javax.servlet .\*

javax.servlet.http .\*

using these interfaces abstraction is promoted through which WODA is achieved

ServletRequest(I)

ServletResponse(I)

HttpServletRequest(I)

HttpServletResponse(I)

ServletContext(I)

ServletConfig(I)

void init(ServletConfig config) throws SE,IOE

}

void doXXXX(HttpServletRequest request, HttpServletResponse response)throws SE,IOE{

}

//request => QueryString data in the form of key-value pair

//response=> PrintWriter object will be there with empty response

w.r.t tomcat server the implementation classnames are as shown below

===========================================================

Implementation class of config is :: org.apache.catalina.core.StandardWrapperFacade

Implementation class of context is :: org.apache.catalina.core.ApplicationContextFacade

Implementation class of request is :: org.apache.catalina.connector.RequestFacade

Implementation class of response is :: org.apache.catalina.connector.ResponseFacade

Eg: WODANature

Different types of scope and attributes in Servlet

======================================

scope refers to the accessability of a variable.

a. local scope => restricted inside method

b. global scope=> available in all the methods of a particular task.

In Servlet we have 3 types of scope

1. request

2. session (HttpSessionTracking)

3. application/context

There are 3 types of parameters(k,v) possible in servlet

a. Form parameters (QueryString[k,v])

b. ServletInitializationParameters (ServletConfig[k,v])

c. ContextInitializationParamters (ServletContext[k,v])

The above 3 parameters type are read-only.from the servlet we can perform only read operation, we cannot modify remove values based on our requirement.so we say parameter type of data is not best suited for sharing the data between component of the webapplication.

parameter data => both key and value should be String.

To resolve this problem we should go for "attributes" type of data.based on our requirement we can add the data, remove the data and we can share the data between the components of the application.

attribute data => key should be String, value can be any object.

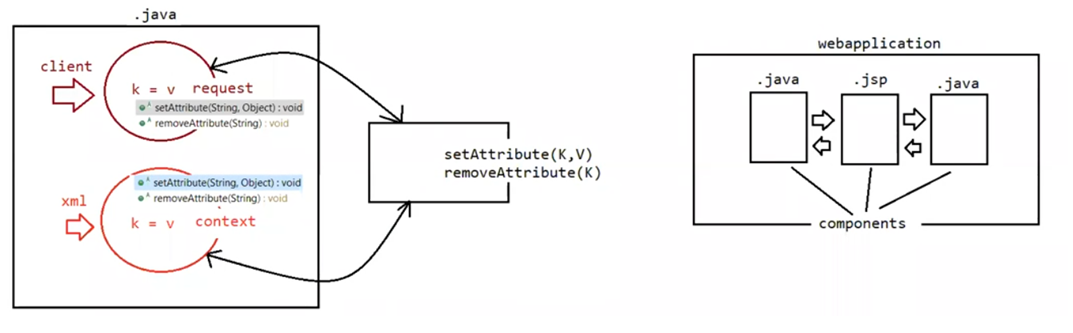
Based on our requirement we need to store the attributes in particular scope.

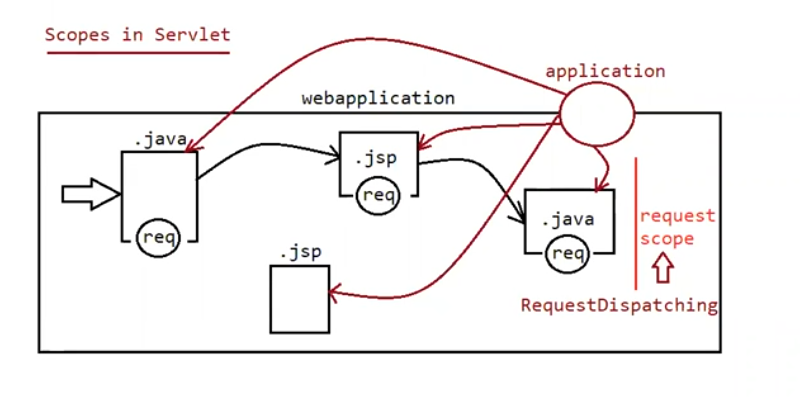
In Servlet we have 3 types of scope

1. request

2. session (HttpSessionTracking)

3. application/context





1.request

=> This scope is maintained by ServletRequest /HttpServletRequest object.

=> This scope will start at the time of request object creation (before calling service())

#> This scope will destroy at the time of request object destruction (after calling service())

=> The data stored in the request object will be available for all components which are processing that request.

3. application/context

=> This scope is maintained by ServletContext object.

=> This scope will start at the time of context object creation (during deployