map certain request url with certain servlet comp class .

note:: Here we are requesting SErvletContainer to allow request going to "DateServlet" t comp private area for the request url "/first"

note:: step1 to step5 completes the development of web application...

step6) Start Tomcat server

Go To <Tomcat_home>\bin directoy and tomcat10.exe file..

step7) Deploy the web application (DateApp)

copy G:/DateApp folder to <Tomcat_home>\webapps folder..

step8) Test the web application...



=>The Servlet comp is identified with its mapped url, So we must use that url to send request to servlet comp..i.e Servlet comp is not identified with its class name or logical name..

=>The modifications done in Servlet comp Soruce code in the deployed web application will reflect only after recompilation of servlet comp and reloading of the web applicaiton..

To recompile servlet comp class

E:\Tomcat 10.0\webapps\DateApp\WEB-INF\classes>javac -d . DateServlet.java

To reload the web application

open tomcat home page (http://localhost:3030) -->Manager App ----> username:admin , password: admin

Go to DateApp web application --> reload (stop and starting the web application)

DateApp	None specified		true	0	Start	Stop	Reload	Undeploy
					Expire sessions	with idle ≥ 30		

Different reasons to get 404 error

404:: requested resource not found

- a) wrong request url in the browser address bar
- b) Wrong Deployment Directory structure (wrong folder names or locations)
- c) for wrong entries in web.xml file
- d) Wrong web.xml file name or location
and etc..

In Servlet comp , the message kept in pw.println() goes to browser through res objs, webcontainer and webServer where as the message kept in System.out.println() goes to Tomcat server console as log message/confirmation message..

ServletContainer creates our servlet comp class obj using only 0-param constructor.. So we must place 0-param constructor in our servlet comp class directly (placed programmer) or indirectly(Generated by javac compiler as default constructor) if not placed then we get jakarta.servlet.ServletException: Error instantiating servlet class [com.nt.servlet.DateServlet]

=>The modifications done html files, web.xml file in the deployed web application will reflect automatically i.e there is no need reloading of web application.. where as the modifications done in the soruce code of Servlet comp will reflect only after recompilation of servlet comp and and reloading the web application..

Different reasons to get 500 Error

500:: Servlet Instantitation Problem

- (a) if Servlet comp class is not having public 0-param cosntructor directly or indirectly..
- (b) if the servlet comp class is not taken as public class..
- (c) if wrong class name of package name is specified in <servlet-class> of web.xml file
and etc..

=>if Servlet comp is getting 10 requests then ServletContainer
-> creates 1 object for servlet comp class
-> creates 10 sets of req , res objects representing requests
-> creates 10 threads representing 10 requests

Single instance multiple
threads behaviour..

note:: if each requested related response is delivered back to browser then
that request related thread, (req,res objs) will be destroyed automatically

To prove the above concept practically we need to place the following code in service(-,-) method of servlet comp and we need to give multiple requests simultaneously either diff browsers or different tabs of same browser..

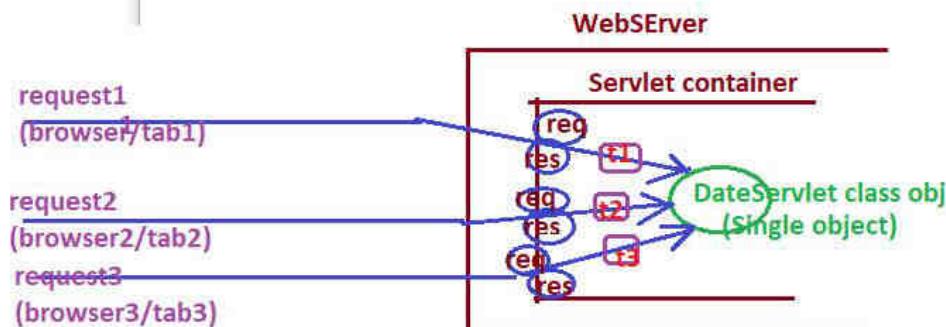
```
pw.println("<br>req obj hashCode::"+req.hashCode());  
pw.println("<br>res obj hashCode::"+res.hashCode());  
pw.println("<br>our servlet class obj hashCode::"+this.hashCode());  
pw.println("<br>current request thread obj hashCode::"+Thread.currentThread().hashCode());
```

browser1 (Edge)	browser2 (Chrome)	browser3(fireFox)
req obj hashCode::753429267 res obj hashCode::573542654 our servlet class obj hashCode::1889589100 current request thread obj hashCode::638582883	req obj hashCode::230661548 res obj hashCode::58225259 our servlet class obj hashCode::1889589100 current request thread obj hashCode::122156764	req obj hashCode::1560122371 res obj hashCode::38645805 our servlet class obj hashCode::1889589100 current request thread obj hashCode::210956223

welcome to Servlet ...

Date and time :: Wed Oct 21 17:26:00 IST 2020

t1,t2,t3:: threads..



Servlet Technology is giving rules to construct web.xml file having fixed tag names and attribute names in form of DTD document or XSD document.

DTD → Document Type Definition (Old – bad)

XSD → XML Schema definition. (Latest – Best)

XML file (with out XSD/DTD rules)

all tags and attributes are user-defined

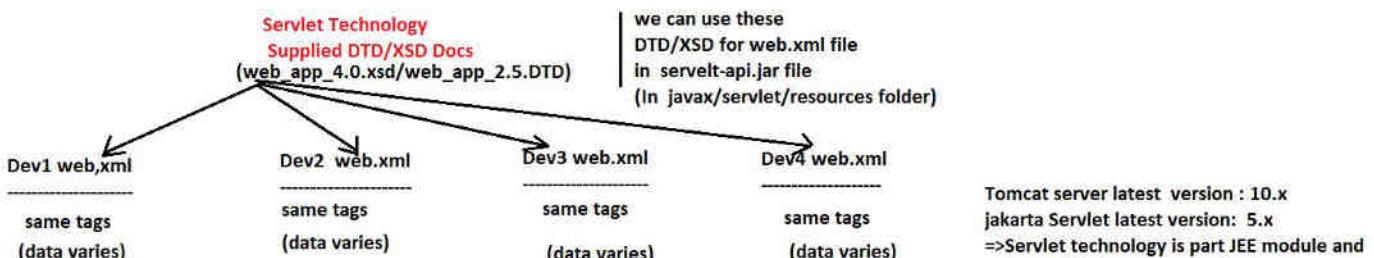
(It is like getting bio-data with out format)

XML file (Based on XSD/DTD rules)

all tag names, attribute names, structure and etc.. are fixed as collected from XSD/DTD rules..

(It is like getting bio-data with format)

If every developer uses his own tags while creating web.xml then the Servlet container can not apply meanings for the tags and tags data.. So Servlet technology is giving rules in the form DTD/XSD documents to all the developers to make all the developers developing their web.xml file based common rules, tags attributes and etc.. So Servlet container can apply standard meaning these fixed tags..



=>if XML document is satisfying basic syntax rules (Strictly typed rules) then it is called "well-formed" XML document..

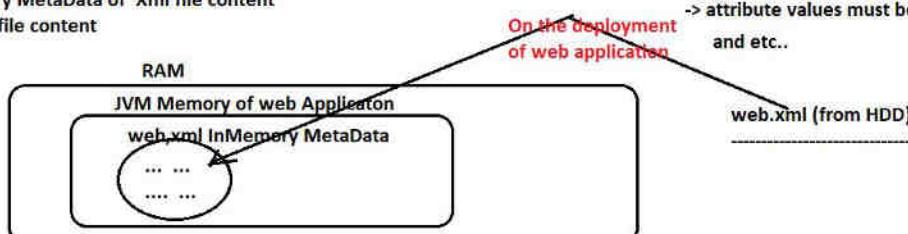
=> If XML document(file) is satisfying DTD/XSD doc rules then it is called valid XML document..

=> XML parser is Java based software that can perform the following operations

- a) Loading XML document
 - b) Checking well-formed, valid or not?
 - c) Reading XML file content
 - d) Create InMemory MetaData of XML file content
 - e) Processing XML file content
- and etc..

Strictly typed rules/Basic syntax rules

- > tag, attribute names are case-sensitive
- > tags must be nested properly
- > special chars are not allowed in the tag, attribute names
- > attribute values must be quoted.
- and etc..



eg: SAX Parser, DOM Parser, JDOM Parser, DOM4J Parser and etc...

SAX :: Simple API for XML Processing

DOM :: Document Object Model

JDOM :: Java Document Object Model

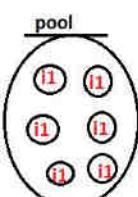
DOM4J :: Document Object Model for Java

Servlet container internally uses one or another XML parser (like SAX parser) to create InMemory Metadata of web.xml file.. by loading and verifying web.xml file..

=> XML Parser creates InMemory MetaData of XML file (like web.xml) in the In Memory where Application runs (that is JVM Memory of RAM) and that MetaData will be used for multiple during the execution of the Application.. This improves the performance becoz it avoids loading of XML files for multiple times, verifying XML files for multiple times..

Note: When the Application is stopped or completed execution.. its JVM memory in the RAM will be vanished.. In that process In Memory MetaData of XML file like web.xml will also be vanished..

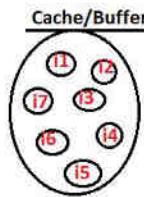
What is the difference between Pool and Cache?



[set of same items]

[Gives the reusability of same items]

[use List collection to implement this]

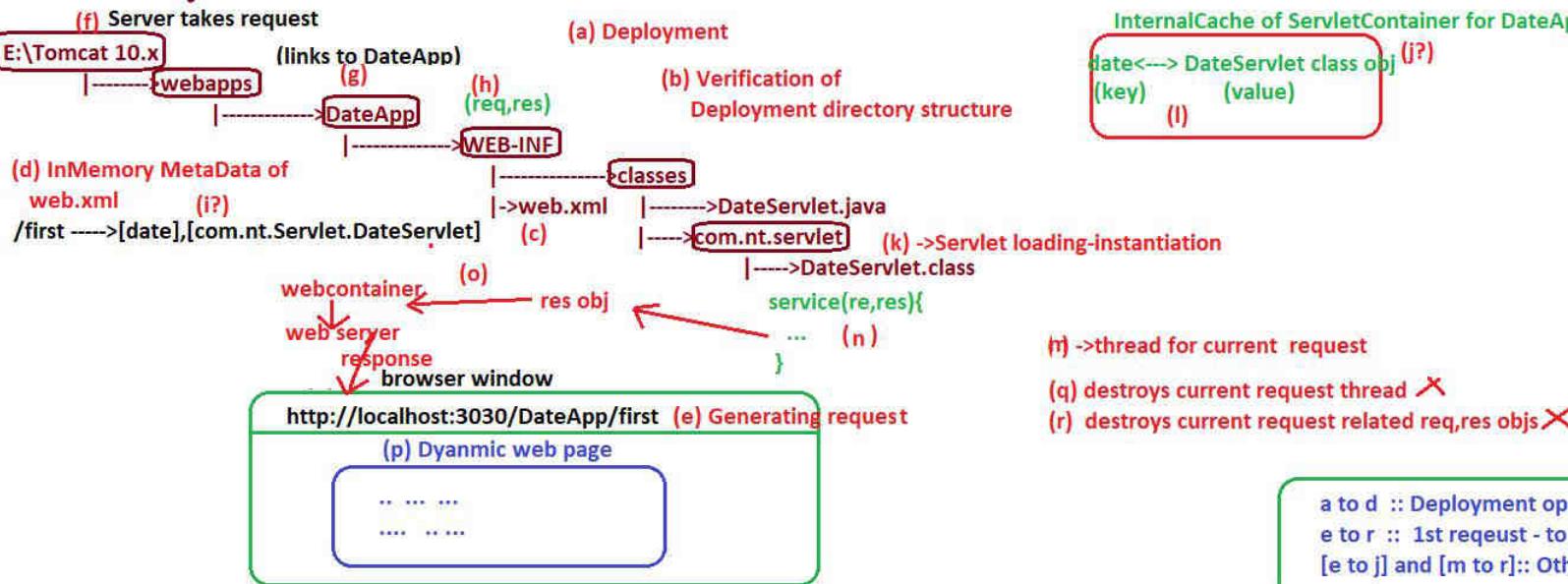


[Set of different items]

[Gives the reusability of different items]

[use Map Collection to implement this]

=> Both Pool and Cache are temporary memory allocating memory in JVM memory of RAM.. So they will be vanished once App execution over..



- Programmer has deployed DateApp web applicaiton in Tomcat server
- Tomcat server or its webcontainer verifies the deployment directory structure of the deployed DateApp web applicaiton
- webcontainer loads web.xml file and checks wheather it is well-formed, valid xml document or not .. if not exception will be thrown..
- webContainer creates InMemory MetaData of web.xml file in the JVM memory of RAM where webcontainer is running the deployed DateApp web application.
- Enduser generates the request by typing request url in the browser address bar
`http://localhost:3030/DateApp/first`
- Becoz of localhost:3030 info request url , the Tomcat server traps and takes the request..
- Tomcat server handovers the request to its WebContainer and this web container maps the request to the deployed "DateApp" web applicaiton based on "DateApp" info of the request URL
- WebContainer creates 1 set of req,res objs for current request
- Based on "/first" of the request url , the WebContaienr seraches in the InMemory MetaData of web.xml file for as Servlet comp /web comp cfg that is mapped "/first" url .. and "com.nt.servlet.DateServlet" as servlet comp class and "date" as the logical name
- webContainer Takes the logical name "date" and verifies for the SERvlet comp class object in internal cache and finds nothing.. (becoz of 1st request to servlet comp)
- WebContainer loads com.nt.servlet.DateServlet class from WEB-IN/classes folder , creates the object, intializes the object..
- Webcontainer keeps com.nt.servlet.DateServlet class object in the Internal cache having logical name (date) as the key and Servlet comp class object(DateServlet class obj) as the value.

- (m) webContainer creates 1 thread for current request.
 - (n) The created thread calls service(-,-) on DateServlet class object having req,res objs as the arguments..
 - (o) service(-,-) process the request and writes the generated output browser as response through res obj --> webCotnainer ---> webserver.
 - (p) Browser recives the response and displays it on the browser as dyamic web page
 - (q) Web container destroys the current request related thread
 - (r) Web container destroys the current request related res,res objs.
-

Q) Can i map servlet comp with multiple url patterns/urls?

Ans) Yes possible as shown below

```
<!--Servlet configuration-->
<servlet>
    <servlet-name>date</servlet-name>
    <servlet-class>com.nt.servlet.DateServlet</servlet-class>
</servlet>

<!--SErvlet mapping-->
<servlet-mapping>
    <servlet-name>date</servlet-name>
        <url-pattern>/first</url-pattern>
        <url-pattern>/second</url-pattern>
</servlet-mapping>
```

=>With the support of mapped url of Servlet comp/web comp we hide the technology of web applicaiton in which it being developed.. This process gives protection to web applicaiton from hackers and jackers..

in web.xml

=====

```
<url-pattern>/nareshit/ameerpet/first.php</url-pattern>
```

the request url :: http://localhost:3030/DateApp/nareshit/ameerpet/first.php

=>web.xml file is called "Deployment Descriptor file" becoz webContainer reads this web.xml file the moment we deploy the java web application.

=>web.xml file is also called as "web applicaiton configuration file" becoz we cfg multiple instructions to WebContainer from this file like

- =>Servlet/jsp/html cfg
- =>Servlet/jsp/html mapping with url/url patterns
- => Welcome/home page cfg
- => Security cfgs
- => Session timeout cfgs
- and etc..

In the Deployment directory structure of Java web application only 4 standard names are there..

they are a) WEB-INF b)classes c) web.xml d)lib (we place inside WEB-INF) and remaining all folder and file names are developer choice

To display image on Servlet comp generated web page

=====

a) place image in public area

E:\Tomat 10.x\webapps



b) add the following line in DateServlet comp class to display the image

```
pw.println("<img src='images/nit.jpg' width='100' height='60'>");
```

contentType/MIME type

MIME :: Multipurpose Internet Mail Extension

text/html, text/plain, text/xml , application/ms-word, application/vnd.ms-excel and etc..

In Servlet comp

```
res.setContentType("<mime type>");
```

eg:

```
res.setContentType("text/html");
```

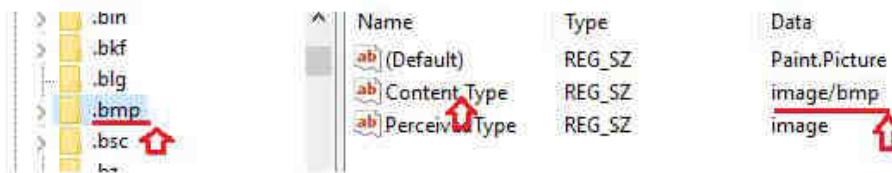
**==>Servlet comp Gives instruction to browser along with response (output) through
res object,WebContainer,webServer to display the received output/response
in the MIME type format.**

note::In most the browsers text/html will be taken as the default MIME type.

=>By using regedit (RegistryEditor) tool we can get MIME types of different file types.

Procedure::

**windows button + r --> type regedit --> expand HKEY_CLASSES_ROOT -->choose extesion and get its
content type in right side panel.**



**windows +shift+s :: to get snippet tool
based screen short**

App3) Develop web application having multiple Servlet comps generating the response in different formats

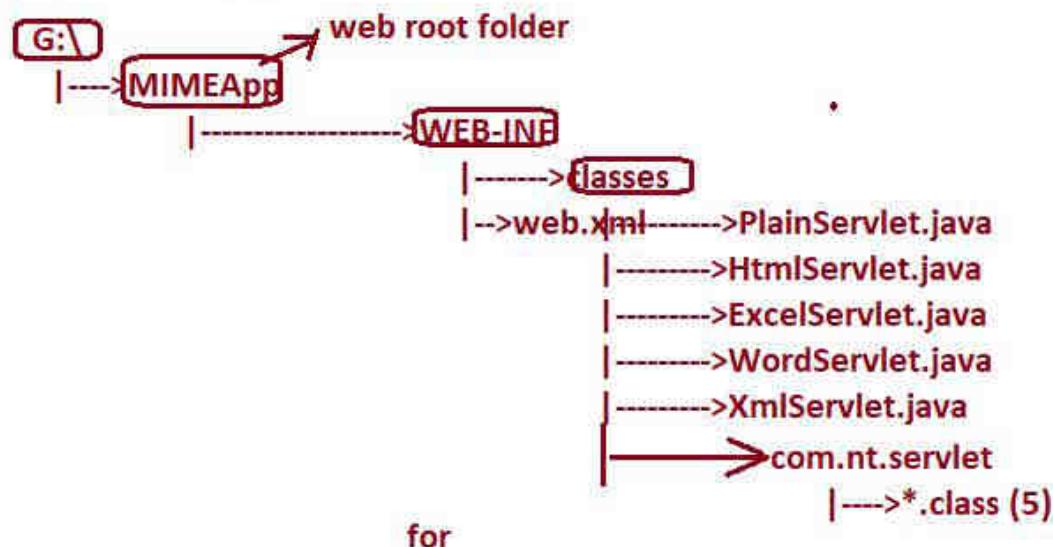
step1) make sure that following s/w setup ready

=>Tomcat 9.x/10.x

=> EditPlus 2/3

=> jdk 1.8/1.9 / any compatible version.

step2) create Deployment Directory Structure



step3) develop the source code Servlet comps..

`jakarta.servlet.GenericServlet(AC)` gives only 1 service(-,-) that is "public void service(ServletRequest req, ServletResponse res)throws SE,IOE"

`jakarta.servlet.http.HttpServlet(AC)` gives 2 service(-,-) methods they are

a)public void service(ServletRequest req,ServletResponse res)throws SE,IOE (Inherited from GenericServlet)
(1st service(-,-) method)

b)protected void service(HttpServletRequest req,HttpServletResponse res)throws SE,IOE -->Direct methof HttpServlet
(2nd service(-,-) method)

ServletRequest,ServletResponse type req,res objs do not not allows us to work with protocol http features

HttpServletRequest,HttpServletResponse type req,res objs allow us to work with protocol http features

`jakarta.servlet.ServletRequest(I)`



`jakarta.servlet.ServletResponse(I)`



```
//HtmlServlet.java
//package creation/delcaration
package com.nt.servlet;

//packages import
import jakarta.servlet.*;
import jakarta.servlet.http.*;
import java.io.*;

public class HtmlServlet extends HttpServlet
{
    //2nd service(-,-) method/protected service(-,-) method
    protected void service(HttpServletRequest req,HttpServletResponse res) throws ServletException,IOException{
        //set response content type
        res.setContentType("text/html");
        //get PrintWriter stream obj from response obj
        PrintWriter pw = res.getWriter();
        //write output message to response obj
        pw.println("<table border='1' bgcolor='cyan' align='center'>");
        pw.println("<tr><th>Team name </th> <th> owner name </th> <th> captain name </th> </tr>");
        pw.println("<tr><td>MI </td> <td> Raliance </td> <td> Rohit sharma </td> </tr>");
        pw.println("<tr><td>DC </td> <td> GMR-JSW </td> <td> Shreys iyer </td> </tr>");
        pw.println("<tr><td>KIXP</td> <td> Preity Zinta</td> <td> KL Rahul </td> </tr>");
        pw.println("<tr><td>CSK </td> <td> Srinivasan </td> <td> Ms dhone </td> </tr>");
        pw.println("<tr><td>SRH </td> <td> Sun network</td> <td> David warner </td> </tr>");
        pw.println("<tr><td>KKR </td> <td> Sharkh khan </td> <td> Eoin morgan </td> </tr>");
        pw.println("<tr><td>RCB </td> <td> Vijay maliya </td> <td> Kohli </td> </tr>");
        pw.println("<tr><td>RR </td> <td> raj kundra </td> <td> steve smith </td> </tr>");
        pw.println("</table>");
        //close stream
        pw.close();
    }//service(-,-)
}//class
```

step4) make sure that servlet-api.jar is added to CLASSPATH env.. variable..

=>same as previous app

step5) compile SErvlet comps...

```
G:\MIMEApp\WEB-INF\classes>javac -d . *.java
```

step6) Create web.xml inside WEB-INF folder having all the 4 servlet comps cfgs and servlet mappings

```
<web-app>
  <!--SErvletCfg --giving servlet comp class details to WebContainer -->
  <servlet>
    <servlet-name>html</servlet-name>
    <servlet-class>com.nt.servlet.HtmlServlet</servlet-class>
  </servlet>
  <!--SErvletMapping --linking Servlet comp with request url/path -->
  <servlet-mapping>
    <servlet-name>html </servlet-name>
    <url-pattern>/htmlurl </url-pattern>
  </servlet-mapping>

  <!--SErvletCfg --giving servlet comp class details to WebContainer -->
  <servlet>
    <servlet-name>excel </servlet-name>
    <servlet-class>com.nt.servlet.ExcelServlet</servlet-class>
  </servlet>
  <!--SErvletMapping --linking Servlet comp with request url/path -->
  <servlet-mapping>
    <servlet-name>excel </servlet-name>
    <url-pattern>/excelurl </url-pattern>
  </servlet-mapping>

  <!--SErvletCfg --giving servlet comp class details to WebContainer -->
  <servlet>
    <servlet-name>word </servlet-name>
    <servlet-class>com.nt.servlet.WordServlet</servlet-class>
  </servlet>
  <!--SErvletMapping --linking Servlet comp with request url/path -->
  <servlet-mapping>
    <servlet-name>word </servlet-name>
    <url-pattern>/wordurl </url-pattern>
  </servlet-mapping>
```

note:: All servlet comps of web application must be configured in web.xml file having unique fully qualified class names, logical names and url /url patterns.

```
<!--SErvletCfg --giving servlet comp class details to WebContainer -->
<servlet>
    < servlet-name>plain </servlet-name>
    < servlet-class>com.nt.servlet.PlainServlet</servlet-class>
</servlet>
<!--SErvletMapping --linking Servlet comp with request url/path -->
<servlet-mapping>
    < servlet-name>plain </servlet-name>
    <url-pattern>/plainurl </url-pattern>
</servlet-mapping>

<!--SErvletCfg --giving servlet comp class details to WebContainer -->
<servlet>
    < servlet-name>xml</servlet-name>
    < servlet-class>com.nt.servlet.XmlServlet</servlet-class>
</servlet>
<!--SErvletMapping --linking Servlet comp with request url/path -->
<servlet-mapping>
    < servlet-name>xml </servlet-name>
    <url-pattern>/xmlurl </url-pattern>
</servlet-mapping>
</web-app>
```

step7) Start Tomcat server

use tomcat10.exe file of <Tomcat_home>/bin directory..

step8) Deploy the web application

copy G:/MIMEApp folder to <Tomcat_home>/webapps folder..

step9) Test the Application..

In browser address bar

=====

http://localhost:3030/MIMEApp/htmlurl
http://localhost:3030/MIMEApp/excelurl
http://localhost:3030/MIMEApp/wordurl
http://localhost:3030/MIMEApp/xmlurl
http://localhost:3030/MIMEApp/plainurl

=>While overriding super class methods in sub class we can use either same modifier or strong modifier (recomanded)

=> while overriding super class protcted methods in sub class we can use the same protected moidifier or stroing public modifier (strong is recomanded for visibility)

=>The protected service(-,-) method HttpServlet can be overridden in our servlet comp class either with same protected modifier or with strong public modifier (recomanded)

=>Since there is possibility of overriding 2nd service(-,-)/protected service(-,-) method of HttpServlet having public modifier in our servlet comps..So it is recomanded to indentify service(-,-) methods as 1st or 2nd service(-,-) methods based their parameter types , not based on their modifers..

In our servlet comp

```
=====
public void service(ServletRequest req,ServletResponse res )throws SE,IOE
(1st service(-,-) method)
```

```
public void service(HttpServletRequest req,HttpServletResponse res) throws SE,IOE
(2nd service(-,-) method)
```

How can u enable auto-refresh on the web page generated by Servlet comp?

Ans) we can give instruction to browser along with response with the support reponse header "refresh" to enable auto-refresh the web page..

In Servlet comp 2nd service(-,-) method (should extend HttpServlet (AC))

```
=====
res.setHeader("refresh", "10"); (or)
res.setIntHeader("refresh",10); --> makes the browser to refresh the
web pag after every 10 seconds..
```

autofresh on the web page is every useful while displaying live game scores, Stock market share values, random advertisements and etc..

=> response header are part of the generated response to browser giving instruction to browser towards displaying the web page...
contentType , refresh, location, server, and etc.. are response headers...

Q) SERVLETREQUEST,HTTPSERVLETREQUEST AND SERVLETRESPONSE ARE THE INTERFACES OF SERVLET API.. THEN HOW CAN SAY REQ, RES ARE THE OBJECTS CREATED SERVLETCONTAINER AS WE KNOW INTERFACES CAN NOT BE INSTANTIATED?

Ans1) req,res objs are not objects of the above said interfaces.. they are the objects of impl classes given by underlying WebContainer/WebServer..

=>req obj is the object of underlying webContainer supplied java class implements jakarta.servlet.http.HttpServletRequest(I) directly or indirectly

=>res obj is the object of underlying webContainer supplied java class implements jakarta.servlet.http.HttpServletResponse(I) directly or indirectly.

note: we do not expose req ,res obj class names in our servlet comp development becoz they are specific to each web container and if we specify them our servlet comp becomz WebContainer/WEBSERVER Dependent servlet comp.. So we refer req, res objs using the common ServletRequest(I)/HttpServletRequest(I) and ServletResponse(I)/HttpServletResponse(I) reference variables.. So that our servlet comp can be moved any WebServer/WebContainer with out any modifications.

In Servlet comp

=====

```
pw.println("<br> req object class name ::"+req.getClass()); //gives org.apache.catalina.connector.RequestFacade  
pw.println("<br> res object class name ::"+res.getClass()); //gives org.apache.catalina.connector.ResponseFacade
```

```
public void service(RequestFacade req, ResponseFacade res) throws SE,IOE{  
    ...  
    ...  
}
```

makes Servlet comp code
specific Tomcat WebServer...
(Tight coupling with Tomcat server)

```
public void service(HttpServletRequest req, HttpServletResponse res) throws SE,IOE{  
    ...  
    ...  
}
```

makes the servlet comp code
as server independent i.e
common to all servers.. nothing
makes servlet comp as WODA

(WODA :: Write Once Deploy Anywhere)

Ans2) => Serilizable object means , it is not object of java.io.Serializable(I), it is the object of java class that implements java.io.Serializable(I).

=>Map object means , it is not object of java.util.Map(I), it is the object of java class that implements java.Util.Map(I).

=>jdbc con object means , it is not object of java.sql.Connection(I), it is the object of java class that implements java.sql.Connection(I).

=>req object means , it is not object of jakarta.servlet.http.HttpServletRequest(I)/jakarta.servlet.ServletRequest(I), it is the object of java class that implements

jakarta.servlet.http.HttpServletRequest(I) /jakarta.servlet.ServletRequest(I)

=>res object means , it is not object of jakarta.servlet.http.HttpServletResponse(I)/jakarta.servlet.ServletResponse(I), it is the object of java class that implements

jakarta.servlet.http.HttpServletRspnset(I) /jakarta.servlet.ServletResponse(I)

Html to Servlet Communication

=>Html is given to develop static web pages , Servlet is given to develop dynamic web pages ,So to get interaction b/w static web pages and dynamic webpages we need to go for html to Servlet communication..

=> Sending request to servlet comp by typing url in the browser address bar is very complex and technical. The non-technical endusers i.e visitors of website can not do that work properly..So it is better to provide GUI env.. like hyperlinks , forms and etc.. for endusers to send request to servlet comp. for this we need **html to servlet communication.**

3 ways of html to Servlet communication

a) Using hyperlinks

=>To send request to servlet comp with out having end user supplied data

eg: getting current trading jobs , show trending news , getAllEmployees (**usecases**)

b) Using forms

To send request to servlet comp having enduser supplied inputs

eg:: Login page/form , registration form/page , payment page/form (**use-cases**)

c) Using Java script

To send request to servlet comp based different java script events that are raised and from diff comps.

eg:: for Onload event (when web page is loaded)

request fetches all countries from servlet comps and puts in selectBox

usecases eg: after choosing new email id , the request goes automatically to check wheather it is already available or not (OnBlur event -> form comp loosing the focus)

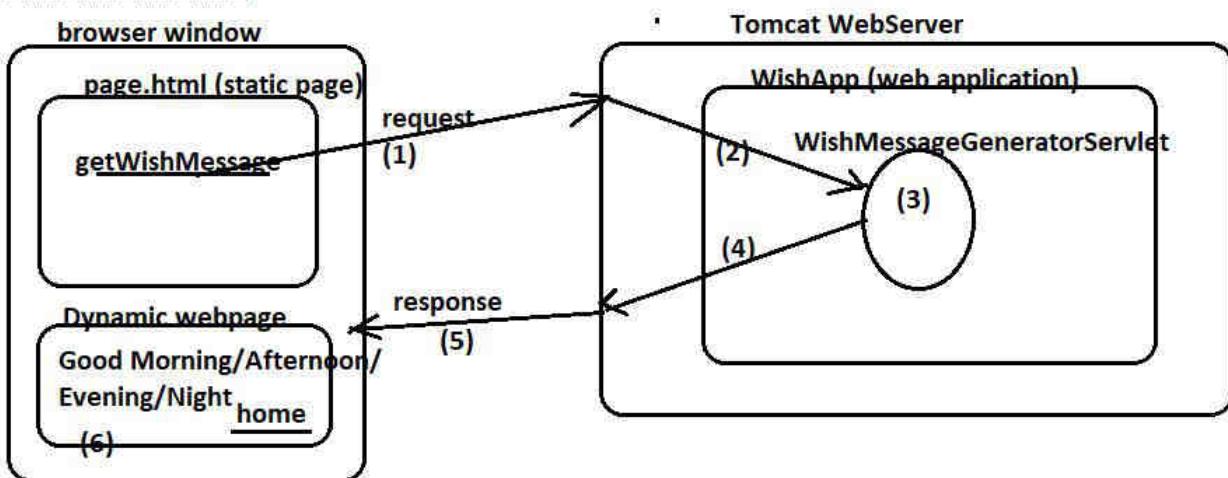
eg:: selecting a country in one select box sends request and gets list of states from Servlet comp and puts them in another select box. (**Onchange Event :: This raises when u select item from the select box**)

1 Html to servlet communication using hyperlinks

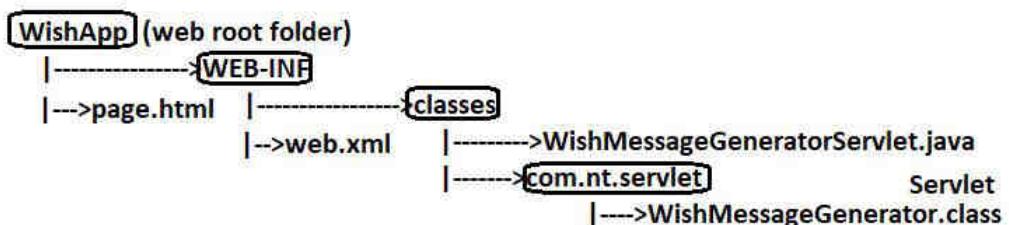
=>For this we need to place the request url of servlet comp as the value of "href" attribute in the <a> tag.

syntax :

Example Application



we can directly create the Deployment directory structure in <Tomcat_home>\webapps folder itself.



WishMessageGeneratorServlet comp is mapped with `"/wishurl"` url pattern,
so the request url for the servlet comp is `"http://localhost:3030/WishApp/wishurl"`

page.html (c)

```
<h1 style="color:blue;text-align:center"> Html to Servlet communication using hyperlinks </h1>
<h2 style="color:red;text-align:center"> <a href="http://localhost:3030/WishApp/wishurl">
getWishMessage </a> </h2>
```

(d)

WishMessageGenerator.java

```
//WishMessageGeneratorServlet.java
package com.nt.servlet;

import jakarta.servlet.*;
import jakarta.servlet.http.*;
import java.io.*;
import java.time.*; //java 8

public class WishMessageGeneratorServlet extends HttpServlet
{
    // 2nd service(-,-) method      (g)
    public void service(HttpServletRequest req, HttpServletResponse res) throws ServletException, IOException{
        //get PrintWriter object
        PrintWriter pw = res.getWriter();
        //set response content type
        res.setContentType("text/html");
        // Get System Time..
        LocalTime time = LocalTime.now();
        //get current hour of the day
        int hour = time.getHour(); // gives in 24 hours format
        //generate wish Message
        String msg = null;
        if(hour < 12)
            msg = "Good Morning";
        else if(hour < 16)
            msg = "Good Afternoon";
        else if(hour < 20)
            msg = "Good Evening";
        else
            msg = "Good Night";
        //write wish message to the response object
        pw.println("<h1 style='color:blue;text-align:center'>" + msg + "</h1>");
        //add home hyperlink
        pw.println("<a href='http://localhost:3030/WishApp/page.html'>home </a>");
        //close stream
        pw.close();
    }//service(-,-)
}//class
//cmd>javac -d . WishMessageGeneratorServlet.java
```

web.xml

```
<web-app>
    <!-- Servlet configuration -->      (b1?)
    <servlet>
        <servlet-name>wish</servlet-name>
        <servlet-class>com.nt.servlet.WishMessageGeneratorServlet</servlet-class>
    </servlet>

    <!-- Servlet Mapping -->
    <servlet-mapping>
        <servlet-name>wish</servlet-name>
        <url-pattern>/wishurl </url-pattern>
    </servlet-mapping>
</web-app>
```

(e)

(a) deployment
operations

request path

request url:: http://localhost:3030/WishApp/page.html (b)

=>While linking the received request to web comp, the ServletContainer always gives first priority to private area web comps like servlet comps , if they are not available mapped with current request url then the same comps will be verified in public area

if the request path/url of servlet comp(private area web comp) and public area web comp (html file) is matching then the request goes to which web comp?

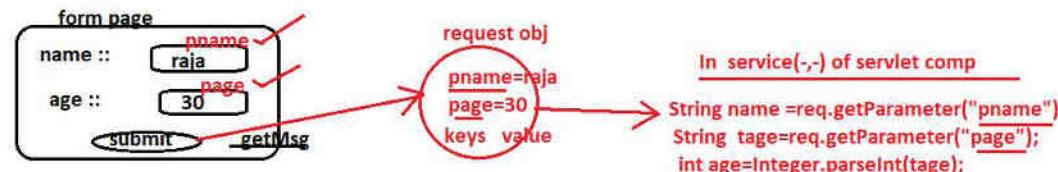
Ans) The request goes to private area web comp first (servlet comp)

java.util.Locale

=====

2.Html to Servlet communication using forms

- => Here request can be given servlet comp having form data supplied by end user
usecases :: registration pages, login page , payment page and etc..
- => we need to place Servlet request url in "action" attribute of <form> tag (as action url) to make form submitted request going servlet comp..
- =>The form data sent by form submission will be stored in to request object automatically having form comp name as the key and value/data as the value.. So we can read that form data in our servlet comp by using request object (like req.getParameter(-) method)



- => Browser can send request to servlet comp in two modes/methods/methodologies
- 1.GET mode** --> Designed to get data from server by sending request (default)
 - > url typed address bar, hyperlinks generates GET mode request to servers/web comps
 - > This mode request can carry 2kb to 8kb (In Most of browsers it is 2kb) data as query data /query String along with the request telling it wants certain type of data from server..

<http://localhost:3030/WishApp/wishurl?type=english &country=india>
query String having type,country as request param names
and english,india as the request param values

2.POST Mode --> Designed to send unlimited amount of data along with request

- > suitable form registration forms , file uploading activities..
- > A form page can send either GET (default) or POST(recomanded)
request request Server /Web comp

A typical form page contains

- mode of request (GET(default)/POST(recomanded))
- action url (target web comp(like servlet comp) request url)
- form comps (like text boxes, password boxes, radio buttons and etc..)
- Buttons (like submit(preferred) , reset and etc..)

sample form page

=====

action url

↓

```
<form action=" <request url of target web comp" method="POST">
    name :: <input type="text" name="pname"> request mode/method
    age :: <input type="password" name="page"> form comps which will become
    <input type="submit" value="send"> request param names and their value will
    </form> become request param values
```

When form is submitted using
this submit Button , the request goes to
that web comp(Servlet comp) whose request url is placed
as the action url in the "action" attribute of <form> tag.

note:: Developing servlet comp having service(-,-) method to process the request is not good practice becoz as a single method we can not differentiate logics for "GET","POST" mode requests..place doXxx(-,-) methods (7 methods) in servlet comp having request processing logic.. since we can place multiple doXxx(-,-) methods at a time in a servlet comp , so we can differentiate logics for different modes of requests

7 doXxx(-,-) methods are (Java methods)

=>Ftp client App /Console client Apps are useful to Host web application on the internet,to remove web application from the internet and also useful to add,remove web comps from the hosted web application..
In this situation , this ftp client tries to send PUT,DELETE and modes of requests..

=> While developing and using Restfull webServices we are going to work with all the 7 modes of request.

7 doXxx(-,-) methods are (java methods)

With all the 7 modes of request.

- ```
=====
```
- doGet(req,res) -----> To process "GET" mode request
  - doPost(req,res) -----> To process "POST" mode request
  - doHead(req,res) -----> To process "HEAD" mode request
  - doTrace(req,res) -----> To process "TRACE" mode request
  - doOptions(req,res) -----> To process "OPTIONS" mode request
  - doPut(req,res) -----> To process "PUT" mode request
  - doDelete(req,res) -----> To process "DELETE" mode request

Different clients like browser , ftp client and etc... togather can give 7 modes of requests to webapplicaitions/webcomps.. But browser can given only two modes of requests web comps/web application ..So while developing webapplications for endusers who operates website using browsers.. we just need to work with doGet(-,-),doPost(-,-) methods..

#### Conclusion::

- => Stop using service(-,-) method to place request processing logic in servlet comp becoz it can not differentiate logics for "GET" , "POST" requests..
- => use doGet(-,-) method having request processing logic in servlet comp to handle the "GET" mode requests coming from hyperlinks, urls typed in browser address bar , forms having method="GET".
- => use doPost(-,-) method having request processing logic in servlet comp to handle the "POST" mode requests coming from forms having method="POST" .

#### signature of 7 doXxx(-,-) methods

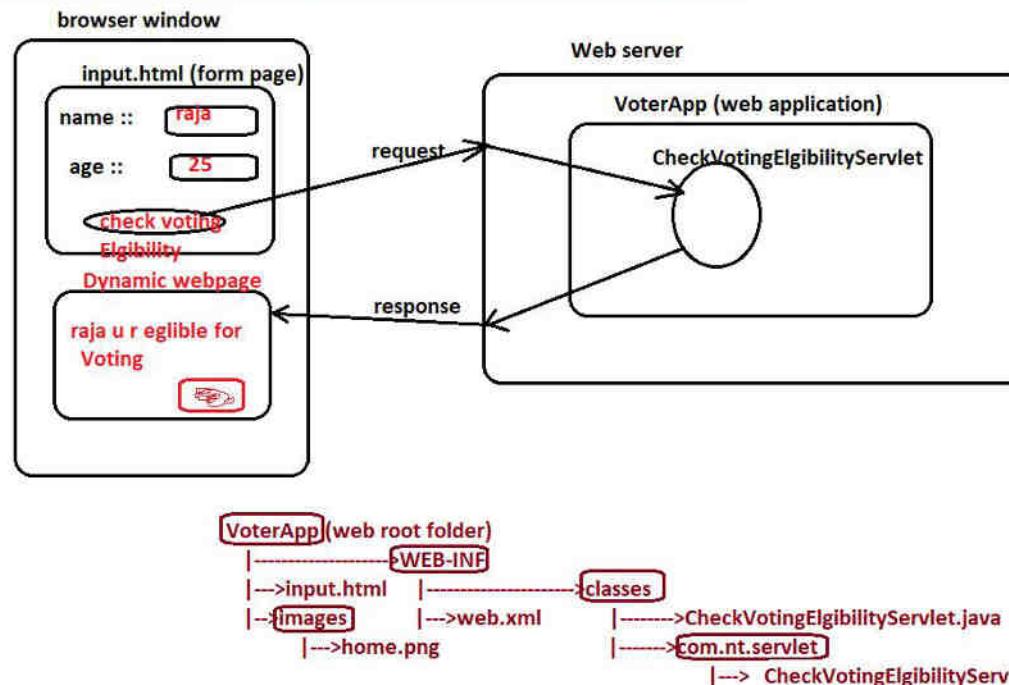
```
=====
```

```
protected void doXxx(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException
```

---

#### Example Web application on Html to servlet communication using form

```
=====
```



=>It is always recommended to take Html form comps as html table content to get control on alignment, styles and etc..

=>ServletContainer does not call doGet(-,-)/doPost(-,-) directly on our servlet comp, they will be called through super class (HttpServlet) service(-,-) method i.e the service(-,-) HttpServlet class internally calls doGet(-,-) method for GET mode request and calls doPost(-,-) for POST mode request.

====Code====

input.html (b)

```

<h3 style="color:red;text-align:center"> Html to Servlet communication using forms </h3>
<h1 style="color:red;text-align:center"> ELECTION COMMISSION OF INDIA </h1>
<body bgcolor="pink">
 <form action="http://localhost:3030/VoterApp/voterurl" method="POST">
 <table border="0" align="center" bgcolor="cyan">
 <tr>
 <td>Person name:: </td>
 <td><input type="text" name="pname"> </td>
 </tr>
 <tr>
 <td>Person age:: </td>
 <td><input type="text" name="page"> </td>
 </tr>
 <tr>
 <td><input type="submit" value="Check Voting Eligibility"> </td>
 <td><input type="reset" value="cancel"> </td>
 </tr>
 </table>
 </form>
 </body>
```

web.xml

```

<web-app>
 <servlet>
 <servlet-name>voting</servlet-name>
 <servlet-class>com.nt.servlet.CheckVotingEligibilityServlet</servlet-class>
 </servlet>
 <servlet-mapping>
 <servlet-name>voting</servlet-name>
 <url-pattern>/voterurl </url-pattern>
 </servlet-mapping>
</web-app>
```

(e)  
MetaData  
will be  
used

### CheckVotingEligibilityServlet.java

```
//CheckVotingEligibilityServlet.java
package com.nt.servlet;
import jakarta.servlet.*; //servlet api
import jakarta.servlet.http.*; //servlet api
import java.io.*; //IO streams api (f) Servlet Loading and Instantiation
public class CheckVotingEligibilityServlet extends HttpServlet
{
 (g) through HttpServlet's service(-,-)
 public void doPost(HttpServletRequest req,HttpServletResponse res)throws ServletException,IOException{
 //set response content type
 res.setContentType("text/html");
 //get PrintWriter
 PrintWriter pw=res.getWriter();
 //read form data (req parameter values from req object)
 String name=req.getParameter("pname");
 int age=Integer.parseInt(req.getParameter("page"));

 //write b.logic or request processing logic
 pw.println("<h1 style='color:maroon;text-align:center'> ELECTION COMMISSION OF INDIA </h1>");
 if(age>=18)
 pw.println("<h1 style='color:green;text-align:center'> Mr/Miss/Mrs. "+name+" you are eligible for Voting </h1>");
 else
 pw.println("<h1 style='color:red;text-align:center'> Mr/Miss/Mrs. "+name+" you are not eligible for Voting wait for "+(18-age)+" years for voting </h1>");
 // add graphical hyperlink for home
 pw.println(" ");
 //close stream
 pw.close();
 }//doPost(-,-)
}//class
```

(h)

(a)  
request url:: <http://localhost:3030/VoterApp/input.html> (from browser )

=>In Html to servlet communication do not give direct request to Servlet comp.. of servlet comp is not ready to process GET mode request , then we will get "405" error.

=>While overriding super class methods in sub classes it is recommended to place @Override on the top of method definitions.. in sub classes, So compiler never takes those methods as the direct methods in sub classes.. This helps to avoid typing mistakes , signature mistakes while overriding super class methods in sub classes.

note:: while overriding GenericServlet/HttpServlet's service(-,-)/doXXX(-,-) methods in our servlet comp it is recommended to add @Override as shown below.

```
@Override
public void doPost(HttpServletRequest req,
 HttpServletResponse res)throws ServletException,IOException{
 ...
 ...
}
```

## Eclipse

=====

type :: IDE for Java based applications development

version :: 2020-09

Vendor :: Eclipse Org

Open Source IDE (Free IDE)

Flavours :: Eclipse SDK (only for Standalone App development)

Eclipse JEE (for all kinds of App development including web applications,  
distributed Apps and etc..)

To download eclipse JEE :: download as zip file from

<https://www.eclipse.org/downloads/packages/release/kepler/sr2/eclipse-ide-java-ee-developers>

-->select windows 64 bit

To install Eclipse JEE :: Extract the zip file and use **eclipse.exe** to launch the IDE

Procedure to develop MarriageApp (checking whether person is eligible for marriage or not  
based on the given name, age and gender) using Eclipse JEE IDE

**step1) Launch Eclipse IDE by choosing the workspace folder**

The folder where Eclipse Projects will be saved.  
(G:\Workspaces\advjava\ntaj414)

**step2) Configure Tomcat Server with Eclipse IDE..**

Eclipse 2020-06 is not compatible with Tomcat 10.x, So keep the installation of  
Tomcat 9.x and configure it with eclipse..

window menu--->preferences ----> server ---> run time env., ---->  
add ---> select apache Tomcat 9.0 --->  
server name :: apache Tomcat 9.0  
Tomcat installation folder :: E:/Tomcat 9.0

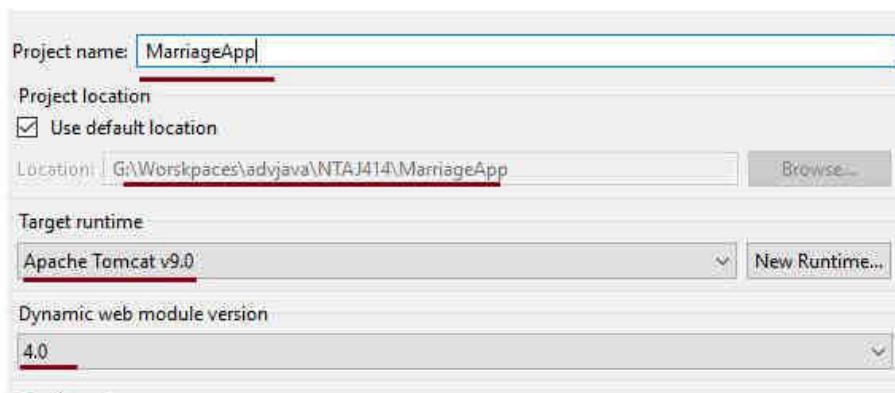
-->ok



In one computer, we can install multiple  
versions of jdk s/ws, eclipse s/ws and  
Tomcat s/ws and etc..

**step3) Create Dynamic web Project in Eclipse IDE by choosing Tomcat server as the target Rumtime env..**

File menu -->new -->Dynamic web project -->  
project name :: MarriageApp  
Target runtime env.. :: Tomcat 9.0  
web module version:: 4.0  
->next -->next --> select web.xml file -->finish.



**step4) Understand Project directory structure that is generated**



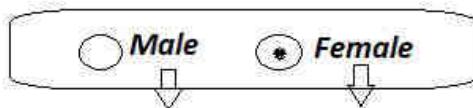
**note:: This is not Standard Deployment Directory Structure.. It is Eclipse Project structure.. to place various classes and files in to the project.. The deploymentn directory will be generated dynamically when we run the Project..**

**step5) place input.html (form page ) in webcontent folder**

**note:: do not take single radio button, becoz once it is selected we can not deslect it. Always take multiple radio buttons having same name to group them into single unit . so if radion button of the group is selected , the other ones will be deselected automatically.**

**note::** The selected radion button label will not go to server as req param value.. the data kept in "value" attribute of selected radion button goes to server as request param value..

```
<input type="radio" name="gender" value="M">Male
<input type="radio" name="gender" value="F" selected>FeMale
```



*Gives "M" or "F" as req param value based on radio button that is selected.*

### input.html

---

```
<body bgcolor="magenta">
 <h1 style="color:red;text-align:center">Shaadi.com</h1>
 <center>

 </center>

 <form action="http://localhost:2020/MarriageApp/marriageurl" method="POST">
 <table bgcolor="aqua" align="center">
 <tr>
 <td>Person name:: </td>
 <td><input type="text" name="pname"> </td>
 </tr>
 <tr>
 <td>Person age:: </td>
 <td><input type="password" name="page"> </td>
 </tr>
 <tr>
 <td>Person Gender:: </td>
 <td>
 <input type="radio" name="gender" value="M">Male
 <input type="radio" name="gender" value="F" checked>FeMale
 </td>
 </tr>
 <tr>
 <td><input type="submit" value="Check Marriage Eligibility"> </td>
 <td><input type="reset" value="cancel"> </td>
 </tr>
 </table>
 </form>
</body>
```

step 6) develop servlet comp in "java resources/src" folder in a java package  
"com.nt.servlet"

**MarriageServlet.java**

```

package com.nt.servlet;
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class MarriageServlet extends HttpServlet {

 @Override
 public void doPost(HttpServletRequest req, HttpServletResponse res) throws ServletException, IOException {
 // set response content type
 res.setContentType("text/html");
 //get PrintWriter
 PrintWriter pw=res.getWriter();
 //read form data (req param values)
 String name=req.getParameter("pname");
 int age=Integer.parseInt(req.getParameter("page"));
 String gender=req.getParameter("gender");
 //write b.logic or request processing logic
 if(gender.equalsIgnoreCase("M")) {
 if(age<21)
 pw.println("<h1 style='color:red;text-align:center> Mr."+name +" u r not eligible for marriage enjoy life </h1>");
 else
 pw.println("<h1 style='color:green;text-align:center> Mr."+name +" u r eligible for marriage , But think once </h1>");
 }
 else {
 if(age<18)
 pw.println("<h1 style='color:red;text-align:center> Miss."+name +" u r not eligible for marriage, be happy </h1>");
 else
 pw.println("<h1 style='color:green;text-align:center> Miss."+name +" u r eligible for marriage , But think thrice </h1>");
 }

 //add graphical hyperlink for home navigation
 pw.println("
 ");
 //close stream
 pw.close();
 }//doPost(-,-)
}//class
```

**step7) Develop web.xml having code to configure Servlet comp and mapping servlet comp with url pattern**

**web.xml**

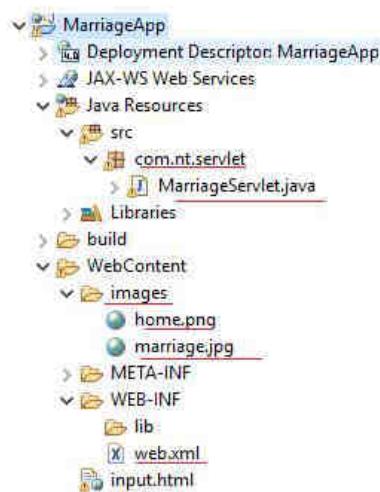
```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns="http://xmlns.jcp.org/xml/ns/javaee"
 xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
 http://xmlns.jcp.org/xml/ns/javaee/web-app_4_0.xsd"
 id="WebApp_ID" version="4.0">
 <servlet>
 <servlet-name>marriage</servlet-name>
 <servlet-class>com.nt.servlet.MarriageServlet</servlet-class>
 </servlet>
 <servlet-mapping>
 <servlet-name>marriage</servlet-name>
 <url-pattern>/marriageurl</url-pattern>
 </servlet-mapping>
</web-app>
```

**step8) Run the Application**

**Right click on project ----> run as --> run on server --->select Tomcat .....**

**request url::**

**http://localhost:2020/MarriageApp/input.html**



When we run the Dynamic web Project .. The eclipse generates the deployment directory structure dynamically in the Tomcat integrated with Eclipse..

G:\Worskpaces\advjava\NTAJ414\.metadata\.plugins\org.eclipse.wst.server.core\tmp0\wtpwebapps  
workspace location.

special folder of  
deployment  
in eclipse Tomcat..

note:: when Tomcat is configured with Eclipse IDE ,the Eclipse will not use that tomcat .. it creates a copy of tomcat in Eclipse workspace folder and use that tomcat..

G:\Worskpaces\advjava\NTAJ414\.metadata\.plugins\org.eclipse.wst.server.core\tmp0

The Location of Eclipse Tomcat with respect to our workspace folder..

To change port number of Eclipse Tomcat

=====

Project Explorer ---> servers --->Tomcat ---> open server.xml and modify "port" attribute value of <Connector> tag

(or)

Go to server tab ---> double click tomcat server ---> then modify "http/1.1" port number

=>By default eclipse web applications use eclipse internal browser .. to change to extenal browser use window-->browser ---> choose the browser (like chrome, firefox and etc..)

---

In eclipse based web application development the modifications done in the soruce code web comps will reflect dyamically becoz the eclipse IDE automatically reloads the web applicaton.. So in eclipse no need to think recompilation and reloading..

=> To make java web application as WODA(Write Once Deploy Anywhere) web application , we need to use relative urls in the place of absolute urls..

```
<form action="http://localhost:3030/MarriageApp/marriageurg">
.... =>absolute url which changed time to time based on the
... dest server port number and host name.. Not recommended to
... use.
}

<form action="marriageurl" >
.... relative url .. which server and host name independent.
... (recommended to use)
</form>
```

JAVA apps are WORA(write Once Run Anywhere)  
java web applications are WODA(Write Once Deploy Anywhere)

---

```
pw.println("home ");
 absolute url (bad practice)

pw.println("home ");
 relative url (good practice)
```

---

### welcome files configuration

---

=>The web page that comes automatically as first page when we give request to web application/website is called home page/welcome page/landing page .. we can configure html files or jsp files or servlet comps as welcome file comps in web.xml file using <welcome-file-list> <welcome-file> tags..

in web.xml

---

```
<welcome-file-list>
 <welcome-file> input.html </welcome-file>
 <welcome-file> input.jsp </welcome-file>
 <welcome-file> home.html </welcome-file>
</welcome-file-list>
```

if multiple welcome files are configured then the web container pickup the welcome files based on the order availability and order of configuration. and remaining files acts as back up welcome files.

=>Only In Tomcat server , if not welcome file is configured then index.html or index.jsp will be taken as default welcome (if both are there index.html will be taken as welcome file)  
(In other servers, there is no support for default welcome files)

=> if web application is having both default welcome files and explicitly cfg welcome files then one of the explicitly cfg welcome file will be taken welcome page or home page

=> if web application is having both default welcome files and explicitly cfg welcome files , but the explicitly cfg welcome files are not physical not there in the web application then that web application run with no welcome file.

=>we can cfg servlet comp as welcome file.. but we should take its url pattern( with out "/") as welcome file

## **Servlet Features**

---

- a.portable
- b.secured
- c.Integratable
- d.expendable
- e.inexpensive
- f.efficient
- and etc..

### **1.portable**

---

=>Servlet comps and its web applications are deployable in all java web servers with out having single modifications in the source code of web comps including servlet comps.. This servlet comps and its web applications as portable or **WODA** (Write Once Deploy AnyWhere).  
=>To continue this feature make sure u r working with "relative urls" and "interface model programming"

### **2.Secured**

---

=> While delivering servlet comps based web application to clients or while hosting same web applications on to the internet .. we just need to place servlet comps based ".class" files.. not .java file.. i.e we can protect servlet comp by hiding their source code.. This makes Servlet comps as secured comps.

=>By providing url pattern to servlet comps ,we hide the technology in which the web comp is developed from endusers/visitors of website. This is also a kind of security.

=>We keep servlet comps source and byte code in the private area of web application's directory structure to protect them from outsiders.. becoz only underlying Webserver/wecontainer can get access to private area web comps. This also provides safety..

### **Integretable**

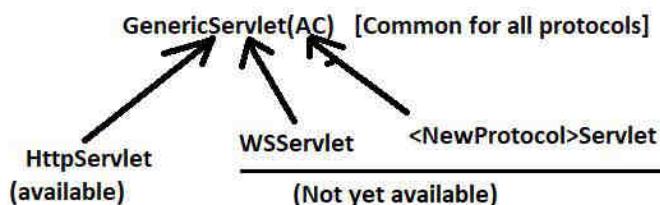
---

=> While developing servlet comps, we not only can use servlet api.. we can also ther apis like networking api, io streams api, utility api and etc..  
=> we can also intergrate servlet comps with hibernate,spring and other frameworks

### **Exapndable**

---

=> servlet technology is expandable to work with different protocols.. currently servlet api supporting protocol http... becoz entire web is designed based on porotocol "http".. if another protocol has come.. in web env.. our servlet technology is expandable to work with that protocol also.



**types**

How many of servlet comps are there?

wrong Answer is a)GenericServlet b) HttpServlet

correct answer is there is a possibility of developing "n" types of servlet comps like

HttpServlet, WSServlet, SMTPServlet and etc.. GenericServlet is not a seperate type of Servlet  
currently not available)

It is common super class for all types of servlet comps as discussed above.. As of now there is only one sub class for GenericServlet that is "HttpServlet" becoz the entire web is running based on the protocol "http".

**InExpensive**

=====

=>Servlet technology is an open technology and most of the vendors are giving servlet based webServers as free webServers.. So the development and deployment of java web applications doesnot need much money.

=> Servlet comps are single instance multiple thread comps.. this indicates servlet comp takes less memory and cpu time during the execution..

**note:: The above two points make servlet technology as InExpensive ..**

**Efficient**

=====

=>Using Servlet technology features , we can develop the web applications having end to end fetures.. i.e servlet technology is offering great features that are required web application development.

**Q) How to make our servlet comp flexible enough to process both GET,POST mode requests either having same logic or different logics**

**There are 3 approaches for this**

**Approach1**

=> provide request processing logic service(-,-) method

```
public class MarriageServlet extends HttpServlet{
```

```
 public void service(HttpServletRequest req,HttpServletResponse res) throws SE,IOE{
```

```

```

```

```

```
 request processing
```

```

```

```
}
```

```
}
```

sysout + ctrl+space :: gives System.out.println()

sout + ctrl+space :: gives System.out.println()

systrace +ctrl+sace :: gives system.out.println(-) with message

**note:: service(-,-) can process bot GET,POST mode/method requests..**

=>GET mode request sends data or gives query data as query String appended to the request url .. So it is not safe.

<http://localhost:3030/MarriageApp/marriageurl?pname=karan&page=34&gender=F>  
(not safe)

=>POST mode request sends data to server as request body (which is internal process) So no query String appended to the request url appears in the browser. It is safe to send data.

<http://localhost:3030/MarriageApp/marriageurl>

**Limitations of service(-,-) method**

a) we can not differentiate logics for GET,POST mode requests i.e if GET , POST mode requests are coming to servlet comp at a time (like hyperlink ,form submission) then we can not differentiate logics in servlet comp.

b) 1st service(-,-) gives ServletRequest, SErvletResponse objs which can not be used to work with protocol "http" features.. (as alternate we can use 2nd service(-,-) method

of

c) service(-,-) is not designed to process 7 modes requests that are coming to servlet comp from different types of clients..

on

d) service(-,-) is not designed based protocol "http" standards.. So it is not recommended to use in all angels..

**note:: To overcome all these limitations of service(-,-) method start using doXxx(-,-) methods (especially doGet(-,-),doPost(-,-) methods)**

## Approach2 (Best practice)

---

Version1: if both GET,POST mode requests want to have same request processing logic

=>Override both doGet(-,-),doPost(-,-) methods and place request processing logic in one method and call that method from another method.

```
public class MarriageServlet extends HttpServlet

 public void doGet(req,res)throws SE,IOE{

 ... //request processing logic
 ...
 }

 public void doPost(req,res)throws SE,IOE{
 doGet(req,res);
 }
}
```

Version2: if GET,POST mode requests want to have two different logics to execute.

=> override doGet(-,-) method having GET mode request processing logic and override doPost(-,-) method having POST mode request processing logic

```
public class MarriageServlet extends HttpServlet{

 public void doGet(req,res)throws SE,IOE{

 //logic for GET mode request..
 (like hyperlink request)
 }

 public void doPost(req,res)throws SE,IOE{

 //logic for POST mode request..
 (like form submission)
 }

}
```

### Approach3 (Extension of approach2) (not recommended)

---

version1:: if both GET,POST mode requests want to execute same request processing logic

Write request processing in user-defined method and call that method from both doGet(,-),doPost(,-) methods

```
public class MarriageServlet extends HttpServlet {
 // user-defined method..
 public void process(req,res) throws SE,IOE{ // it should have same signature of

 ... //request processing logic

 }

 public void doGet(req,res) throws SE,IOE{
 process(req,res);
 }

 public void doPost(req,res) throws SE,IOE{
 process (req,res);
 }
}//class
```

version2:: if GET,POST requests want to execute two different logics

=>write GET mode request procesing logic in one user-defined method and call it from doGet(,-) method, similarly write POST mode request procesing logic in another user-defined method and call it from doPost(,-) method.

```
public class MarriageServlet extends HttpServlet{

 public void processGet(req,res) throws SE,IOE{

 //logic for GET mode request

 }

 public void processPost(req,res) throws SE,IOE{

 //logic for POST mode request

 }

 public void doGet(req,res) throws SE,IOE{
 processGet(req,res);
 }
 public void doPost(req,res) throws SE,IOE{
 processPost(req,res);
 }
}
```

These two methods are use-defined methods but should have same signature of doXxx(,-) methods..

Conclusion:: Approach2 is good.. :: reason is no need placing extra user-defined methods..

with  
Advantages of working doXxx(-,-) methods (doGet(-,-),doPost(-,-) methods)

- a) allows to differentiate logics for both GET ,POST mode requests..i.e if GET,POST mode requests are given to servlet comp at time then we differentiate their logics using doGet(-,-),doPost(-,-) methods
- b) The parameters of doXxx(-,-) are HttpServletRequest, HttpServletResponse objs. so we can work with protocol "http" features..
- c) 7 doXxx(-,-) methods are given to handle 7 modes of requests comming from different types of clients
- d) doXxx(-,-) are designed based on the protocol "http" standards..

#### Form Validations

=>Verifying the pattern and format of form data before it is getting used in the b.logic as inputs is called form validations , otherwise b.logic may give invalid results or exceptions by taking the inputs...

eg:: checking required fields are filledup or not  
      checking wheather age is typed as numeric value  
      and etc..

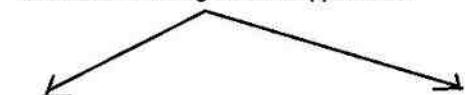
#### What is the difference b/w formvalidation logic and b.logic?

=>In form validation logic , the pattern and format of the form data will be verified. where b.logic/request processing logic takes form data as inputs uses them to process the request and to generate the results..

#### examples

- =>checking credit number is having 16 digits or not (format verified) (form validation)
- =>checking credit number is exsiting/valid number or not (b.logic)
- => checking wheather date vlaue is given in dd/MM/yyyy pattern or not (pattern verification) (form validation)
- => getting sales report before given data value (b.logic)
- => checking wheather age is given as numberi value or not.. and it is in the range of 1 through 150 is form validation logic
- => Checking wheather person is elible to vote or not based on the given age is b.logic
- and etc..

#### Form Validation logic in web applications



=>place java script code in form page as form validation logic  
=>Executes in browser by coming browser along with form page

=>place java code before b.logic in servlet comp as form validation logic  
=> always resides and executes in server by becoming part of Servlet comp code.

=>Decide wheather code is client side or server side code based on the place where is executing .. not based on the place where it is residing.

## Different approaches of writing form validation logics

---

### Approach1) Write only Server side form validation logics

Limitation:: if the form page is rejected for multiple times..then there will be more network round trips b/w browser and web server.

### Approach2) Write only client Side form validation logics

advantage:: reduces network trips b/w browser and webServer , even though form page is rejected for multiple times becoz of form validations

Limitation :: there is a possibility of disabling java script code execution through browser settings..then no form validation take place..

### Approach3) write both client side and server side form validations

Advantages:: reduces network trips b/w browser and webServer , even though form page is rejected for multiple times becoz of form validations

Though client Side java script disabled..the server side java code will take care of form validations..

Disadvantage :: => performance issue becoz of validating same form data at client side and server side

### Approach4) write both client side and server side form validations but enable server side form validations only when client side form validations are not done.. (Best practice)

Advantages:: reduces network trips b/w browser and webServer , even though form page is rejected for multiple times becoz of form validations

Though client Side java script disabled..the server side java code will take care of form validations..

=>No Performance issue becoz server side will executes only when client side form validations are not done.

=> if u r copy pasting Dynamic web project of Eclipse IDE... change its context path using web project settings.. other wise new web application name/context path will not reflect in urls..  
right click on project ---> properties ---> web project settings --->contextPath:: <new context path>--> ....

=> if the method return type is void .. then the return statement placed in method definition will not return the value.. but returns the control caller from the current method definition.. i.e return statement with out value can be used for not executing further statements in the java method definition.

```
public void m1(){
 ...
 ...
 return ;
 ... | These statements will not be executed
 .. }
```

note:: The error messages or success messages given by servlet/jsp comps can not be displayed in the same web page from where requested generated with out taking the support of ajax directly or indirectly.  
ajax :: AsynchronousJavaScriptAndXML (now a days every one using through angular/angularJs/ReactJs

place the following code before b.logic of MarriageServlet.java to perform server side form validations

```
===== //read form data (req param values)
String name=req.getParameter("pname");
String tage=req.getParameter("page");
String gender=req.getParameter("gender");
//form validation logic (server side)
List<String> errList=new ArrayList();
System.out.println("MarriageServlet.doPost(): Server side form validation logic");
if(name==null || name.equals("") || name.length()==0) { //required rule
 errList.add("Person name is required");
}
else if(name.length()<5) { //min length rule
 errList.add("Person name must have minimum 5 characters");
}

int age=0;
if(tage==null || tage.equals("") || tage.length()==0) { //required rule
 errList.add("Person age is required");
}
else {
 try{
 age=Integer.parseInt(tage); // numeric value rule
 if(age<1 || age>125) { //range rule
 errList.add("Person age must be given 1 through 125");
 }
 }
}
```

#### java8 features

=====  
=>forEach() method  
=>lambda expression  
=> functional interface  
=> :: reference operator  
=> Stream API  
=> Date Time API  
and etc...

```

 }//try
 catch(NumberFormatException nfe) {
 errList.add("Person age must be numeric value");
 }//catch
}//else
//Display form validation error messages
if(!errList.isEmpty()) {
 errList.forEach(msg->{ //java8 foreach() method with Lamda expression
 pw.println("" +msg+ "");
 });
 return;
}//if

```

---

#### Client Side form validation using java script

What is the different b/w java and java script?

##### java

- a) It programming language given by sun Ms/Oracle corp
- b) it is object oriented programming language
- c) To execute JAvA code,we need JRE/JVM
- d) In web application in can be in servlet/jsp comps as server side code.
- e) can be used to all kinds of applications like standalone Apps, web application, distributed Apps and etc..

##### java script

- a) It is scripting language whose code must be embedded with html tags directly or indirectly for execution and it is given by sunMs/oracle corp + Netscape.
- b) It is object based (no Inheritance and polymorphism) language
- c) To execute Java Script code , we need java script engine of browser s/w
- d) In web application , it can be used along with html code as client side code..
- e) can be used only in web application development.

##### JavaScript

=====

- =>It is based on DOM (Document Object Model).. i.e every tag will be treated as an object.. having methods and properties(attributes).
- =>It is giving lots of built-in objects.. like document, window, navigator, history,frames and etc..
- =>It supports event driven programing i.e we can execute logics based on the events that are raised like onclick, onblur, ondblclick, onchange and etc..
- => Java script not strictly typed
- => Java script is case-sensitive and etc..

## **Client side form validations in MarriageApp input.html form page using Javascript**

---

=>person name is required  
=>person name must have min of 5 chars  
=>person age is required  
=>person age must be numeric value  
=>person age must be there between 1 through 125

### **To define function in JS**

```
function <function-name>(params,...){

 ...
}
```

### **To declare variables in JS**

---

```
var <variablename>;
or
let <variablename>;
```

=>parameter data types and variable data types will be decided dynamically based on the values that are assigned.  
=>No return type is required for the function.. but function can return any value.

### **input.html (form page with client side form validations)**

---

```
<script language="JavaScript">
 function validate(frm){
 //read form data
 let name=frm.pname.value;
 let age=frm.page.value;
 let flag=true;
 //client side form validation logic
 if(name==""){ //required rule
 alert("person name is required");
 frm.pname.focus(); //text box gains the focus
 flag=false;
 }
 else if(name.length<5){ // min length
 alert("person name must have min of 5 chars");
 frm.pname.focus(); //text box gains the focus
 flag=false;
 }
```

```

if(age==""){ //required rule
 alert("person age is required");
 frm.page.focus();
 flag=false;
}
else if(isNaN(age)){ //numer value rule
 alert("person age is not a number");
 frm.page.focus();
 flag=false;
}
else if(age<1 || age>125){ // age range rule
 alert("person age must be through 1 through 125");
 frm.page.focus();
 flag=false;
}
return flag; //true --> no form validations , false for form validation errors
}//function

```

```

<body bgcolor="magenta">
<h1 style="color:red;text-align:center">Shaadi.com</h1>
<center>

</center>

```

```

<form action="marriageurl" method="POST" onsubmit="return validate(this)">
```

table bgcolor="aqua" align="center">

```

<tr>
 <td>Person name:: </td>
 <td><input type="text" name="pname" ></td>
</tr>
<tr>
 <td>Person age:: </td>
 <td><input type="text" name="page" ></td>
</tr>
<tr>
 <td>Person Gender:: </td>
 <td>
```

(Here keeping "return" is very important before js function call becoz it takes validate(-) function result (true/false) and gives to browser and browser uses that to check form validation errors are there (false) or not there(true) ..if not there it sends the request server based on action url.. otherwise it blocks the request in the browser itself)

```

 <input type="radio" name="gender" value="M">Male
 <input type="radio" name="gender" value="F" checked>Female
 </td>
</tr>
<tr>
 <td><input type="submit" value="Check Marriage Eligibility"></td>
 <td><input type="reset" value="cancel"></td>
</tr>

```

```

</table>
</form>
</body>

```

To disable java script using Mozilla FireFox

type "about:config" in browser address bar ---> accept risk ---> search java script ---> go to javascript.enabled key ---> change the value to false

To disable java script using chrome browser

Chrome "customize control button" : -->settings --> search for java script  
-->go to site settings ---> java script --->blocked.

---

HTML 5 is supplying some form validation rules.. like required,min,max,maxlength and etc..

Working with them are having some limitations

- a) Very less form validations are given
- b) Can not customize form validation error messages
- c) writing some validation logics through java script and some other logics through html 5 form validations does not look good.

and etc..

note:: Write all client side form validations using JAvascript..

```
<input type="text" name="pname" required maxlength="10">
input type="text" name="page" min="1" max="125">
```

=>To improve resuability of JS code across the multiple webpages given by different comps and to hild java script code soruce visibility from browser's view resoruce option.. it recomanded place java script code in .js file and link to file to multiple web comps..

input.html

=====

```
<script language="JavaScript" src="js/validation.js">
</script>
```

refer

=> innerHTML/outerHTML

=>document.getElementById(-)

webcontent

|-->js

|-->validation.js

```
function validate(frm){
```

...

...

...

...

```
}
```

=>In recent versions of html like html4/5 the <font> tag deprecated and as alternate two tags are given

- a) <span> --> To apply styles on 1 line of text
- b) <div> --> To apply styles on multiple lines of text..

note:: we can get any tag as object based on id by using document.getElementById(-) method as part of JS DOM Programming.. and we can replace body to that tag using innerHTML/innerText or we can replace total tag with new content using outerHTML/outerText.

```
document.getElementById("nameErr").innerHTML="person name is required";
```

=By using hidden box support the form page can send signal to Server/Servlet comp along with the request wheather client side JS form validations are performed or not.. if performed then it will disable server side form validations otherwise it will execute server side form validations..

(Approach4 based form validations)

input.html (c)

```
<script language="JavaScript" src="js/validation.js">
</script>

<body bgcolor="magenta">
<h1 style="color:red;text-align:center">Shaadi.com</h1>
<center>

</center>

<form action="marriageur" method="POST" onsubmit="return validate(this)">
<table border="1" align="center">
<tr>
<td>Person name:</td>
<td><input type="text" name="pname" ></td>
</tr>
<tr>
<td>Person age:</td>
<td><input type="text" name="page" ></td>
</tr>
<tr>
<td>Person Gender:</td>
<td>
<input type="radio" name="gender" value="M">Male
<input type="radio" name="gender" value="F" checked>Female
</td>
</tr>
<tr>
<td><input type="submit" value="Check Marriage Eligibility"></td>
<td><input type="reset" value="cancel"></td>
</tr>
</table>
<input type="hidden" name="vflag" value="no"/>
</form>
</body>
```

validation.js

```
(f)
function validate(frm){
 alert("client side...")
 //empty the form validation error messages
 document.getElementById("nameErr").innerHTML="";
 document.getElementById("ageErr").innerHTML="";

 //read form data
 let name=frm.pname.value;
 let age=frm.page.value;
 let flag=true;
 //client side form validation logic
 if(name==""){ //required rule
 document.getElementById("nameErr").innerHTML=<i>person name is required</i>;
 frm.pname.focus(); //text box gains the focus
 flag=false;
 }
 else if(name.length<5){ // min length
 document.getElementById("nameErr").innerHTML="person name must have min of 5 chars";
 frm.pname.focus(); //text box gains the focus
 flag=false;
 }
 if(age==""){ //required rule
 document.getElementById("ageErr").innerHTML=<i>person age is required</i>;
 frm.page.focus(); //text box gains the focus
 flag=false;
 }
 return flag;
}
```

```
</body>
```

```
if(age==""){ //required rule
 document.getElementById("ageErr").innerHTML="person age is required";
 frm.page.focus();
 flag=false;
}
else if(isNaN(age)){ //numer value rule
 document.getElementById("ageErr").innerHTML="person age must be numeric value";
 frm.page.focus();
 flag=false;
}
else if(age<1 || age>125){ // age range rule
 document.getElementById("ageErr").innerHTML="person age must be between 1 through 125";
 frm.page.focus();
 flag=false;
}
//change vflag (hidden box) value to "yes" to indicate client side form validations are done
frm.vflag.value="yes";
(g) return flag; //true --> no form validations , false for form validation errors
}//function
```

*MarriageServlet.java*

```
(1)
public class MarriageServlet extends HttpServlet {
 @Override
 (k)
 public void doPost(HttpServletRequest req, HttpServletResponse res) throws ServletException, IOException {
 System.out.println("MarriageServlet.doPost()");
 // set response content type
 res.setContentType("text/html");
 //get PrintWriter
 PrintWriter pw=req.getWriter();
 //read form data (req param values)
 String name=req.getParameter("pname");
 String tage=req.getParameter("page");
 String gender=req.getParameter("gender");
 String vstatus=req.getParameter("vflag");
 //execute server side form validation logic only when client side JS is disabled
 int age=0;
 if(vstatus.equalsIgnoreCase("no")){
 //form validation logic (server side)
 List<String> errList=new ArrayList();
 System.out.println("MarriageServlet.doPost(): Server side form validation logic");
 if(name==null || name.equals("") || name.length()==0){ //required rule
 errList.add("Person name is required");
 }
 else if(name.length()<5){ //min length rule
 errList.add("Person name must have minimum 5 characters");
 }
 if(tage==null || tage.equals("") || tage.length()==0){ //required rule
 errList.add("Person age is required");
 }
 else{
 try{
 age=Integer.parseInt(tage); // numeric value rule
 if(age<1 || age>125){ //range rule
 errList.add("Person age must be given 1 through 125");
 }
 } //try
 catch(NumberFormatException nfe){
 errList.add("Person age must be numeric value");
 }
 }
 }
 }
}
```

*web.xml*

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns="http://xmlns.jcp.org/xml/ns/javaee"
 xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
 http://xmlns.jcp.org/xml/ns/javaee/web-app_4_0.xsd"
 id="WebApp_ID" version="4.0">
 <servlet>
 <servlet-name>marriage</servlet-name>
 <servlet-class>com.nt.servlet.MarriageServlet</servlet-class>
 </servlet>
 <servlet-mapping>
 <servlet-name>marriage</servlet-name>
 <url-pattern>/marriageurl</url-pattern>
 </servlet-mapping>
 <servlet>
 <servlet-name>welcome</servlet-name>
 <servlet-class>com.nt.servlet.WelcomeServlet</servlet-class>
 </servlet>
 <servlet-mapping>
 <servlet-name>welcome</servlet-name>
 <url-pattern>/welcomeurl</url-pattern>
 </servlet-mapping>
```

```

 errList.add("Person age must be numeric value");
 } //catch
} //else
//Display form validation error messages
if(!errList.isEmpty()) {
 errList.forEach(msg->{ //java8 foreach() method with Lamda expression
 pw.println("" +msg+"");
 });
 return;
} //if
else {
 age=Integer.parseInt(age); // To convert String age value to numeric value
} //else

//write b.logic or request processing logic
System.out.println("MarriageServlet.doPost(): Server side b.logic");
pw.println("<h1 style='color:blue;text-align:center'> Shaadi.com </h1>");
if(gender.equalsIgnoreCase("M")) {
 if(age<21)
 pw.println("<h1 style='color:red;text-align:center'> Mr." +name + " u r not eligible for marriage enjoy life </h1>");
 else
 pw.println("<h1 style='color:green;text-align:center'> Mr." +name + " u r eligible for marriage , But think once </h1>");
}
else {
 if(age<18)
 pw.println("<h1 style='color:red;text-align:center'> Miss." +name + " u r not eligible for marriage, be happy </h1>");
 else
 pw.println("<h1 style='color:green;text-align:center'> Miss." +name + " u r eligible for marriage , But think Thrice </h1>");
}

//add graphical hyperlink for home navigation
pw.println("
 ");
//close stream
pw.close();
} //doPost(-,-)

@Override
public void doGet(HttpServletRequest req, HttpServletResponse res) throws ServletException, IOException {
 System.out.println("MarriageServlet.doGet(-,-)");
 doPost(req,res);
}
} //class

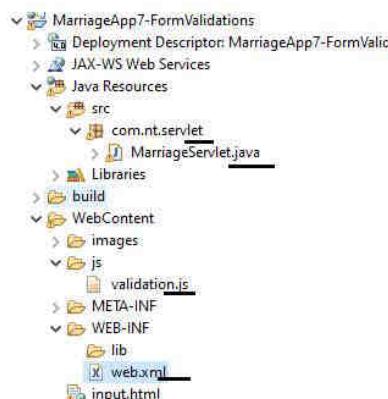
```

URL in browser address bar

(a)

<http://localhost:3030/MarriageApp-FormValidations>

(we are assuming no form validation  
errors and JS code is executed)



```

<welcome-file-list> (b)
 <welcome-file>input.html</welcome-file>
 <welcome-file>home1.html</welcome-file>
 <welcome-file>input1.jsp</welcome-file>
 <welcome-file>home1.jsp</welcome-file>
</welcome-file-list>

</web-app>

```

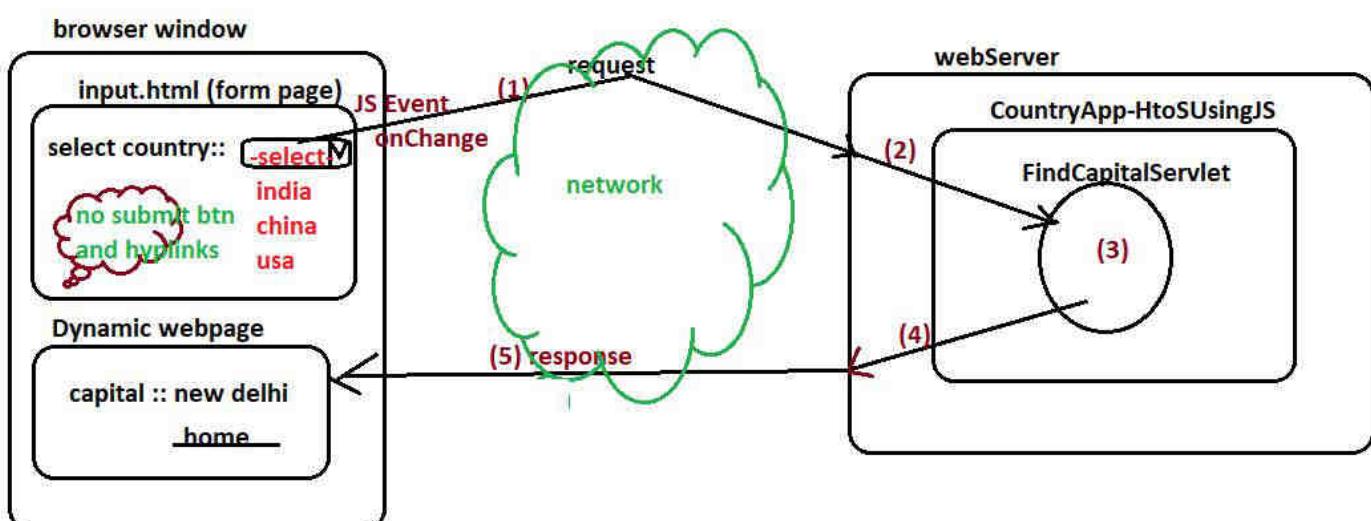
## HTML to servlet communication using Java Script

=>It is very useful to send requests to Servlet component html file not by using hyperlink or submit button .. just by using various events raised in web page or form comps like onBlur, onLoad, onChange, onFocus, onClick, onKeyDown, onKeyUp and etc...

eg:: after choosing new password .. it should give whether password is weak or strong for onBlur (losing focus) event raised on password box.

eg:: After selecting the country from the select box the request should go and should capital on country for onChange event (changing the item of select box)

(While working these uses cases of java script event based request submission we need to take support ajax to inorder to receive response and display response in the same webpage from where request is generated)



=>we can use form obj.submit() method to send request to destination web comp

formal args/params

```
public int sum(int x,int y){
 return x+y;
}
```

```
int a=10;
int b=20;
sum(a,b);
```

note:: formal params and  
actual params need not  
to match.

actual params/args

While working the following comps of the form page.. the select item visible text/data will not go to server as request param value.. The value/data kept in "value" attribute of selected item will go to server as request param value.

=>radio buttons

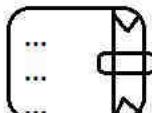
=>check boxes

=> Select box / combo Box / DropDown box



(Allows to select 1 item at a time)

=> List Box



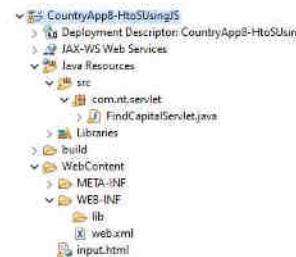
(Allows to select more than 1 item at a time  
by holding ctrl key)

#### Example App code

```
=====
input.html (c)
=====
<body bgcolor="lightblue">
 <h1 style="color:red;text-align:center"> Html to Servlet communication Using JS </h1>
 <form action="countryurl" name="frm" method="POST">
 Select country :: <select name="country" onchange="send()> (e)
 (d) select on time
 <option value="" selected>-select--</option>
 <option value="0"> India </option>
 <option value="1"> China </option>
 <option value="2"> USA </option>
 <option value="3"> Germany </option>
 <option value="4"> Japan </option>
 </select>
 </form>
</body>
<script language="JavaScript">
 function send(){ (f)
 frm.submit(); //JS method invoked form object (frm)
 } (g)
</script>
</html>
```

#### web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://xmlns.jcp.org/xml/ns/javaee"
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app_4_0.xsd" id="WebApp_ID"
version="4.0">
 <display-name>CountryApp8-HtoSUsingJS</display-name>
 <welcome-file-list>
 <welcome-file>input.html</welcome-file>
 </welcome-file-list>
 <servlet>
 <servlet-name>capital</servlet-name>
 <servlet-class>com.nt.servlet.FindCapitalServlet</servlet-class>
 </servlet>
 <servlet-mapping>
 <servlet-name>capital</servlet-name>
 <url-pattern>/countryurl</url-pattern>
 </servlet-mapping>
</web-app>
```



#### FindCapitalsServlet.java

```
package com.nt.servlet;

import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
(j)
public class FindCapitalServlet extends HttpServlet {
```

```
@Override (k)
public void doPost(HttpServletRequest req, HttpServletResponse res) throws ServletException, IOException {
 //get PrintWriter
 PrintWriter pw=res.getWriter();
 //set response content type
 res.setContentType("text/html");
 //read form data
 int countryCode=Integer.parseInt(req.getParameter("country"));
 //write b.logic /request processing logic
 String capitals[]={ "New Delhi", "Beijing", "WashingtonDC", "Berlin", "Tokyo"};
 //write response /output to response obj
 pw.println("<h1 style='color:red;text-align:center'> Capital City name is ::"+capitals[countryCode]+"</h1>");
 //add home hyperlink
 pw.println("
home");
 //close stream
 pw.close();
}
//doPost(--)

@Override
public void doGet(HttpServletRequest req, HttpServletResponse res) throws ServletException, IOException {
 doPost(req,res);
}
//doGet(--)

}//class
```

(a)  
request url :: <http://localhost:3030/CountryApp8-HtoSUsingJS/>

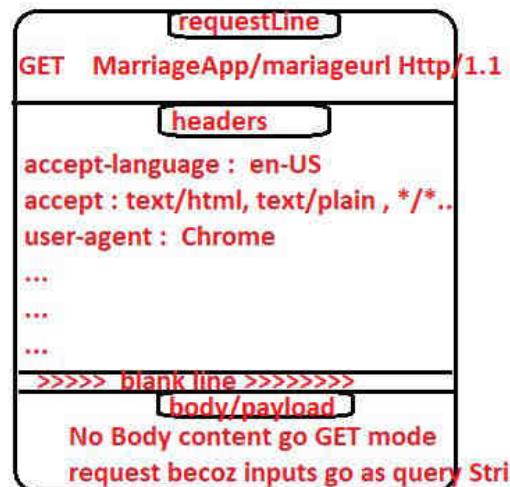
---

Q) After getting webpage on to browser.. i modified the source code of webpage using view-source option. Can u tell me wheather webcontent will be effected or not?

Ans) will not reflect in the web page.. becoz after modification.. it we say refresh. the request server .. and the server web content executes which is not modified..

<http://localhost:3030/MarriageApp/mariageurl?pname=raja&page=30&gender=M> (request url) (GET)

Http request structure

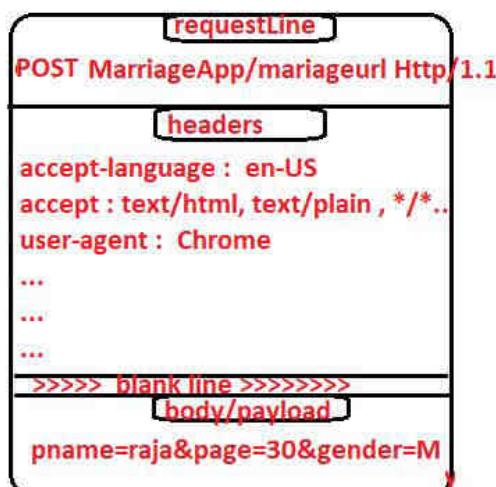


GET mode request carries enduser supplied form data/other data as query String.

After receiving request based on different actions done by end user like typing url in the browser address bar, clicking on hyperlink, submitting form raising java script event and etc.. the browser generated request structure putting some extracted info as shown in the diagrams..

<http://localhost:3030/MarriageApp/mariageurl> (request url) (POST)

Http request structure



POST mode request carries enduser supplied form data/other data as request body content /payload

=>requestLine contains basic info about the currently generated request.

=>request header names are fixed and given by protocol "http" specification and automatically carry more info about browser or client setup along with request.. To read these request header values in our servlet component we need to use request.getHeader(-) and etc..

=>HttpServletRequest object provides multiple getXxx() methods to read various details from request object that like request line info, request header info, request params info and other misc info..

```
public abstract java.lang.String getAuthType();
public abstract jakarta.servlet.http.Cookie[] getCookies();
public abstract long getDateHeader(java.lang.String);
public abstract java.lang.String getHeader(java.lang.String);
public abstract java.util.Enumeration<java.lang.String> getHeaders(java.lang.String);
public abstract java.util.Enumeration<java.lang.String> getHeaderNames();
public abstract int getIntHeader(java.lang.String);
public default jakarta.servlet.http.HttpServletMapping getHttpServletMapping();
public abstract java.lang.String getMethod();
public abstract java.lang.String getPathInfo();
public abstract java.lang.String getPathTranslated();
public default jakarta.servlet.http.PushBuilder newPushBuilder();
public abstract java.lang.String getContextPath();
public abstract java.lang.String getQueryString();
public abstract java.lang.String getRemoteUser();
public abstract boolean isUserInRole(java.lang.String);
public abstract java.security.Principal getUserPrincipal();
public abstract java.lang.String getRequestedSessionId();
public abstract java.lang.String getRequestURI();
public abstract java.lang.StringBuffer getRequestURL();
public abstract java.lang.String getServletPath();
public abstract jakarta.servlet.http.HttpSession getSession(boolean);
public abstract jakarta.servlet.http.HttpSession getSession();
public abstract java.lang.String changeSessionId();
public abstract boolean isRequestedSessionIdValid();
public abstract boolean isRequestedSessionIdFromCookie();
public abstract boolean isRequestedSessionIdFromURL();
and etc...
```

Methods of

[javax.servlet.http.HttpServletRequest\(\)](#)

### What is the difference between GET and POST mode requests?

#### GET

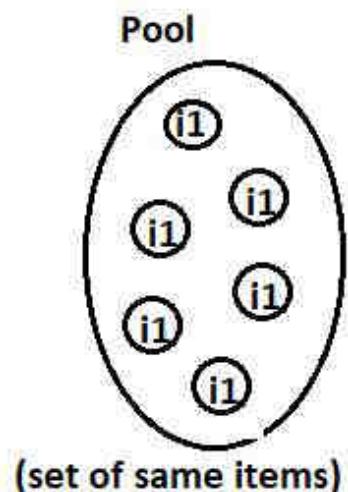
- a) This is default mode ,if no mode is specified for the request.
- b) It is designed to "GET" data from server by sending request
- c) It can query data from Server by sending limited amount of query data(input) along with the request that 2kb to 8kb (In many browsers it is 2kb)
- d) Sends query data to server along with request as queryString appended to the request url (refer diagram)
- e) GET Mode request Structure does not contain request BODY/Payload (refer diagram)
- f) Here there is data no Security/Secracy as form data appears in the browser address bar as query String
- g) Does not support file upload activity
- h) can send only text data along with request.  
In General GET
  - i) PERforms read operation on server
  - j) GET mode requests are safe to repeat. (Idempotent)  
(Using refresh or back button )
  - k) To processs GET mode requests in servlet comp use service(-,-) or doGet(-,-) method
  - l) GET Mode request is bit faster to go server
  - m) Does not support data encryption
  - n) GET Supports bookmarking of web pages (in all situations)
  - o) GET Support caching /buffering web page content
  - p) GET is recommended for hyperlinks based requests
  - q) To generate GET mode request mode requests
    - i) typing url in the browser address bar
    - ii) clicking on hyperlinks
    - iii) `<form action="....">`  
...  
`</form>`
    - iii) `<form action="..." method="GET">`  
...  
`</form>`

#### POST

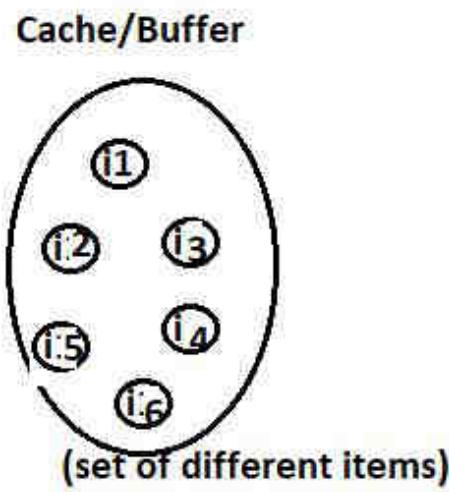
- a) This is not a default mode
  - b) It is desinged to POST data server along with request
  - c) it can POST/send unlimited amount of data to server along with request
  - d) Sends data to server along with request as request body/payload (refer diagram)
  - e) POST Mode request Structure contains request BODY/Payload (refer diagram)
  - f) The Data security/secracy is possible with POST mode request becoz data goes as request body
  - g) supports file uploading
  - h) can send both text and binary data(images,audio,vedio andetc..) along with the request.
  - i) Performs both read and write/update operation on the server. (In General)
  - j) POST mode requests are not safe to repeat. (non-Idempotent)
  - k) To process POST mode requests in servlet comp use service(-,-) or doPost(-,-) methods
  - l) POST mode request is bit slow compare to GET mode request
  - m) supports ..
  - n) POST does not support in most of situations becoz there is no query String in the url
  - o) POST does not support becoz it is not having query String in the request url
  - p) POST recommand form related submit button, java script related requests.
  - q) To generate POST mode request
- ```
<form action="...." method="POST">
...
</form>
```

note:: Certain web page will come , not just becoz of request url ,also becoz of the query String inputs.. So in bookmarking we need both url and query String..

=>To Book mark any web in google chrome press star symbol of browser address bar or Ctrl+D.



(Gives the reusability of
same items)



(Gives reusability of
different items..)

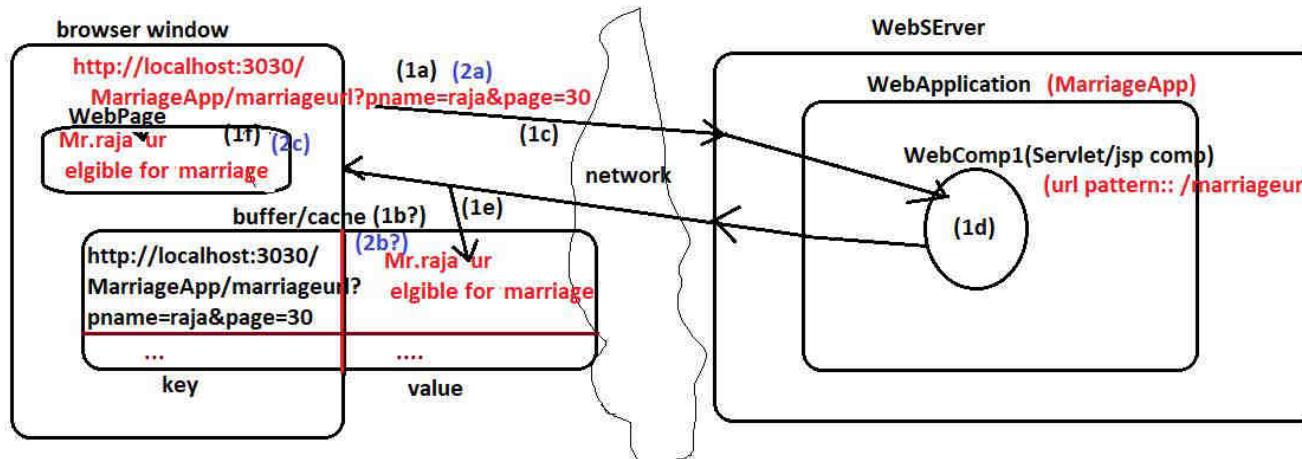
Cache/Buffer

=====

=> It is a temporary memory that holds data for temporary period..

=> In Client-Server App the cache/buffer at client side holds server supplied data and uses that data across the multiple same requests .This process reduces the network round trips between client and server ..

=> the cache/buffer at browser side holds web server/web application supplied data and uses that data across the multiple same requests .This process reduces the network round trips between browser and web applicaton(webServer) ..



(1a) to (1f) :: 1st request

(2a) to (2c) :: other than 1st request like 2nd/3rd/4th request.. and etc..

(all requests are
same requests i.e they
have same request url and
query string data)

=>Once we close the browser window.. the buffer/cache will be vanished automatically..

=>The buffer/cache can have multiple request urls with different req param as keys their result/response data as values.

Limitation of buffer/cache

=====

If the web comp of web application changing its result/response not based on inputs ..but based on time of request generation.. better disable buffer/caching at browser side for that web comp generated response/result.

eg1:: The web comp that displays Live game score details like crickinfo.com

eg2:: The web comp that displays Stock market share details like sharekhan.com

=>To instruct browser not to keep certain web comp results/response in the buffer/cache use

"cache-control"(http 1.1) or pragma(http1.0) response header .

code in servlet comp::

```
res.setHeader("cache-control","no-cache"); (or)  
res.setHeader("pragma","no-cache")
```

note:: GET mode request url contains queryString and the keys in browser cache/buffer expects request url.. So GET more request support caching/buffering where as POST request does not support becoz the POST mode request url does not contain query String.

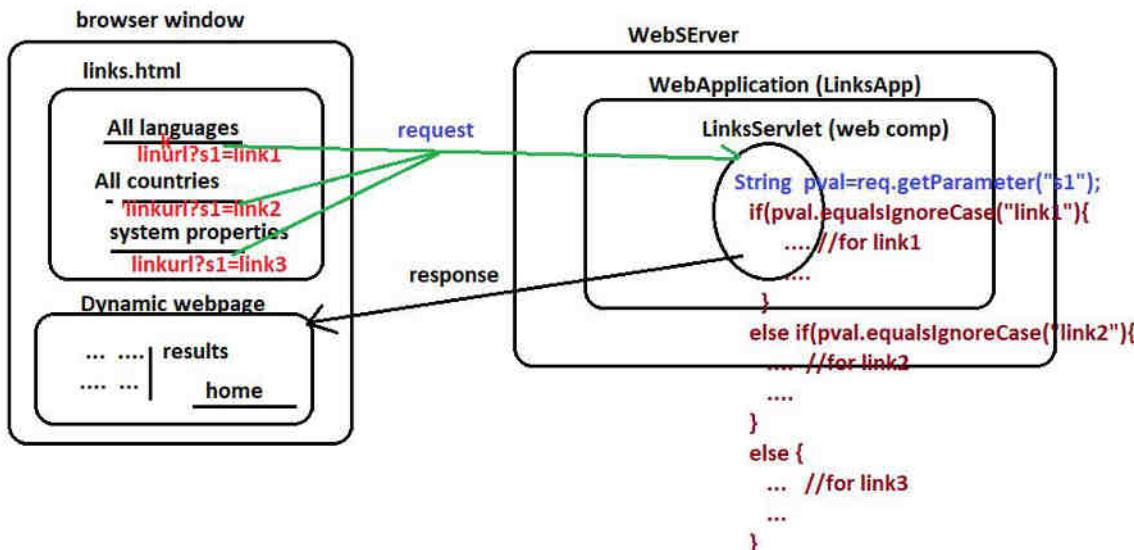
`<a href="wishurl" go ` (Does not carry any thing along with request)

`<a href="wishurl?sno=101&sname=raja" go ` (Carries static/fixed additional data along with request
becoz we appended query string to the request url)

In Servlet comp

```
int no=Integer.parseInt(req.getParameter("sno")); //gives 101
String name=req.getParameter("sname"); //gives raja
```

How to differentiate logics in Servlet comp , if multiple hyperlinks are giving request to same servlet comp?



=> we need append query String to the "href" url of of hyperlinks having same request param name and with different values . we can take this additional request param value as the criteria value in servlet comp to differentiate the logic.

input.html

```
<a href="linkurl?s1=link1"> all languages </a>
<a href="linkurl?s1=link2"> all countries </a>
<a href="linkurl?s1=link3"> system properties </a>
```

=>Locale means language+country

eg:: en-US (english as it speaks in USA)
fr-FR
de-DE (German as it speaks in Germany)
fr-CA
hi-IN (Hindi as it speaks in India)
and etc..

Each java.util.Locale class obj represents one Locale.. and This class gives static method called `Locale.getAvailableLocales()` which returns `Locale` class object []

```
Locale locales[] = Locale.getAvailableLocales();
```



To get only countries

```
=====
for(Locale l: locales){
    S.o.p(l.getDisplayCountry());
}
```

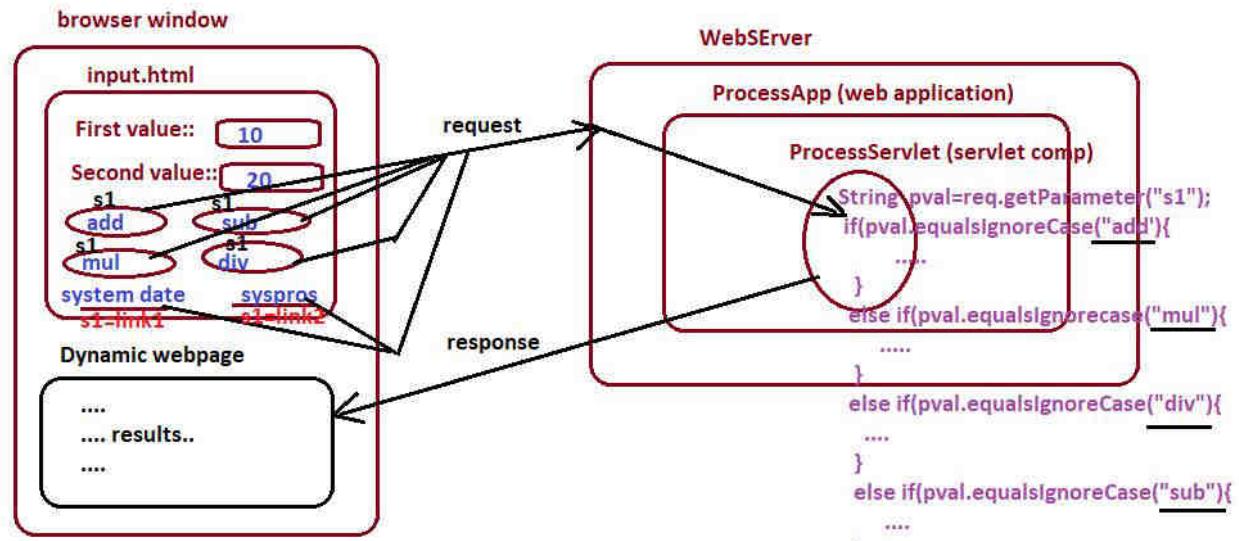
To get only languages

```
=====
for(Locale l: locales){
    S.o.p(l.getDisplayLanguage());
}
```

To get System properties

```
=====
Properties props=System.getProperties();
```

How to differentiate logics in servlet comp when multiple hyperlinks and submit buttons are giving request same servlet comp



give same name and different captions to submit buttons also give additional params to hyperlinks having submit button name as additional req param name(s1). we can use this additional req param value (s1) as the criteria value to differentiate the logics..
(refer the diagram)

```
<form action="..." method="...">
...
<input type="submit" name="s1" value="send">
```

=>Sends the submit caption to server as req param value
s1=send

```
<form action="...." method="....">
...
<input type="submit" value="send">
```

=>It does not send the caption of submit to server as the request param value.

PeocessApp10-WorkingMultipleSubmitButtons-hyperlinks (EDWP)

```
|---->java resources
|--->src
|---->com.nt.servlet
|---->ProcessServlet.java
|---->webcontent
|---->input.html
|---->WEB-INF
|--->web.xml
```

note:: the hyperlinks added to form page does not carry form comps data when they generate request..

Different types html form comps

standard form comps

textbox ()
password (()
textarea (<textarea> ... </textarea>)
radio buttons (<input type="radio" ...)
single checkbox (<input type="checkbox" ..)
grouped checkboxes (<input type="checkbox" ..)
select box /drop down box <select><option> ... </option> </select>
List box <select multiple><option> ... </option> </select>
file upload <input type="file" ...>
submit button <input type="submit" ..>
reset button <input type="reset" ...>
standard button <input type="button" ..>
hidden <input type="hidden" ...>

Html5 form comps

=====

date
datetime-local
time
datetime
color
search
tel
number
url
range
month
week
image
email
and etc..

<input type=".....">

note:: if form comp/rquest param is having only one value then

use req.getParameter(-) to read the form comp value (for most of form comps this is enough)

return String obj

note:: if form comp/rquest param is having multiple values then

use req.getParameterValues(-) method to read the form comp values (for List Box, grouped checkboxes)

returns String[]

In form page::

Hobbies: <input type="checkbox" name="hb" value="reading">Book Reading
<input type="checkbox" name="hb" value="watching">Watching movies
<input type="checkbox" name="hb" value="listening">Listening to music
<input type="checkbox" name="hb" value="roaming">Travelling

In servlet comp::

String hobbies[] = req.getParameterValues("hb");



if all checkboxes selected the hb[] contains "reading", "watching", "listening", "roaming"
as the element values..

Working different types of form comps

=>Html basic + htm5 togather is offering 30+ form comps..

=>Html 5 is offering new comps.. also giving some comps as as text boxes with builtin- form validations.

=>Except ListBox,checkbox comps the remaining the comps will send either empty string or default value as the request parameter value even though they are not selected or not filledup with values , Where as for list box and check comps the values will give to server as request parameter values only when they are selected otherwise no request parameter will be formed for them. So for these comps we must non-select state explicitly.. (i.e we must assign on or other value at servlet comp if they are not selected)

```
String ms=req.getParameter("ms");
if(ms==null)           //for checkbox
    ms="single";
```

```
String hobbies[] =req.getParameterValues("hb");
if(hobbies==null)
    hobbies=new String[] {"no hobbies are selected"}; //for group for checkboxes
```

```
String languages[] =req.getParameterValues("languages");
if(languages==null)
    languages=new String[] {"no languages are selected"}; List Box
```

Servlet Life cycle

=>Life cycle means all activities from start to end or birth to death.

=>Human Life cycle contains

- >Birth
- >start of schooling
- >end of schooling
- > getting job
- > marriage
- >retirement
- > death

as the life cycle activities or life cycle events

=> Servlet Contains manages servlet comp life cycle having the following life cycle events/activities

->Servlet Instantiation Event (1)

(Raises when Servletcontainer creates our servlet class obj)

-> request processing event (2)

(raises for every request, when the servlet comp is ready to process
the request)

-> Destruction Event (3)

(raises when the servlet comp is about to destroy by SErvletcontainer)

=>Keeping track of Servlet comp activies /events form its birth to death (object creation to
Object destruction) is called Servlet life cycle.

=>The methods that will be executed servlet container automatically for life cycle events
are called Servlet life cycle methods or container callback methods.

1) init(*ServletConfig cg*) --> Life cycle method Instantitation Event

signature :: public void init(*ServletConfig cg*)throws ServletException

=>It is one time executing block of servlet comp life cycle.

=>Programmer overrides this method in servlet comp class

having initilaization logic like creating jdbc connection with Db s/w.

=> init() method also there in servlet api.. but it is not life cycle method , It is given as
convience method for programmer..



note:: *ServletConfig obj* is 1 per *Servlet class obj*
and will be useful to pass addtional info to servlet comp
and to gather more about servlet comp.

2) service(HttpServletRequest req, HttpServletResponse res) -> Life cycle method of request processing event

signature :: public void service(HttpServletRequest req, HttpServletResponse res) throws ServletException, IOException
(1st service(-,-) method)

=> Once Servlet Comp instantiation and initialization is completed and when it is ready to process then this request processing event will be raised and service(-,-) life cycle method will be called.

note:: This repeatedly executing block of servlet comp life cycle.. It executes for every request ,So we place request processing logic(b.logic) in this method.

note:: 2nd service(-,-) method and 7doXxx(-,-) are not life cycle methods.. they are given as convience methods for programmers to work protocol http features.

note:: 2nd service(-,-) and 7doXxx(-,-) are called through 1st service(-,-) method directly or indirectly by Servlet Container..

3) destroy() -- Destruction Event

=> signature :: public void destroy()

=> When ServletContainer is about to destroy our servlet comp class object ,then destruction event will be raised and this life cycle method destroy() will execute.

=> This is one time executing block of servlet comp life cycle.

=> By overriding this method programmer keeps uninitialization logics like closing jdbc connection and releasing non-java resources from Servlet comp.

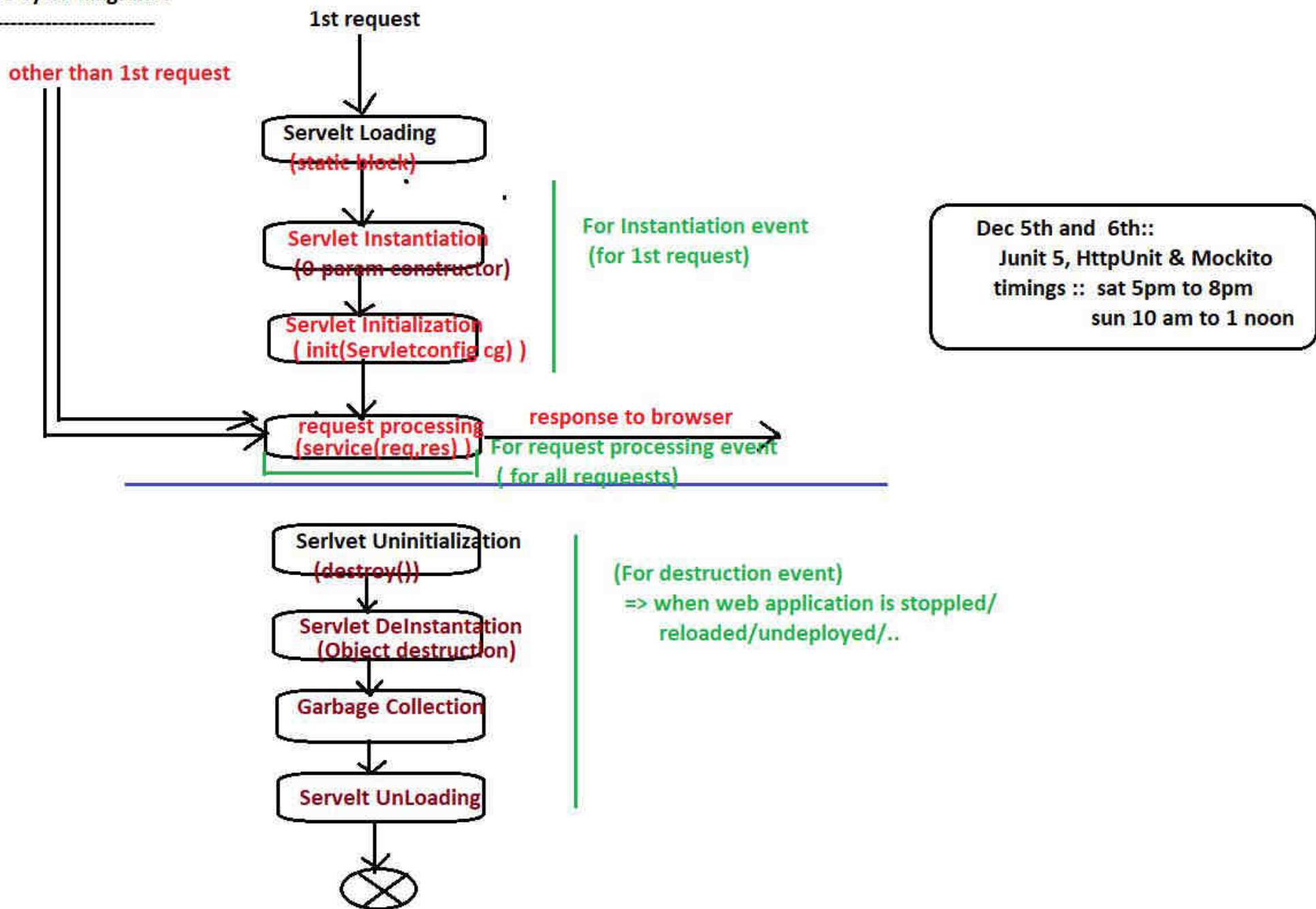
connection to DB ,file and etc..



=> Servlet Container is giving servlet life cycle methods to programmers for two reasons

- To override servlet life cycle methods to place programmer choice logics in servlet comp.. So programmer logics will execute even though the entire life cycle servlet comp is taken care by servlet container
- To get access ServletContainer created objects in our servlet comp as the parameters of servlet life cycle methods. like init(-) method Servletconfig obj and service(-,-) method gives HttpServletRequest, HttpServletResponse objs

Servlet life cycle diagram2



=>While working with Eclipse Tomcat (Tomcat integrated with Eclipse IDE) we can not see tomcat home page or Manager page .. i.e we can not perform stop , reload and undeploy activities on the web applications. But we can deploy web application developed in Eclipse IDE in out side Tomcat Server (external server) by using export option.. (For this external Tomcat java version and Eclipse web application java version must be compatible)
(the one that is chosen during the installation of Tomcat) (Check by using Project Facets
java version from web application properties)

Right On Eclipse web applicationProject --> export --> war file -->
Destination :: E:/Tomcat 9.x/webapps folder -->select all checkboxes (optional) -->Finish..

note:: we can stop web application or web server but servlet comp directly.. but by stopping web application or webServer we are indirectly destroying the servlet class object.

note:: ServletContainer does not use init(-) method logic to create our servlet class object.. It is having its own internal logic to create our servlet class object. So programmer uses init(-) method for writing his choice intialization logics

note:: ServletContainer does not use destroy() method logics to destroy our servlet class object.. It is having its own internal logic to destroy our servlet class object. So programmer uses destroy() method for writing his choice uninitialization logics.

=>SERvletContainer/WebContainer does not use JVM's Garbage Collector to destroy our servlet class objs ,it will use its own Garbage collector for the same becoz Servletcontainer to destroy our servlet class objects even though it is in utilization forcefully when required.. but JVM Gargbage Collector can destroy the objs only when they are completely free/idle having no reference.

Limitations of "new" operator

=====

Test t=new Test();

note:: new Operator creates the object at runtime , but expects the presence of the java class from compile time onwards.. i.e we can not use new operator to create the object of java class whose class name comes to Application dynamically at runtime. For this we need to use newInstance() method of java.lang.Class .

Test t=(Test)Class.forName("Test").newInstance();
Loads the java class at runtime creates the object of loaded class at runtime

ServletContainer does not use "new" operator create our servlet class object becoz it servlet class name from web.xml file will be collected dynamically at runtime and object should created at runtime itself. So it uses newInstance() method for object creation.

LcTestServlet lc=(LcTestSERvlet)Class.forName("com.nt.servlet.LcTestServlet").newInstance();
Loads our servlet class at runtime by collecting from the InMemory MetaData of web.xml ->creates the object of the loaded servlet class at runtime.

what happens if we call destroy() method from service(-,-) method?

what happens if we call init() method from service(-,-) method?

what happens if we call destroy() method from init(-) method?

What happens if we call destroy() method from service(-,-) method?

Ans) Servlet class obj will not destroyed.. but the logics of destroy() method executes along with service(-,-) method logics

note:: When the life cycle event is raised , the life cycle method will be called automatically.. since we call life cycle method manually the life cycle event will not be raised .

What happens if the init(-) is called from servcie(-,-) method?

Ans) the logics init(-) method also executes from service(-,-) method.. but new object for servlet comp class will not be created.

what happens if destroy() is called from init(-) method?

Ans) the logics of destory() method will execute along with init(-) method

What is pre-instantiation or eager/early instantiation of servlet comp?

Ans) Generally first request given servlet comp participates in servlet loading, Servlet Instantiation and Servlet Intializatioon before performing request processing.. This is called Lazy Instantiation of Servlet comp becoz ServletContainer is creating our servlet class object lately only when we give first request. The other than first requests given to servlet comp directly participates in request processing. due to this the response time(time taken to get output) of first request is bit high compare to other than 1st requests.

To minimize first request's response time and to equalize with other than 1st request , we need to make Servlet container to create our servlet class object either during server startup or during the deployment web application by enabling <load-on-startup> on the servlet comp .. this called eager/early/pre -instantiation of servlet comp.. (Makeing servlet container to create our servlet class object either during server startup or during the deployment web application irrespective of wheather it recieves the requests or not)

```

<servlet>
    <servlet-name>lcs</servlet-name>
    <servlet-class>com.nt.servlet.LCTestServlet</servlet-class>
    <load-on-startup>1</load-on-startup>
</servlet>          <load-on-startup> priority value (it can any positive number or zero)
<servlet-mapping>
    <servlet-name>lcs</servlet-name>
    <url-pattern>/lcurl</url-pattern>
</servlet-mapping>

```

note: It is not mandatory that we must enable <l-o-s> on every servlet comp.

note: it is recommended to enable <l-o-s> on such a servlet that must be requested guaranteeably after the deployment of the web application.

eg: servlet comp giving home page, servlet comp main menu page,
Servlet comp that traps all requests (FrontController)

when Servlet Container creates our servlet class obj?

if the <load-on-startup> is not enabled

- a) when servlet comp gets first request
- b) when servlet comp gets first request after restarting/reloading/redeploying the web application.
- c) when servlet comp gets the first request after restarting server.

if <load-on-startup> is enabled

- a) during the deployment of web application or during the sever startup
(For Hot Deployment) (For cold deployment)

When SErvletcontainer destroys our servlet class obj?

Ans)

- a) when the web application is stopped/reloaded/undeployed
- b) when the webServer is stopped/crashed
- c) if our servlet class obj is continuously idle for long time.

note:: Servletcontainer does not give any special priority towards the destruction of our servlet class object .. even though <load-on-startup> is enabled on it.

What is the difference b/w COLD deployment and HOT Deployment?

Ans) if we deploy the web application when server is in stopped mode..
then it is called COLD deployment.

if we deploy the web application when server is in active/running mode..
then it is called HOT deployment.

<load-on-startup> priority value

=>if multiple servlet comps of a web application are enabled with <load-on-startup> then their order of pre-instantiation will be decided based on the <load-on-startup> priority value.

=>High value indicates low priority.

=>Low value indicates high priority.

=> -ve value ignores the enabled <load-on-startup>

Servlet1,Servlet2,Servlet3,Servlet4 are comps of WebApplication1

	<l-o-s> value	<l-o-s> value	<l-o-s> value	
Servlet1	10 (III)	5 (II)	10	Server decides the order..
Servlet2	20 (IV)	16 (III)	10	(Generally the random order will be chosen)
Servlet3	5 (I)	0 (I)	10	
Servlet4	6 (II)	-5 (Ignores <l-o-s>)	10	

note:: taking <l-o-s> with out priority value throw an exception.

LcTestServlet -->1 (i)

WordServlet-->5 (ii)

ExcelServlet--> 20 (vi)

PlainServlet-->8 (iii)

XmlServlet---> 15 (v)

HtmlServlet --> 10 (iv)

Can we place main(-) method in our servlet comp class?

Ans) Yes , we can place .. but that will not executed becoz it is not the life cycle method of servlet comp class.

Q) How Servlet comp is executing with out main(-) method?

Ans¹) Servlet container is executing servlt comp through life cycle methods.. Since main(-) method is not life cycle method of servlet comp .. it is not required in servlet comp class.

Ans²) only standalone Apps begin their execution by JVM using main(-) method... Servlet comp is not a standalone App .. It is web comp whose execution is taken care by SErvletContainer througuh life cycle methods, So main(-) method is not required in Servlet comp as main(-) is not life cycle method of servlet comp.

Ans³) JVM starts the webServer by calling main(-) method.. This main(-) creates SErvlet Container --> ServletContainer manages the given SErvlet comp life cycle through life cycle methods . So main(-) is not required in SErvletcontainer and also in Servlet comps.. i.e main(-) will be there only in the class that starts server.. In Tomcat server that class name is org.apache.catalina.startup.BootStrap.

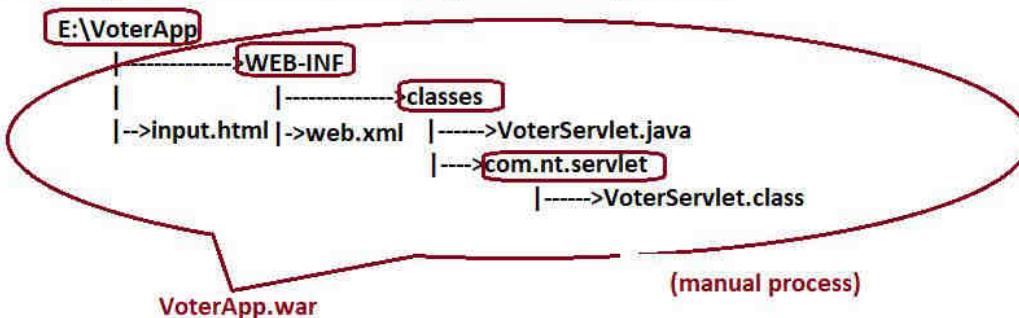
Clicking on tomcat9.exe file --> goes to <Tomcat_home>\bin\bootstrap.jar--> from META-INF/manifest.mf file it collects org.apache.catalina.startup.BootStrap name as the main class name and call main(-) method on that class--> The main(-) creates ServletContainer (Catalina) ----> Servlet container manages the given servlet comps life cycle (by creating and managing the objects)

To use online decompiler --> launch decompiler.com --> select u r jar file (bootstrap.jar) ---> Decompile -->get the source code

=>We can deploy web applicaiton in any server either in the form of directory or in the form war file.

war file --> web application archive (jar file for web application having the extension of .war)

=>We need to prepare war file on the Deployment directory structure



E:\VoterApp> jar cvf VoterApp.war .

c-->create archive

v-->verbose mode

f-->specify archive file name (jar file name)

=> To prepare war file from Eclipse Dynamic web project use "export" war file option

Right click on Eclipse Dynamic web Project ---> export -->war file ---

Destination:: <Select the location where to get the war file> ---> select all check boxes---> finish.

There are 3 approaches for the deployment of web application in web server

a) **Hard Deployment**

(copy directory or war file to Server Deployment folder/location like <Tomcat_home>\webapps folder)

b) **Console Deployment**

Use Admin console screen of server (like Tomcat manager) to deploy the web application either in the form of directory or in the form of war file.

c) **Tool Based Deployment (Best)**

=>Use IDE like Eclipse or maven tool or gradle tool for deployment.

Performing Console deployment of web application in Tomcat server

step1) prepare war file representing the entire web application

refere the above VoterApp.war

step2) start tomcat server manually using <Tomcat_home>\bin\tomcat9.exe file

step3) Open the Tomcat Manager/Admin page and deploy the web application..

Tomcat home page (<http://localhost:3030>) ---> Manager App

-->submit username :: admin password:admin ----> Go to war file to deploy section ---> choose the war file --->

VoterApp.war (This step keeps the given war file in <Tomcat_home>\webapps folder and also extracts that file)

note:: when war file is deployed , the war file name becomes the context path of the web application.

step4) Test the web application..

URL :: <http://localhost:3030/VoterApp/input.html>

To undeploy the web application through admin console of Tomcat

open Tomcat manager /admin screen --> Go to ur web application (VoterApp) --> undeploy.

note:: For Hard deployment of web application in Tomcat server copy war file(like VoterApp.war) or directory(VoterApp) to <Tomcat_home>\webapps folder

What is the difference b/w COLD Deployment and HOT deployment?

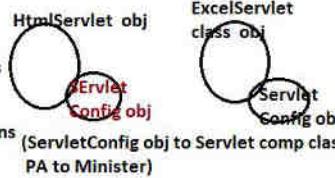
=>if we deploy the web application when the server is running mode.. then it is called HOT Deployment..

=> If we deploy the web application when the server is in stopped mode then it is called COLD Deployment..

note:: HardDeployment /ToolBased Deployment can be HOT or COLD Deployment where as Console Deployment must always be Hot Deployment.

ServletConfig object

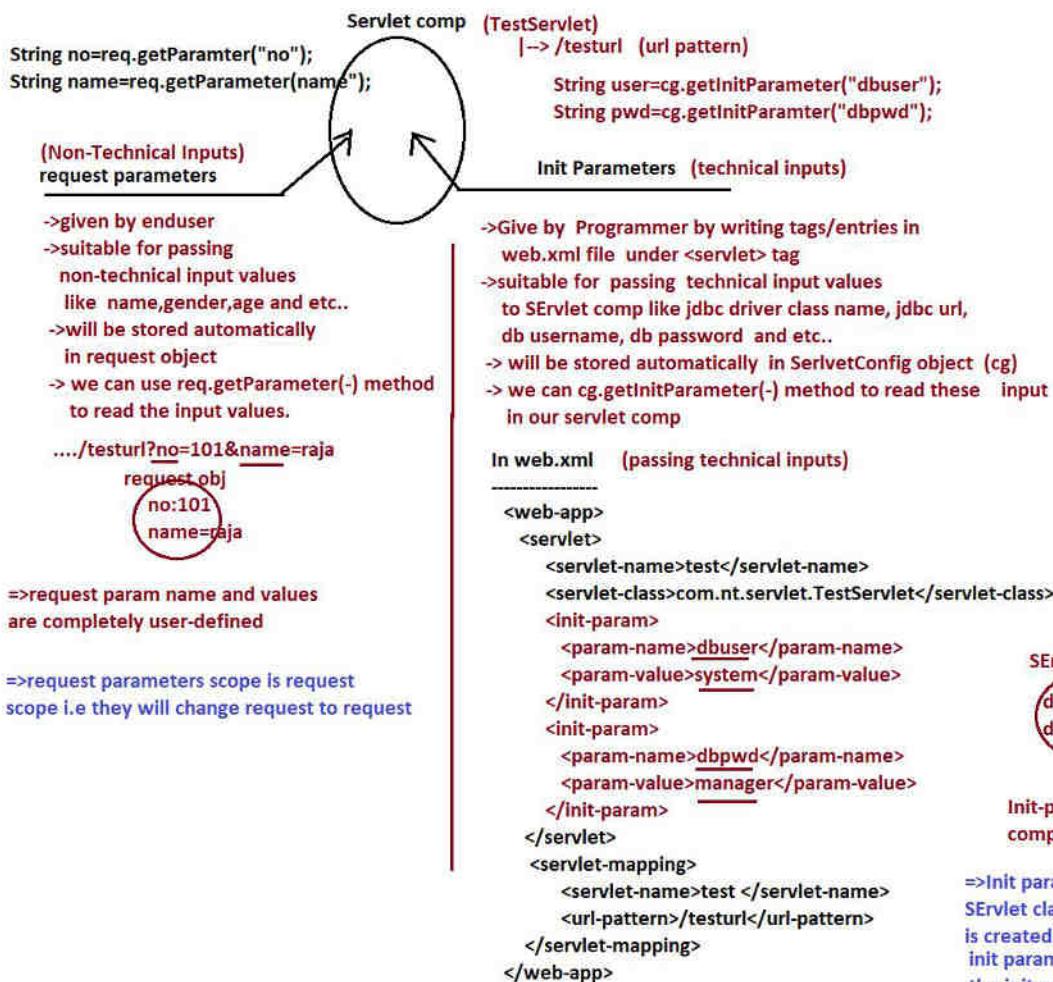
- =>It is created on 1 servlet class object basis
- =>ServletContainer creates this object during the deployment of web application or during Server startup based servlet cfg done in web.xml file.
- =>ServletConfig will be assigned to respective Servlet class object by calling init(ServletConfig) method , the moment Servlet class object is created by ServletContainer.
- =>It is called right hand object to our Servlet class object becoz Servlet class can use that object to pass additional info Servlet class from out side and to gather more about Servlet comp class.
- =>ServletConfig object means, it is object of java class that implements javax.servlet.ServletConfig().



(ServletConfig obj to Servlet comp class obj is like PA to Minister)

=>Hard Coding inputs in Servlet comp is bad practice.. So we should go for soft coding ..

- => We can pass input values to Servlet comp (softcoding) in two ways
 - a) As request parameters (form data)
 - b) As ServletInit Parameters (from web.xml)



Deployment --> loading of web.xml --> InMemory Meta data of web.xml --> <some operations..>
-> checks servlet configurations --> found n servlet configurations --> creates n ServletConfig objs on 1 per
servlet comp basis --> each ServletConfig obj holds its respective Servlet comp's init param values --> pre-instantiation of
servlet comps on which <load-on-startup> is enabled --> initialization of same servlet comps by assigning
their servlet config objects to servlet comps takes place by calling init(Servletconfig) methods.

ServletContainer creates which Object first?

- a) ServletConfig obj
- b) Servlet class object

Ans) (a) --> With respect to above discussions the answer is (a)

=> we can not write messages to browser from init() and destroy() methods becoz to do that we need response object based PrintWriter and
response obj is not visible in the init() and destroy() methods.

=> we can write message to browser from service(-,-) and doXxx(-,-) methods becoz response object PrintWriter is visible and
accessible in those methods..

=> req,res objs not visible and accessible in init() and destroy() methods .. they are just specific to service(-,-) and doXxx(-,-) methods..

Example code on Servlet init parameters

in web.xml

```
<servlet>
    <servlet-name>lctest</servlet-name>
    <servlet-class>com.nt.servlet.LCTestServlet</servlet-class>
    <init-param>
        <param-name>dbuser</param-name>
        <param-value>system</param-value>
    </init-param>
    <init-param>
        <param-name>dbpwd</param-name>
        <param-value>manager</param-value>
    </init-param>
    <load-on-startup>10</load-on-startup>
</servlet>
<servlet-mapping>
    <servlet-name>lctest</servlet-name>
    <url-pattern>/lctest</url-pattern>
</servlet-mapping>
```

In the init() of LCTestServlet

```
public void init(ServletConfig cg) {
    System.out.println("LCTestServlet.init()");
    System.out.println("db user init param value:"+cg.getInitParameter("dbuser"));
    System.out.println("db pwd init param value:"+cg.getInitParameter("dbpwd"));
}
```

note: when we write JDBC code Servlet comp .. instead of hardcoding jdbc properties in
in Servlet comp.. it is recommended to get them from web.xml file as init parameter values
through ServletConfig object..

we
Why we are not preferring properties file to collect the jdbc properties.. why preferring to gather from web.xml file as init
param values?

Ans) Instead of taking separate or additional properties file .. it is recommended to work with the already available web.xml to pass
such technical info (any how web.xml is part of web application development and we can not avoid it)

Is our servlet comp class singleton java class?

Ans) The Java class that allows us to create only one object in all situations is called singleton Java class ...
Though class is allowing to create more objects.. if we are creating one only for that class then that
class not called as singleton java class.

=> ServletContainer creates only 1 object for our servlet comp class .. though class allows to create multiple objs.
So we can say servlet comp class not is not singleton java class.

Two init methods of servlet life cycle

=>Servlet api is offering two init() methods we can use both methods for placing Programmer/application specific initialization logic like creating jdbc con object and etc....

1st init() method or init(-) with parameter method or init(ServletConfig cg) method

Signature: `public void init(ServletConfig cg) throws ServletException`

=> This servlet life cycle method for instantiation event.

=> This method is originally declared in `javax.servlet.Servlet()` and implemented in `javax.servlet.GenericServlet(AC)` and inherited to `javax.servlet.http.HttpServlet(AC)`

=>Servletcontainer hanovers the Servletconfig object to its respective Servlet class obj by calling this `init(ServletConfig)` method

2nd init() method or init() with no parameter method

signature :: `public void init() throws ServletException`

=>This is not a life cycle method of Servlet comp .

=>This is given as **convience method** to programmer to place initialization logic.

=> Instead of `init(ServletConfig)` method to place initialization logics.. It is recommended to place logics in `init()` method by taking as convinence method.

=>This method is direct concrete method with no body (null method) in `javax.servlet.GenericServlet(AC)`

note:: `HttpServlet` is not having any `init()` method.. but as the sub class `GenericServlet` class it inherits and uses `init()` method of `GenericServlet (AC)`

GenericServlet.java (abstract class)

```
=====
public abstract class GenericServlet implements javax.servlet.Servlet{

    private ServletConfig config;
    //1st init() method
    public void init(ServletConfig cg) throws ServletException{
        this.config=cg; // ServletConfig obj initialization
        init();
    }
    //2nd init() method
    public void init() throws ServletException{ | Empty method definition or
        } | null method definition

    public ServletConfig getServletConfig(){
        return config;
    } | our servlet comp classes can call this method
        | to get Access to the above initialized Servletconfig obj

    public abstract void service(ServletRequest req,ServletResponse res) throws SE,IOE;
    ...
    ... //other methods
    ...
}
```

There are 3 approaches to place init() methods in our servlet comp:::

**Approach1:: override init() method (no-param method) and place
application/programmer specific intialization logics. (Best)**

**Approach2:: override init(Servletconfig) method (1-param method) and
place application/programmer specific intialization logics.**

**Approach3:: override init(Servletconfig) method (1-param method) and
place application/programmer specific intialization logics and call super.init(cg) method**

```
public class Test{
    public void m1(){ (b)
        m2(); (c)
    }
    public void m2(){
        ..
    }
}
```

public class Demo extends Test{

```
    public void m2(){ (d)
        ..
    }
}
```

**Demo d=new Demo();
d.m1(); (a)**

**Approach1:: override init() method (no-param method) and place
application/programmer specific intialization logics. (Best)**

```

GenericServlet.java (abstract class)
=====
public abstract class GenericServlet implements javax.servlet.Servlet{

    private ServletConfig config;
        //1st init() method      (d)
    public void init(ServletConfig cg) throws ServletException{
        this.config=cg; // ServletConfig obj initialization
        init(); (e)
    }
        //2nd init() method
    public void init() throws ServletException{           | Empty method definitation or
        }                                              | null method definitation

    public ServletConfig getServletConfig(){           | our servlet comp classes can call this method
        return config;                                | to get Access to the above initialized SERvletconfig obj
    }

    public abstract void service(ServletRequest req,ServletResponse res) throws SE,IOE;
    ...
    ... //other methods
    ...
}

our Servlet comp class
=====
(a)1st request (no <l-o-s>) (b) ServletLoading and Instantiation (c)Instantiation event and calling
public class LCTestServlet extends GenericServlet/ HttpServlet           init(ServletConfig cg) method
                                                                on our Servlet class obj)

    public void init(){           | our intialization logic like creating jdbc con object
        //our intialization logic like creating jdbc con object
        ...
        (f)
        ...
        ServletConfig cg=getServletConfig();
    }

    public void service(ServletRequest req,ServletResponse res) throws SE,IOE{
        ...
        (g)
        ...
        //request processing logic
        ...
        ServletConfig cg=getServletConfig();
    }
}

```

note:: In this approach we need not intialize Servletconfig object manually (taking SERvletconfig object given by Servletcontainer assigning to one instance variable) that job will be taken care by super class init(ServletConfig) method (GenericServlet class init(ServletConfig) method).. To access Super class intialized ServletConfig object in our servlet comp we need to call getServletConfig() in various method.

=> To call super class public methods in sub class and to call same class methods there is no need of object/reference variable.. we can call those methods directly..

note:: init(ServletConfig) method is life cycle method for instantiation event .. But recommended method to place init() (init no param method) becoz it make programmer not to worry about initializing ServletConfig obj manually (GenericServlet init(Servletconfig) method takes care it), So we need to write only our initialization logics in the init() method (no param method) ..So it's called convenience method given to programmers

<urname><uniqueid> : pavan-NAJ012

What happens if do not place any init() method in our servlet comp class?

Ans) The super class (GenericServlet) init(ServletConfig) and init() method methods execute (as Init(ServleConfig) internally calls init() method)

What happens if place both init() and init(ServletConfig) in our servlet comp class?

Ans) init(ServletConfig) method executes.. becoz it is life cycle method of Servlet comp and init() (no param) will not execute..

Approach2:: override init(Servletconfig) method (1-param method) and place application/programmer specific initialization logics.

(Bad becoz Programmer may forget the explicit initialization of ServletConfig object)

GenericServlet.java (abstract class)

```
=====
public abstract class GenericServlet implements javax.servlet.Servlet{

    private ServletConfig config;
    //1st init() method
    public void init(ServletConfig cg) throws ServletException{
        this.config=cg; // ServletConfig obj initialization
        init();
    }
    //2nd init() method
    public void init() throws ServletException{
    }

    public ServletConfig getServletConfig(){
        return config;
    }

    public abstract void service(ServletRequest req,ServletResponse res) throws SE,IOE;
    ...
    ... //other methods
    ...
}
```

LCTestServlet.java (Our Servlet class)

(a) 1st request (no <l-o-s>) (b) Sevlet Loading and Instantiation (c) Instantiation Event –calls init(ServletConfig method).

```
public class LCTestServlet extends HttpServlet{
    private ServletConfig cg;
    public void init(ServletConfig cg){
        this.cg=cg; //ServletConfig object Initialization
        .... //our initialization logic like creating jdbc con object
        .... use cg here..
        ...
    }

    public void service(ServletRequest req,ServletResponse res) throws SE,IOE{
        .... (e)
        .... //request processing logic
        .... //use cg here
    }
}
```

note:: We need to intialize ServletConfig object explicitly in the overrider init(ServletConfig) method of SErvletComp class inorder use to that ServletConfig object in other methods like service(-,-) method. If we forget to do that work then accessing servletConfig object in service(-,-),destroy() methods becomes not possible.. In this calling getServletConfig() of GenericServlet class returns null becoz we have not given any chance to execute GenericSevlet's init(-) to ServletConfig object there.

Approach3:: override init(Servletconfig) method (1-param method) and place application/programmer specific intialization logics and call super.init(cg) method

(Bad Approach becoz programmer may forget to call super.init(cg) method)

GenericServlet.java (abstract class)

```
=====
public abstract class GenericServlet implements javax.servlet.Servlet{

    private ServletConfig config;
    //1st init() method      (f)
    public void init(ServletConfig cg) throws ServletException{
        this.config=cg; // ServletConfig obj initialization
        init(); (g)
    }
    //2nd init() method
    public void init() throws ServletException{ (h)           Empty method definiton or
        }                                     null method definiton

    public ServletConfig getServletConfig(){           our servlet comp classes can call this method
        return config;                                to get Access to the above initialized SERvletconfig obj
    }

    public abstract void service(ServletRequest req,ServletResponse res) throws SE,IOE;

    ...
    ... //other methods
    ...
}
```

LCTestServlet.java (Our Servlet class)

(a) 1st request (no <l-o-s>) (b) Sevlet Loading and Instantiation (c) Instantiation Event --calls
public class LCTestServlet extends HttpServlet{ init(ServletConfig method).
 (d)
 public void init(ServletConfig cg) throws ServletException{
 super.init(cg); (e) =>use cg

 //our initialization logics
 (i) like creating con obj
 }

 public void service(ServletRequest req, ServletResponse res) throws SE,IOE{
 (j)
 ServletConfig cg=getServletConfig();
 }
}

=>HEre we need not to initialize ServletConfig object explicitly in our servelt comp „, becoz super class
init(ServletConfig) is getting chance to execute becoz super.init(cg) placed in our servlet comp..So
we can call getServletConfig() in all the methods of Servlet comp to get access to ServletConfig object.

=> But programmer may forget to call super.init(cg) method our init(Servletconfig) method of Servlet comp.. This leads
problems.. So it is not a good approach

Understanding two service(-,-) methods and 7 doXxx(-,-) method Servlet api

1st service(-,-) method /public service(-,-) method

signature :: public void service(ServletRequest req,ServletResponse res) throws ServletException,IOException

=> It is life cycle method for request processing event.

=> This is originally declared in javax.servlet.Servlet(I),

maintained as abstract method in javax.servlet.GenericServlet(AC) and implemented in
javax.servlet.http.HttpServlet having logic to convert ServletRequest,ServletResponse objs
to HttpServletRequest,HttpServletResponse objs and calling 2nd service(-,-) or protected service(-,-)
method.

note:: It is not a recommended method to place request processing logic

2nd service(-,-) method /protected service(-,-) method

signature protected void service(HttpServletRequest req,HttpServletResponse res) throws ServletException,IOException

=> It is not a life cycle method for request processing event. It is given convenience method for programmers
to work with request processing logic

=> It is direct concrete method of javax.servlet.http.HttpServlet(AC). This method contains logic to get
current request request method/mode and to call one of the 7 doXxx(-,-) methods..

note:: This is not good method to place our request processing logic in our servlet comp.

7 doXxx(-,-) methods (doGet(-,-),doPost(-,-),doDelete(-,-),doHead(-,-),doOptions(-,-),doTrace(-,-),doPut(-,-))

Signature :: protected void doXxx(HttpServletRequest req,HttpServletResponse res) throws SE,IOE

=> These direct concrete methods of javax.servlet.http.HttpServlet(AC) having to return 405 error
response to browser indicating our servlet comp is complete to process the request..

is

note:: It is recommended to place request processing logic by overriding doXxx(-,-) methods in our servlet (especially doGet(-,-)/doPost(-,-))

=> GenericServlet is having only one service(-,-) method that is public service(-,-)/1st service(-,-) method
that is abstract method

=> HttpServlet is having only two service(-,-) methods and 7 doXxx(-,-) methods as concrete methods..

HttpServlet.java (Abstract class)

```
public abstract class HttpServlet extends GenericServlet{

    //1st service(-,-) method (5)
    public void service(ServletRequest req,ServletResponse res) throws SE,IOE{
        HttpServletRequest request=(HttpServletRequest)req;
        HttpServletResponse response=(HttpServletResponse)res;
        //calling 2nd service(-,-) method
        service(request,response); (6)
    }

    //2nd service(-,-) method (7)
    protected void service(HttpServletRequest req,HttpServletResponse res) throws SE,IOE{
        //get current request method/mode
        String method=req.getMethod();
        if(method.equals("GET"))
            doGet(req,res); (8)      calls 7 doXxx(-,-)based on current request mode
        else if(method.equals("POST"))
            doPost(req,res);
        else if(method.equals("HEAD"));
            doHead(req,res);
        ....
        ....
    }

    public void doXxx(HttpServletRequest req,HttpServletResponse res) throws SE,IOE {
        .... (7 methods)
        .... //logic to 405 error response to browser..
        ....
    }

    .... //other method definitions
}

//class
```

```
//LCTestServlet.java (ourServlet comp)
=====
(1) GET mod request from browser
    public class LCTestServlet extends HttpServlet{           Container
        (2) creates
        (3) ServletRequest
        (4) Since Servlet obj ready, the request processing
            event raises and calls 1st service(-,-) having ServletRequest,
            (life cyle method) ServletResponse objs but not found in our
            Servlet comp
        ServletResponse objs
        if created completes all instantiation formalities

        doPost
        public void (HttpServletRequest req,HttpServletResponse res) throws SE,IOE{
            ... //request processing (9) ✓
            ...
        }

        doGet
        public void doGet(HttpServletRequest req, HttpServletResponse res) throws SE,IOE{
            ...
            ...
        }

        service
        public void service(ServletRequest req,ServletResponse res) throws SE,IOE{
            super.service(req,res); ✓
            ...
        }
}
```

GenericServlet (abstract class)

|----> public abstract void service(ServletRequest req, ServletResponse res) throws SE, IOE

HttpServlet (abstract class)

|----> public void service(ServletRequest req, ServletResponse res) throws SE, IOE { }

|----> protected void service(HttpServletRequest req, HttpServletResponse res) throws SE, IOE { }

|----> protected void doXxx(HttpServletRequest req, HttpServletResponse res) throws SE, IOE { }

7methods

9 methods

=> In order to use the logic of certain Java class only accessible through its sub classes.. not directly then that class should be taken as abstract class.. Servlet API designers want to see the logic of HttpServlet class accessible only through sub classes (our servlet classes) .. not accessible directly.. So they have given HttpServlet class abstract class.

=> Since HttpServlet is abstract class .. all the logic of that class must be accessed only through sub classes .. not directly..

Note:: 2nd service(-,-) and 7doXxx(-,-) methods will not be called by ServletContainer or any outsider directly.. they will be called within HttpServlet class
or from its sub classes (like our servlet classes) .. For this they have them as ^{given}protected methods.. to apply access restrictions..

Assume the client makes a HTTP request based on GET method in the following situations

- 1. If our Servlet class contains both doGet(-,-) and doPost(-,-)**
 - a. public service(-,-) of HttpServlet
 - b. protected service(-,-) of HttpServlet
 - c. doGet(-,-) of our Servlet class
- 2. If our Servlet class contains only doPost(-,-)**
 - a. public service(-,-) of HttpServlet
 - b. protected service(-,-) of HttpServlet
 - c. protected doGet(-,-) of HttpServlet(as our class does not contain doGet(-,-))
 - d. 405 response back to client.
- 3. If our Servlet class overrides public service(-,-) method, and contains doGet(-,-), doPost(-,-).**
 - a. public service(-,-) of our Servlet class [doGet(-,-) of our Servlet class will not get invoked, because the control did not pass on to protected service(-,-) of HttpServlet].
- 4. If our Servlet class overrides public service(-,-) method, and does not contain doGet(-,-)**
 - a. public service() of our Servlet class
No 405 response back to client.
- 5. If our class overrides public service(-,-) method and it makes a call to super.service(-,-) method, and contains doGet(-,-), doPost(-,-)**
 - a. public service(-,-) of our Servlet class
 - b. public service(-,-) of HttpServlet(because of super.service(-,-))
 - c. protected service(-,-) of HttpServlet
 - d. doGet(-,-) of our Servlet class
- 6. If our class overrides public service() method, and it makes a call to super.service(-,-) method, and contains only doPost(-,-)**
 - a. public service(-,-) of our Servlet class
 - b. public service(-,-) of HttpServlet(because of super.service(-,-))
 - c. protected service(-,-) of HttpServlet
 - d. protected doGet(-,-) of HttpServlet
 - e. 405 response back to client

=>In the execution of our Servlet Component don't let the control going to doXxx(-,-) methods of javax.servlet.http.HttpServlet Class because they always generate **405 error response page** indicating our servlet class is totally **incomplete** to process the request.

Q. If both service (--) methods are placed in our Servlet Component then which method will be executed?

Ans: Since public service(--) method is life cycle method of our Servlet Component only public service(--) method will be executed.Even though public service(--) method is life cycle method it is recommended to place request processing logic of our Servlet Component by using doXxx(--) methods because they are defined based on protocol Http standards ,Moreover if they are not placed properly in our servlet ,the super class doXxx(--) methods simply send 405 error to client indicating the problem.

Q. When all the methods of pre-defined HttpServlet Class, are concrete methods why the class itself is given as an abstract class?

Note: In java abstract class can have only abstract methods or only concrete methods or mix of both

- javax.servlet.http.HttpServlet Class is abstract, even though none of the methods within it are abstract it is because, it contains seven doXxx() methods, to match seven ways of making HTTP request(GET/POST/DELETE/OPTIONS/TRACE/PUT/HEAD). These methods are the request processing methods of a HttpServlet, just like service(--) method for a GenericServlet.
- Since there is only one request-processing method in GenericServlet, it is defined as abstract in javax.servlet.GenericServlet Class, which makes the developer making his class to extend javax.servlet.GenericServlet Class, to provide implementation only for one method,But in the case of javax.servlet.http.HttpServlet Class, if all **9 (7+2 methods)** request-processing methods are defined as abstract, then every developer who creates a child class for it, has to provide implementations for all the **9** methods, which is quite an issue. So the specification has made javax.servlet.http.HttpServlet Class to contain implementations for all the 9 methods, but they made the javax.servlet.http.HttpServlet Class itself as abstract, which means no **Developer/ServletContainer** can create an instance of it directly.

The seven http request methods/methodologies

The client can send request to web component of the web application in seven different ways by using seven different Http methods. To process these methods based request in our Servlet Component we can override and use 7 different **doXxx(--) methods**.

- **GET (default)**
- **POST**
- **HEAD**
- **PUT**
- **DELETE**
- **TRACE**

- **OPTIONS**

GET:

Default request method, designed to get data from server by generating request without data or with limited amount of data (max. of 2kb-8kb).

POST:

Can send request with unlimited amount of data and gathers data from server as response. The response of GET based or POST based request contains everything including response body like response headers, miscellaneous information, etc...

HEAD:

Same as GET but the HEAD based request related response does not contain response body. HEAD based requests are useful to test whether web component is present or not. Note: Even though there are 7 request methods the most regularly used request methods in real world while developing java websites are GET, POST.

PUT:

This is capable of allowing client to place new file or web component in already deployed web application of web server. In real projects after placing websites in the web server of ISP(Internet Service Providers) machine we use FTP(File Transfer Protocol) application from our computers to maintain that website. This FTP application uses PUT method request to add new file or new web resource component in that ISP machine website.

DELETE:

Allows client to send a request having the capability to delete file or web component of web application in the server. FTP application uses this delete method to delete web page or document or anything from the hosted web application of the ISP machine based web server.

TRACE:

This **trace** method request returns all the debugging messages and flow of execution details regarding the request and response of certain web components.

OPTIONS:

The **options** method based request given to web resource component determines using which Http request methods that this servlet can be request from client.

For example:

If our Servlet Component overrides doGet(--) method as shown below then the OPTIONS method based request given to the Servlet Component returns the following response

Allow: HEAD, GET, OPTIONS, TRACE

Eg:

```
public class TestServlet extends HttpServlet
```

```
{  
    public void doGet(HttpServletRequest, HttpServletResponse res) throws  
        ServletException, IOException  
    {  
        .....  
        .....  
    }  
}
```

Note: “POST”,“PUT” are non-idempotent. “GET”,“HEAD”,“OPTIONS”,“TRACE”,“DELETE” are idempotent.

Note:

- Since doXxx(--) method are given to place 7 types of request processing logic for 7 modes of request given by client and also allows to work with HttpServletRequest, HttpServletResponse object. It is recommended to used doXxx(--).
- Using browser we can send only GET or POST methodology request. So while developing regular website we our develop our Servlet Component just having doGet(--) and doPost(--) method to process the request.
- Hyperlink can generate only GET methodology request where as form page can generate GET or POST methodology . requests, due to this we just use doGet(--). doPost(--) in our Servlet Component to process the request.
- PUT and DELETE and other request methodologies will be utilized only in the ftp application development which can be used website hosting and web application maintaince.

Hosting of java web application

=>It is all about keeping java web application on to the internet having domain name (website name)
For this we need to take the support of DomainRegistrar company like goDaddy, jelastic(java), indiarocks, i9, indiabricks and etc..

Domain Registrar gives following 3 services

- a) Selling/allocating space in the webserver /App server of Static IP Adress(fixed) machine.
- b) Selling domain names/web site names
- c) AMC (Annual Maintenance Contract) of website.

jelastic.com

- |--> Cloud based Domain registrar. (cloud-taking hardware/software infrastructure on rental basis)
- |--> Allows 14 trial period based email id.
- |--> supports java based hosting ..

Godaddy is not supporting java hosting as of now.

Procedure for hosting /placing java web applicaiton (website) on to the internet

step1) register with jelastic.com by submitting email id ..

=>jelastic.com ----> services ---> java hosting ---> submit email id : ... (rightside of the screen) ...
=> Open the submitted email account --> open jelastic email id --> click on link to choose the password and to complete the registration process

step2) create the env.. in jelastic

Choose Tomcat 9.x --> choose open jdk 14 --> choose NITHYD as env.. name -->create.

step3) create war file representing the web applicaiton...

Right click on Dynamic eclipse Project --> export -->war file --> select the folder to save the war file (Desktop --> MarriageApp.war)

step4) upload and deploy the war file.. to Tomcat server of Env..

Jelastic console screen --> upload -->select the war file .. (MarriageApp.war) --> go to war file --> deploy to --> select select env.. name(NITHYD) --> provide context path :: MarriageApp

step5) Test the web applicaiton...

URL :: <http://srk27.cloud.interhostsolutions.be/MarriageApp/input.html>

Maintenance of hosted Project/Web application

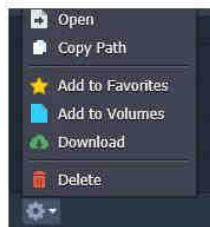
(Working with hosted Application)

To modify the source code of html file

Go to domain (NITHYD) ---> go to Application server(Tomcat) ---> config (wrench symbol) ---> go webapps (downside) ---> Go to MarriageApp ---> go input.html file ---> modify source code ---> save. and refresh browser window where the application is running..

To delete the any web comp of hosted web application

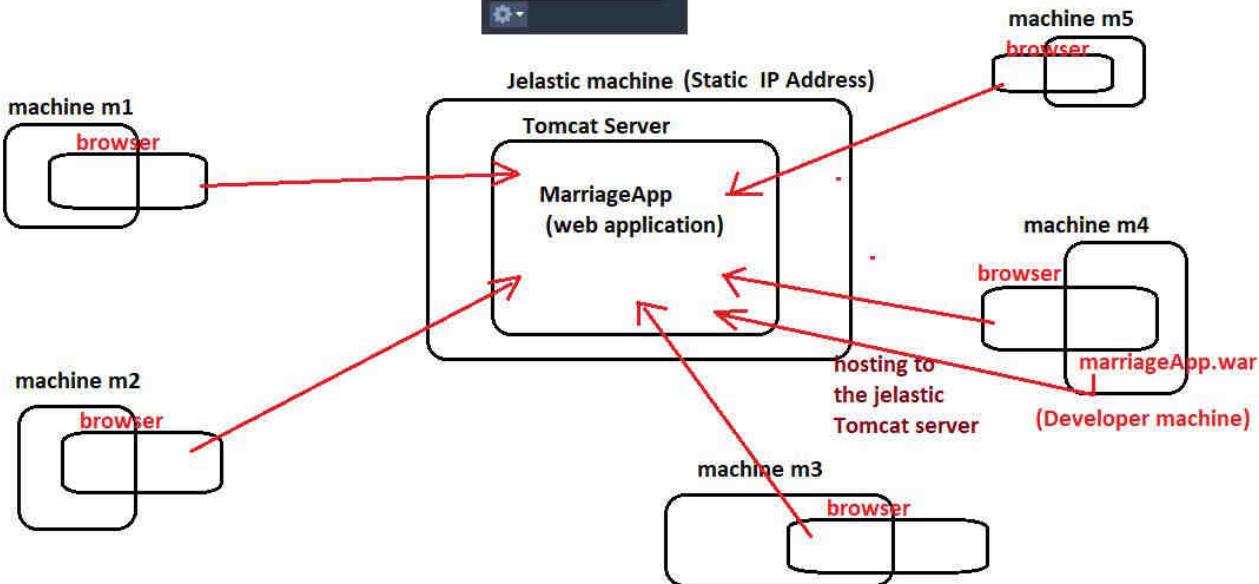
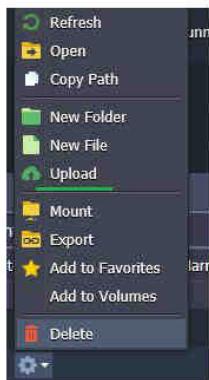
Go to the webcomp location (as show as above) ---> use addtionality button ---> delete ..



add

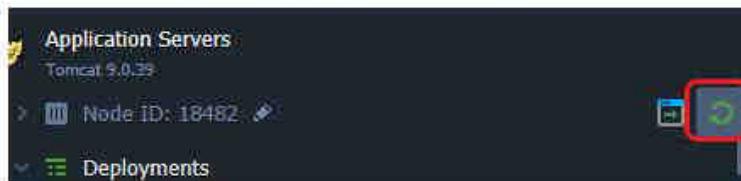
To new web comp to the hosted web application

=> Go to parent folder where u want to upload --> use addtionality button --> upload --> browse and select html file.



To modify the source code of Servlet comp in the hosted web application

- a) delete existing .class file of servlet comp from the hosted web application
- b) modify source the servlet comp in Eclipse Application and test it (Local modifications)
- c) gather .class file of Locally modified servlet comp (from Eclipse Tomcat server folder like wtpwebapps)
- d) upload locally upded .class file of servlet comp to Jelastic hosted Marriage Web application to WEB-INF/classes/com/nt/servlet folder
- e) restart the Tomcat server node.. from the env..



=>if modifications are happening in the huge scale.. then update all modifications to Local Eclipse Application.. over the certain time period (like for 20 days like that...) prepare the war file and replace existing hosted war file with new war file.

Understanding different Http Request methods /modes/methodologies

- GET
- POST
- HEAD
- PUT
- DELETE
- OPTIONS
- TRACE

CONNECT (reserved for future)

Understanding Http methods/modes

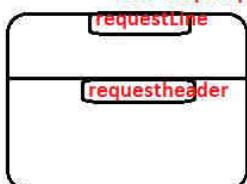
protocol http latest version : 1.1	
1.GET	http 1.0
2.POST	
3.HEAD	
4.PUT	
5.DELETE	http 1.1
6.TRACE	
7.OPTIONS	
8.CONNECT	(Reserved for future)

GET

====

- => Given to give the request to web server and to GET response from server
- => This is default request mode
- => This http request does not contain body becoz input data/query data along with request goes to server as query string appended to the url.

GET-HttpRequest



No request body
form data /request parameters
go to server as queryString appended to the request url

=> This is request idempotent (safe to repeat request)

=> This request related response contains all 3 parts
a) response line b) response headers c) response body

GET -HttpResponse

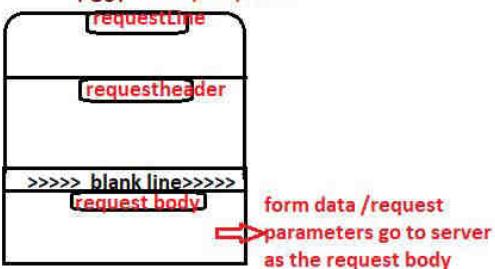


POST

====

- => Not the default request mode
- => Can carry unlimited data along with request as request body content
- => request structure contains a)request line b)request headers c) request body
- => given to send/post data to server along with the request and also get data from server
- => This request is not idempotent (not safe to repeat the request)

POST -HttpRequest



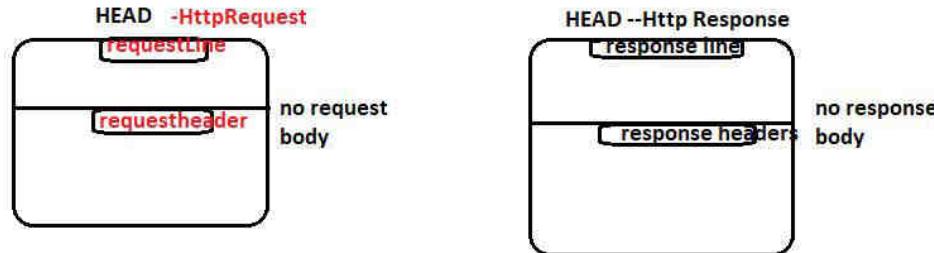
POST -HttpResponse



HEAD

====

- =>Same as GET but ... HEAD request mode related response does not contain reponse body.
- => It is given for checking wheather certain web comp is available or not (active or not)



As Enduser Cleint ,the browser can generate only GET,POST mode requets.

As admin Cleint , the jelastic cpanel, ftp clients can generate all the 7 modes of request.

PUT

====

- => Useful to generate request to place new comps / web applications in webServer..
- => Admin clients like jelastic cpanel , ftp clients will use this option in order to provide env.. for uploading of web applications or new web comps to exiting web applications.
- => PUT mode request is idempotent becoz it will try to put the same web comp by replacing the existing web comp with out any side effect (It is safe to repeat the request)
- =>In reamaing all aspects PUT mode request is same as POST mode request.

- => if we are developing normal web applictions to serv endusers then we need to work with doGet(-,-),doPost(-,-) methods
- => if we are developing cpanel,ftp client applictions to serv admin then we need to work with all the 7 doXxx(-,-) methods.

DELETE

=====

- =>Useful to generate request for deleting existing web comps of the web applicaiton or web applicaiton itself.
- => Admin clients like jelastic cpanel, ftp clients will use this DELETE mode request to delete the existing web comps and also to undeploy the web applicaitons..
- =>DELETE mode request is non idempotent.. if we try delete already deleted web comp by repeating request then we get exception error.
- DELETE
- =>In reamaing all aspects mode request is same as POST mode request.

TRACE

=====

- =>Useful for admin clients to trace/monitor various that are happening from request to response for a web comp..
- =>Useful for debugging and to know reasons for successful /failure execution of web comps..
- =>Useful to know how many forwards /redirects are happened for given request related response generation.
- => TRACE mode request is Idempotent.. i.e safe to repeate the request..
- => In remaining all aspects it is same as GET mode request.

OPTIONS

=====

=>if we OPTIONS mode request to any web comp then it will determine and give different request modes that possible to give.

=>This is useful to analyze web comp and to know what different request modes that should be used to generate the request to certain web comp.

For example

=====

```
public class TestServlet extends HttpServlet{  
  
    public void doGet(,-)throws SE,IOE{  
        ...  
        ...  
    }  
}
```

=>if we give OPTIONS mode request to this web comp
then get the following response
allow :: GET,TRACE,HEAD,OPTIONS

```
public class TestServlet extends HttpServlet{  
  
    public void doPost(,-)throws SE,IOE{  
        ...  
        ...  
    }  
}
```

=>if we give OPTIONS mode request to this web comp
then get the following response
allow :: POST,TRACE,OPTIONS

weblogic

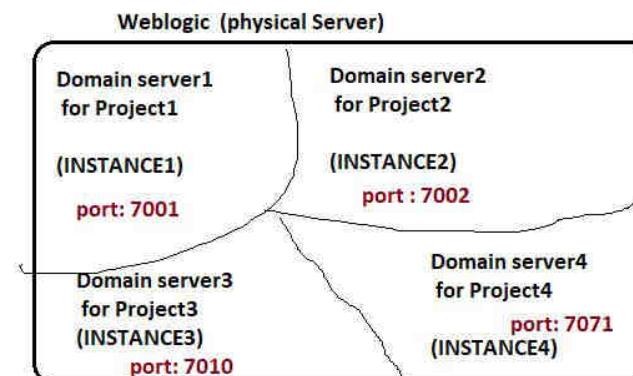
=====

type :: Application server (web Server++)
version :: 12c (compatible with jdk 1.7+)
vendor :: BEA Systems /oracle corp
default port number :: 7001

Allows to create domains .. Every domain is 1 instance of weblogic acting as separate application server..
(The multiple projects in company will use the multiple domains of same weblogic server)

commercial S/w

<https://www.oracle.com/middleware/technologies/weblogic-server-installers-downloads.html>
[weblogic-12.1.3-kepler-installer-win32.exe](#) (get this setup file)
(search for this setup file)



supports all 3 modes of deployment like console deployment, hard deployment and tool based deployment.

Procedure to create domain server/instance in weblogic 12c s/w

search and launch oracle weblogic configuration wizard --> select basic domain --> type domain name (WLNTAJ414Domain) as the directory name-->next-->next --> choose username ,password (javaboss, javaboss1) -->next-> choose other jdk Location (max java8) --> next -->select adminstration server --> next--> choose port number -->next--> create..

Procedure to perform console deployment of java web application in weblogic domainserver

step1) prepare war file representing the web application..

note1:: make sure all .class files are generated by using java8 compiler

note2:: Remove XSD import statement from web.xml file.

VoterApp.war

```
E:\VoterApp\WEB-INF\classes>set path=C:\Program Files\Java\jdk1.8.0_31\bin  
E:\VoterApp\WEB-INF\classes>set classpath=E:\Tomcat 9.0\lib\servlet-api.jar  
E:\VoterApp\WEB-INF\classes>javac -d . VoterServlet.java  
E:\VoterApp>jar cf VoterApp.war .
```

step2) start weblogic domain server (WLNTAJ414Domain)

go to C:\Oracle\Middleware\Oracle_Home\user_projects\domains\WLNTAJ414Domain folder and use startWeblogic.cmd file.

step3) open admin console of weblogic domain server by submitting username,password

<http://localhost:7070/console> --> submit username:::javaboss and password :::javaboss1 --> login

step4) upload the war file and deploy the war file..

Admin console screen --> deployments --> install --> upload your files --> select the war file (VoterApp.war) -->next-->next->next->next-->save

step5) Test the web application..

<http://localhost:7070/VoterApp/input.html>

note:: To undeploy the web application from admin console screen

Go to admin console screen --> deployments --> select the webapplicaiton (VoterApp) --> delete.

=>To perform HARDeployment web application in weblogic domain server copy war file (VoterApp.war)

or directory (VoterApp) to

C:\Oracle\Middleware\Oracle_Home\user_projects\domains\WLNTAJ414Domain\autodeploy folder

<Weblogic_home>

(Domain directory
name)

<http://localhost:7070/VoterApp/input.html>

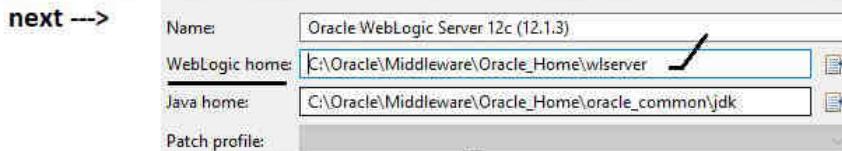
Procedure to configure weblogic domain server with Eclipse IDE (try with eclipse 2019 versions)

step1) install oracle weblogic server adapter /tool/plugin with eclipse IDE

window menu-->preferences -->server --> runtime env.. ----> add --> oracle --> oracle weblogic server tools -->next --> accept terms and conditions --> restart IDE.

step2) cfg weblogic domain server with eclipse IDE.

window menu-->preferences -->server --> runtime env.. ----> add --> oracle --> oracle weblogic server --> select local server-->



next

Domain directory: C:\Oracle\Middleware\Oracle_Home\user_projects\domains\WLNTAJ414

select domain server

Finish --> apply and close

Procedure to configure weblogic domain server with Eclipse IDE (try with Eclipse 2018 versions)

step1) install oracle weblogic server adapter /tool/plugin with eclipse IDE

window menu-->preferences -->server --> runtime env.. ----> add ---> oracle ---> oracle weblogic server tools -->next --> accept terms and conditions --> restart IDE.

step2) cfg weblogic domain server with eclipse IDE.

window menu-->preferences -->server --> runtime env.. ----> add ---> oracle —> oracle weblogic server —> select local server-->

next -->

Name: Oracle WebLogic Server 12c (12.1.3)
WebLogic home: C:\Oracle\Middleware\Oracle_Home\wlserver
Java home: C:\Oracle\Middleware\Oracle_Home\oracle_common\jdk
Patch profile:



next

Domain directory: C:\Oracle\Middleware\Oracle_Home\user_projects\domains\WLNTAJ414

select domain server

↓
Finish ----> apply and close

GlassFish

type :: Application server
version :: 5.x (comptible with jdk1.8+)
vendor :: Sun Ms/Oracle corp
allows to create domain servers (Instances of GlassFish server)
Domain port numbers :: http (8080) , admin console (4848)
default domain name :: domain1
domain1 username :: admin
domain1 password : adminadmin

To download s/w :: Download s/w as zip file from <https://javaee.github.io/glassfish/download>

To install GlassFish s/w :: Extract the zip file. (glassfish-5.0.zip)

Procedure to create user-defined domain server GlassFish 5.x

```
E:\glassfish-5.0\glassfish5\glassfish\bin>set path=C:\Program Files\Java\jdk1.8.0_31\bin
E:\glassfish-5.0\glassfish5\glassfish\bin>asadmin create-domain --adminport=4545 --user=testuser GFNTAJ414Domain
Enter the admin password [Enter to accept default of no password]>testuser
Enter the admin password again>testuser
...
...
...
Domain created successfully
```

Procedure to change http port number of GlassFish domain server

=>Go to E:\glassfish-5.0\glassfish5\glassfish\domains\GFNTAJ414Domain\config folder domain.xml file
and modify "port" attribute value of first
<network-listener port="6677" protocol="http-listener-1" transport="tcp" name="http-listener-1" thread-pool="http-thread-pool">
</network-listener>

Procedure to perform console deployment in GlassFish5.x domain server

step1) create war file (make sure all java classes compiled jdk 1.8) (VoterApp.war)

step2) start Glassfish domain Server..

E:\glassfish-5.0\glassfish5\glassfish\bin>asadmin start-domain GFNTAJ414Domain

step3) Open admin console of GlassFish domain Server

http://localhost: 4545

username:testuser

password:testuser

login

step4) Deploy the war file..

admin console screen --> applications -->deploy -->select the war file (VoterApp.war) --> ok.

step5) Test the web application..

http://localhost:6677/VoterApp/input.html

(or)

Admin console -->deployments --> go to war file (VoterApp) --> launch

=>For undeployment of web application using

admin console

Admin console screen --> deployments -->

VoterApp -->undeploy.

Procedure to hard deployment of web application in Glassfish Domain server

copy war file (VoterApp.war) or directory (VoterApp) to

E:\glassfish-5.0\glassfish5\glassfish\domains\GFNTAJ414Domain\autodeploy folder

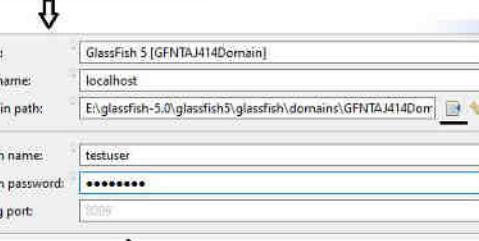
To test the web application.

http://localhost:6677/VoterApp/input.html

Procedure to configure GlassFish domain server with Eclipse IDE

window menu --> preferences --> server -->run time env-->

add --> GlassFish --> Glassfish 5.0 -->



Finish.

Differences between Webserver and Application server?

<u>WebServer</u>	<u>Application Server</u>
a) It is developed based on Servlet,jsp technologies	a) It is developed based on Servlet,jsp, EJB, Jndi, JTA, Java mail,JMS and etc.. technologies
b) It can be used only to deploy ,manage and execute web applications (war file)	b) It can be used to deploy ,manage and execute web applications(war file) , EJB comps (jar file), Enterprise Applications (ear file).
c) Gives only ServletContainer ,jsp Container	c) Gives Servlet container, jsp container, EJB container, JMS server , Mail Server and etc..
d) Understands only protocol http, https	d) Understands multiple protocols like http,https, iiop, jrmp, and etc..
e) Light weight server	e) Heavy weight server
f) does not allow to create domains	f) Allows to create domains
g) Suitable for small scale web applications	g) Suitable for medium scale, large scale web applications and also for enterprise Applications ,mailing Apps, JMS Applications
h) eg:: Tomcat , jetty , resin and etc..	h) weblogic ,GlassFish , Wildfly , Oracle10gAS, Jrun and etc..
i) gives less no.of middleware services	j) Gives more no.of middleware services.
j) can handle limit no.of clients/reqets at a time	j) can handle huge no.of requests/clients at a time.

ear file = war file +war file +...
ear file = jar file+ jar file +...
ear file = war file + jar file + ..

war --> web application archive
jar --> java archive (ejb comp)
ear -> Enterprise Application archive

Application Server = webserver + EJB Container + More Middleware services..

=>Upto Tomcat 6.x , It is web server

=> from Tomcat 7.x , It can be called as Application server becoz more facilities are added from 7.x version

Servlet To DB s/w communication

It is required for

- => To save inputs coming to Servlet comp permanently (SignUp activity)
 - => To save results/outputs generated by servlet comp permanently (online shopping)
 - => To get data from Db s/w to use in b.logic or to display for enduser (reports)
 - => To modify the data of DB s/w as need for Application (Changing items prices during offer period)
 - => To perform delete on the data (closing bank account, closing membership and etc..)
- note:: Servlet has perform CURD/CRUD Operations on Db s/w data by using jdbc code.

There are 4 approaches of placing jdbc code in servlet comp

Approach1 (Bad Approach)

- => create JDBC con object in the init() method of servlet comp.
- => Use JDBC con object in the service(-,-)/doXxx(-,-) method to create other jdbc objs and to develop persistence logic
- => close jdbc con in the destroy() method

Sample Servlet

```
public class TestServlet extends GS/HS{
    private Connection con;
    public void init(){
        //create jdbc con
        ...
    }
    public void service(-,-)/doXxx(-,-)throws SE,IOE{
        //use jdbc con to create other jdbc objs
        // and to develop persistence logic
        ...
    }
    public void destroy(){
        //close jdbc con
        con.close();
    }
}
```

In any jdbc code

- => To create jdbc con obj
- => Use jdbc con obj to create other jdbc objs and to develop persistence logic
- => Close jdbc con

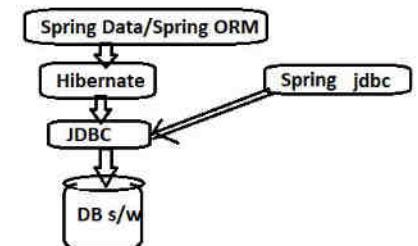
advantages::

- => All the requests coming to servlet comp will use single jdbc connection to interact with Db s/w
- So the performance will be good.

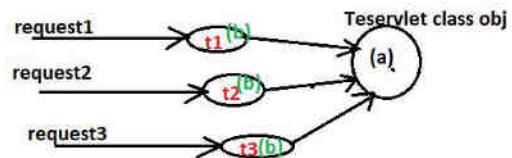
disadvantages::

- => Here jdbc con obj must be taken as instance variable of Servlet comp class, So it is not thread safe by default., We need to use Synchronization concepts to make it as thread safe.
- => If any simultaneous request related thread calls con.rollback() method then other threads related persistence operations will be disturbed.

- =>Jdbc code in standalone App makes the standalone App talking to Db s/w
- =>Jdbc code in Web comp makes the Web App talking to Db s/w
- =>Jdbc code in Spring App makes the Spring App talking to Db s/w
- =>Jdbc code in WebService comp makes the Distributed App talking to Db s/w and etc..



```
public class TestServlet extends GS/HS{
    private int a;
    public void service/doXxx(-,-)throws SE,IOE{
        int b;
        ....
    }
}
```



=> Every Thread gets its own copy of Local variable (b) , So Local variable any method including service(-,-)/doXxx(-,-) method are thread safe by default.

=>All threads acting on Servlet class obj will work on single copy of instance variable (a).. So instance variable java class like servlet class are not thread safe by default, To make them thread safe in service(-,-) method we need to use synchronisation concepts.

Approach2 :: (good Approach)

- =>create jdbc con obj in the service(-,-)/doxxx(-,-) methods
- =>use jdbc con obj to create other jdbc objs
and to develop persistence logic in service(-,-)/doXxx(-,-) methods
- => close jdbc con obj in service(-,-) /doXxx(-,-) methods

```
public class TestServlet extends HS/GS{  
  
    public void service(-,-)/doXxx(-,-)throws SE,IOE{  
        //create jdbc con obj  
        ...  
        //use jdbc con obj and to create other jdbc objs  
        ...  
        ...  
        //close jdbc con object  
        ...  
    }  
}
```

Advantages::

- a) Here jdbc con obj is Local variable to service(-,-)/doXxx(-,-) methods , So it is thread safe by default
- b) if any simultaneous request calls con.rollback() method it does not affect other request related persistence operations becoz every request/thread is using its own copy of jdbc con object.

DisAdvantage:

- > Since every request is using separate jdbc con to interact with Db s/w the performance will be degraded.

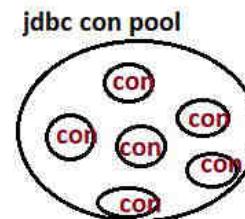
JDBC Con pool

=====

=>Jdbc con pool is a factory that contains set of readily available jdbc con objs before actually being used.

The advantages of jdbc con pool are ::

- a) Reusability of pooled jdbc con objs
- b) With minimum jdbc con objs, we can make maximum clients /requests talking to Db s/w
- c) Programmer need not worry about creating jdbc con objs, managing jdbc con objs and destroying the jdbc con objs.. all these operations will be taken care by jdbc con pool itself



Every application server and most of web servers like Tomcat are giving provision to work Server managed jdbc con pools

Approach3::

- =====
- => Get jdbc con object from Server managed jdbc con pool being from service(-,-)/doXxx(-,-) method.
 - => Use that jdbc con object to create other jdbc objs and to develop persistence logic being from service(-,-)/doXxx(-,-) method
 - => close or return jdbc con object back to jdbc con pool being from service(-,-)/doXxx(-,-) method.

Advantages

- =====
- a) we can take all the advantages of jdbc con pool (end to end)
 - b) The reference varible that holds server managed jdbc con pool is local variable to service(-,-)/doXxx(-,-) method , So it thread safe variable

disadvantage::

- use
a) We should use that server that supports server managed jdbc con pool

DAO class (Data Access Object)

- =====
- => The java class that separates persistence logic from other logics and makes that persistence logic as reusable logic and flexible logic to modify is called DAO..
 - => DAO class should not have any logic other than persistence logic even c=a+b is wrong in DAO class
 - => if project is having <100 db tables ,then we should take DAO classes on 1 per db table basis
 - => if project is having >=100 db tables ,then we should take DAO classes on 1 per related db tables..
(4-5 db tables together)

=>DB s/w like oracle ,mysql, postgreSQL comes under Data storage technologies.

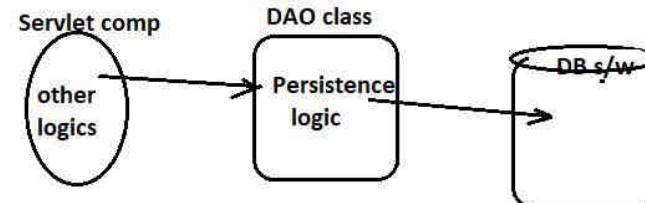
=>The Technologies using which we can develop persistence logic are called as Data Access Technologies like jdbc,hibernate, iBatis,spring jdbc , spring data and etc..

=> The java class that contains Data Access technology's persistence logic is called DAO class.

note:: Instead of writing persistence logic directly in servlet comp, it is recommended to write in separate java class called DAO make servlet comp taking with DAO class

Approach4

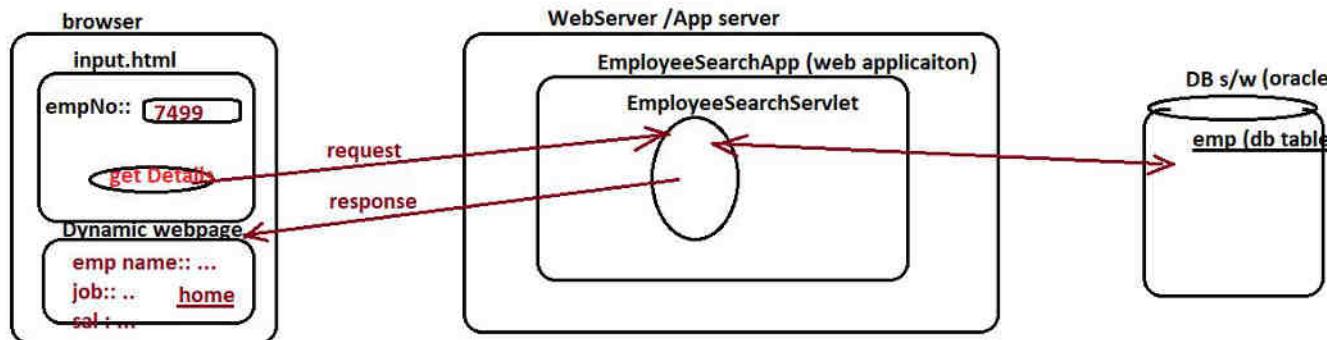
- =====
- a) create DAO class having Persistence logic
(It can use either direct jdbc con obj or pooled jdbc con object)
 - b) In servlet comp , create DAO class object and invoke methods on it to execute DAO class persistence logic being from servlet comp.



Conclusion:: if the web application is small scale then use
Approach2 (less than <10 web pages)

if the web application is large scale or medium scale then use
Approach4 (more than >=10 web pages)

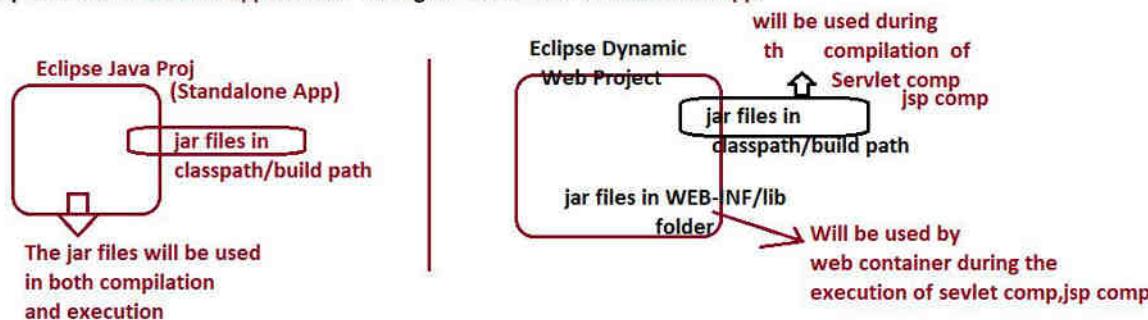
Servlet to Db s/w communication using Approach2 (writing jdbc code in service(-,-)/doXxx(-,-) method)



=>The jar files added to CLASSPATH/BUIDPATH of Eclipse Dynamic web project will be used by Eclipse Java Compiler to recognize third party api during the compilation of servlet comp ,where as the jar files added WEB-INF/lib folder of java web application will be used by WebContainer to recognize and use third party api during the execution of web comps like servlet comps, jsp comps.

=>if java web application use third party api (like oracle thin driver) then that third party api related jar files (like ojdbc6.jar) must be added to CLASSPATH/build path and must also added to WEB-INF/lib folder.

=>The STandalone App/Project of Eclipse IDE uses the jar files added to build path/classpath during the compilation of standalone app and also during the execution of standalone App.

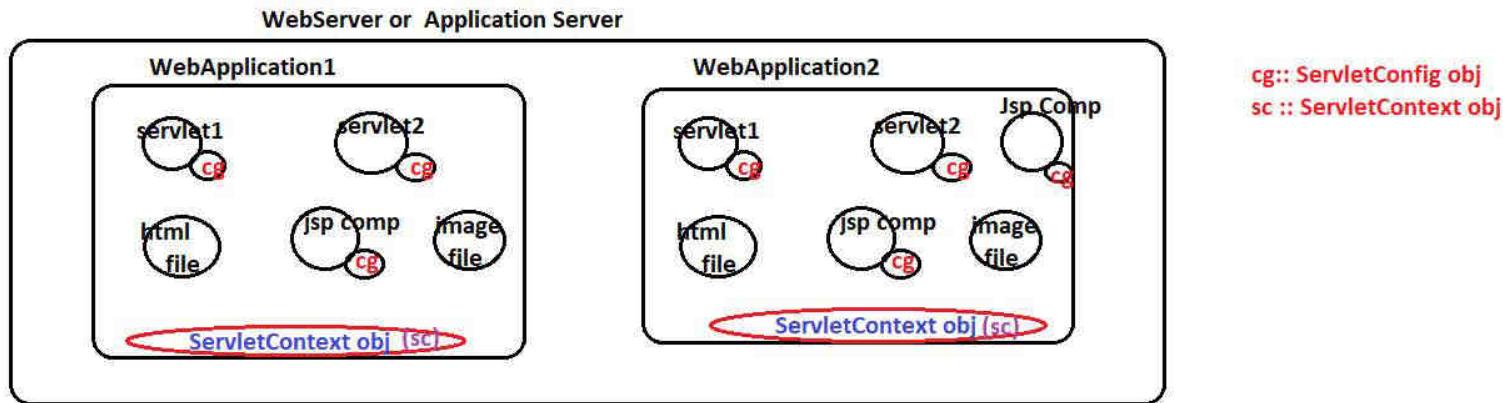


=>Instead of placing jar files to WEB-INF/lib folder manually, we can use Deployment Assembly option of Eclipse IDE's Dynamic web Project to do the same i.e it copies the select jar files of buildpath to WEB-INF/lib folder..

Right click Dynamic Web Project --> Properties ---> Deployment Assembly ---> add ---> java buildpath entries --> select jar files from Build path -->

What is the difference between ServletConfig and ServletContext objs?

ServletConfig obj --> It is 1 per Servlet comp /jsp comp class obj
ServletContext --> it is 1 per web application



ServletConfig obj

- => It is 1 per servlet class obj/jsp comp.
- => It is right hand object to our servlet class obj becoz Programmer uses this object to gather info about servlet comp and to pass info servlet comp from outside.
- => ServletContainer creates this object during the deployment of web application and assigns to respective Servlet class obj as part Servlet Initialization by calling init() having Servletconfig obj as the argument value.
- => Servletconfig obj means it is the object of underlying Servletcontainer supplied java class that implements javax.servlet.ServletConfig(). Its impl class name will change webcontainer to Web container or Server to server.
- => Servletcontainer destroys ServletConfig obj along with Servletclass object
- => Using Servletconfig obj
 - >we can get logical name (instance name) of Servlet comp
 - >we can get init param values of Servlet comp that are placed in web.xml file.
usecase:: passing jdbc properties from web.xml as technical input values to servlet comp

note:: Since Servletcontainer creates our servlet class obj using 0-param constructor, So programmer can not pass his choice data/values as part of servlet initialization through constructor.. To overcome this problem init(Servletconfig) method is given as life cycle method ,so programmer can pass init param values through Servletconfig obj as initial data/values to servlet comp.

jdbc properties

- >driver class name
- >jdbc url
- >db username
- >db password

(These are technical inputs , So only Programmer can give them)

Servletcontext obj

- =>It is 1 per web application
- =>Servletcontainer creates this object on the deployment of web application and destroys this object when the web application is stopped or reloaded/undeployed
- =>It is called global memory of the web application becoz if we keep any data in this object and that data can be accessed in all web comps through out a web application.
- =>It is the object of java class given underlying Servletcontainer implementing javax.servlet.ServletContext(I)
- => Using ServletContext obj , we can
 - > get underlying server details
 - > get all servlet comp names
 - > servlet api version supported by underlying server
 - > Global init params/context param values kept in web.xml file
(servlet init params are specific to each servlet comp where as global init params/context param are common for all web comps of web applicaiton) note:: if multiple servlet comps are looking form same jdbc properties from web.xml file then pass them as global init parameter with the support of ServletContet obj
- > get underlying web name/context path
- => perform dynamic registration of servlet comp (cfg servlet comp with url pattern using java code) and etc..

In a web server 10 java web applicaiton are deployed , In that 6 web applications are in running mode and 4 web applicaitons are in stopped mode .. Can u tell me how many Servletcontext objs are currently available in the web server?

- a) 10 b) 6 c)1 d)4

correct answer :: (b) (10-4 = 6)

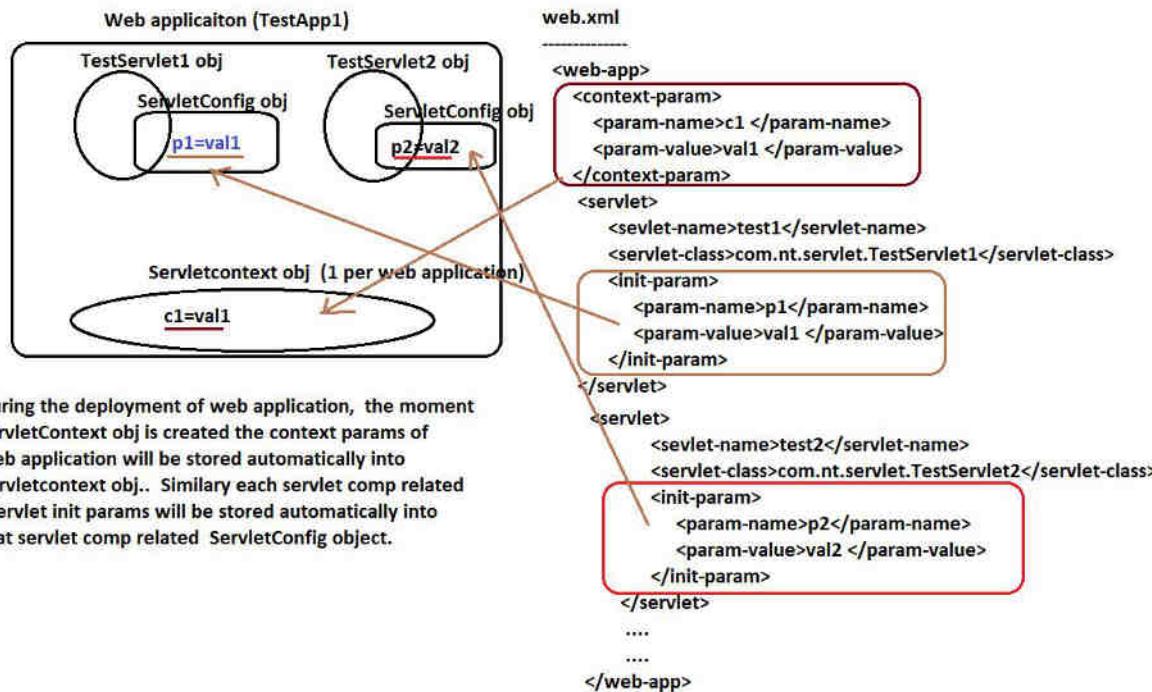
Q) In a web application total 10 servlet comps are there , In that 3 servlet comps are already requested and another 3 servlet comps are enabled with <load-on-startup> , Can you tell me total how may ServletConfig objs are currently avaialble?

- a)10 b) 3 c) 6 d) 1

Corrent answer :: (a)

Q) Can i give same name to ServletInit Param and context param (global init param)?

Ans) yes , we can becoz init params allocates memory in Servletconfig obj and context params allocate memory in SerlvetContext obj



Use Case of Servletinit parameters

=>Instead of hardcoding jdbc properties in directly in Servlet comp, it is recommended to gather them from web.xml file as servlet init parameter values (Soft coding) with the support of servelt config obj becoz the are technical input values..

Code in web.xml file

```

<servlet>
    <servlet-name>emp</servlet-name>
    <servlet-class>com.nt.servlet.EmployeeSearchServlet</servlet-class>
    <init-param>
        <param-name>driverClass</param-name>
        <param-value>oracle.jdbc.driver.OracleDriver</param-value>
    </init-param>
    <init-param>
        <param-name>jdbcUrl</param-name>
        <param-value>jdbc:oracle:thin:@localhost:1521:xe</param-value>
    </init-param>
    <init-param>
        <param-name>dbuser</param-name>
        <param-value>system</param-value>
    </init-param>
    <init-param>
        <param-name>dbpwd</param-name>
        <param-value>manager</param-value>
    </init-param>
</servlet>

```

Code in Servlet comp

```

//get Access to Servletconfig obj
ServletConfig cg=getServletConfig();
String driver=cg.getInitParameter("driverClass");
String url =cg.getInitParameter("jdbcUrl");
String username=cg.getInitParameter("dbuser");
String pwd=cg.getInitParameter("dbpwd");
//write jdbc code
Class.forName(driver);
//establish the connection
con=DriverManager.getConnection(url,username,pwd);

```

Being from one servlet comp, can we read init param values of another servlet comp?

Ans) No .. not possible .. we get null values.. becoz ^{s are} ServletInit parameter specific to each Servlet comp as servlet init param allocate

Memory in ServletConfig obj (which is specific to each Servlet comp)

Q) Can have same name for two init parameters of two different servlet comps having different values?

Ans) Yes , we can have becoz Each Servlet's init parameter allocates memory in respective separate ServletConfig object.

in web.xml

=====

```
<servlet>
  <servlet-name>test</servlet-name>
  <servlet-class>com.nt.servlet.TestServlet</servlet-class>
  <init-param>
    <param-name>dbuser</param-name>
    <param-value>mahesh</param-value>
  </init-param>
</servlet>
```

```
<servlet>
  <servlet-name>emp</servlet-name>
  <servlet-class>com.nt.servlet.EmployeeSearchServlet</servlet-class>
  <init-param>
    <param-name>dbuser</param-name>
    <param-value>system:</param-value>
  </init-param>
</servlet>
```

Q) What happens ,if two init params of servlet comp are having same name with different values?

Ans) only first value will be taken as the final init param value.

```
<servlet>
  <servlet-name>test</servlet-name>
  <servlet-class>com.nt.servlet.TestServlet</servlet-class>
  <init-param>
    <param-name>dbuser</param-name>
    <param-value>mahesh1</param-value>
  </init-param>      This will be taken as final value
  <init-param>
    <param-name>dbuser</param-name>
    <param-value>mahesh</param-value>
  </init-param>
</servlet>
```

Q) How can we get our Servlet class obj instance name(object name) created by Servlet Container?

Ans) we can get that information with the support of ServletConfig obj by calling
cg.getServletName() method ... (the logical name given <servlet-name> tag becoz
Servlet instance (object name) internally)

```
pw.println("<br> SErvlet class object name::"+cg.getServletName());
```

note: ServletConfig obj is the object of underlying webContainer supplied java class that implements javax.servlet.ServletConfig()
we can get that class name using cg.getClass(); In tomcat server catalina.jar gives the class name of ServletConfig obj ...

```
pw.println("<br> SErvletConfig object class name::"+cg.getClass());
```

↓
Gives org.apache.catalina.core.StandardWrapperFacade

(This class implements Servletconfig()
directly or indirectly)

How to pass input values to servlet comp from outside the servlet comp?

=>There are two ways of passing

a) as request parameters (nothing but form data)

[Useful to get non-technical input input values from endusers like name,age,address and etc..]

=>Will be stored in request object automatically, so we can use req.getParameter(-) method to read them being from Servlet comp

=> specific to each request (so the scope is request scope)

b) As servlet init parameters/Servletcontext params

[Useful to get technical input values from Programmer like jdbc properties]

=>Servlet Init params are specifc each servlet comp and they will be stored in ServletConfig obj , to read init parameter value use cg.getInitParameter(-) method.

=> ServletContext params are common for multiple servlet comps and they will be stored in Servletcontext obj, To read context param values use sc.getInitParameter(-) method.

Multiple Servlet comps of a web application are looking to interact with Db s/w by using same jdbc driver

Then what is best way to get jdbc properties in the all the servlet comps?

a) Hard code jdbc properties in multiple servlet comps (Hard coding bad pratice)

b) get jdbc properties from enduser as request parameters(form data) (enduser can not give technical input values)

c) get jdbc properties from web.xml as servlet init parameters (we need same init params for multiple servlet comps)

d) get jdbc properties from web.xml as servlet context parameters (Ans ::d)

| ---> We can jdbc properties as context params only once in web.xml file and they can be used in all servlet comp

getting jdbc properties from web.xml as context param values

code in web.xml file

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://xmlns.jcp.org/xml/ns/javaee"
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app_4_0.xsd"
id="WebApp_ID" version="4.0">
    <display-name>EmployeeSearchApp13</display-name>
    <welcome-file-list>
        <welcome-file>input.html</welcome-file>
    </welcome-file-list>

    <context-param>
        <param-name>driverClass</param-name>
        <param-value>oracle.jdbc.driver.OracleDriver</param-value>
    </context-param>
    <context-param>
        <param-name>jdbcUrl</param-name>
        <param-value>jdbc:oracle:thin:@localhost:1521:xe</param-value>
    </context-param>
    <context-param>
        <param-name>dbuser</param-name>
        <param-value>system</param-value>
    </context-param>
    <context-param>
        <param-name>dbpwd</param-name>
        <param-value>manager</param-value>
    </context-param>
    <servlet>
        ...
    </servlet>
    <servlet-mapping>
        ...
    </servlet-mapping>
```

In Servlet comps

//get Access to ServletContext obj

```
ServletContext sc=getServletContext();
String driver=sc.getInitParameter("driverClass");
String url =sc.getInitParameter("jdbcUrl");
String username=sc.getInitParameter("dbuser");
String pwd=sc.getInitParameter("dbpwd");
//write jdbc code
Class.forName(driver);
//establish the connection
con=DriverManager.getConnection(url,username,pwd);
```

Can i give same to context param and servlet init param ?

Ans) Yes , we can give

note:: Init param allocates memory in **ServletConfig** obj
and Context param allocates memory in **ServletContext** obj

What happens if context params are having same name with different values?

Ans) The last value in the list will be taken as final value.

note:: we do not create following objects.. these objects are created by ServletContainer but we can access these objs in our servlet comp by using different techiques

- a) our servlet class obj (we can "this" to access our servlet class obj)
- b) request obj | [These two objects can be accessed as the
- c) response obj | parameters of service(-,-)/doXxx(-,-) methods]
- d) ServletConfig obj
- e) ServletContext obj

Different ways of accessing ServletConfig obj?

Approach1) as init(-) method parameter

```
public class TestServlet extends HttpServlet{  
    ServletConfig cg;  
    public void init(ServletConfig cg){  
        this.cg=cg;  
        ... use cg here  
    }  
    public void service(-,-)/doXxx(-,-) throws SE,IOE{  
        //use cg here  
        ....  
    }  
}
```

Bad Approach , we may forget initialize Servletconfig obj

Approach2) Using getServletConfig() method (Best)

```
public class TestServlet extends HttpServlet{  
  
    public void init(){  
        ServletConfig cg=getServletConfig();  
        //use cg here  
    }  
    public void service(-,-)/doXxx(-,-)throws SE,IOE{  
        ....  
        Servletconfig cg=getServletConfig();  
        //use cfg here  
    }  
}
```

Different ways of getting access to ServletContext obj?

Approach1) Using cg.getServletContext();

In any method of servlet comp

```
ServletConfig cg=getServletConfig();  
ServletContext sc=cg.getServletContext();
```

Approach2) Using getServletContext() of GenericServlet class (Best)

```
ServletContext sc=getServletContext();
|-->This method interlly uses approach1 code
|--> This is called convience method to the programmer
```

In GenericServlet.java

```
public ServletContext getServletContext() {
    return this.getServletConfig().getServletContext();
}
```

Approach3) Using reg.getServletContext(); method (Servlet api 3.0)

// Only from service(-,-)/doXxx(-,-) methods of servlet comp

```
ServletContext sc=req.getServletContext();
```

is

What the advantage of writing initialization logic in the init() method of servlet comp rather writing in the constructor though both are one time executing blocks of servlet comp?

Ans) ServletConfig obj is assigned to Servlet class obj using init(-) method of servlet comp..

With out servletConfig obj or request obj we can not access ServletContext object..

note:: The code placed in the constructor of Servlet comp can not get access to

Servlet Init param, context param values becoz constructot does not have access to ServletConfig, ServletContext objs.

The code placed in init() method of servlet comp can get access to Servlet Init param, context param values becoz init() is having access to ServletConfig, ServletContext objs

note:: if we place logic to create JDBC con obj in the init() method of servlet comp

then we can get jdbc properties from web.xml either as init params or
as context params

note:: if we place logic to create JDBC con obj in the constructor of servlet comp

then we can not get jdbc properties from web.xml either as init params or
as context params

=>ServletContext obj means , it is not object javax.servlet.ServletContext(). It is the object of

java class that implements javax.servlet.ServletContext(). [It is class org.apache.catalina.core.ApplicationContextFacade in Tomcat server]
(see in catalina.jar)

Using ServletContext obj we can gather more info current web application, underlying server and its web comps

```
ServletContext sc=getServletContext();
pw.println("<br> Server info ::"+sc.getServerInfo());
pw.println("<br> server api version ::"+sc.getMajorVersion()+"."+sc.getMinorVersion());
pw.println("<br> Context path of web application::"+sc.getContextPath());
pw.println("<br> Absolute Path of input.html::"+sc.getRealPath("/input.html"));
pw.println("<br> Absolute path of web application::"+sc.getRealPath("/")); //=>Web root folder
pw.println("<br> MIME type of input.html::"+sc.getMimeType("/input.html"));
```

Two types of jdbc con pools

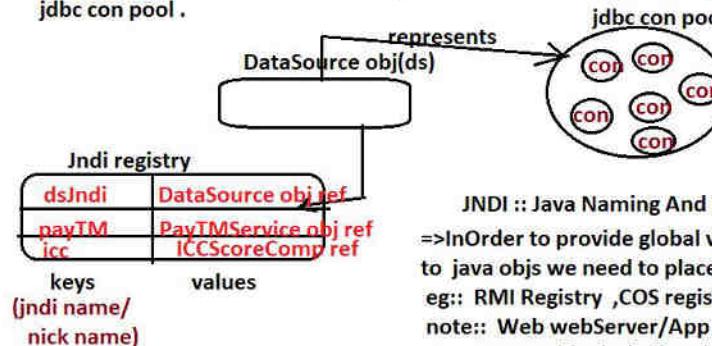
1. standalone jdbc con pool

=>useful in
standalone Apps
(The apps that are
running outside the server)
eg:: apacheDBCp, HikariCP
and etc..

2. server Managed JDBC con pool

=> This jdbc con pool is created and managed
by underlying webServer /Application server..
=> useful for web applications that deployable and
executable in webServer/Application server
eg:: Weblogic managed jdbc con pool
Tomcat managed jdbc con pool
Wildfly managed jdbc con pool

DataSource obj represents Jdbc con pool and acts as entry point to get jdbc con objs from
jdbc con pool .



JNDI :: Java Naming And Directory Interface

=>InOrder to provide global visibility and accessibility
to java objs we need to place them in Jndi registry
eg:: RMI Registry ,COS registry

note:: Web webServer/App server gives one built-in jndi registry
eg: weblogic jndi registry, Tomcat jndi registry, Wildfly jndi registry...

Java App -----> JDBC API (java.sql ,javax.sql and etc..) (Using JDBC API code , we can perform DB s/w like insert,update,delete and select operations)

Java app -----> JNDI API (javax.naming and sub.packages) (Using JNDI API code , we can perform JNDI operations on JNDI registry like Bind,Unbind,Rebind and lookup operations)

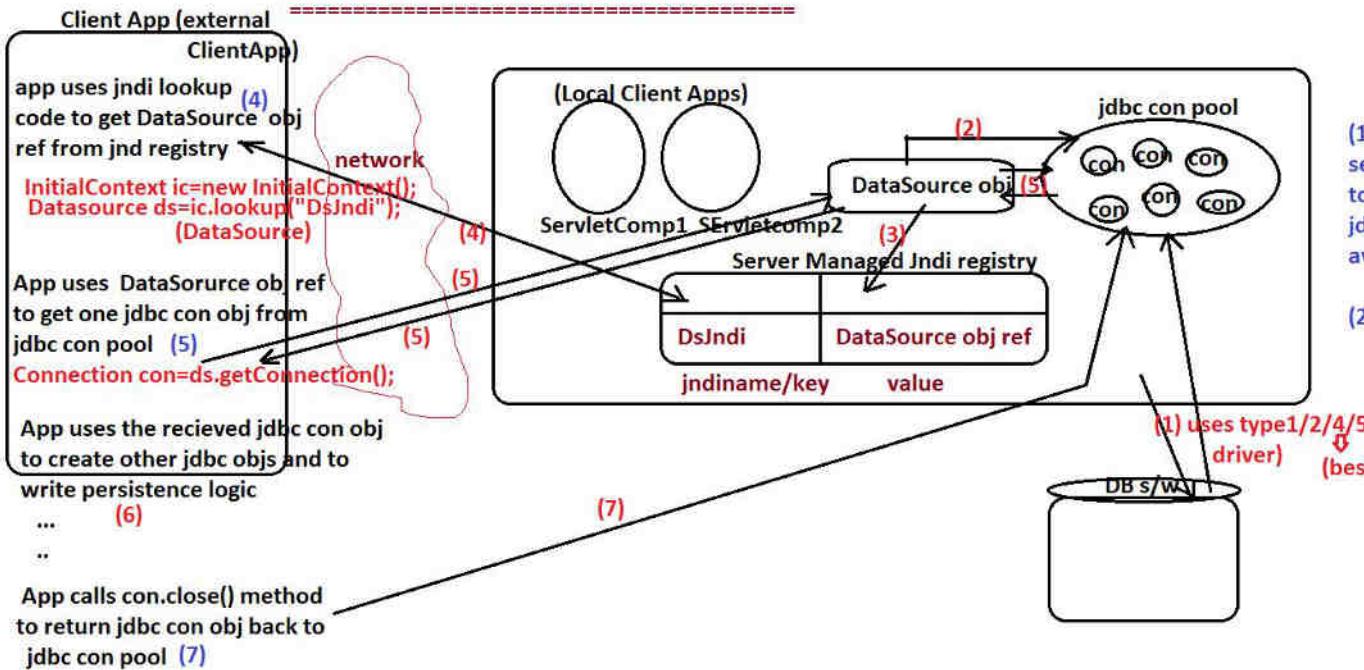
Bind Operation:: keeping obj/obj ref in the Jndi registry having nick name

UnBind Operation:: removing obj/obj ref in the Jndi registry having nick name

Lookup Operation:: Accessing obj/obj ref from Jndi registry

Rebind Operation:: Replacing existing object with new obj in the jndi registry

Understanding Server Managed jdbc con pool

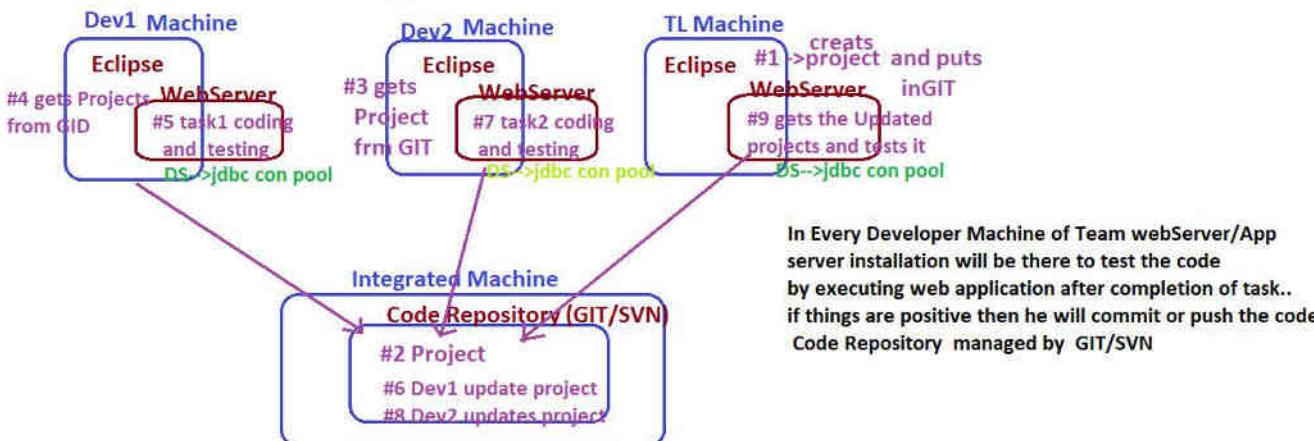


(1) TL/PL makes WebServer/Application server to use Type1/2/4/5 jdbc driver to interact with Db s/w and to create jdbc con pool having set of readily available jdbc con objects

(2) TL/PL make webServer/Application server to create DataSource pointing to jdbc con pool

(3) TL/PL keeps DataSource obj reference in jndi registry for global visibility having jndi name/nick name(DsJndi)

- =>DataSource obj means it is the obj java class that implements javax.sql.DataSource()
- => In One Server we can have multiple jdbc con pools either for same Db s/w or for different DB s/w and representing these multiple jdbc con pools , we need to take multiple Datasource objs
- =>All Jdbc con objs in jdbc con pool represents connectivity with same Db s/w..
- eg:: jdbc con pool for oracle means all the jdbc con objs in jdbc con pool represents connectivity with same oracle DB s/w
- eg:: jdbc con pool for mysql means all the jdbc con objs in jdbc con pool represents connectivity with same mysql DB s/w



Procedure to create jdbc DataSource obj pointing jdbc con pool for oracle in the Domain server of Weblogic.

step1) start weblogic domain server (WLNTAJ414domain)

```
use C:\Oracle\Middleware\Oracle_Home\user_projects\domains  
\WLNTAJ414Domain\startWeblogic.cmd file
```

step2) Open Admin console screen weblogic domainser ..

```
http://localhost:7070/console --> submit username :: javaboss  
                                password :: javaboss1
```

step3) Create JDBC dataSource pointing to jdbc con pool for oracle

Admin console --> sevices -->datasoruces -->new -->generic DataSoruce -->

DSname :: ds1 (logical name)

JndiName :: DsJndi

DB type : oracle -->next --> choose Oracle Thin driver for service connections -->next

db name: xe , host name: localhost , port number :: 1521 , db username:: system , db pwd ::manager -->next -->

Test DBconfiguration --> select Admin server --> finish.. go to jdbc con pool tab and specify con pool properties..

Initial Capacity:	<input type="text" value="5"/>
Maximum Capacity:	<input type="text" value="30"/>
Minimum Capacity:	<input type="text" value="10"/>

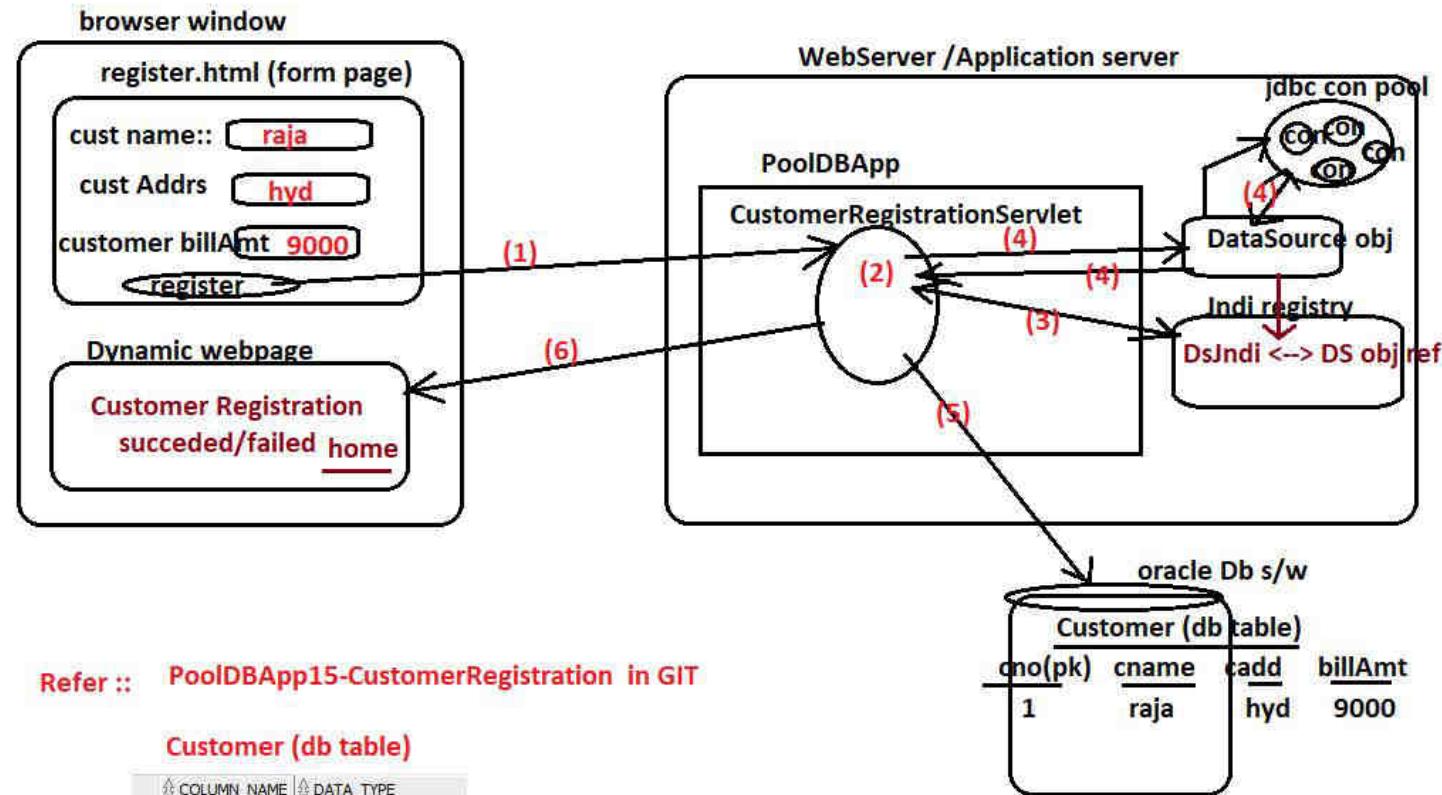
-->save ..

note:: The above process creates Datasource pointing to jdbc con pool for oracle and keeps Datasoruce obj ref in Jndi registry having Jndi name : DsJndi



Admin console --> enviroment --> servers --> admin server -->view Jndi Tree -->
search for DsJndi

App
Example on Servlet to DB s/w communication using Approach3 (Working with Jdbc con pool)

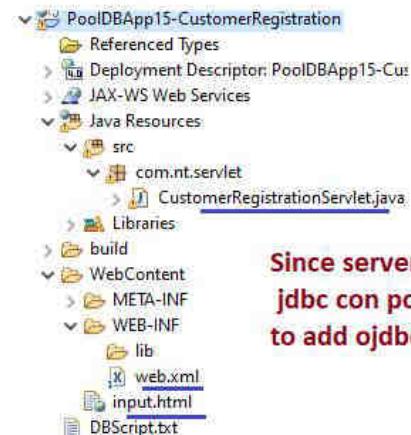


Refer :: PoolDBApp15-CustomerRegistration in GIT

Customer (db table)

COLUMN_NAME	DATA_TYPE
1 CNO (pk)	NUMBER(10,0)
2 CNAME	VARCHAR2(20 BYTE)
3 CADD	VARCHAR2(20 BYTE)
4 BILLAMOUNT	FLOAT

```
CREATE TABLE "SYSTEM"."CUSTOMER"
(
    "CNO" NUMBER(10,0) NOT NULL ENABLE,
    "CNAME" VARCHAR2(20 BYTE),
    "CADD" VARCHAR2(20 BYTE),
    "BILLAMOUNT" FLOAT(126),
    CONSTRAINT "CUSTOMER_PK" PRIMARY KEY ("CNO"));
```



Since server creates con objs in jdbc con pool, So we need not to add ojdbc6.jar file in WEB-INF/lib folder

Procedure to create jdbc con pool for oracle and dataSource in GlassFish domain server

=====

step1) copy odbc6/7/8/9/10.jar to <GlassFish_home>\domains\GFNTAJ414Domain\lib\ext folder

step2) start GlassFish domain server

E:\glassfish-5.0\glassfish5\glassfish\bin>asadmin start-domain GFNTAJ414Domain

step3) Open admin console of Domain server

http://localhost:4545 ---->
t
login username:: tesuser
login password :: testuser
-->login

step4) create jdbc con pool for oracle..

admin console --- resources --->jdbc --->jdbc con pools --->new --->
pool name ---> data source type :: java.sql.DataSource ----> DB Vendor : oracle --->

pool1

Pool Settings

Initial and Minimum Pool Size:	<input type="text" value="8"/> Connections	Minimum and initial number of connections maintained in the pool
Maximum Pool Size:	<input type="text" value="32"/> Connections	Maximum number of connections that can be created to satisfy client requests
Pool Resize Quantity:	<input type="text" value="2"/> Connections	Number of connections to be removed when pool idle timeout expires
Idle Timeout:	<input type="text" value="300"/> Seconds	Maximum time that connection can remain idle in the pool
Max Wait Time:	<input type="text" value="60000"/> Milliseconds	Amount of time caller waits before connection timeout is sent



Select	Name	Value
<input type="checkbox"/>	TNSEntryName	
<input type="checkbox"/>	Description	
<input checked="" type="checkbox"/>	User	system
<input type="checkbox"/>	MaxStatements	0
<input checked="" type="checkbox"/>	DatabaseName	xe
<input type="checkbox"/>	ImplicitCachingEnabled	false
<input type="checkbox"/>	NetworkProtocol	tcp
<input checked="" type="checkbox"/>	URL	jdbc:oracle:thin:@localhost:1521:xe
<input type="checkbox"/>	ConnectionCacheName	
<input type="checkbox"/>	DataSourceName	OracleDataSource
<input type="checkbox"/>	LoginTimeout	0
<input checked="" type="checkbox"/>	ServiceName	xe
<input type="checkbox"/>	ServerName	
<input type="checkbox"/>	ONSConfiguration	
<input checked="" type="checkbox"/>	DriverType	thin
<input checked="" type="checkbox"/>	PortNumber	1521
<input type="checkbox"/>	ExplicitCachingEnabled	false
<input checked="" type="checkbox"/>	Password	manager

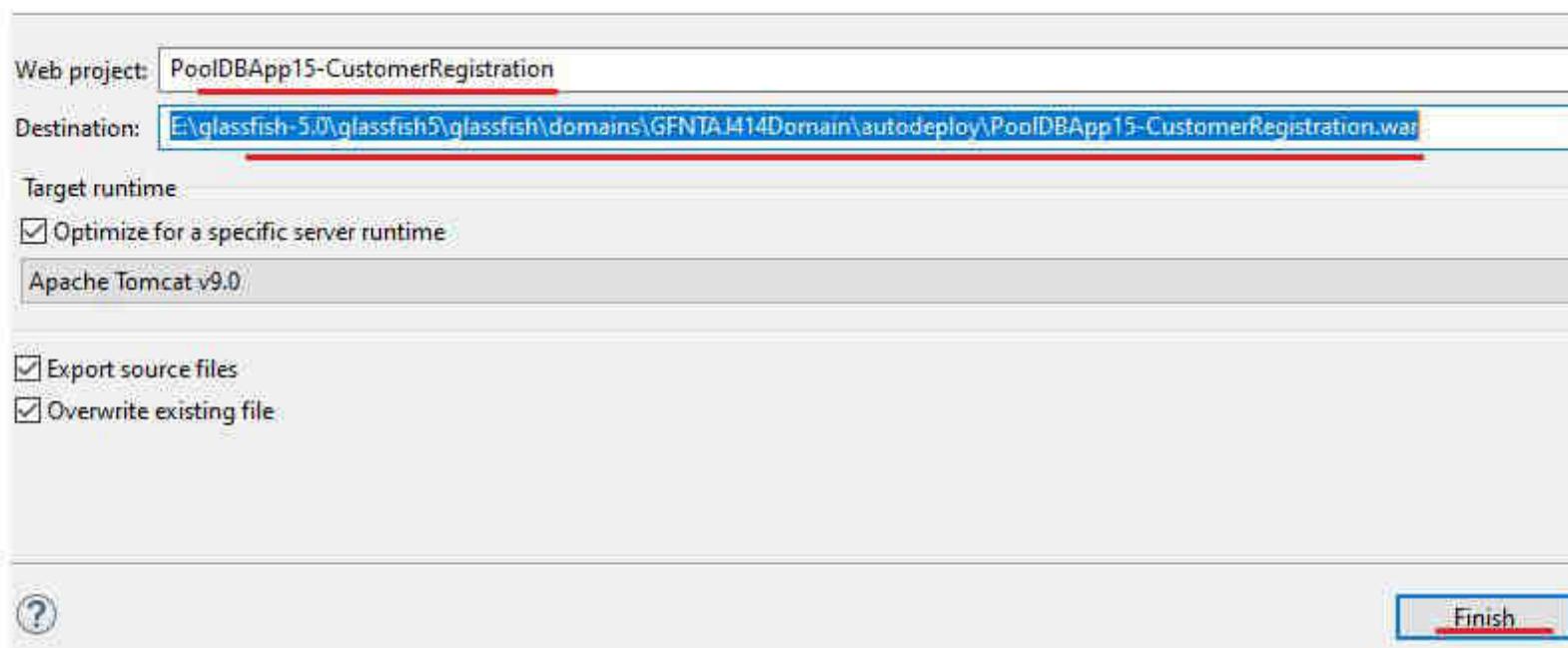


Finish ---> launch pool1 ---> ping --> u should get ping succeded.

step4) create Jdbc DataSource pointing to the the above jdbc con pool

admin console ----> resources ---->JDBC ---->JDBC Resources ---->
Jndi name:: DsJndi
pool name :: pool1
-->ok

step5) Run PoolDBApp15-CustomerRegistration app by exporting its war file GlassFish Domain server
Righ Click on Eclipse Dynamic web Project --->export ---> war file--->



To test the application::

<http://localhost:6677/PoolDBApp15-CustomerRegistration/input.html>

Procedure to create JDBC con pool for oracle and JDBC DataSource in Tomcat 9.x server

step1) make sure that tomcat-jdbc.jar and ojdbc6/7/8/9/10.jar files are added to <Tomcat_home>\lib folder
note:: From Tomcat7 tomcat-jdbc.jar file is coming along with Tomcat server installation..

step2) configure Tomcat server eclipse IDE once again through it is already configured..

step3) Collect <Resource> tag from Tomcat docs and place in <Tomcat_home>\conf\context.xml
(For external Tomcat server jdbc con pool) or place in Eclipse Project Explorer --> servers --> (for eclipse tomcat) Tomcat 9 --> context .xml file under <context> tag having entries related oracle jdbc driver ..

=>When Tomcat server is configured with Eclipse IDE, Eclipse creates its own copy Tomcat server in workspace.. So to create jdbc con pool, Ds in that Eclipse Tomcat use Project Explorer --> servers --> Tomcat -->context.xml file

=>To create jdbc con pool,Ds in external Tomcat server then use <Tomcat_home>\conf\context.xml

note:: When weblogic ,GlassFish servers are configured with Eclipse IDE.. then Eclipse does not its own copy those servers.. It still uses same Orginal Weblogic , GlassFish server..

note:: After developing and testing web application if u remove and add its server in Eclipse IDE .. then we need relink the web applicaiton with server

Right click on webProject ----> build path -->configure build path --> libraries tab --> add Library --> server run time -->select tomcat.

In context.xml (under <context> tag)

```
<Resource name="DsIndi" auth="Container"
    type="javax.sql.DataSource" driverClassName="oracle.jdbc.OracleDriver"
    url="jdbc:oracle:thin:@localhost:1521:xe"
    username="system" password="manager" maxTotal="20" maxIdle="10"
    maxWaitMillis="50000"/>
```

To get sample <Resource> from Tomcat docs

file:///E:/Tomcat%209.0/webapps/docs/jndi-datasource-examples-howto.html
(or)
<Tomcat_home>/webapps/docs/jndi-datasource-examples-howto.html

step4) restart the server..

step5) Run the PoolDBApp15-CustomerRegistration in Tomcat server of Eclipse by Run As -->server -->tomcat 9 optionn

note:: we need to add adtional pre-fixes for the Datasource Jndi name of Tocmat server that is java:/comp/env/DsIndi
Fixed prefix

Wildfly (Old JBoss server)

type :: Application server
vendor :: RedHat
version :: 2.1.x (compatible with jdk1.8+)
Open source
To download software :: <https://www.wildfly.org/downloads/>
default Ports :: 9990 (admin console)
8080 (for http requests) (changable)
=>Does not allow to create domains..
=>Allows only Console and Tools Based Deployment (No HardDeployment support)
To install wildfly server :: extract the zip file..

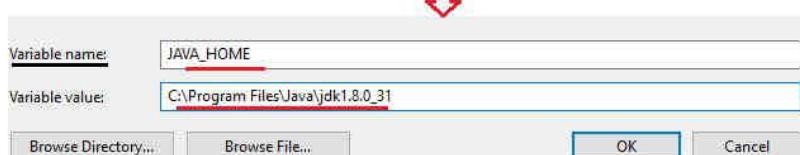
To change http port number of Wildfly server

Go to E:\WildFly21.0Soft\wildfly-21.0.2.Final\standalone\configuration\standalone.xml file
and modify <socket-binding name="http" port="\${jboss.http.port:8080}"/>
8899

To add admin user to Wildfly server

step1) make sure JAVA_HOME env.. variable created in my computer pointing to Java8 installation folder..

This PC /Mycomputer -->properties --> advanced system settings ---->
env.. variables --> system variables -->new



step2) Add User by <Wildfly_home>\bin\adduser.bat file



What type of user do you wish to add?
a) Management User (mgmt-users.properties) ✓
b) Application User (application-users.properties)
[a]:
Enter the details of the new user to add.
Using realm 'ManagementRealm' as discovered from the existing property files.
Username : testuser1
Password recommendations are listed below. To modify these restrictions edit the a
- The password should be different from the username
- The password should not be one of the following restricted values {root, admin,
- The password should contain at least 8 characters, 1 alphabetic character(s), 1
password : testuser1*
re-enter Password : testuser1*
...
...

Procedure to create JDBC DataSource pointing to JDBC con pool for oracle in Wildfly server

step1) make sure JAVA_HOME env.. variable created in my computer pointing to Java8 installation folder.. (refer above)

step2) start Wildfly server..

use E:\WildFly21.0Soft\wildfly-21.0.2.Final\bin\standalone.bat file

step3) open admin console of wildfly server..

<http://localhost:9990/> --> submit username :: testuser1
password :: testuser1*

step4) Deployojdbc6/7/8/9/10.jar to wildfly server to make it available in server..

Admin console -->Deployments --> (+)add --> upload deployment --> browser and select ojdbc6/7/8/9/10.jar file --> next --> finish -->close.

step5) Create DataSource pointing to jdbc con pool for oracle

Admin console -->configuration -->Datasoruce --> start --> subsystems -->
datasources & drivers --> data soruces --> (+)->add data soruce --> select oracle -->next-->
DataSoruce name:: oracleDs , jndi name:: java:/DsJndi1 --> enter jdbc properties
jdbc url:: jdbc:oracle:thin:@localhost:1521:xe
db username:: system
db password :: manager
next --> testConnection --> next -->Finish .

Procedure to deploy Java web application in Widfly server

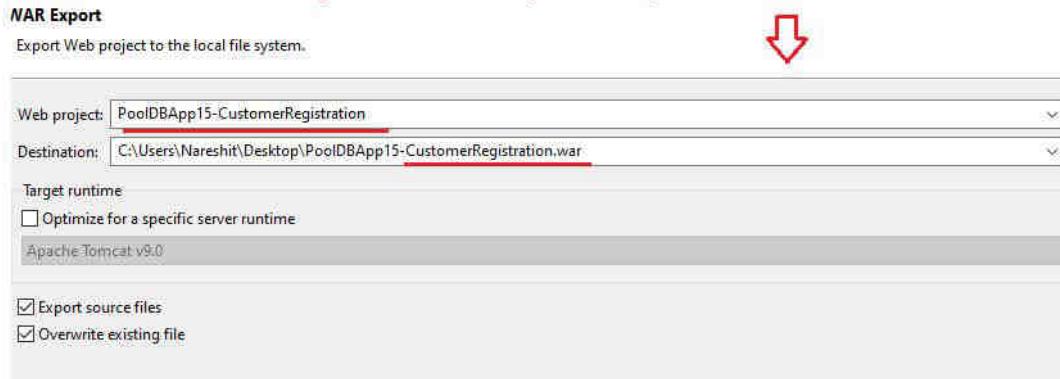
step1) Keep PoolDBApp15-CustomerRegistration App ready with modified jndi name:

java:/DsJndi1

step2) make sure that **fixed prefix** Java version of project is changed to jdk1.8

step3) prepare war file representing the web application..

Right click on the Project --> export -->war file -->



step4) Open admin console of server and deploy the war file..

Admin console ----> deployments --start ----> (+)add -->upload deployment --->
browse and select the war file ---> finish..

step5) Test the web applicaiton..

<http://localhost:8899/PoolDBApp15-CustomerRegistration>

Java Bean And usage::

<https://www.youtube.com/watch?v=io2TciOcF2Y>

VO class , DTO class and BO class

=====

<https://www.youtube.com/watch?v=GOvak6awDrl>

HttpUnit :: 11am to 12:30 noon

(old meeting id)

pre-requisites :: Junit5 and servlet basics

Java Bean

=> The java class that is developed by following some standards is called java bean.

The standards are

=>The java Class must taken as public class

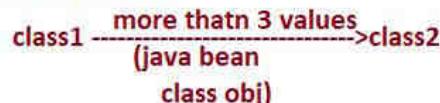
=> It is recomanded to implement Serializable(I)

=> All member variables must be taken as private and non-static (bean properties)

=> Every bean property must have 1 setter method and 1 getter method

=> we must have 0-param constructor directly(given by programmer) or indirectly(generated by javac)

note:: Java bean class should always be used as helper class to pass multiple values from 1 main class another main class.



Based on the kind of data we place in java bean , there are 3 types of java beans

a) VO class (Value object class)

-->Carries inputs or outputs

b) DTO class (Data transfer object class)

--> Carries data to transfer between two classes same project and between two classes of two different projects.

c) BO Class (Business object class)

--> Carries /holds persistable data (To be inserted to the db table as record)
or Persistent data (collected from db table record)

--> it will be taken as 1 per Db table basis, So there must be compatitable b/w BO class properties and Db table cols...

Limitation of placing multiple logics in single java class (single Servlet comp class)

=> since we mixup multiple logics by placing them in single comp.. Code becomes clumsy and clean separation b/w the logics will be missed.

=> The modifications done in one kind of logics will effect other kind of logics..

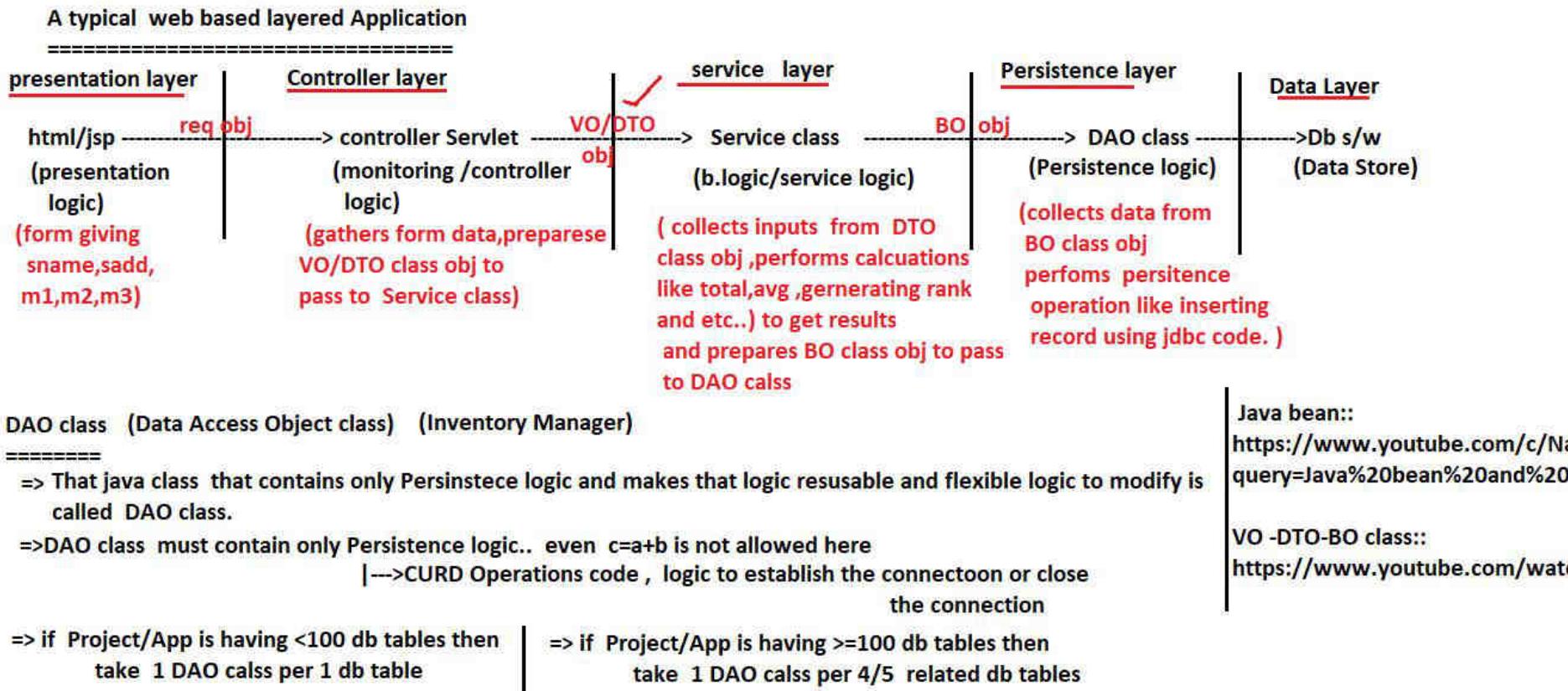
=> The maintainance(Bug fixing) and enhacement(new features) of the Project/App becomes very complex.

=> Parallel development is not possible .. So the productivity is very poor.

=> It is not industry standard approach to develop the s/w Applications..
and etc..

=>To overcome these problems , take support of layered applications .. Which separate different logics in different layers.

A Layer is logical partition of the App/project that contains one or more classes representing certain logic .



- => This contains two parts
 - a) query Part
 - =>All SQL Queries will be declared at top of class as private static final String variable values(String constants) in uper case letters for easy identification and modification.
 - b) Code Part
 - => Multiple methods representing multiple persistence logics (CURD Operations)
- => Generally gets input values from Service class in the form of BO class obj (More than 3 values)
- =>We can write Persistence logic by any Persistnece technology or framework like JDBC, Spring JDBC, Hibernate , spring ORM , Spring data and etc..

Java bean::

[https://www.youtube.com/c/NareshIT/search?
query=Java%20bean%20and%20its%20](https://www.youtube.com/c/NareshIT/search?query=Java%20bean%20and%20its%20)

VO -DTO-BO class::

<https://www.youtube.com/watch?v=GOvak6awDrI>

Service class (Accountant)

- =====
- => It is a java class that contains only b.logic /service logic (calculations, analyzations and etc..)
 - => It is generally 1 per module
 - => It also maintains additional services like TransactionManagement ,Security ,Logging and etc..
 - => gets inputs from Controller class in the form VO /DTO class obj (more than 3 values)

Controller class (Supervisor)

- =====
- => In web application env.. it is servlet comp .. having monitoring /supervising logics
 - => Contains logics to map /link certain html/jsp requests to certain service class and vice-versa.
 - => It is generally 1 per Project controlling or monitoring all the activities of the web application
 - => gathers inputs from html/jsp comps and passes them to Service class as VO /DTO class obj

htm/jsp comps (Beautician)

- =====
- => These comps contains presentation logic (logic to gather inputs from enduser and logic to display outputs to enduser)
 - => These can multiple in numbers (as per the need having forms ,hyperlinks to generate the requests to display the received outputs)

Hospital Project (1 controller servlet/class)

- |--->InPatient (module1)
- |--->OutPatient (module2)
- |---> Payroll and Finance (module3)
- |---> Pathology(Labs) (module4)
- |---> Medicine (module5)
- |---> BloodBank (module6)
- |---> OT services (module7)

7 service classes

=>While Layer developing apps , we need to take

a) DAO Interface and Impl class , So that we can write diffrent logics in different impl classes

|--->DAO Impl class1 having JDBC persistence logic

|--->DAO Impl class2 having Hibernate Persinstence logic

|--->DAO Impl class3 having spring JDBC Persinstence logic

b) Service Interface and Impl class , So that we can write different logics in different impl classes

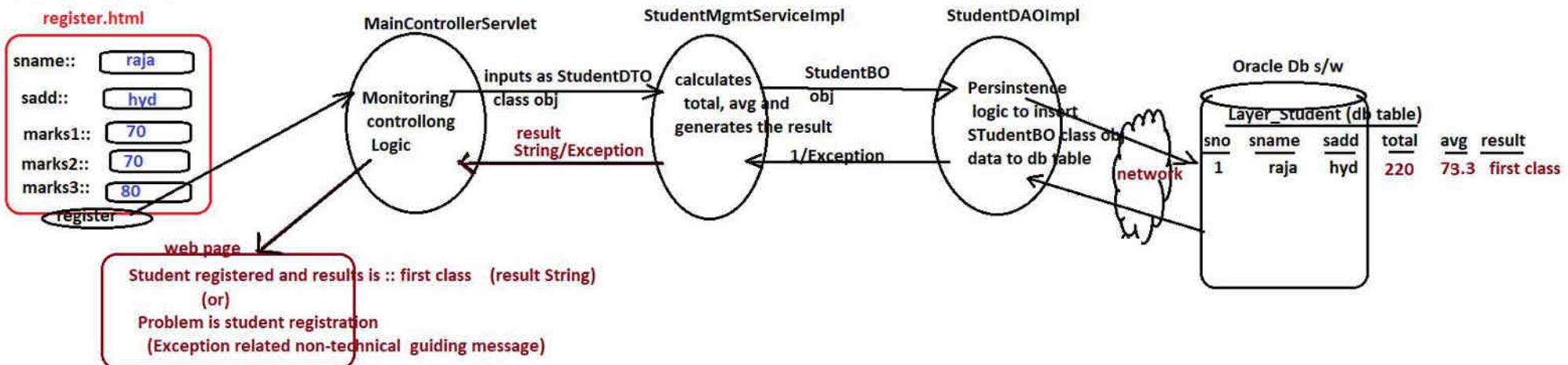
|---> service Impl class1 having b.logic by using algorithm1

|---> service Impl class1 having b.logic by sing algorithm2

Based on Core java's
run time polymorphism
or Loose coupling concept.

c) Controller Servlet is already implementing javax.servlet.Servlet(I) indirectly and it is
one per Project , So there is no need of taking seperate interface ..

Example Application



How do u pass/propagate the exception raised in DAO class to ControllerServlet/browser?

Ans) Do not just catch and handle exception in DAO class usin try /catch block becoz when
exception is raised the controller goes catch block and the exception will be eaten/suppressed there..

DAO class

```
public int insert () {  
    try{  
        ...  
        ... //jdbc code to insert the record  
        ...  
    }  
    catch(Exception e){  
        ...  
    }  
}//insert() method
```

In DAO ,service classes do not catch and handle exception using try/catch.. always prefer declaring the exception to be thrown using "throws" clause , so the exception raised in DAO class method will be passed to Service class method and will be passed Controller Servlet .. then controller Servlet comp catches and handles the exception using try /catch and displays exception related non-technical messages to enduser using pw.println(-) method.

(note:: PrintWriter (pw) is not visible /creatable in Service,DAO calss.. it can be used only used in servlet comp through response obj. PrintWriter pw=res.getWriter())

Exception propagation

DAO class code

```
p int insert() throws Exception{  
... //jdbc code  
...  
}
```

service class code

```
public String generateResults() throws Exception{  
    int count=dao.insert();  
    ....  
}
```

controller sevlet code

```
public void doGet/doPost(..., ...) throws ServletException, IOException{  
    ....  
    try{  
        String result=service.generateResults();  
        pw.println(result);  
    }  
    catch(Exception e){  
        pw.println("Problem in student registration");  
    }  
}
```

DAO and service classes are ordinary java classes , So they do not have response obj to create PrintWriter i.e they can not write messages to browser directly.

Stroy board of our Layed applicaton

(Persistnece laeyer)

IStudentDAO.java

```
public interface IStudentDAO{  
    public int insert(StudentBO bo) throws Exception;  
}
```

StudentDAOImpl.java

```
public class StudentDAOImpl implements IStudentDAO{  
  
    public int insert(StudentBO bo) throws Exception{  
        .... //read data from BO class obj  
        .... // and write that data to Db table  
        .... // as record using JDBC code  
  
        ....  
        return 1/0;  
    }  
}
```

Stroy board of our Layed applicaton (Persistnece laeyer)

IStudentDAO.java

```
public interface IStudentDAO{  
    public int insert(StudentBO bo) throws Exception;  
}
```

StudentDAOImpl.java

```
public class StudentDAOImpl implements IStudentDAO{  
  
    public int insert(StudentBO bo) throws Exception{  
        .... //read data from BO class obj  
        .... // and write that data to Db table  
        .... // as record using JDBC code  
  
        ....  
        return 1/0;  
    }  
}
```

Service layer

IStudentMgmtService.java

```
public interface IStudentMgmtService{  
  
    public String generateResult(StudentDTO dto) throws Exception;  
}
```

note:: if class is not having state (no member variables) or read only
then it is recomanded to create only 1 object for that class and use it
for multiple times to avoid memory wastage.. (Fixed)

=>Generally DAO ,Service classes contain either no state or
fixed state .. So we prefer creating only one object for those
java classes.

StudentMgmtServiceImpl.java

```
public class StudentMgmtServiceImpl implements IStudentMgmtService{  
  
    private IStudentDAO dao;  
  
    public StudentMgmtServiceImpl(){  
        dao=new StudentDAOImpl();  
    }  
  
    public String generateResult(StudentDTO dto) throws Exception{  
        int total=dto.getM1()+dto.getM2()+dto.getM3();  
        float avg=total/3.0f  
        String result=null;  
        if(avg<35)  
            result="fail";  
        else if(avg<50)  
            result="third Grade";  
        else if(avg<60)  
            result="Second Grade";  
        else  
            result="First Grade";  
  
        //prepare StudentBO class obj having sname,sadd , total,avg,result  
        StudentBO bo=new StudentBO();  
        bo.setSname(dto.getSname());  
        bo.setSadd(dto.getSadd());  
        bo.setTotal(total); bo.setAvg(avg); bo.setResult(result);  
  
        //use DAO  
        int count=dao.insert(bo);  
        if(count==0)  
            return "studnet registration failed";  
        else  
            return "studnet registration succeded";  
    }  
}
```

b.logic

Service layer

IStudentMgmtService.java

```
public interface IStudentMgmtService{  
    public String generateResult(StudentDTO dto) throws Exception  
}
```

note:: if class is not having state (no member variables) or read only
(Fixed)
then it is recommended to create only 1 object for that class and use it
for multiple times to avoid memory wastage..

=>Generally DAO ,Service classes contain either no state or
fixed state .. So we prefer creating only one object for those
java classes.

StudentMgmtServiceImpl.java

```
public class StudentMgmtServiceImpl implements IStudentMgmtService{  
    private IStudentDAO dao;  
    public StudentMgmtServiceImpl(){  
        dao=new StudentDAOImpl();  
    }  
    public String generateResult(StudentDTO dto) throws Exception{  
        int total=dto.getM1()+dto.getM2()+dto.getM3();  
        float avg=total/3.0f  
        String result=null;  
        if(avg<35)  
            result="fail";  
        else if(avg<50)  
            result="third Grade";  
        else if(avg<60)  
            result="Second Grade";  
        else  
            result="First Grade";  
        //prepare StudentBO class obj having sname,sadd , total,avg,result  
        StudentBO bo=new StudentBO();  
        bo.setSname(dto.getSname());  
        bo.setSadd(dto.getSadd());  
        bo.setTotal(total); bo.setAvg(avg); bo.setResult(result);  
        //use DAO  
        int count=dao.insert(bo);  
        if(count==0)  
            return "student registration failed";  
        else  
            return "student registration succeeded";  
    }  
}
```

b.logic

Controller Servlet (Controller layer)

```
#4
public class MainControllerServlet extends HttpServlet{

    private IStudentMgmtService service;
    public void init(){
        service=new StudentMgmtServiceImpl();
    }
    #7
    public void doGet(req,res) throws SE,IOE{
        //read form data
        ...
        //create StudentDTO class obj having form data or inputs
        ...
        try{
            // use service      #8
            #15   String result=service.generateResult(dto);
            pw.println("RESULT::"+result); #16
        }
        catch(Exception e){
            e.printStackTrace();
            pw.println("Problem is student registration");
        }
    }//doGet(--)

    #5
    public void doPost(req,res) throws SE,IOE{
        doGet(req,res); #6
    }
}//class
```

Form page (student_register.jsp)

```
=====
#2
<form action="controller" method="POST">
    ...
    ... //form comps
    ...
    <input type="submit" value="register"/>
</form>      #1
```

=====

web.xml
=====

```
=> cfg MainControllerServlet with url pattern /controller
by enabling <load-on-startup> #3
```

=>The code placed in static block of the class is one-time executing block having the following limitations

- a) we can use only static variables in the static block
- b) Exception is not possible from static block

=>if web.xml is not coming along with dynamic web project creation.. then we can generate it seperately by using Eclipse Options..

Right click on project --> JEE tools --->Generate Deployment Descriptor Stub.

=>No Sequences in support in mysql.. In order generate PK col dynamically , we need to take the support of **auto increment constraint**.

```
CREATE TABLE `ntaj414db`.`layered_student` (
  `sno` INT NOT NULL AUTO_INCREMENT,
  `sname` VARCHAR(20) NULL,
  `sadd` VARCHAR(20) NULL,
  `total` INT NULL,
  `avg` FLOAT NULL,
  `result` VARCHAR(20) NULL,
  PRIMARY KEY (`sno`),
  UNIQUE INDEX `sno_UNIQUE` (`sno` ASC) VISIBLE;
```

=>To create JDBC Data Source pointing to JDBC con pool for mysql in tomcat server, we need to add the following <Resource> tag in Context.xml file

```
<Resource name="DsIndi-mysql" auth="Container"
  type="javax.sql.DataSource" driverClassName="com.mysql.cj.jdbc.Driver"
  url="jdbc:mysql://localhost:3306/NTAJ414db"
  username="root" password="root" maxTotal="20" maxIdle="10"
  maxWaitMillis="50000"/>
```

=>jar files added to <Tomcat_home>\lib folder (external Tomcat server) will reflect to Eclipse Tomcat server automatically if they were added before configuring Tomcat server with Eclipse otherwise they will not reflect. In that situation , we need to add jars files to the Project's WEB-INF/lib folder seperately..

In Layered Apps, it is recommended to enable <load-on-startup> on servlet comp.. to perform eager instantiation on Servlet comp class ,service class and DAO class.

Assignment ::Employee Registration Layered app to calculate gross,netSalaries based on the given ename,eadd,basicslary details..

Annotation driven servlet mapping

```
//@WebServlet(value="/testurl") ✓  
//@WebServlet(value={"/testurl1","/testurl2"}) ✓  
//@WebServlet({"/testurl1","/testurl2"}) ✓  
//@WebServlet(urlPatterns={"/testurl1","/testurl2"}) ✓  
//@WebServlet(name="test", value="/testurl", loadOnStartup=1) ✓  
@WebServlet(name="test", value={"/testurl1","/testurl2"}, loadOnStartup=1) ✓  
public class TestServlet extends HttpServlet{  
    ....  
    ....  
}
```

If we configure Servlet comp having both annotation driven and xml driven servlet mappings then what happens?

Ans) => if annotation driven and xml driven servlet mappings are taken with the same logical name
then xml driven servlet mapping url pattern will dominate annotation driven mappings related url pattern.

=> if annotation driven and xml driven servlet mappings are taken with two different logical names
then that Servlet comp can be requested using both annotation driven and xml driven mappings related url patterns.

=> In annotation driven servlet mapping, if do not provide logical name then the fully qualified class name of servlet comp class will become default logical name..

=> In xml driven servlet mapping, if do not provide logical name in <servlet-name> tag then we get errors..

note:: the logical name of servlet internally becomes object name (refer variable name)
when Servlet container creates object for servlet comps..

```
@WebServlet(urlPatterns={"/testurl1","/testurl2"},name="test1",loadOnStartup = 1)  
public class TestServlet extends HttpServlet { ... }
```



```
Class c=Class.forName("com.nt.servlet.TestServlet");  
TestServlet test1=(TestServlet)c.newInstance();  
object name(reference variable name)
```

To know Servlet comp class obj name (logical name) dynamically being from servlet comp class
then we can use.

```
Servletconfig cg=getServletConfig();  
pw.println("<br><b> logical name of servlet comp is ::"+cg.getServletName()+"</b>");
```

note: if we configure Servlet comp having both xml driven and annotation driven different url patterns
the settings done in web.xml for servlet comp takes place if we give the request using xml driven url pattern,
Similarly the settings done in annotations for servlet comp takes place , if we give the request using annotation driven url patterns..

Q) What happens if we give same url pattern in xml driven cfgs and annotation driven cfgs?

ans) if logical name is same in both places then annotation driven cfgs takes place otherwise error will be raised.

=>Using @WebInitParam annotation, we can pass input values to Servlet comp as servlet init param values.. But we do not have the flexibility of modifications becoz we are placing them directly in the source code of servlet comp...

```
@WebServlet(name="test1",urlPatterns = "/testurl1") initParams = {@WebInitParam(name="p1",value="val1"),
@WebInitParam(name="p2",value="val2")
})  
public class TestServlet extends HttpServlet {  
....  
....  
}
```

note:: if want overide or modify the Annotation driven cfgs , with out touching the source code of servlet comp then we can write same cfgs with different values using web.xml entries.

Can we avoid web.xml in servlet based web application development?

Ans) yes we can avoid.. in few situations.. but not recommended becoz the still web.xml is required for following cfgs as annotations are not given for the same cfgs.

- a) context params
 - b) welcome cfgs
 - c) Security cfgs
 - d) Session Mgmt cfgs
- and etc..

(still we need to use
web.xml for all these
cfgs)

=>Thumb rule in servlet programming

=> First prefer configure things by using Annotations.., if not possible then go for web.xml based cfgs.

Servlet 3.0+ onwards we are having supports for annotations

The List of annotations are

- a) @WebServlet
- b) @WebFilter
- c) @WebListener
- d) @WebInitParam
- e) @MultipartConfig
- f) @HttpMethodConstraint
- g) @HttpConstraint
- h) @HandleTypes

related
security cfgs.

File uploading & downloading

uploading

--> It is the process of selecting file from client machine (browser machine) file system and sending that file server machine (webSErver) file system

downloading

-->Reverse is called downloadng....

=>Usecase:: MartriMony App, Social networking Apps, job portal Apps, profile mgmt app..

For file uploading

(a) take file uploading file comp

```
<input type="file" name="f1">
```

(b) take form page having method="POST" and enctype="multipart/form-data"

To send unlimited
data along with
request

To send diff MIME types
of content along with the
request.

(c) use req.getInputStream() to read the uploaded file content..

```
InputStream is=req.getInputStream();
OutputStream os=new FileOutputStream("dest_file.txt");
IoUtils.copy(is,os);
```



- (a) we can not get the name of uploaded file ,So we can not save file on server machine file system with the same name
- (b) Size restrictions, overriding restrictions, type restrictions should be handled manually.
- (c) Still commons-io-<ver>.jar is required as third party api

To simply this process, we can work with Third party api called **javazoom api**

javazoom api jar files (javazoom.net)

=====

uploadbean.jar (main jar file)

stuts.jar

cos.jar (Dependent jar files)

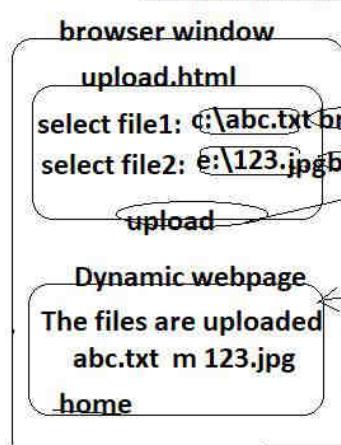
note:: uploadbean.jar classes are using the classes and interfaces of cos.jar,struts.jar

api

if servlet comp uses third party ,then that third party api related main jar file like **uploadbean.jar** should be added to classpath but we need both main and dependent jar files (3 jar files) to WEB-INF/lib folder...

Client Machine File System

server Machine File System



c:\abc.txt
e:\123.jpg

important classes in java zoom api

(a) MultiPartFormDataRequest

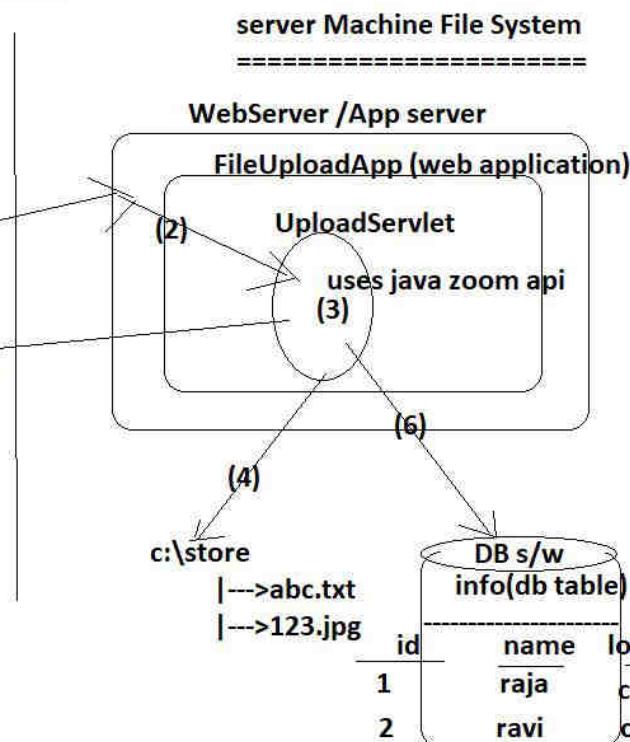
--> wrapper around request obj holding both text data and binary data..
(uploaded files)

(b) UploadBean

- |-->To perform file uploading
- |-->To specify destination folder
- |-->To specify size,type and overriding restrictions

(c) UploadFile

|-->To get information each uploaded file like name,size ,type and etc..



```

FileUploadApp (EDWP)
|-->java resources
    |---->src
        |---->com.nt.servlet
            |---->UploadServlet.java
|---->webcontent
    |---->upload.html
    |---->WEB-INF
        |---->lib
            |---->*.jar (javazoom apy 3 jars)
|---->web.xml

```

place uploadbean.jar in
build path of project..

important classes of javazoom api

(a) MultipartFormDataRequest

=>Wrapper around request object holding the uploaded files

(b) UploadBean

==>completes teh file uploading by slecting dest folder, restrictions and etc..

(c) UploadFile

==>Represents each uploaded file..

map (Hashtable object)	
ephoto	UploadFile object
eresume	UploadFile object

CREATE SEQUENCE "SYSTEM"."ENO_SEQ" MINVALUE 1 MAXVALUE 10000 INCREMENT BY 1 START WITH 41 CACHE 20 NOORDER NOCYCLE ;

File downloading

(a) response downloading (it is not regularly used one)

--->Here the response of the web comp will be downloaded to the browser as file

---> add this code in servlet/jsp comp to make its output/response as downloadable file

eg:: res.addHeader("Content-Disposition","attachment;fileName=abc.txt");



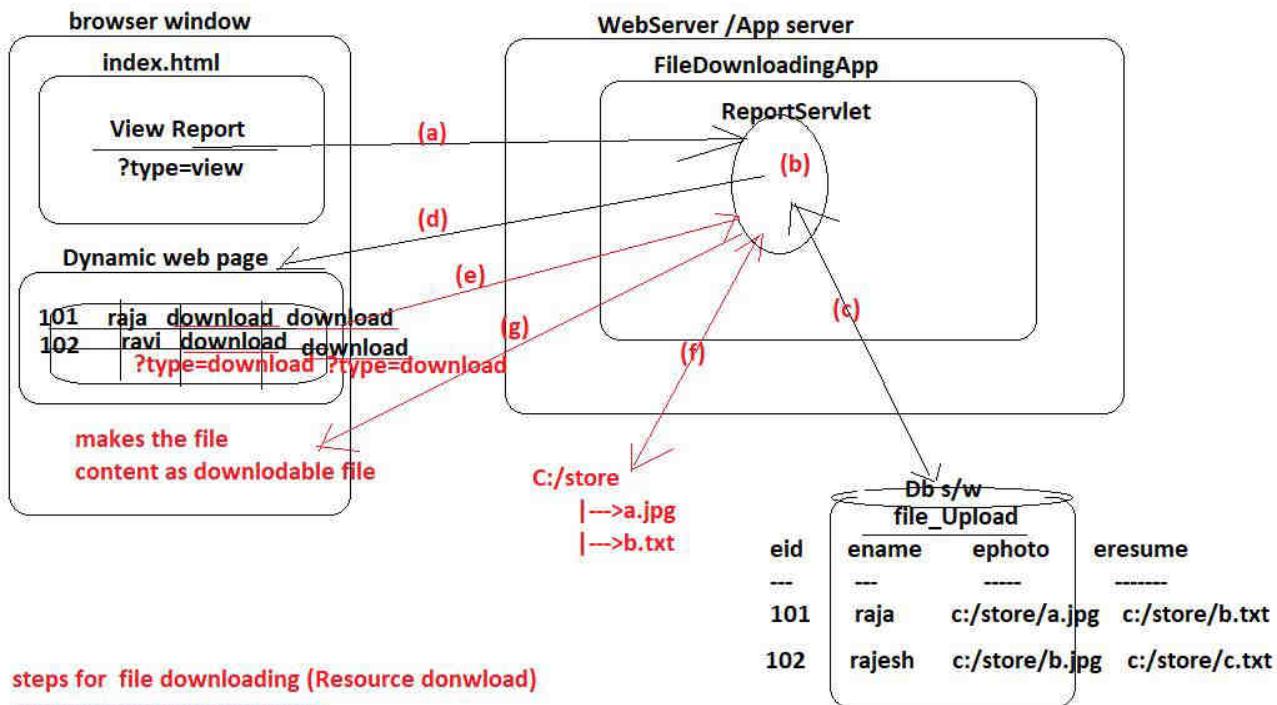
Gives Instruction to browser to make the received output as downloadable text document (abc.txt")

if the header value is "attachment" ---> the output of web comp
will become downloadable file

if the header value is "inline" (default) ---> the output of web comp
will come on browser itself..

(b) Resource downloading (very much used in realtime)

Here the files in server machine file system which already there, were uploaded will be downloaded like pdf, software, audio ,vedio files downloading..



steps for file downloading (Resource download)

(a) Create java.io.File class object to locate the file and to get the length of the file

```
File file=new File("c:/store/b.txt");
int lenght=file.length();
```

(b) set file lenght as response content length
res.setContentLength(lenght);

(c) set file MIME type as response content type
ServletContest sc=req.getServletContext();
String mimeType=sc.getMimeType(file.getAbsolutePath());
res.setContentType(mimeType!=null?mimeType: "application/octet-stream");

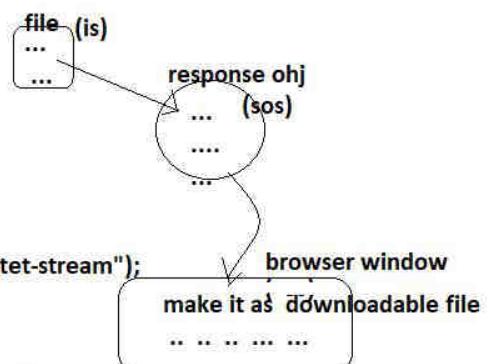
(d) create InputStream pointing to file
InputStream is=new FileInputStream(file);

(e) gives downloading instruction to browser using "Content-Disposition"

```
res.addHeader("Content-Disposition", "attachment;filename=file.getName());
```

(f) create OutputStream pointing to response
ServletOutputStream sos=res.getOutputStream();

(g) copy file content to response object.
IoUtils.copy(is,sos); (add commons-io.jar file to build path)



```
(h) close streams  
    is.close();  
    sos.close();
```

Pointing response object we can take two streams

(a) PrintWrtier

==>It is character stream
==> Good for writing text data to response object
eg:: PrintWriter pw=res.getWriter(); pw.println(" hello ");

(b) ServletOutputStream

==>It is byte stream
==>Good for writing both text data and binary data..
eg:: ServletOutputStream sos=res.getOutputStream();
 sos.println(" hello ");

note:: we can not both at a time .. if one stream is take we get
error for other stream..

Procedure to develop App using gradle

=====

step1) make sure that Build ship plugin is installed ... in Eclipse (built-in plugin)

ste2) create Gradle Project and convert it web application....

```
File -->project -->gradle --> gradle project -->  
    project name :: FileDownloadApp -->  
    select gradle wrapper --> select auto sync --> .... (gives standalone project)
```

right click on the project -->properties --> Project facets --> select dynamic web module..

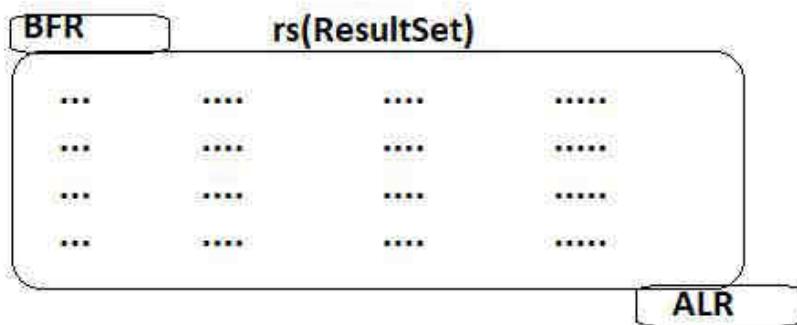
step3) write following code .. in the build.gradle

```
build.gradle  
=====  
plugins {  
    id 'war'  
}  
  
repositories {  
    mavenCentral()  
}  
  
dependencies {  
    // https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api  
    implementation group: 'javax.servlet', name: 'javax.servlet-api', version: '4.0.1'  
  
    // https://mvnrepository.com/artifact/org.apache.commons/commons-io  
    implementation group: 'org.apache.commons', name: 'commons-io', version: '1.3.2'  
  
    // https://mvnrepository.com/artifact/com.oracle.database.jdbc/ojdbc6  
    implementation group: 'com.oracle.database.jdbc', name: 'ojdbc6', version: '11.2.0.4'  
  
}
```

step4) create packages in src/main/java folder ...

In Maven/gradle.. the jar files to added to classpth/build
path will copied to WEB-INF/lib folder dynamically...

```
FileDownLoadApp
|--->src/main/java
    |--->com.nt.servlet
        |---->ReportServlet.java
|--->webcontent
    |---->index.html
    |---->WEB-INF
        |-->web.xml (create it manually)
|--->build.gradle
```



note:: In new versions of Gradle the webcontent folder of Gradle web application is not reflecting in Deployment directory structure by default.. So use deployment assembly option as shown below..

Right click on Project ---> properties ---> Deployment assembly ---> Add -->folders---> select webcontent-->

Different types of servlet url patterns

=>We keep servlet comp private area of the web application.. and we map with url pattern /url in web.xml or using @webServlet.. to make servlet container allowing the browser generated request to servlet comp.

According to servlet specification we can have 3 types of url patterns

- a) Exact Match url pattern
- b) Directory match url pattern
- c) Extension Match url pattern

Exact Match url pattern

=>This url pattern must begin with "/" and should not contain "*" .. Can have multiple words seperated with "/" ..

eg: <url-pattern> /test1 </url-pattern> (or) @WebServlet("/test1")

request urls (browser window)

http://localhost:3030/TestApp/test1 (valid)
http://localhost:3030/TestApp/test1.c (invalid)
http://localhost:3030/TestApp/abc/test1 (invalid)
http://localhost:3030/TestApp/test1/test1 (invalid)
http://localhost:3030/TestApp/test1/test2/test3.c (invalid)

note:: while working with exact url pattern
there will be only one valid request url requesting
that servlet comp..

note: So far we worked with
Exact match url pattern

url patterns help programmers to hide the technology web application from outsiders..

Can we provide multiple urls /url patterns to Servlet comp?

Ans) Yes, we can do

```
<servlet-mapping>
    <servlet-name>html</servlet-name>
    <url-pattern>/hturl</url-pattern>
    <url-pattern>/hturl1</url-pattern>
</servlet-mapping>
```

(or)

@WebServlet({"/hturl","/hturl1"}) (or)

@WebServlet(value={"/hturl","/hturl1"}) (or)

@WebServlet(urlPatterns={"/hturl","/hturl1"})

other example exact match url patterns

```
=====  
eg1: <url-pattern>/test1/test2 </url-pattern> (or) @WebServlet("/test1/test2")  
eg2: <url-pattern>/test1/test2.c </url-pattern>(or) @WebServlet("/test1/test2.c")  
eg3: <url-pattern>/test1.cpp </url-pattern> (or) @WebServlet("/test1.cpp")
```

Directory Match url pattern

```
=====  
This url pattern must start with "/" and should end with "*" symbol
```

```
eg:: <url-pattern>/x/y/*</url-pattern>
```

request urls from browser window

```
=====  
http://localhost:2020/TestApp/x/y/test1 (valid)  
http://localhost:2020/TestApp/x/y/test1.cpp (valid)  
http://localhost:2020/TestApp/x/y/a/b/test1.x (valid)  
http://localhost:2020/TestApp/x/y/abc.do (valid)  
http://localhost:2020/TestApp/x/test1.c (invalid)  
http://localhost:2020/TestApp/a/b/x/y/test1.c (invalid)  
http://localhost:2020/TestApp/y/x/test1 (invalid)  
http://localhost:2020/TestApp/x/y/ (valid)
```

Directory

other match url patterns

```
=====  
eg1: <url-pattern>/x/* </url-pattern>  
eg2: <url-pattern>/test1/* </url-pattern>  
eg3: <url-pattern>/test1/test2/* </ur-pattern>  
eg4: <url-pattern>/x.c/y.d/* </url-pattern>
```

Extension Match Url pattern

```
=====  
syntax:: *.extesion>
```

```
eg: <url-pattern>*.do </url-pattern> --> @WebServlet("*.do")
```

request urls from the browser window

```
=====  
http://localhost:2020/TestApp/abc.do (valid)  
http://localhost:2020/TestApp/abc.x (invalid)  
http://localhost:2020/TestApp/123/456/abc.z (invalid)  
http://localhost:2020/TestApp/123/456.do/abc.do (valid)  
http://localhost:2020/TestApp/.do (valid)  
http://localhost:2020/TestApp/123/456.do/abc.xyz (invalid)  
http://localhost:2020/TestApp/.do/.do (valid)
```

other examples of extension match url pattern

eg1: <url-pattern> *.c </url-pattern>
eg2: <url-pattern> *.d </url-pattern>
eg3: <url-pattern> *.naresh </url-pattern>
eg1: <url-pattern> *.123 </url-pattern>

Find out the type of url pattern?

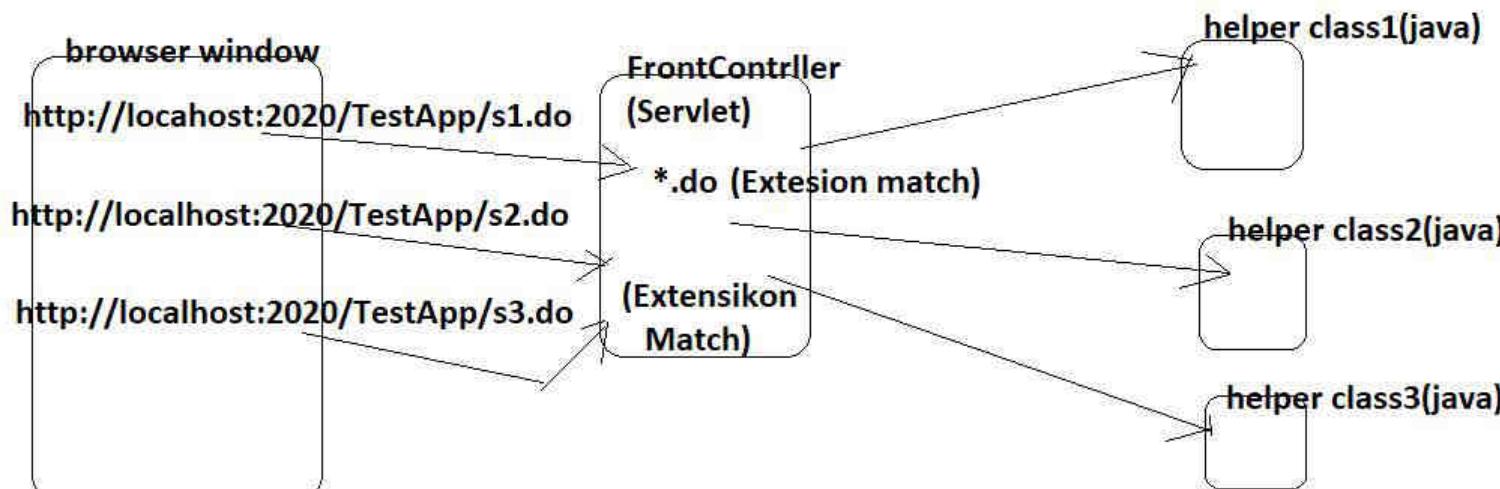
<url-pattern>/test1/test2/test3.xyz </url-pattern> (Exact match url pattern)

Find out the url pattern type?

<url-pattern>/x/y/*.c </url-pattern> (invalid formation of url pattern)

FrontController Servlet

The special servlet comp in the webapplication .that traps all the requests, apply common system services and delegates the controller to appropriate java classes...i.e this servlet should take diff request ruls based requests.. For that this FrontController servlet comp should be mapped directory match or extension match url pattern...



If clash /ambiguity comes among the 3 types url patterns based servlet comps then the Servlet container gives priority in the following order

- 1)Exact Match
- 2)Directory Match
- 3) Extension Match

Servlet1 url pattern	/test1/test2/123.c	(Exact match url pattern)
Servlet2 url pattern	/test1/test2/*	(Directory match url pattern)
Servlet3 url pattern	*.c	(Extension match url pattern)

request url ::: http://localhost:2020/TestApp/test1/test2/123.c

This request goes to Servlet1 comp
as Exact match url pattern gets high priority.

what happens if multiple servlet comps are having same url pattern?

=> url patterns to servlet comp must be unique with in the web application.. So exception will be raised..

Ans) java.lang.IllegalArgumentException: The servlets named [html] and [plain] are both mapped to the url-pattern [/test1] which is not permitted

what happens if multiple servlet comps are having same logical name?

Ans) The Last Servlet comp in that list according to the cfg done in web.xml file will execute.. and that servlet comp can be requested with multiple url patterns..

internalCache	
html	PlainServlet.class obj

if it is annotation env.. only one Servlet comp in that list will execute and remaining servlet comps will 404 error.. That one servlet comp will be picked up based Alphabetic order (whose name starts with less alphabet-asdening)

Can we configure Servlet comp with out url pattern?

(or)

Is any there any default url pattern for Servlet comp?

Ans) We can not configure servlet comp with out url pattern... i.e there is not default url pattern for servlet comp... So we must map servlet comp with url pattern...

Can we can cfg servlet comp with out logical name? (or)

Is there any default logical name for servlet com?

=> if cfg servlet comp in web.xml with out logical name then error will come.. if we do same by using annotation (@WebServlet) then the fully qualified servlet class name will be taken as default logical name..

```
@WebServlet(value="/test1")
public class PlainServlet extends HttpServlet{
    ...
    ...
}
```

here "com.nt.servlet.PlainServlet" class name will be taken as logical name..

What is Default Servlet Comp?

Ans) The servlet comp that executes for any invalid url given in the browser address bar is called Default Servlet .. Generally this servlet comp will be configured with "/" url pattern

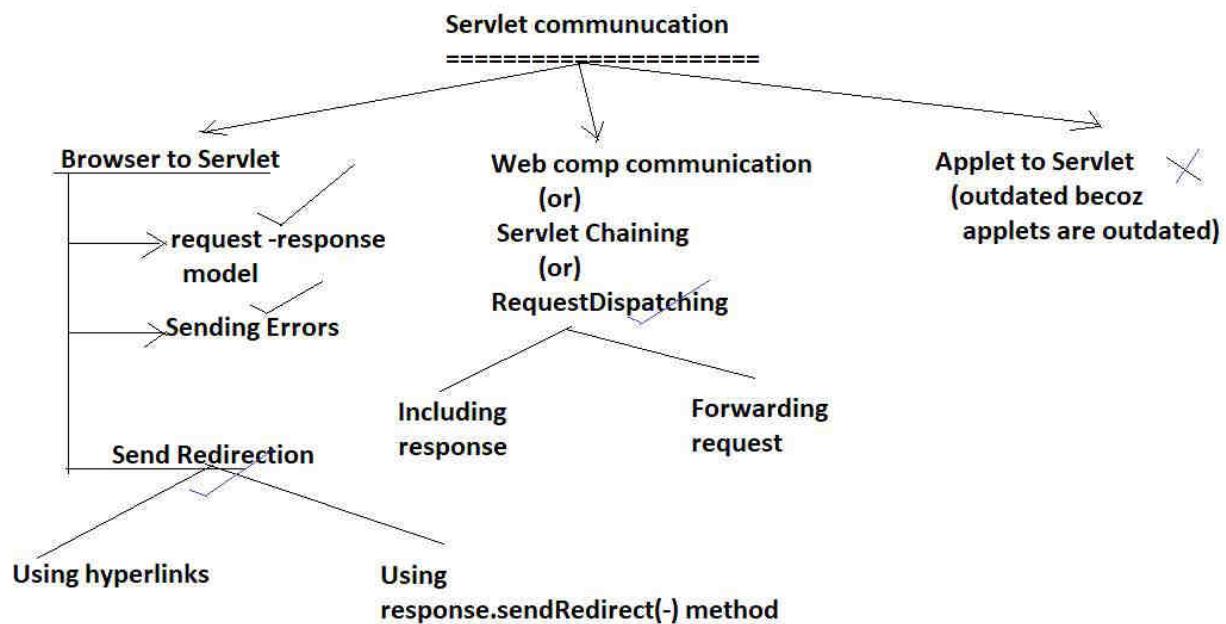
note:: welcome file will execute if do not place any virtual path in the request url where as default servlet /web comp will come for any invalid url.....

http://localhost:2020/TestApp -->This time welcome file executes
http://localhost:2020/TestApp/aafafdsf (invalid url) --> this time default Servlet comp executes..

ServletCommunication

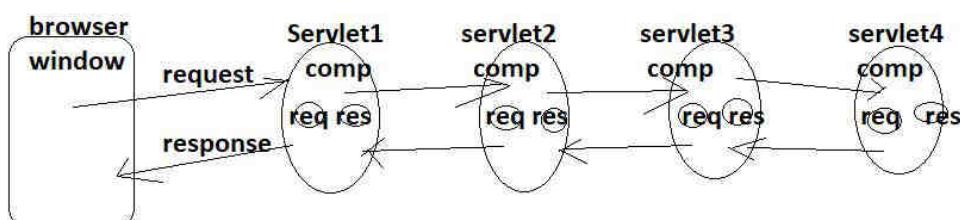
It is all about knowing

- => how servlet comp takes request from diff types of clients like browser, applet and etc..
- => how servlet comp delivers the success/ error response back to the browser
- => how servlet comp interacts with other Local or remote web comps..



WebComps Communication / SERvletChaining / RequestDispatching

=> So far the request given to servlet comp is processed by that servlet comp directly.. Processing request by taking multiple servlet comp in a chain is called **Servlet chainging** or **request processing**



All the servlet comps that are in chaining /communicatin will use same set of req,res objs while processing a request..

Doing servlet communication by creating destination servlet class object in source servlet comp and calling service(-,-)/doXxx(-,-) method on that object is bad pratice.. becoz ServletContainer has to manag servlet life cycle.. by creating object and calling life cycle methods .. mot programmer....

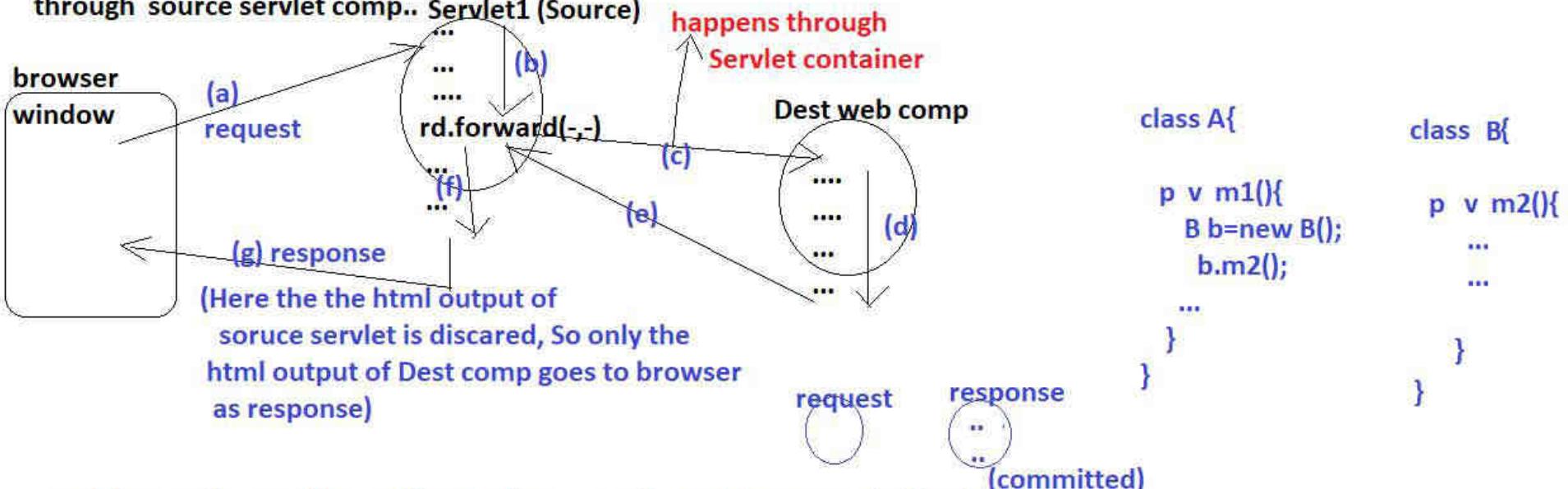
To overcome this problem we can RequestDispatcher object.. which makes the source servlet comp talking to dest servlet comp throgh ServletContainer.. So the both Source and dest comps life cycle will be managed by container dynamically..

two ways of Request Dispatching

(a) Forwarding request

=> source servlet comp calls forward(req,res) on RequestDispatcher obj (rd)

=> Here the source servlet comp's html output (pw.println(-) messages) will be discarded and only the output of dest comp goes to browser as response through source servlet comp.. Servlet1 (Source)



=> rd.forward(req,res) based forwarding request servlet communication is good to develop ErrorServlet comps..

=> The Servlet comp that executes only when the exception is raised in the main Servlet comps and useful to display non-technical guiding messages to enduser by discarding partial/incomplete outputs generated by the main servlet comps..

=> when rd.forward(-,-) is executed in source comp

i) The already added output to response obj of source servlet will be discarded

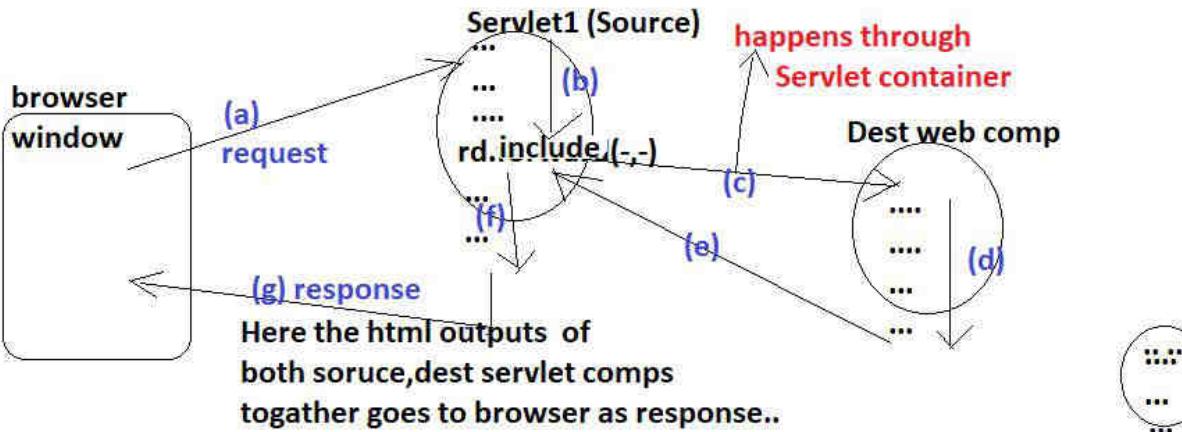
ii) control goes to specified dest comp through servlet container and executes that comp.. and also that dest comp output to response object and commits output to response object..

iii) transfers the control back source servlet comp.. having committed output in the response obj

b) including response

=>For this we use `rd.include(-,-)` method..

=> Here no html output will be discarded .. more ever the html outputs of both source servlet comp and dest comp together goes to browser as response.



usecase:: instead of placing same header ,footer logics in every main servlet comp.. keep them seperated web comps .. and include their outputs in main web comps using `rd.include(-,-)` methods calls..

RequestDispatcher object means ... it is the object of java calss that implements `javax.servlet.RequestDispatcher()`.

Different ways of creating RequestDispatcher object

=====

Approach1) using request object

In Servlet1 Comp(Source servlet comp)

`RequestDispatcher rd=req.getRequestDispatcher("/s2url");`
optional
url pattern of dest servlet comp

`rd.include(req,res)`

(or)

`rd.forward(req,res);`

->if the dest comp is html /jsp file of public area
then we can pass either file name or its url
pattern as the argument value..

Approach2 Using ServletContext object

In servlet1 comp

```
ServletContext sc=getServletContext();           mandatory
RequestDispatcher rd=sc.getRequestDispatcher("/s2url");
                                                 url pattern of dest servlet comp
rd.forward(req,res);
(or)
rd.include(req,res);
```

->if the dest comp is html /jsp file of public area
then we can pass either file name or its url
pattern as the argument value..

Approach3) Using ServletContext obj

In Servlet1 comp(source servlet comp)

```
ServletContext sc=getServletContext();
RequestDispatcher rd=sc.getNamedDispatcher("s2");      dest
                                                       logical name of servlet comp
rd.include(req,res);
(or)
rd.forward(req,res);
```

->if the dest comp is html /jsp file of public area
then we can pass their logical names given in web.xml file..

What is the difference b/w `getRequestDispatcher()` and `getNamedDispatcher()` method?

`getRequestDispatcher()`

- (a) invokable on both request, servletContext objs
- (b) if the dest comp is ServetComp , then it expects the url pattern of dest servlet comp as the argument value
- (c) if the dest comp is public area jsp file name then we can pass either jsp file or its url pattern as the argument value..
- (d) if the dest comp is public area html file then we can pass html file name or its url pattern as the argument value..

`getNamedDispatcher()`

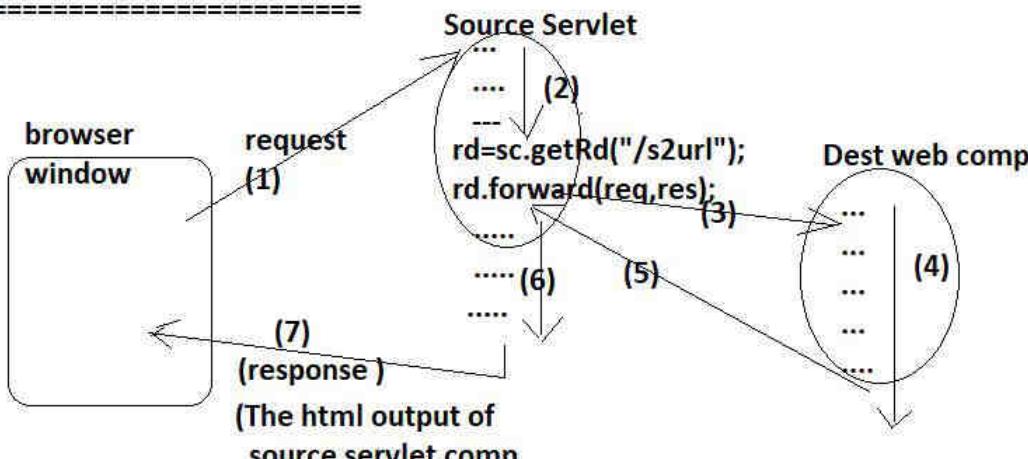
- (a) invokable only on ServletContext object
- (b) if the dest comp is ServetComp , then it expects the logical name of dest servlet comp as the argument value
- (c) if the dest comp is public area jsp file name then we can pass the logical name of jsp file as the argument value..
- (d) if the dest comp is public area html file then we can pass html file logical name given in the web.xml file as the argument value.

What is the diff b/w creating RequestDispatcher obj using request obj and creating RequestDispatcher obj using ServletContext object?

=>The "request" object based "RequestDispatcher" obj allows us to keep the source servlet comp and dest web comp in the same web application...

=>The "ServletContext" object based "RequestDispatcher" obj allows us to keep the source servlet comp and dest web comp either in the same web application or in the two different web applications of same server.. (but not in two diff web applications of two different servers
For this use SendRedirection concept)

Understand rd.forward(-,-)



keypoints

- =>performs the forwarding request mode of servlet communication
- =>The source servlet comp and dest comp will share req,res objs . So the req data coming to Source servlet comp is visible and accessible in dest web comp.
- =>Source servlet comp can pass additional data to dest web comp using request attributes..
- =>The source servlet comp and dest comp can be there either in same web application or in two diff web applications of same server..
- =>The Dest comp can be a servlet comp , jsp file or html file..
- =>The entire html output of source servlet comp will be discarded, i.e only the html output of dest comp goes to browser as response through servlet comp..
- =>if we place multiple rd.forward(-,-) method calls in Source servlet comp the we will get IllegalStat Exception..
- =>It is always recommended to place rd.forward(-,-) in source servlet comp as the conditional statement to execute..
- => Usecase:: ErrorServlet configuration..
 - >the Servlet comp that executes only when exception is raised in other servlet comp is called Error servlet comp and it is useful to display exception related messages as non-technical guiding messages to enduser...

Q) what is the need of ErrorServlet, when we can do the same work by using pw.println() statements placed in the catch(-) block..

Ans) The message printed in pw.println() will print non-technical guiding messages end to user. but it can not discard partial/incomplete outputs generated by Source servlet comp. But same thing is possible while working with ErrorServlet comp.

sample code of ErrorServlet configuration

ErrorServlet.java

```
public class ErrorServlet extends HttpServlet/GenericServlet{

    p v doGet(req,res)throws SE,IOE{
        //get PrintWrtier
        ....
        .....
        //display message
        pw.println("<h1 style='color:red'>Interanl Problem --Try gain </h1>");

        //close stream
        pw.close();
    }

}
```

MainServlet comp

```
public class EmployeeSerachServlet{
    p v doGet(req,res)throws SevletException,IOException{
        //get PrintWriter
        ..
        try{
            .... ,may
            ... //jdbc code that raise exception..
            ...
            ...
        }
        catch(Exception e){
            RequestDipatcher rd= sc.getRequestDispatcher("/s2ul");
            rd.forward(req,res);
            ...
        }
    }
```

=>we can configure jsp files , html files in web.xml file to get url patterns and logical names...

```
<!-- html cfg -->
<servlet>
  <servlet-name>h1</servlet-name>
  <jsp-file>/myError2.html</jsp-file>
</servlet>

<servlet-mapping>
  <servlet-name>h1</servlet-name>
  <url-pattern>/errurl2</url-pattern>
</servlet-mapping>

<!-- Jsp cfg -->
<servlet>
  <servlet-name>j1</servlet-name>
  <jsp-file>/myError1.jsp</jsp-file>
</servlet>

<servlet-mapping>
  <servlet-name>j1</servlet-name>
  <url-pattern>/errurl1</url-pattern>
</servlet-mapping>
```

Q) What happens if he place multiple rd.forward(,-) method calls in the same source servlet comp?

=> Since the first rd.forward(,-)method call commits the outputs response object.. so calling second rd.forward(-,-) asks to discard the already committed outputs.. which is not possible.. So the IllegalStateException will be raised..

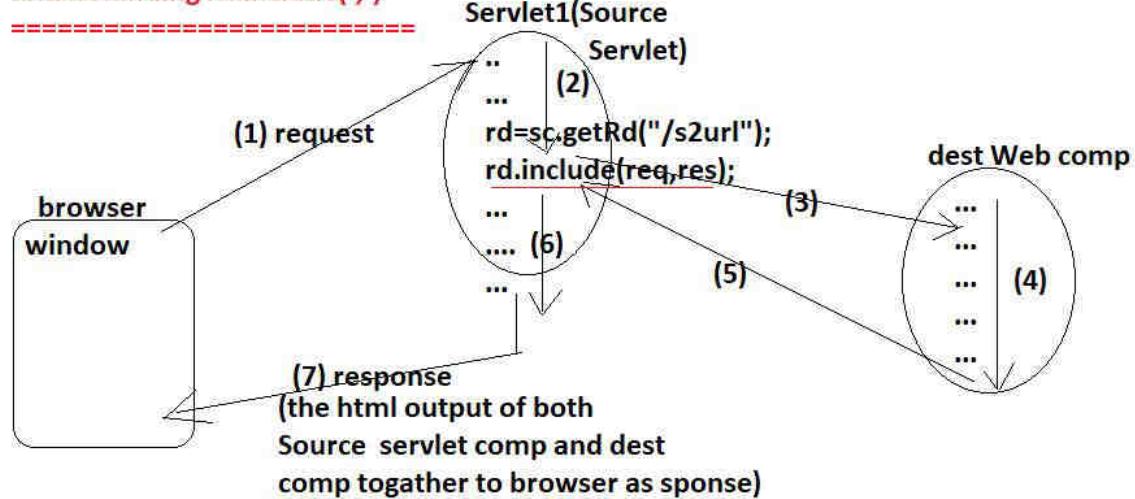
Q) Can we use multiple error servlets/pages/files for multiple exceptions?

Ans) Possible.. In multiple catch blocks , call multiple different rd.forward(,-) method calls on one ErrorServlet/pages/file for each Exception..

note:: pw.close();

=>This method not only the closes the stream..and also the commits the outputs to the response object i.e after pw.close() we can not write further messages to response object.

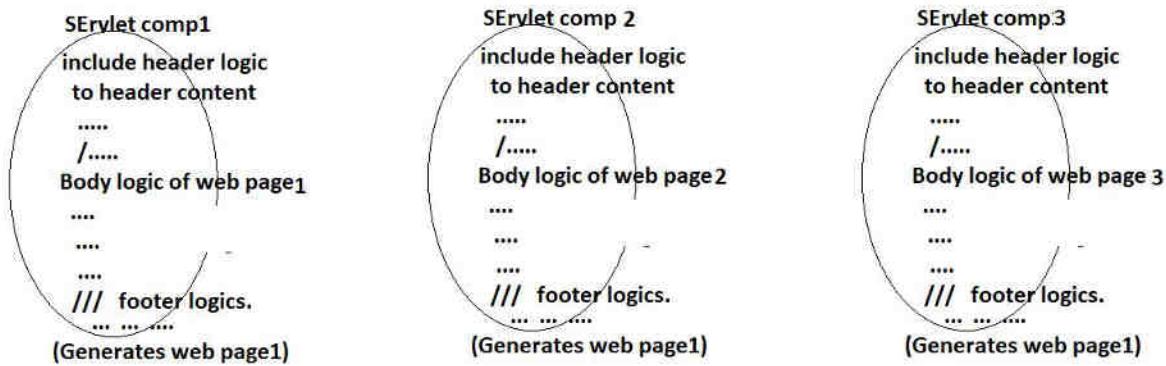
understanding rd.include(,-)



keypoints

- =====
- (a) Performs the including response mode of servlet communication
 - (b) Source servlet and dest comp use same request and response objects
So req data (form data) ..coming to source servlet comp can be used in dest servlet comp..
 - (c) Source servlet comp and dest comp can be there in same web application or
can be there in two diff web applications of same server.
 - (d) No html output will be discarded.. more over all outputs of both source
and dest comps will go to browser as final response
 - (e) The dest comp's output will be included to the source servlet comp output in the place
where rd.include(-,-) is called.
 - (f) use case:: Compositie view Desing pattern
 - (making common logics as reusable logics (header ,footer logics)
by keeping them in seperate web comps and by
including thier outputs to source servlet comp 's outputs..)

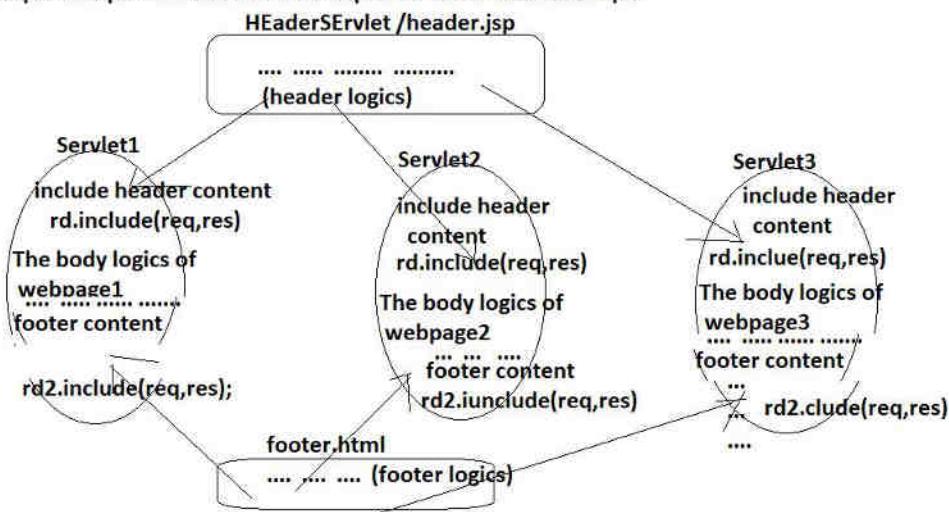
Problem::



note::Here header ,footer logics are not reusable.. though they
are same in all web comps..

Solution : Composite view desing pattern

According this .. seperate common logics(header ,footer logics) to other
helper comps.. and include their ouputs in main Servlet comps.



Here the view (web page) comes as the output given by multiple web comps together
as the united.. So this process is named as composite DP.

output

Sample code

=====

HeaderServlet.java

```
public class HeaderServlet extends HttpServlet{  
  
    public void doGet(HttpServletRequest req, HttpServletResponse res) throws ServletException, IOException {  
        // get PrintWriter  
        ...  
        // set response content type  
        ...  
        // display messages  
        pw.println("<h1 style='color:red;text-align:center'> N A R E S H Technologies </h1>");  
  
        // do not close stream  
        // pw.close();  
    }  
  
    public void doPost(HttpServletRequest req, HttpServletResponse res) throws ServletException, IOException {  
        doGet(req, res);  
    }  
}
```

footer.html

=====

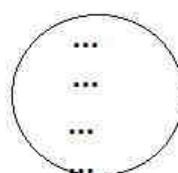
```
<br><br><br><br>  
<b><i> © all rights reserved 2010-20 </i></b>
```

main servlet comp

=====

public class EmployeeSearchServlet extends HttpServlet{

```
    public void doGet(HttpServletRequest req, HttpServletResponse res) throws ServletException, IOException {  
        // get PrintWriter  
        ...  
        // set response content type  
        ...  
        try{  
            // include header content  
            RequestDispatcher rd = req.getRequestDispatcher("/headurl"); ...  
            rd.include(req, res);  
        }  
    }  
}
```



```

//body content
.....
.... //jdbc code that
.... gives body content
.....

//include footer content
RequestDispatcher rd2=req.getRequestDispatcher("footer.html");
rd2.include(req,res);
}

catch(Exception e){
    RequestDispatcher rd=req.getRequestDispatcher("/errorurl");
    rd.forward(req,res);
}
}

p v doPost(req,res) throws SE,IOE
    doGet(req,res);
}

}

```

Q) What happens if we execute both rd.forward(--) and rd.include(--) methods in the same source servlet comp?

Ans) rd.forward(--) not only discards the original output of source servlet comp.. it also discards the other included outputs ...

multiple

Q) what is happens if we place rd.include(--) method calls in the same source servlet comp?

Ans) multiple outputs collected from multiple destination comps will be included to the source servlet comp's output.

note1:: while working with rd.forward(--) or rd.include(--) we should not place pw.close() or another method that closes stream .. before calling method.. becoz it throws IllegalStateException

note2:: do not close stream in the dest comp.. when source comp interacts with destination comp by rd.include(--) method , due to this only dest comp output will be added to response object.. and the remaining output will not be added..

What is the diff b/w rd.forward(--) and rd.include(--)?

rd.forward(req,res)

- (a) performs forwarding request mode of communication.
- (b) The html output of source servlet comp will be discarded only the output dest comp goes to browser response.
- (c) commits the response the moment this method is called method
- (d) use case:: for error servlet cfg
- e) if we place multiple rd.forward(--) methd calls in sisngle source selrvet comp the IllealStateException will be raised

rd.include(req,res)

- (a) Performs the including repsonse mode of communication
- (b) The html outputs of both soruce serlvet comp and dest serlvet corp togather goes to browser as response..
- (d) does not the commit the response
- (d)usecase:: composite view design patteern
- (e) placing multiple rd.include(--) methods will get and include the output from multiple destination comps..

Cross Context Communicattion

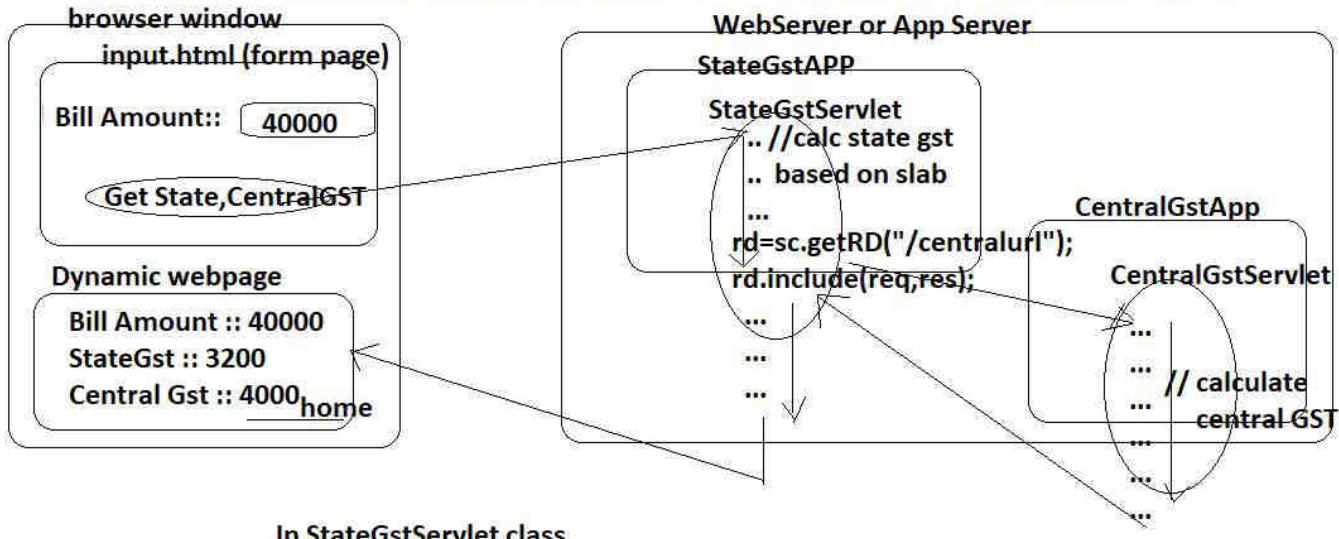
(or)

Communication b/w two diff web applications of same server

useCases:: forwaring the request given to amazon.in to amazon.com

(here both Web Apps must
be there in the same server)

=>For this Cross-Context Communication , we need ServletContext obj based RequestDispatcher.



In StateGstServlet class

```
//get access to ServletContext obj of current web application
ServletContext sc=getServletContext(); //StateGstApp
//get access to ServletContext obj of Dest web application
ServletContext sc1=sc.getServletContext("/CentralGstApp"); //CentralGstApp
//create RequestDispatcher object
RequestDispatcher rd=sc1.getRequestDispatcher("/centralurl");
rd.include(req,res);
```

StateGSTApp

```
|---->java resources
|-->src
|--->com.nt.servlet
|---->StateGstServlet.java
|--->WEB-INF
|--->web.xml
```

CentralGstApp

```
|---->java resources
|--->src
|--->com.nt.servlet
|---->CentralGstServlet.java
|--->WEB-INF
|--->web.xml
```

apps in the

note:: we must deploy both same in server..

note:: Weblogic , Glassfish, wildfly servers support CrossContext communication without any addtional settings

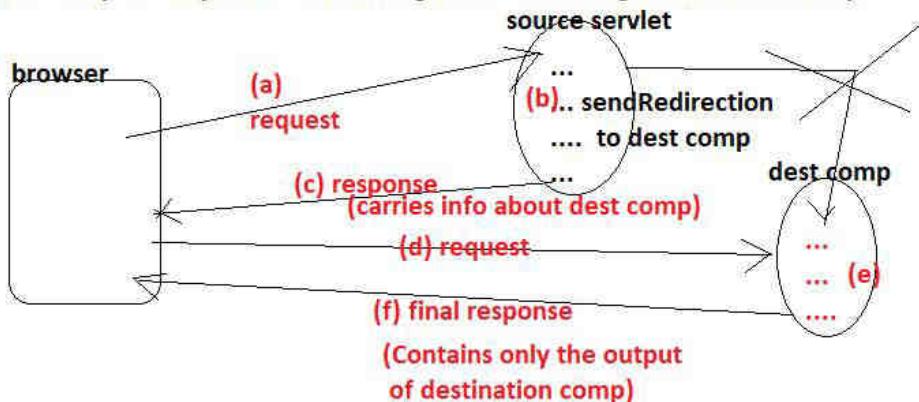
note:: Tomcat support cross Context Communication with addtional settings

- place crossContext="true" in the <Context> tag of Context.xml
(Go to servers section of eclipse project explorer)
- call sc.getServletContext() method as shown below
ServletContext fc=sc.getServletContext("/CentralGSTApp")

Limitations of RequestDispatcher based Servlet Communication

- (a) Dest comp must be there as html file or jsp file or servlet comp .. i.e we can not take asp, asp.net, php ,python comps as dest comps
- (b) source Servlet comp and dest comp can not be there in two different web applications of two different servers..
- (c) "crossContext" communication is possible any few servers .. In some server we need enable special settings for cross context communication..

To overcome these problems use sendRedirection.. which makes source servlet comp taking to local or remote dest comp through browser.. i.e source servlet comp will not talk with dest comp directly, It will talk through browser having network round trip.



=>Since browser can give request to any location and any technology web comp based on the given url info.. So the dest web comp in sendRedirection app can be there in any technology and in any location.(any server of any machine)

UseCases of SendRedirection

- (a) if company A acquires company B , so the requests given to company B website like B.com will be redirected to company A website like A.com using sendRedirection

eg:: Oracle corp has acquired sun Ms , So the request given to sun.com will be redirected to a page of Oracle.com

eg:: the request given to rational.com will be redirected to a page of IBM.com

eg:: the request given to gmail.com will be redirected to a page of google.com

- (b) To use Remote website services begin from local website pages...

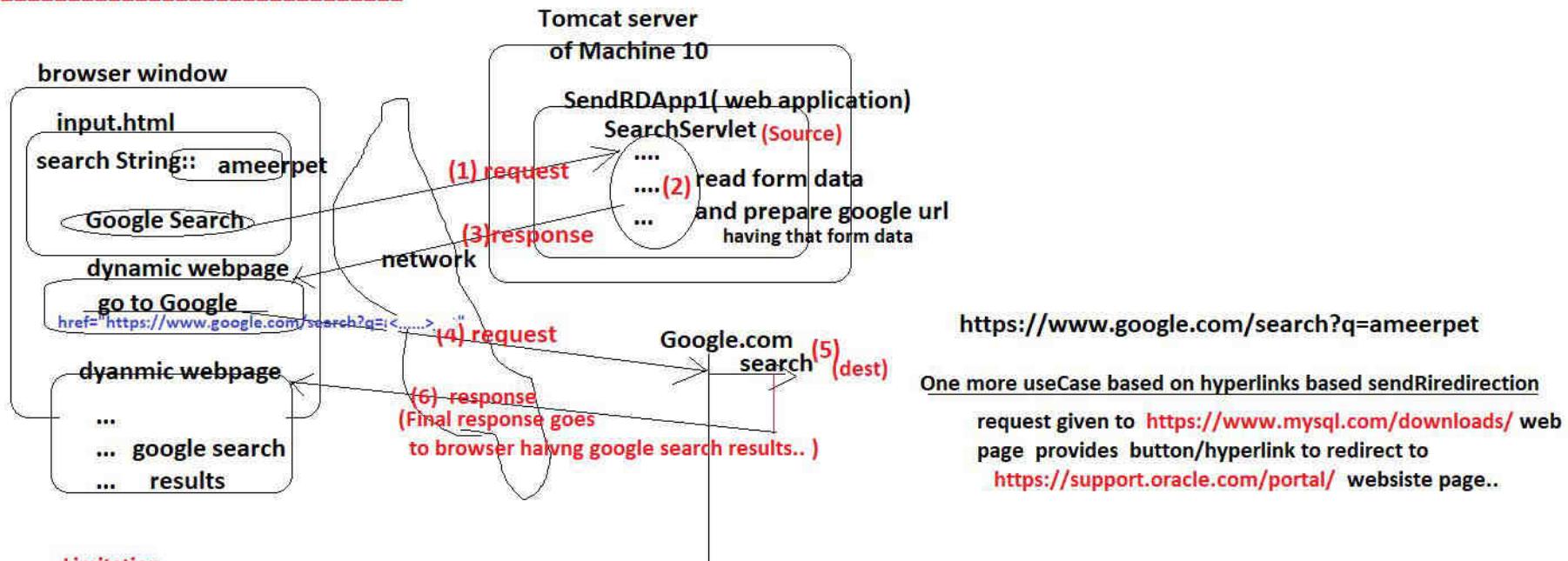
eg: search box of any website internally uses Google search services

eg: using google maps in our web application/websiste.

Two ways of implementing sendRedirection

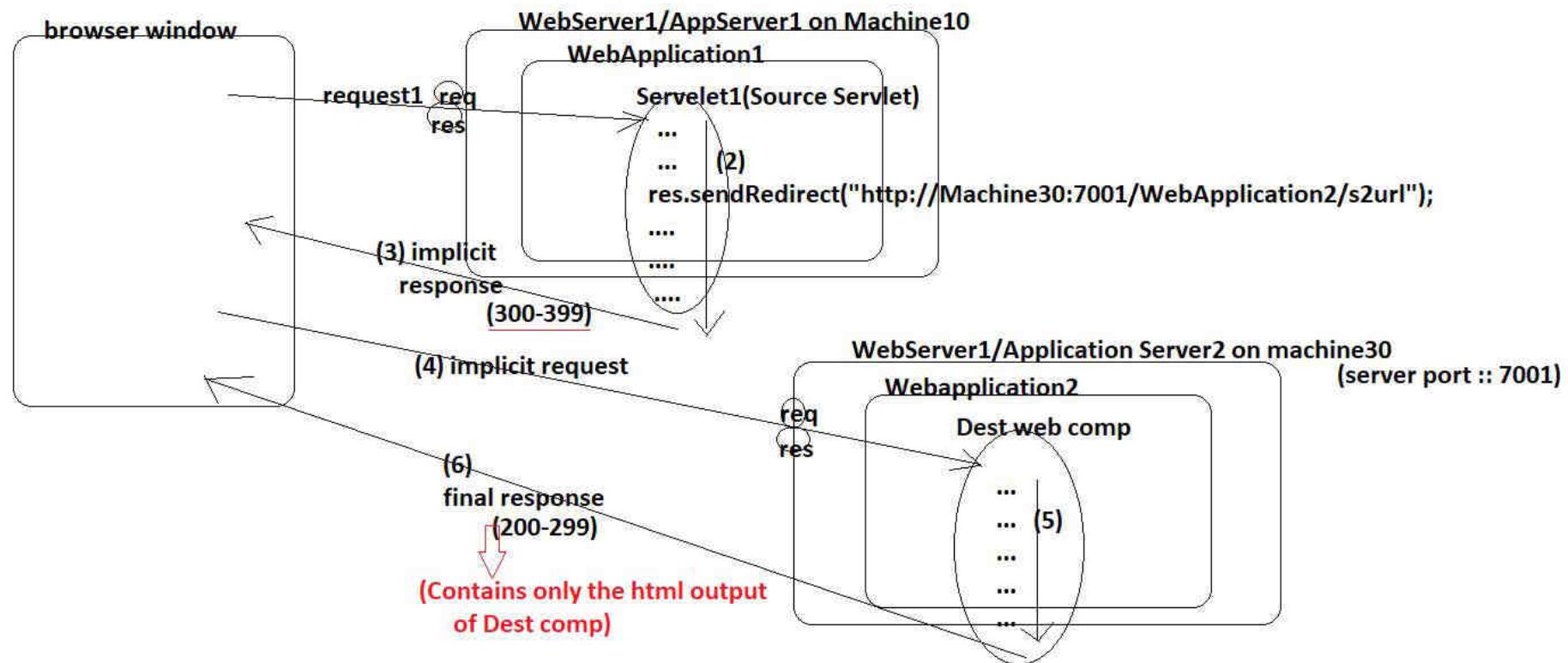
- (a) By using hyperlinks
- (b) By using `res.sendRedirect(-)` method (best)

Send Redirection using hyperlinks



```
SendRDApp1
|--->java resources
    |-->src
        |--->com.nt.servlet
            |---->SearchServlet.java
|--->webcontent
    |---->input.html
    |--->WEB-INF
        |---->web.xml
```

SendRedirection using res.sendRedirect() method



key points

- a) performs sendRedirection based servlet communication..
- b) Source Servlet interact with Dest comp through browser
- c) Source servlet and dest comp will not use same req,res objs so the request data/form data comming to servelt is not visible and accessible in dest comp..
- d) To pass additional data from source servlet to dest comp append query String to the url of `res.sendRedirect()` method.
in source servlet comp
`res.sendRedirect("http://localhost:7001/WebApplication2/s2url?sno=101&sname=raja")`

In Dest comp

```
int no=Integer.parseInt(req.getParameter("sno")); //gives 1001  
String name=req.getParameter("sname"); //gives raja
```

- e) Dest comp can be there in any technology like servlet,jsp, php,asp,asp.net and etc..

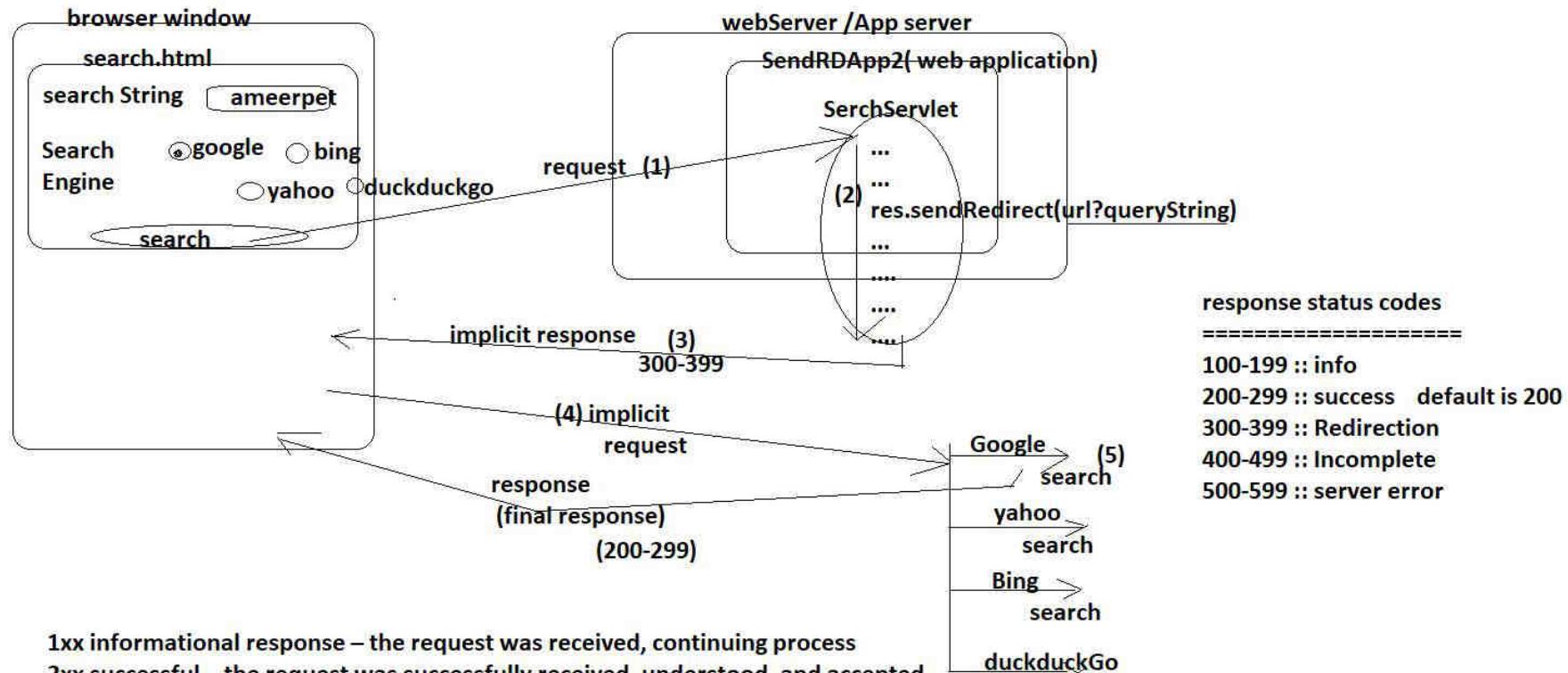
- f) Source servlet and dest comp can be there in same web application or in two diff web applications of same server or in two diff web applications of two diff servers.. staying on same machine or in different machines..
- g) All the statements in Source serlet comp executes , but their output will be discarded.. since res.sendRedirect(-) is executed the generated response to browser from Servlet comp contains status code 300-399 having the url kept in res.sendRedirect(-) method.
- h) if source servlet and dest comp are there in same web application then we can use relative url in res.sendRedirect(-) method , otherwise we need to use absolute url

`res.sendRedirect("s2url"); //relative url`

`res.sendRedirect("http://machine30:7001/WebApplication2/s2url"); //absolute url`

Example App

=====



1xx informational response – the request was received, continuing process

2xx successful – the request was successfully received, understood, and accepted

3xx redirection – further action needs to be taken in order to complete the request

4xx client error – the request contains bad syntax or cannot be fulfilled

5xx server error – the server failed to fulfil an apparently valid request

Passing data among the web comps of the web application

How to pass data from source servlet comp to dest web comp?

if source Servlet comp and dest web comp are there in the same web application

(a) request attributes

=>use if the source servlet comp and dest comp are using same req,res objs.

(b) session attributes

=>use if the source servlet comp and dest comp are getting request from same browser s/w of a client machine.

(c) application attributes /ServletContext attributes

=>Use if the source servlet comp and dest comp are not using same req,res objs and not getting request from same browser.

if the source servlet comp and dest comp are not there in same web application

d) append query String to the url of res.sendRedirect(-) method.

res.sendRedirect(url? p1=val1&p2=val2&p3=val3)

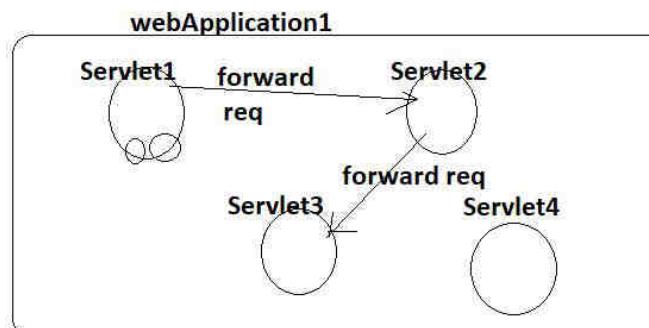


request attributes

=>Servlet attributes are the logical names which hold values as objects having scope.. These are no way related to html/xml tag attributes.

=>These attributes allocate memory in request obj.. and visible and accessible through out that request object..

=>Request attributes scope is "request scope" i.e request attributes are visible through out request cycle (indicates they are specific to each request)



=>Here the **Servlet1,Servlet2,Servlet3** are using same req,res objs when we give request to **Servlet1** comp.. So the **req** attribute created in the "servlet1" is visible and accessible in **Servlet2,Servlet3** comps.. but not in **servlet4** comp.

To create request attribute

```
=====
req.setAttribute("name","raja");
req.setAttribute("age",30);
```

attr	attr
name	value

attr name must be string... value
must be object. if u pass simple value
the it will be converted to wrapper class
object using auto boxing feature.

To create modify request attribute value

```
=====
req.setAttribute("name","raja1");
req.setAttribute("age",31);
```

setAttribute(-,-) method creates new attribute
if the attribute is not already available in the
specified scope otherwise it will modify the
existing attribute value

To read request Attribute value

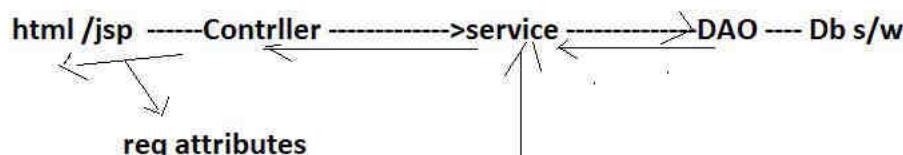
```
=====
String name=(String) req.getAttribute("name");
int age= req.getAttribute("age");
```

Here Integer Wrapper class obj is converted into simple int using autounboxing

To remove request Attributes

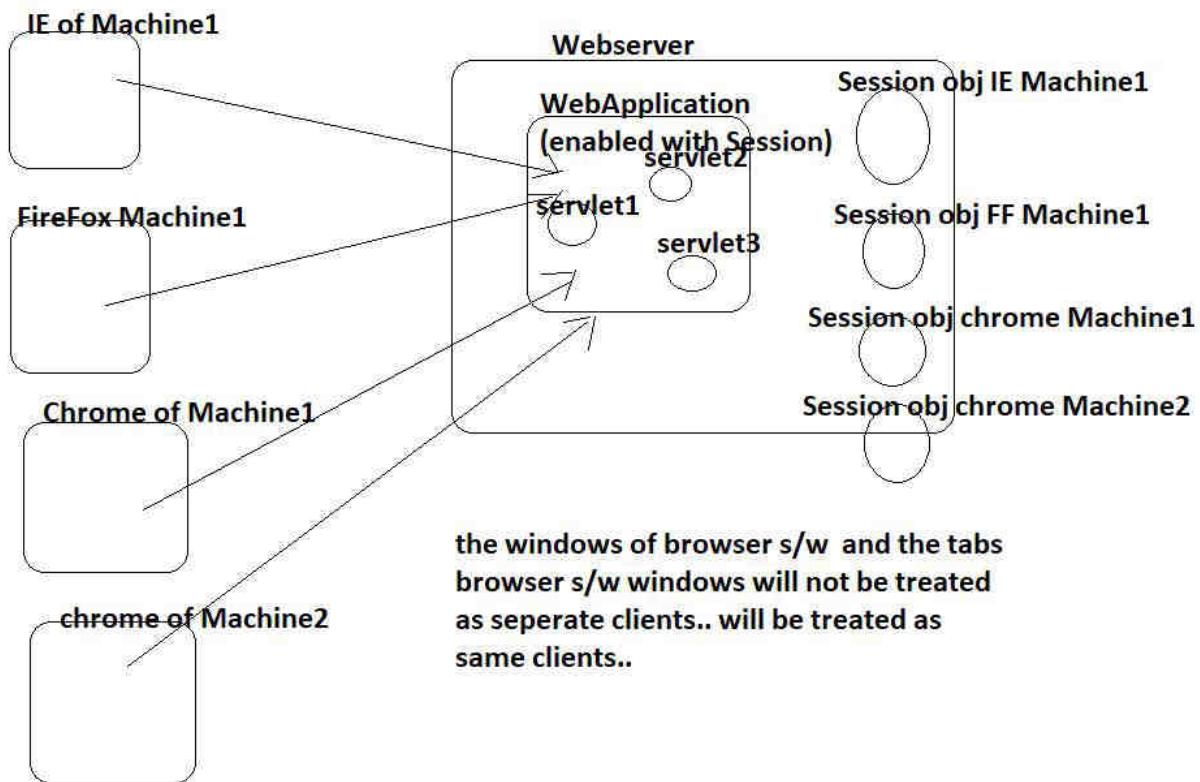
```
=====
req.removeAttribute("name");
req.removeAttribute("age");
```

UseCase:: In layered Apps ,the controller servlet comp the gets results from service class.. and we can pass those results to dest html/jsp page having the support of request attributes.



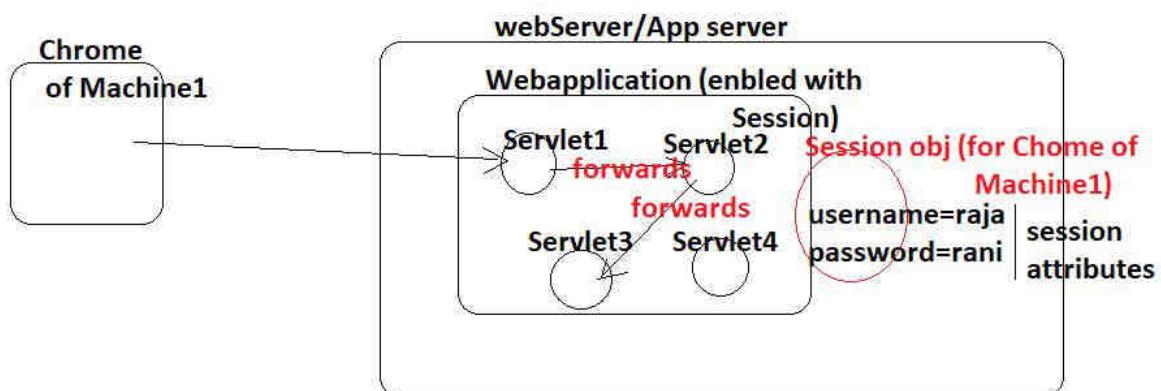
Session Attributes

- => Session attributes are allocated in HttpSession object (Session object)
- => Session object is allocated in the Server on 1 per browser software of each client machine



=> Session object and its session attributes are visible in all web components of web application irrespective of the request/response objects they are using.. but they must get request from that browser for which the session object is created in the server..

=> Session attributes scope is "session" scope i.e. session attributes are visible in all web components of web application but specific to a browser software of a client machine..



=>The session attributes created in Servlet1 by getting request from chrome browser of machine1 are visible and accessible in all the web comps of web application irrespective of req,res objs they are using.. but they must get request from same **chrome browser of machine1**

=> Gmail, fb websites remembers the username,password details as session attribute values.. i.e they are specific each browser s/w of a client machine..

To create Session attributes

```
=====
HttpSession ses=req.getSession(); //creates /locates Session obj for browser s/w of a client machine
ses.setAttribute("uname","raja");
ses.setAttribute("pwd","rani");
```

To modify session attribute values

```
=====
ses.setAttribute("uname","Raja1");
ses.setAttribute("pwd","rani1");
```

To read session attribute values

```
String un=(String)ses.getAttribute("uname"); //gives "Raja1"
String pass=(String) ses.getAttribute("pwd"); //gives "rani1"
```

To remove session attribute

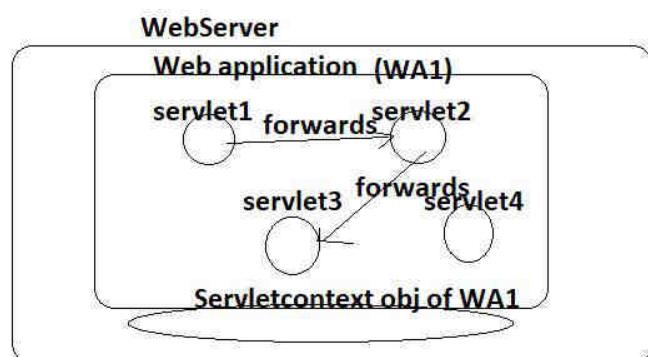
```
=====
ses.removeAttribute("uname");
ses.removeAttribute("pwd");
```

ServletContext attributes /application attributes

=>These attributes allocate memory in ServletContext object which is 1 per each web applicaiton.

=>ServletContext object and its attributes are visible in all the web comps of a web application irrespective of req,res objs they are using and irrespective of the browser s/w from which request is coming.

=> ServletContext attributes scope is "applicationscope" i.e they are globally visible with in the web application irrespective of any conditions..



=> The Servletcontext attributes created in Servlet1 comp are visible and accessible in all the web comps of web application..irrespective of any condition.

usecase :: To store request count for the entire web application , we need to take "count" ServletContext attribute.. (counting no.of visits happening to web site)

To create ServletContext attributes

```
ServletContext sc=getServletConext();
sc.setAttribute("count",10);
```

To modify ServletContext attribute values

```
ServletContext sc=getServletConext();
sc.setAttribute("count",20);
```

To read Servletcontext attribute values

```
int count=(Integer)sc.getAttribute("count");
```

To remove Servletcontext attribute values

```
sc.removeAttribute("count");
```

Different scope to keep data in servlet programming

- b) page/servlet scope (specific to each servlet comp --Instance variables in Servlet comp)
- a) request scope (specific to each request)
- b) session scope (specifc to each browser s/w of client machine)
- c) application scope (visible within a web application)

SEssion Tracking /SEssion Management

Need

===== Two types form pages

a) static form pages

- > html files having form designing
- > here the content of form page is fixed.
- > so far we working with static form pages

b) dynamic form pages

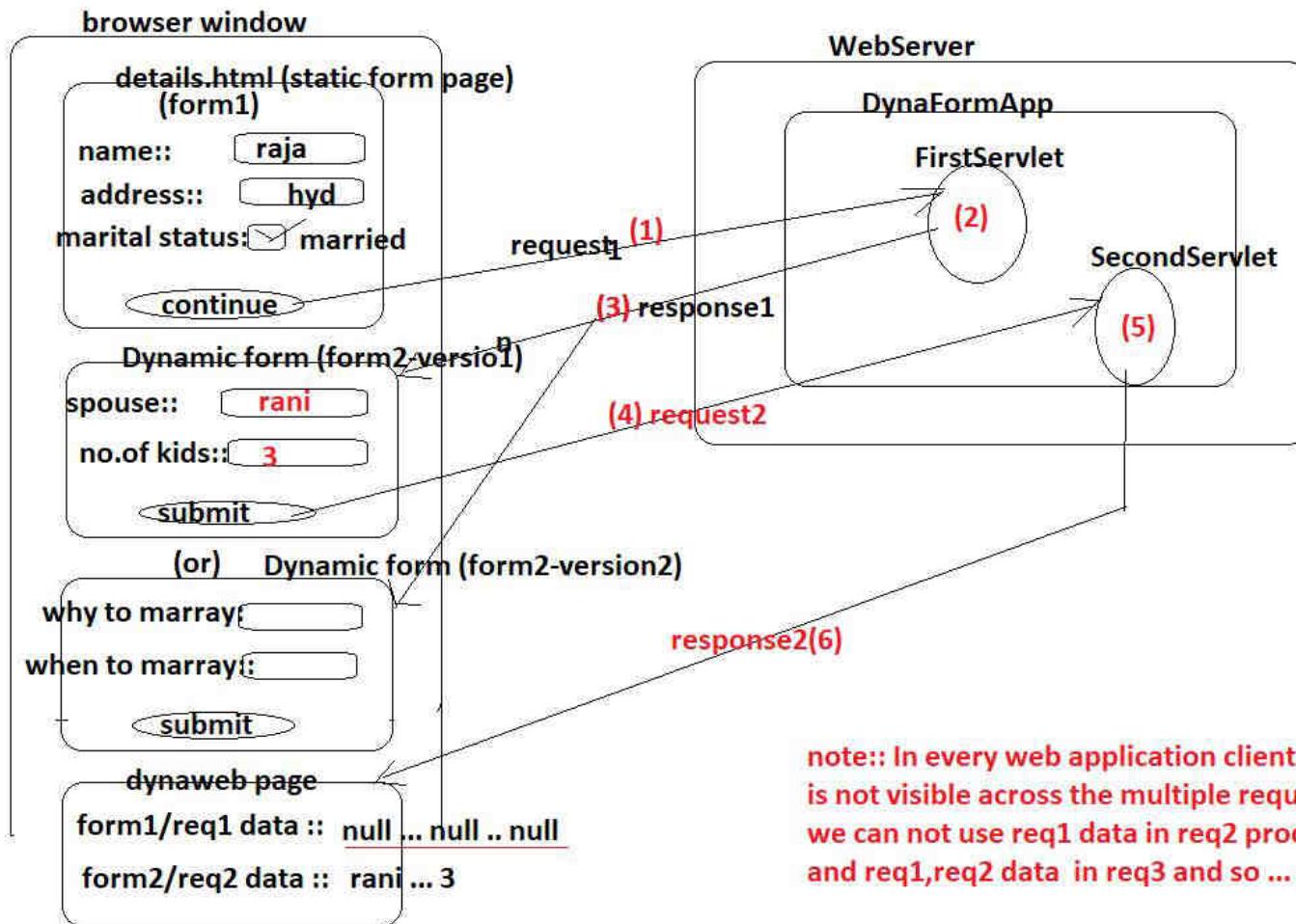
- > It is the form page that comes as the response of dynamic web comps like Servlet/jsp comp
- > These form pages contain dynamic content..
place <form> <input> and etc., in pw.println() methods..
- =>collecting multiple inputs from end user by using single static form page is
bad pratice becoz enduser will be forced
to answer both necessary and unneccesary
questions..

details.html (static form page)

name::	<input type="text"/>
addrss::	<input type="text"/>
marital status:	<input type="checkbox"/> married
spouse::	<input type="text"/>
no.of kids::	<input type="text"/>
when do want to mary?	<input type="text"/> ^u
why do want to mary?	<input type="text"/> ^u
<input type="button" value="submit"/>	



=>collect info from enduser by using both static ,dynamic form pages like gather initial info using static form page.. based inputs given to static form page start gathering remaining details using dynamic form page as shown below..



=>by default all web applications are stateless i.e Web application can not use client data across multiple requests..

What is a Session?

=> set of related and continuous operations performed on the web application by a user or a client (browser s/w of client machine) is called session..

eg:: => Login to logout in FB , Gmail

=> Start of playing game to end of playing game in online on a particular day

=>Start of java class to end of java class on a particular day

=> start of zoom meeting to end of zoom meeting on a particular day

What is stateless web application and what is statefull web application?

if the web application is not capable remembering client data across multiple requests during a session then it is called stateless web application.. i.e web application can not use previous requests data while processing current request.. By default all web applications are stateless.

if the web application is capable remembering client data across multiple requests during a session then it is called statefull web application.. i.e web application can use use previous requests data while processing current request.. By adding Session tracking/management techniques on web application we can make them as stateful web applications,,

Why the web applications are stateless by default?

Ans) Web applications are stateless becoz the protocol http is designed as stateless..i.e for every request separate connection will be created b/w browser s/w and ^{not} webServer where the web application is running. Due to this 1 connection/request data can be used in another connection/ request , This makes web application as stateless.. This advantage of state less behaviour is no client can engage connection with server for long time and can keep connection in idle state for long time... So no wrong business practices are allowed...

=>The stateless protocol http makes the web application as stateless web application.. To make such web application stateful web application we need to add session tracking/session management support.. (It is all about remembering client data across the multiple requests during a session)

The session tracking techniques are
=====

- (a) hidden form fields (unofficial)
- (b) Http cookies
- (c) Http session with cookies
- (d) Http session with URL Rewriting

note: The connection created b/w browser server will be closed automatically for

- (a) success response (or)
- (b) error response (or)
- (c) for connection timeout

Hidden Form fields (Session tracking technique1)

In html form page

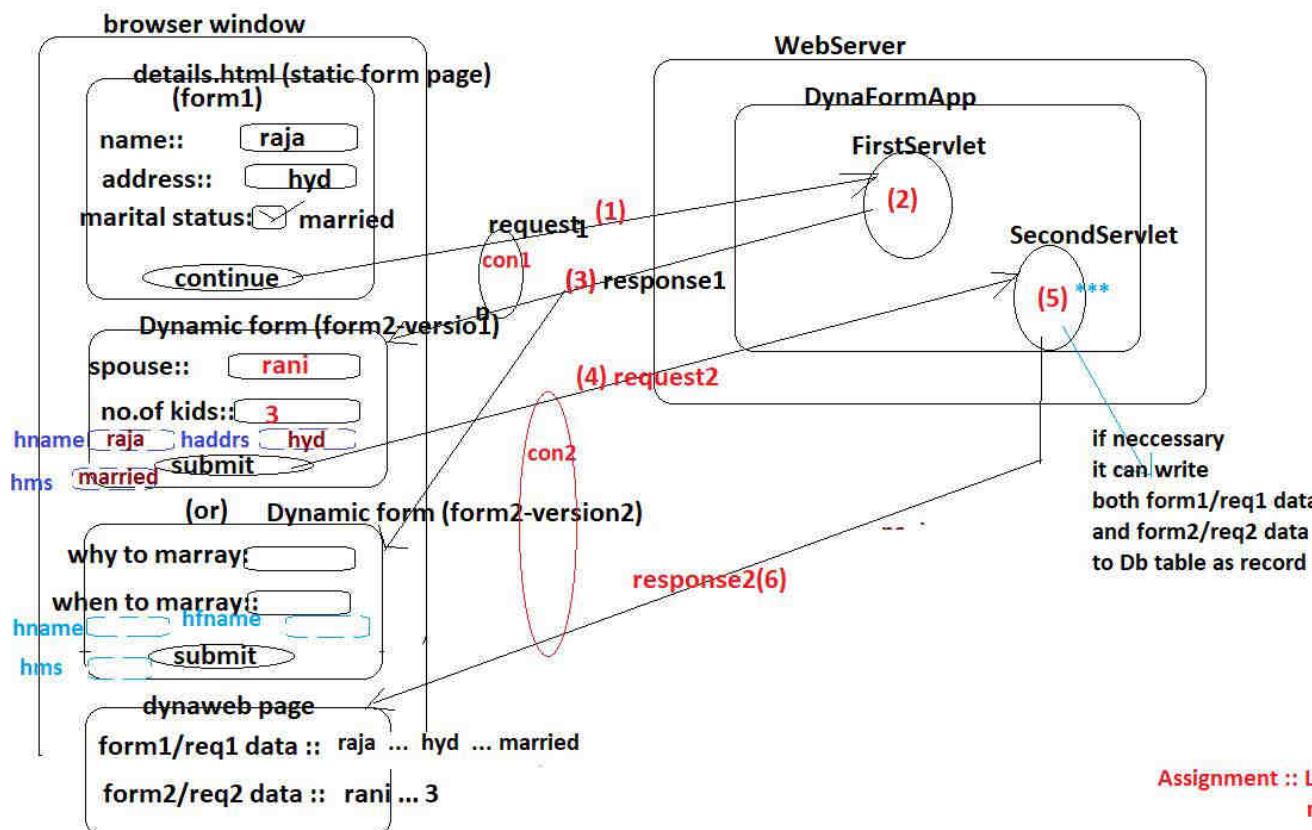
```
<input type="hidden" name="h1" value="hi"/>
```

In servlet comp

```
String s1=req.getParameter("h1"); //gives hi
```

Hidden Form files based Session tracking

(Makes the web application stateful web application)



***:: SecondServlet is performing Sessiontracking i.e It can read form1/req1 data from hidden box values while proressing form2/req2 data..

Assignment :: List of 10 different category usecases where u have noticed the session tracking?

Advantages of Hidden Form fields Session Tracking

- a) Basic knowledge of html is sufficient
- b) This works with all servers and all web technologies
- c) easy to learn and easy to implement

DisAdvantages

- (a) No data secrecy (we can see hidden box values using view source option)
- (b) Hidden boxes can hold any text data / String values..
- (c) Hidden boxes values will travel across the multiple requests.. So it will increases network traffic.
- (d) While creating each dynamic form page, we should add previous forms data as hidden boxes this gives burden programmer.. (while getting last dynamic form we need to all previous multiple forms data as hidden boxes)
- (e) It is not an official /standard Session tracking technique...

10 uses-cases where the Session tracking is observed

- a) Login- Logout in any web application (eg:: fb logic, gmail login and etc..)
- b) Registration process using multiple forms (eg: nuakri.com)
- c) Adding Items to shopping cart
- d) Online Payment
- e) Online Gaming
- f) Online ticket booking..
- d) Online stock trading..
- e) Online examinations..
- f) Online Banking like withdraw,deposite and etc...
- g) Online bill payments..
- h) Online Direct advertisements ..
(rendering the adds based on activities of the user)
- i) Online Visa Processing/applying
and etc..

Q) What is the diff b/w Session Management and State management?

In SessionManagement , the web application remembers the client data across the multiple requests but that data is specific to each client and that data will be ready to vanish after data session (i.e that data can be used by a client only during the session ,not after the session)

eg:: Hidden Form files, http cookies , HttpSession with cookies
and HttpSession with URL rewriting and etc. are session tracking techniques..

In State Management , the web application remembers the clients data but that data is not specific to each client and will not be vanished.. after session.. i.e this data can be used by any client at any time..

eg:: writing each request data to db table or as servlet Context attributes falls under state management..

Http Cookies (Session tracking technique2)

=>Cookies are small textual infomations that allocate memory at client side remembering client data across the mutliple requests.. Cookies are created in web application (server side) and they will be sent to client (browser) along with the response and cookies come back to Web application from client along with the rqeuest..

Two types cookies

(a) InMemory Cookies /PerSession cookies

- >Allocates in the In the InMemory where Browser runs..
- > Once the browser window is closed, these cookies will be destroyed automatically
- > No expiry time
- >very useful for session tracking

b) Persistence cookies

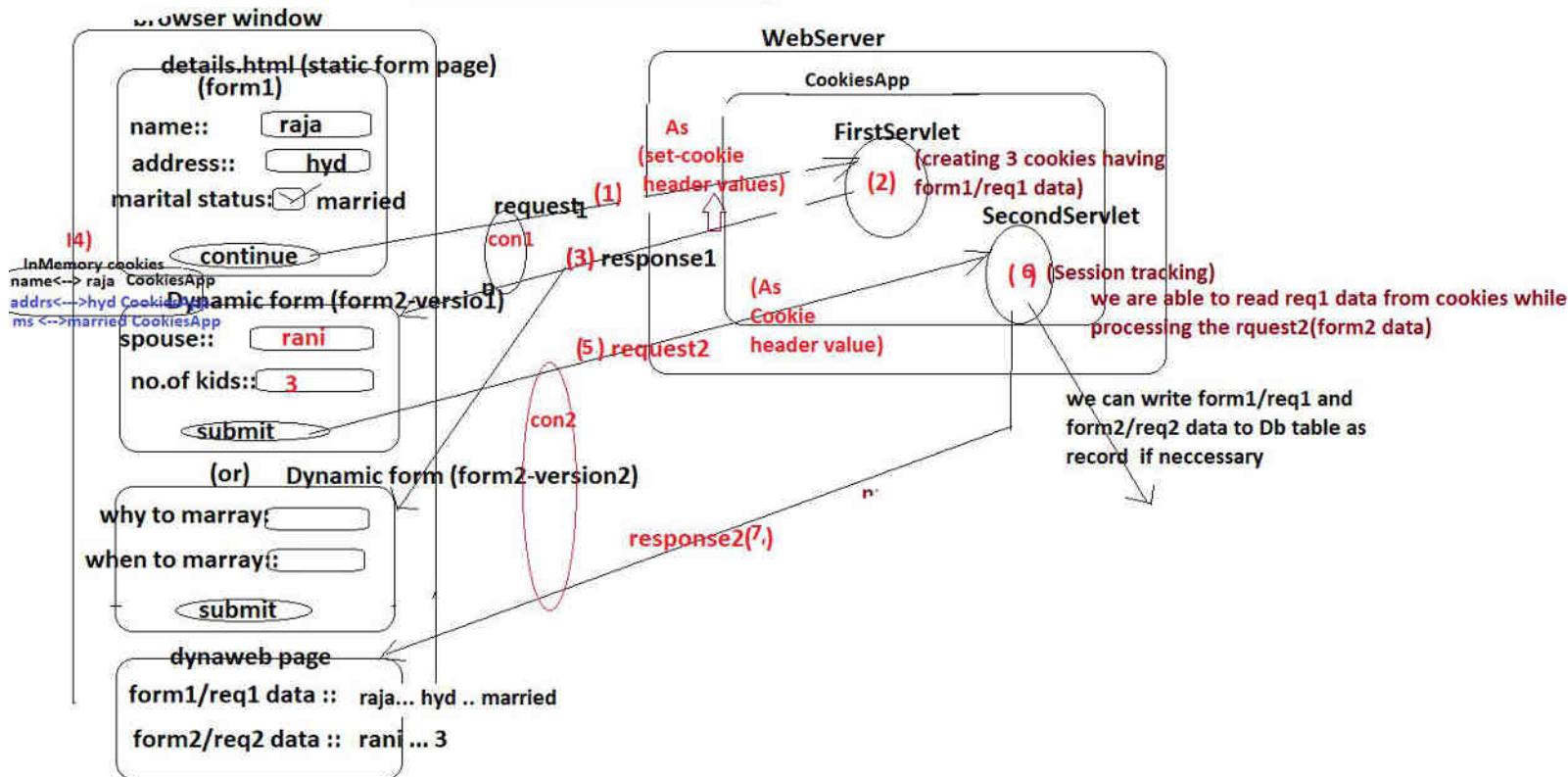
- =>Allocates Memory in the client machine file system (hard disk) having expiry time.
- =>These cookies will not destroyed once the browser is closed.. but will be destroyed if the expiry time is completed.
- => these cookies are not suitable for session tracking.. becoz these are rembering client data even after closing browser (after ending the session)
- =>These are very useful to enable "rember me" facility in the web applicaiton..

Every Cookie contains

- a) cookie name (String)
- b) cooke value (String)
- c) Domain/website name (String)

"set-cookie" (response header) --> WebServer to browser/client machine
"cookie" (request header) --> browser to webServer

Cookies based Session tracking



Note: The cookies belonging to certain web application goes back to web application from browser or client machine when they give request to same web application from browser..

Cookie API (javax.servlet.http.Cookie)

=====

To create cookies

```
Cookie ck1=new Cookie("ap","A-V-K");
Cookie ck2=new Cookie("TS","HYD");
```

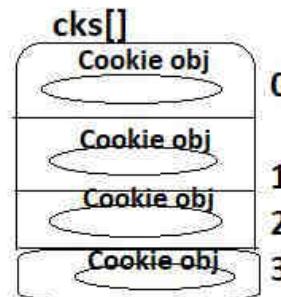
`res.addCookie(ck1); res.addCookie(ck2);` | (These are InMemory Cookies)

```
Cookie ck3=new Cookie("mh","mumbai");
Cookie ck4=new Cookie("UP","luknow");
ck3.setMaxAge(1800); //secs
ck4.setMaxAge(1800); //secs
res.addCookie(ck3); res.addCookie(ck4);
```

| (These are persistent cookies)

To read cookie values

```
Cookie cks[] = req.getCookies();
if(cks!=null){
    for(Cookie ck:cks){
        S.o.p(ck.getName()+" "+ck.getValue());
    }
}
```



To modify cookie values

```
ck1.setValue("a-v-k1");
ck2.setValue("newHyd");
```

To delete cookies

We can not delete cookies manually..becoz the cookies are allocating memory at client side..

- > InMemory cookies will be destroyed once the browser window is closed
- > Persistence cookies will be destroyed once the expiry time is over.

To get domain name of cookies

```
String s1=ck1.getDomain();
String s3=ck1.getDomain();
```

To get the Max age

```
int m1=ck1.getMaxAge() ; //gives -1
int m2=ck3.getMaxAge() ; //1800 secs..
```

To comment to the comment

```
ck1.setComment("holds Ap hyd");
```

To get the comment of the cookies

```
String c1=ck1.getComment();
```

App to feel the diff b/w InMemory , Persistence cookies practically?

CookiesApp1

```
|--->java resources
|   |--->src
|   |   |--->com.nt.servlet
|   |   |   |--->CreateCookiesServlet.java (4 cookies :: 2+2)
|   |   |   |--->ShowCookiesServlet.java (Display all the 4 cookies)
|--->webcontent
|   |--->WEB-INF
|   |   |--->web.xml
```

Advantages and disadvantages of http cookies

advantages

- >These cookies allocate memory at client side and do not burden to server.
- > All server and all web technologies support cookies
- > Supports both session management (In Memory cookies) and State management(persistence cookies)

disadvantages

- > cookies travel over the network along with the request,response i.e they add network traffic
- > We must work http protocol env.. (HttpServlet)
- > Using Browser settings ,either we can block cookies or we can delete cookies then session tracking fails.
- >there restriction on no.of cookies and size of cookie
 - The browser is expected to support 20 cookies for each Web server, 300 cookies total, and may limit cookie size to 4 KB each.
- >Using browser settings we can see the cookies data.. (only in few browsers)

To block , view or remove cookies using Chrome browser

chrome control panel(: ---> settings -->search for cookies ---> go to cookies and site data

- a) use block All cookies radion button
- b) see all cookies -> search for cookies to view and delete them

To block or remove cookies from Firefox browser

fire fox control panel  -->options -->search for cookies

- a) select custom --> select cookies -->select block all
- c) go to cookies and site data --> manage -->search for appropriate and remove them

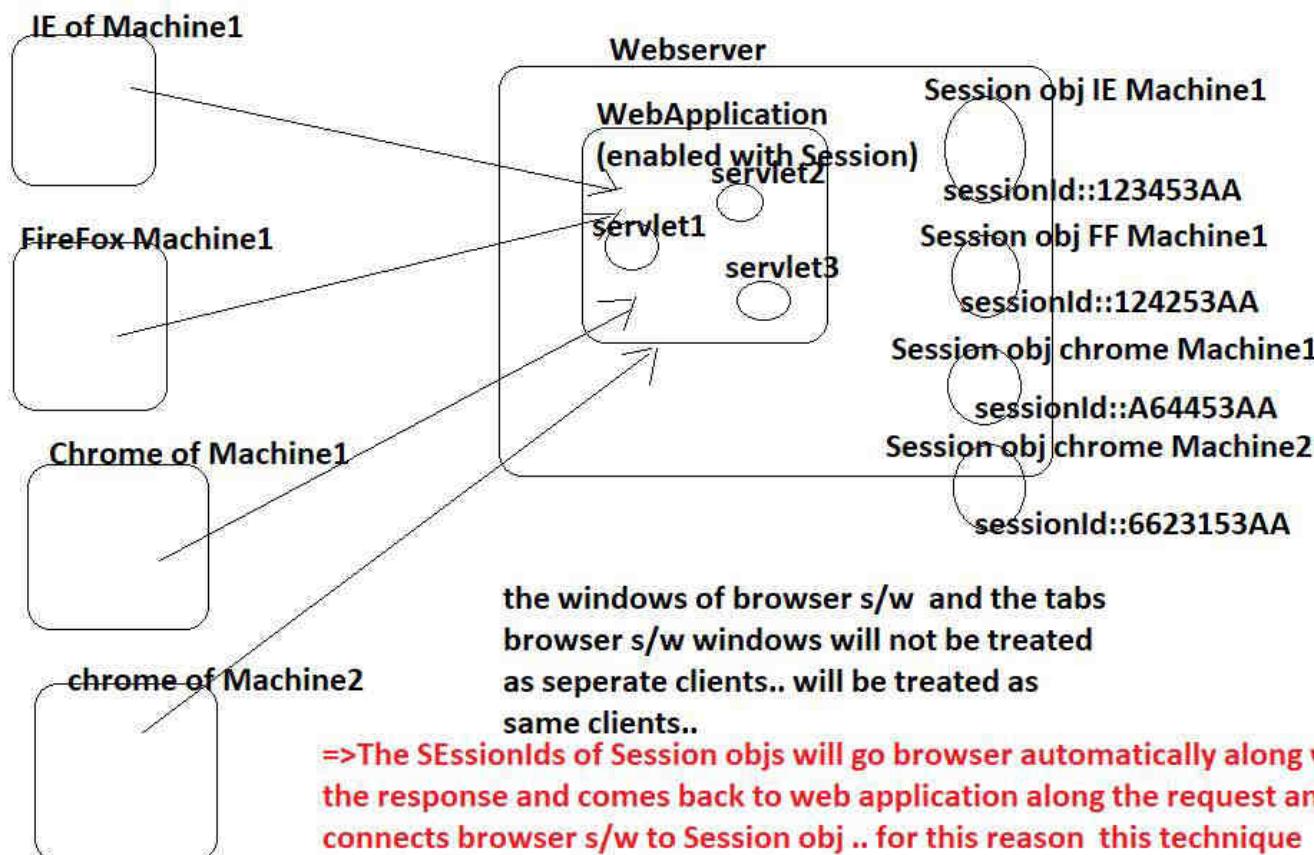
Http Session with Cookies

=> HttpSession object is created in web server on 1 per browser s/w of each client machine..

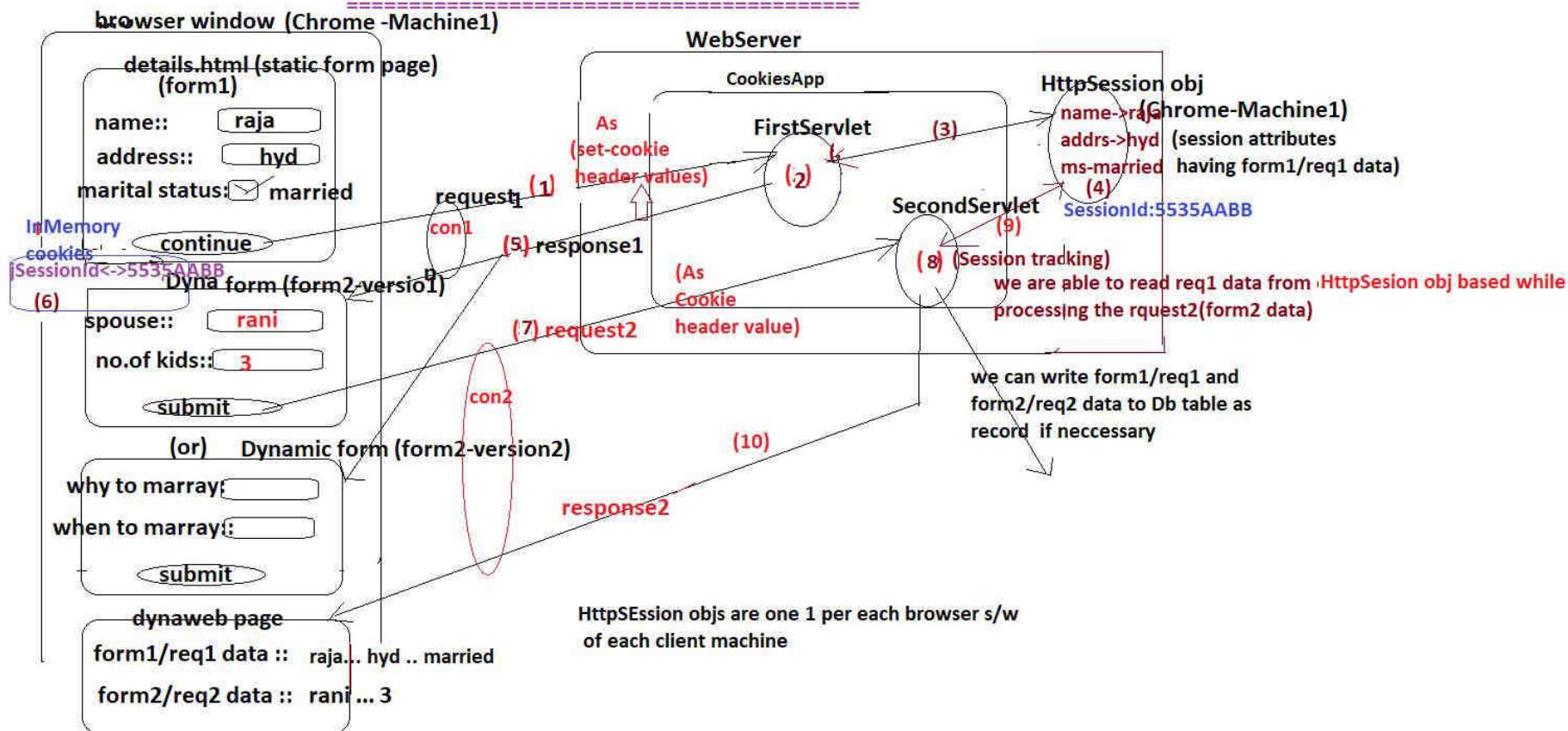
=> HttpSession obj contains data in the form of session attributes ..

=> HttpSession object and its session attributes are visible in all the web comps of web application , but they must get request from that browser for which session obj and session attributes are created i.e session obj and its session attributes are global with in the web application but specific to a browser s/w of client machine.

=> HttpSession object/Session object is nothing but it object of java class that implements javax.servlet.http.HttpSession(I).



HttpSession with Cookies based Session tracking



Session API (working with javax.servlet.HttpSession(I))

To create/Locate Session object for browser s/w of client machine being from Servlet comp

a) `HttpSession ses=req.getSession();`

(Locates/creates)

=> this Method creates new Session object in the server if session object is not already there for current client machine's current browser s/w otherwise it gets access to already available HttpSession object

b) `HttpSession ses=req.getSession(true);`

same as (a)

c) `HttpSession ses=req.getSession(false);`

=>this method does not create new Session object .. it makes ServletComp received always accessing existing Session object i.e it makes request always joining existing session.. if the session object is not available then it returns "null".

For FB login :: use (a) or (b)

for operations like posting , commenting , checking envets an etc.. use (c)

To invalidate the Session

=====

To make the SEssion object as inactive object
and also ready for garbage collection

(a) By closing current browser s/w window that is having session object on the server..

When we close the browser s/w or window , the InMemory cookies will be deleted including the cookie that is having Session Id.. So the session will not be continued..

b) By Calling ses.invalidate() method in servlet comp..
(useful for logout operation)

c) When Session's idle timeout period /Max inactive interval period is completed..

(In most of the servers the default is 30)

(a) Using programmatic approach (java code)

ses.setMaxInactiveInterval(1200); // 1200 seconds

(b) Declaration approach (xml code)

in web.xml

<session-config>
 <session-timeout>10</session-timeout>
</session-config> mins

if we specify the Session timeout in both programmatic and declarative Approches then what will be taken as the final value?

Ans) Programmatic becoz java code of serlet comp executes after InMemory Medata creation of web.xml , So the servlet comp settings will override web.xml's session timeout period.

To get Sesison Id

=====

String id=ses.getId();

To know wheather current req is creating new session object or not?

boolean b=ses.isNew();

if this method called Before the Session id of the SEssion object goes its browser from web application then it returns `true`.. otherwise returns "false".

//To get access to ServletContext obj

=====

ServletContext sc=ses.getServletContext();

To get Session creation time

```
=====
long ms=ses.getCreationTime();
Date d=new Date(ms);
```

To know the lastAccessed Date and time of Session obj

```
=====
long ms=ses.getLastAccessedTime()
Date d=new Date(ms);
```

We can put data inSession object in the form of attributes for this

we can use ses.setAttribute(-,-), ses.getAttribute(-,-),
ses.removeAttribute(-) method

=> **putValue(-,-), getValue(-) and removeValue(-)** are deprecated methods
of the above methods..

Advantages

=====

=> Works with all web technologies and servers

=> session attributes allocate memory the Session object of server..

Session attribute values will not travel over the network along with the
request,response So there will be less network traffic

=> In Session Attributes , we can keep any objects as the attribute values including
Strings..

=> We have control on to start session and to end session

=> we can specify idle timeout period for session

DisAdvantages

=====

=>Session objs allocate memory at server side ,So they give burden to server

=>This technique fails ,if the cookies restricted coming to browser or if the
cookies are deleted in the middle of the session .

HttpSession with URL Rewriting

=====

=> Gives to solve cookies related problem of 3rd technique..

=> In this we do not use cookies to send Session Id along with the request
and response.. But we send session Id appending the urls which goes
to browser from web application along with the response and comes back
to web application along with the request .. like **dynamic form action url,**
dynamic hyperlink href url and etc..

surl;jSessionId=4534545DFF322DA

=>we can do this work with method call

res.encodeURL("surl"); ---> gives surl;jSessionId= 4534545DFF322DA

```
pw.println("<form action='"+res.encodeURL("surl")+"' method='post'>");  
....  
....  
pw.println("</from>");
```

Advantages and DisAdvantages of HttpSession with URL Rewriting

=>same as HttpSession with cookies .. But this technique works though cookies are restricted coming to browser or cookies are deleted in the middle of browsing session.

Conclusion on SessionTracking

=>if the Session data is insensitive and the web application is having huge no.of customers then go for Http cookies

eg:: Online shopping cart , Google search results, job portal registration

=>if the Session data is sensitive and the web application is having less no.of customers then go for HttpSession with URL Rewriting

eg1:: Storing login username, password of banking app

eg2:: Storing login username, password of Stock trading app..

=> In one web application we can add both techniques like Http cookies and HttpSession with URL rewriting ..

eg::: In flipkart.com use http cookies for shopping cart activities .. and use http session with url rewriting for holding login credentials..

What is the diff b/w GenericSERvlet and HttpServlet?

GenericSERvlet	HttpServlet
=>It is protocol independent	=>it is specific to "protocol" http
=> It is abstract class with one abstract method	=> It is abstract class with no abstract method.
=>Support only hidden form fields based session tracking	=> supports all the 4 techniques of session trackings
=>we can not use "http" protocol features like autorefresh	=> we can use there
=>gives only one service(-,-) method as req processing method	=>gives 2 service(-,-) methods and 7 doXxx(-,-) methods as req processing methods..

Servlet Listeners (Event handling in Servlet programming)

Event :: An action performed on the comp or object

Event Handling :: Executing logic when the event is raised is called Event Handling.

Event Listener :: The comp that listens and handles the event by providing event handling method.

note:: Life cycle methods and event hadling will be called automatically based on the events that are raised..

To perform event handling we need 4 details

- (a) Source object (Like Button)
- (b) Event class (Like ActionEvent)
- (c) Event Listener (Like ActionListener)
- (d) Event Handling method (like actionPerformed(-) method)

ServletListeners allow Event handling on 3 objects of Servlet programming (request,session,servletContext)

=>Event handling on request object allows to keep track of when request obj is created and destroyed.this helps to findout the request procesing time of each request

=>Event handling on session obj allows to keep track of when session obj is created and destroyed.this helps to findout the session duration of a user..

=>Event handling on ServleteContext obj allows to keep track of when ServletContext obj is created and destroyed.this helps to monitor the web application execution..

source object	Event class	Listener Interface	EventHandling method
ServletReuquest obj	ServletRequestEvent	ServletRequestListener	requestInitialized() requestDestroyed()
HttpSession obj	HttpSessionEvent	HttpSessionListener	sessionCreated() sessionDestroyed()
ServletContext obj	ServletContextEvent	ServletContextListener	contextInitialized() contextDestroyed()

=ServletListeners are useful to enable monitoring on the Web applications..

=>sc.log("<msg>") method writes messages to current day's log file..

(<tomcat_home>\logs\localhost-<date>.txt)

=>To develop classes as listener classes take them as implementation classes of XxxListener interfaces

=>Every Listener must be cfg either using <listener> ,<listener-class> tags or using @WebListener annotation.

Upto servlet 3:: XxxListener (I) are normal java interfaces with no default methods

From servlet 4:: XxxListener (I) are java8 interfaces with default methods

During the deployment of the web application, the Servletcontainer creates the objects in the following order

- (a) All Listener classes objs (if Listeners are cfg)
- (b) ServletContext object
- (c) FilterConfig objects (if filters are cfg)
- (d) Filter classes objects (if filters are cfg)
- (e) Servletconfig objs
- (f) <load-on-startup> enabled ServletClasses object objs

java web frameworks give lots of built-in frontcontroller servlet comps...but cfg them with directory match or extension match url patterns is responsibility of the programmer

Struts ---->ActionServlet

spring MVC ---->DispatcherServlet

JSF ----> FacesServlet

=>The <load-on-startup> enabled on the FrontController Servlet comp performs pre-Instantiation & pre-initiatlization of FrontController SErvlet either during server startup or during the deployment of the web application,

=>The init() method of DispatcherServlet contains logic to create IOC container of type WebApplicationContext by taking WEB-INF/<DSlogicalname>servlet.xml as spring bean cfg file.

DispatcherServlet cfg in web.xml file

=====

```
<web-app>
  <servlet>
    <servlet-name>dispatcher</servlet-name>
    <servlet-class>org.sf.web.servlet.DispatcherServlet</servlet-class>
    <load-on-startup>2</load-on-startup>
  </servlet>
  <servlet-mapping>
    <servlet-name>dispatcher</servlet-name>
    <url-pattern>*.htm</url-pattern>
  </servlet-mapping>
</web-app>
```

(1) Deployment of Spring MVC App

(2) ServletContainer verifies Deployment
Directory structure

(3) ServletContainer loads
web.xml and creaes InMemory
MetaData after verifying for
well-formed or not , valid or
not

(4) Deployment Activities like SErvletContext
obj, ServletConfig object creation for
DS(DispatcherServlet) takes place..

(5) ServletContainer observes the <load-on-startup> enabled
DS in web.xml inMemory Data...

(6) ServletContainer loads and instantiates DS during the deployment
itself (pre-instantiation of DS)

spring-webmvc-<ver>.jar
(contains DispatcherServlet class and
we add this class to WEB-INF/lib folder of
Spring MVC App with other dependnet
jars)

ServletContainer creates Dispatcher
SErvlet class obj where as IOC container
created by the DispatcherServlet creates
the Spring Bean class objects like
handler classes

we cfg handler classes , service classes,
DAO classes as spring beans in spring
bean cfg file..

(7) As part of DS initilaization the happens during deployment (pre-initialization) the init() method DS executes and it creates IOC container of type WebApplicationContext by taking WEB-INF/<DSlogicalname>-servlet.xml (in our case WEB-INF/dispatcher-servlet.xml) as spring bean cfg file..

Here the SErvletConfig obj will also be initialized with DispatcherServlet

(8) IOC container loads spring bean cfg file and checks for well-formness and valid ness and also create InMemory MetaData of spring bean cfg file...

(9) Since the container is ApplicationContext IOC container,it performs pre-instantiation /eager Instantiation of singleton scope spring beans like handler classes, service classes , DAO classes and etc... And also completes dependency injectios among those classes.. keeps their spring bean class objs in the internal cahce IOC container..

note: 1 to 9 steps DS intiatiation and Intialization activities...

natarazjavaarena@gmail.com

Spring MVC Flow

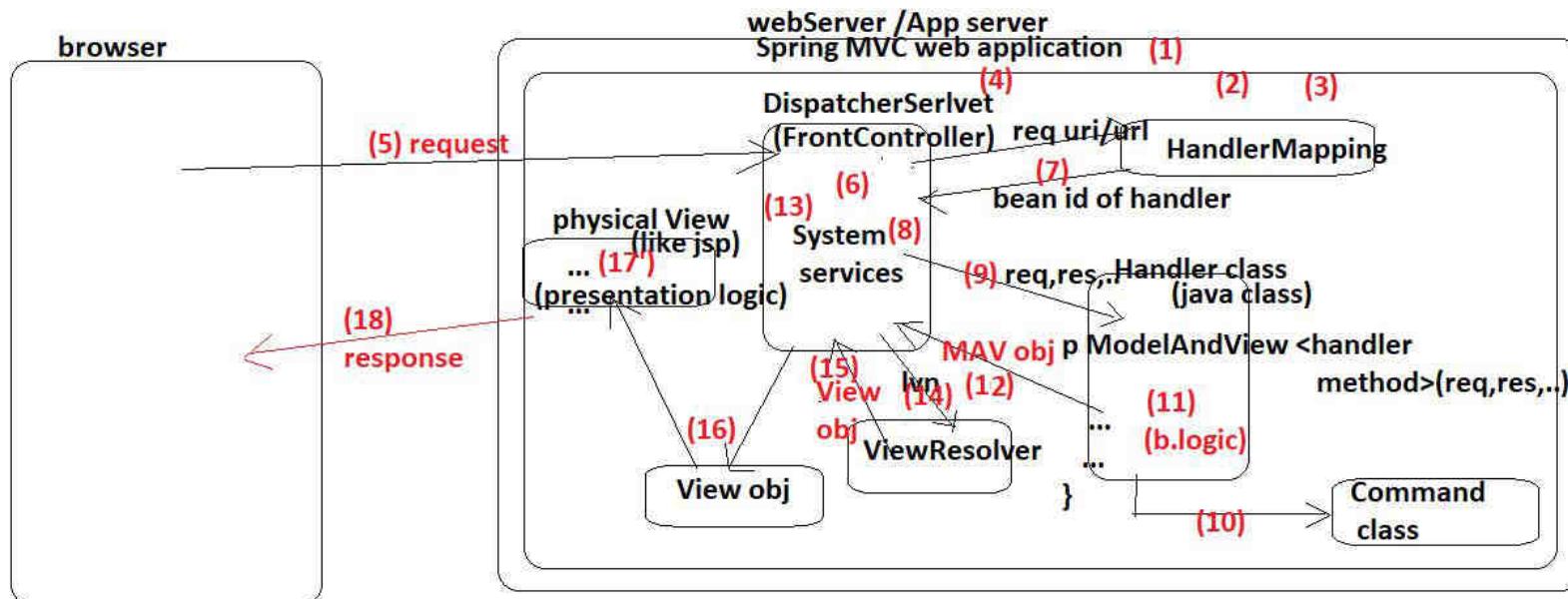
=====

Spring MVC framework is designed around DispatcherServlet ... so the entire flow is based DispatcherServlet navigation flow..

==>SpringMVC the java class/Java bean that holds form data is called command class..

==>DispatcherServlet maps multiple incoming requests urls with multiple handler classes by taking the support of "HandlerMapping" component

==>DispatcherServlet map lvn to physical view name with the support of "ViewResolver" component..



- (1) Programmer deploys the spring MVC Web application in WebServer or Application Server
- (2) ServletContainer verifies the Deployment Directory structure web application...
- (3) The spring MVC web application's deployment activities will happen like creating InMemory metadata of web.xml , creating ServletContext object and creating ServletConfig object for DS and etc...
- (4) DispatcherServlet related pre-instantiation and pre-initialization activities will take place like create DS object, creating IOC container , pre-instantiation of singleton scope beans and etc...
- (5) Browser gives request to Spring MVC web application..
- (6) As front controller, the DispatcherServlet traps and takes the request and applies common system services..
- (7) DS handovers the request to HandlerMapping comp. This comp takes incoming request uri and maps/links with Handler class and also returns that handler class bean id back to DS
- (8) DS receives bean id of Handler class from Handler Mapping gives DS managed IOC container by calling getBean(-) to get Handler class object..
- (9) DS calls handler method on handler class object having necessary arguments like req,res and etc.. objs
- (10) Handler class receives the request and writes the form data(req param values) to command class object(request wrapping) if the request is generated by form page otherwise this step will not take place.
- (11) Handler class either directly process the request by using b.logic or communicates with service,DAO classes to process the request and also keeps b.logic results ,logical view name (lvn) in ModelAndView class object
- (12) Handler method of Handler class returns MAV object (ModelAndView class obj) as having results and model data and lvn as view to DS
- (13) DS receives MAV object and collects model data from it and keeps in request scope
- (14) DS collects logical view name from ModelAndView object and gives View Resolver
- (15) View resolver resolves/identifies the name and location of physical view (like jsp) based on the received lvn and returns View object (View(I) implementation class obj) having the name^{and} location^{of}Physical View comp.
- (16) DS calls render() method on View object to send navigation to Physical View comp.
- (17) The physical View comp like jsp receives results/model data from request scope and formats the results using presentation logics
- (18) Physical View comp sends formatted results to browser as response.

place

Spring MVC recommends to jsp files inside WEB-INF folder(private area), not outside of WEB-INF folder(public area)

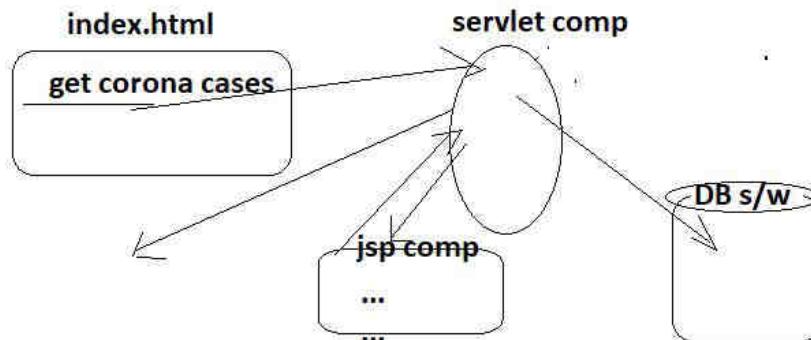
Reasons are

- (a) if jsp is placed in public area then there is chance requesting that jsp file by specifying its name directly in the address bar request url or in href of <a> or in action of <form> tags ,all these things may reflect jsp file names in the browser address bar and it may give chance to hackers to know the technology in which web site is developed and helps them hacking the web application..

FirstApp
|--->WEB-INF
|-->abc.jsp (wrong practice)

FirstApp
|--->WEB-INF
|--->pages
|--->abc.jsp (good practice)

(b)



if jsp is designed to display request scope data gathered from servlet comp and if that jsp page is there in public area then there is a possibility of giving direct to jsp page without giving to servlet comp and this process may display ugly or null values on the browser... To overcome this problem place jsp page in private area and stop direct requests jsp files..

Code based Flow of Spring MVC App (Xml driven cfgs)

DispatcherServlet is FrontController and Handler classes are called controller classes...

we can develop spring MVC in 4 approaches

- (a) using xml driven cfgs
- (b) using annotation driven cfgs
- (c) using 100% java code /java config driven cfgs
- (d) spring boot driven cfgs

Handler class/Controller class

- => It is java class either request processing logic directly or having logic to delegate the request to service class.
- => Java class must implement org.springframework.web.Mvc.Controller(I) directly or indirectly to make it as controller /handler..
- => Every Handler class must be cfg in spring bean cfg file having bean id.

MyController.java

```
=====
public class MyController implements Controller{  
  
    public ModelAndView handleRequest( req,res )throws Exception{  
  
        // b.logic  
        Date d=new Date();  
        return new ModelAndView("result","sysDate",d);  
    }  
}
```

lvn model attribute name, value.

in dispatcher-servlet.xml

```
<!--Controller/Handler class cfg -->  
<bean id="mc" class="pkg.MyController"/>
```

Handler Mapping

=====

=> This comp maps incoming requests with handler classes i.e it lks incoming request urls/uris with Handler classes bean ids and returns those bean ids back to DispatcherSErvlet..

=> All Handler Mappings are java classes implementing org.sf.web.servlet.HandlerMapping(l).. Every HandlerMapping must be cfg in spring bean cfg file...

--> The popular HandlerMapping classes are BeanNameUrlHandlerMapping,SimpleUrlHandlerMapping and etc...

in dispatcher-servlet.xml

=====

```
<!-- handler mapping cfg -->  
<bean id="suhm" class="org.sf.web.servlet.handler.SimpleUrlHandlerMapping">  
    <property name="mappings">  
        <props>  
            <prop key="home.htm">mc</prop>  
            ....  
        </props>  
    </property>  
</bean>
```

<value> or valuee attribute :: for simple values injection
<ref> or ref attribute :: for Bean objs injection based on bean ids
<list> :: for injecting values java.util.List type bean property
<set> ::for injecting values java.util.Set type bean property
<map> :: for injecting values java.util.Map type bean property
<props> :: for injecting values java.util.Properties type bean property

ViewResolver

==> This comp takes lvn from DispatcherServlet returns View object back to DS having physical view name and location.. and DS uses this View obj to transfer/render the control to View comp..i.e it maps lvns with physical view names with locations.

==> All ViewResolvers are classes implementing org.sf.web.servlet.ViewResolver (I)... The readymade view resolvers are InternalResourceViewResolver,XmlViewResolver,ResourceBundleViewResolver and etc..

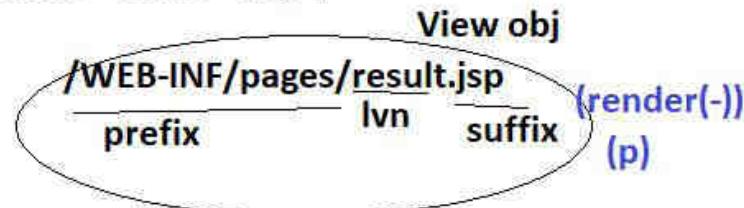
useful to locate the view comps that are internal to web application (jsp /servlet comps placed in private area)

==>The InternalResourceViewResolver and some ViewResolvers can get name and location of private jsp comps /servlet comps eventhough they are not cfg in web.xml having url pattern..

dispatcher-servlet.xml

(n)

```
<bean id="irvr" class="org.sf.web.servlet.view.InternalResourceViewResolver">
    <property name="prefix" value="/WEB-INF/pages/"/>
    <property name="suffix" value=".jsp"/>
</bean>
```



```

result.jsp (WEB-INF/pages) (q)
=====
<%@page isELIgnored="false"%>
    result is :: ${sysDate} (r)
        EL code

```

DispatcherServlet cfg in web.xml file

```

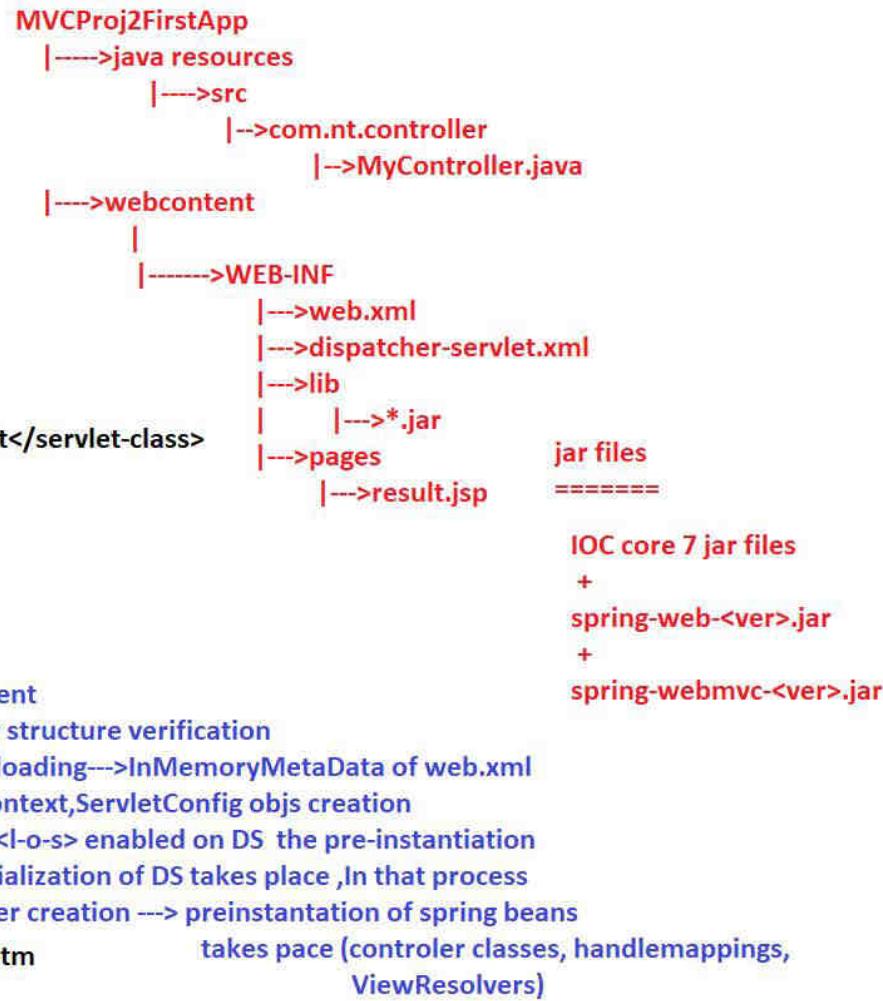
=====
<web-app>
    <servlet>
        <servlet-name>dispatcher</servlet-name>
        <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>
        <load-on-startup>2</load-on-startup>
    </servlet>
    <servlet-mapping>
        <servlet-name>dispatcher</servlet-name>
        <url-pattern>*.htm</url-pattern>
    </servlet-mapping>
</web-app>

```

request url from browser

<http://localhost:3030/MVCProj2-FirstApp/home.htm>

(f)



index.jsp (webcontent)

```

=====
<jsp:forward page="home.htm"/>

```

request url from browser

<http://localhost:3030/MVCProj2-FirstApp/>

=>Spring MVC gives `org.sf.web.mvc.servlet.DispatcherServlet` as pre-defined Front Controller Servlet (available `spring-web-<ver>.jar` file)

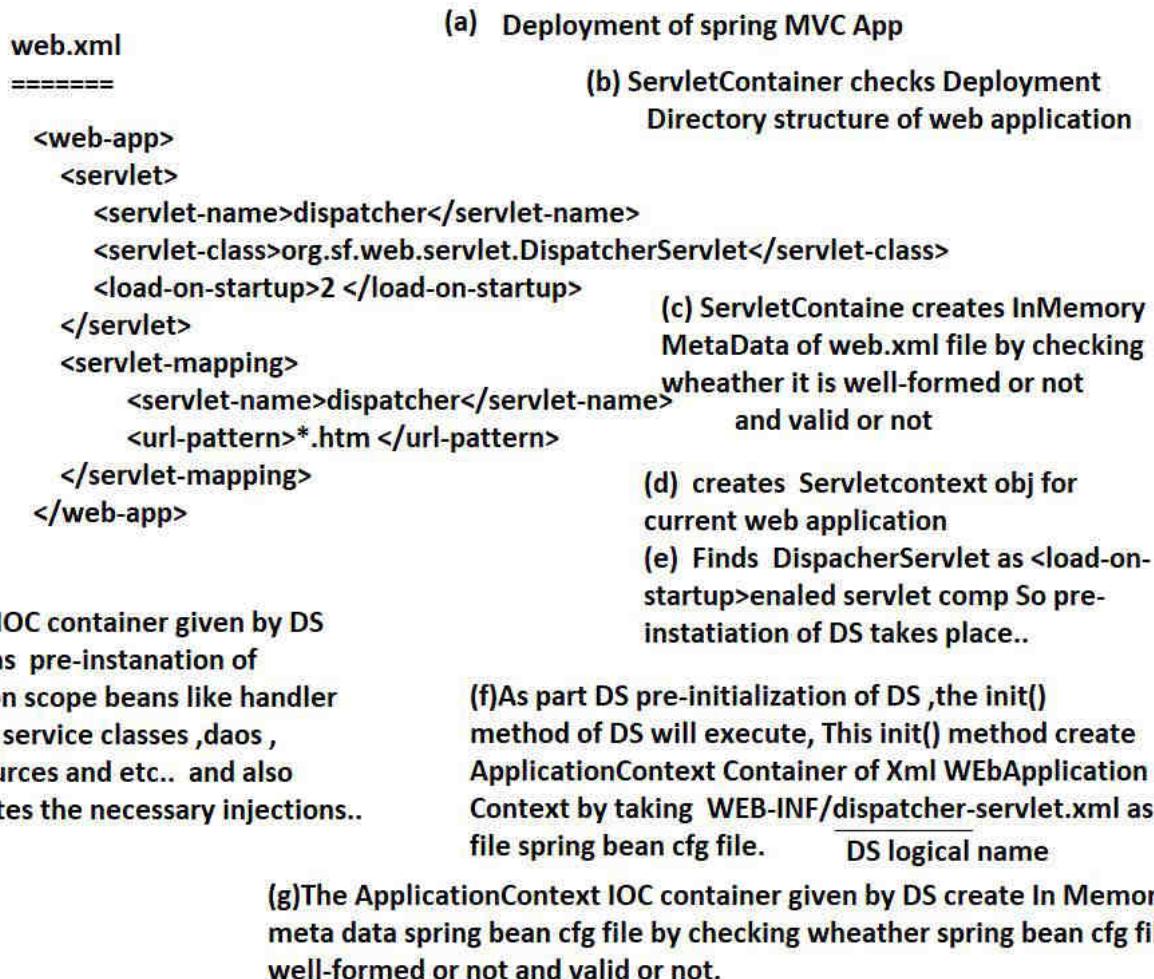
=> Though `DispatcherServlet` is pre-defined Servlet comp, its cfg in `web.xml` file is mandatory.. in order to make Servlet container recognizing and managing the life cycle of DS..

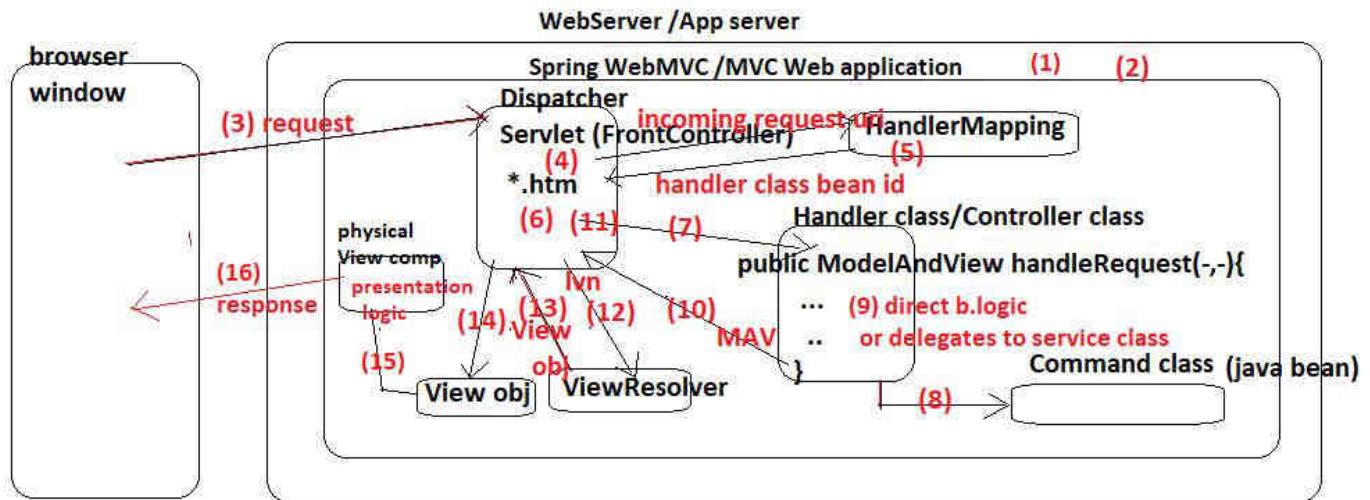
The `init()` method of `DispatcherServlet` contains logic create `XmlWebApplicationContext` IOC container by taking `<DSlogicalname>-servlet.xml` of WEB-INF folder as spring bean cfg file.. This IOC container contains cfg of Handler classes, Service classes , DAO classes, DAta source classes as spring beans ..

=>Servlet container of Spring MVC app takes care the life cycle management of FrontController `DispatcherServlet` and jsp comps.

=> IOC container created DS takes care of the handler, service, dao and etc.. classes bean life cyclement and their injections..

=>It is always recomanded to cfg DS in `web.xml` by enabling `<load-on-startup>` in order to see DS,spring Beans(singlscope) pre-instantiation either during server startup or during the deployment of web application..





- (1) Programmer deploys Spring webMVC web application in webServer /App server..
- (2) All pre-instantiation and pre-initialization related activities will take place on DispatcherServlet either during server startup or during the deployment of spring WebMVC Web application.. as discussed in DispatcherServlet's internal flow..
- (3) Browser generates request to spring mvc web application with or without form
- (4) DispatcherServlet traps , takes the request and applies common system services on the request
- (5) DS handovers the request to HandlerMAppling comp and this comp takes incoming request uri , maps with handler class and returns Handler class bean id back to DispatcherServlet.
- (6) DS receives the bean id of Handler class and gets Handler class obj from Internal IOC container created by DS itself by calling ctx.getBean(-) method.
- (7) DS Calls the handler method handleRequest(-,-) on Handler class object
- (8) Handler class will write the received from data (req params) to Command class object.
- (9) The handler method of handler class either directly process the request or delegates to service class to process the request..
- (10) Handler method returns ModelAndView object having b.logic generated result as Model Data and logical View Name as the View..
- (11) DS receives ModelAndView obj and keeps model data (results of b.logic) in request scope
- (12) DS collects logical View name (lvn) from ModelAndView object and gives ViewResolver comp
- (13) ViewResolver resolves/identifies physical View based on the received logical view name and returns View Object having physical view name and location.
- (14) DispatcherServlet calls render(-) method on received View object and renders & the control to the physical View comp
- (15)
- (16) In Spring MVC physical view is an abstract entity i.e we can take html or jsp , spring bean or freemarker or velocity and etc.. comps as view comps .. This physical view comp executes presentation logic and formats the results by collecting from request scope and sends the results to browser as response..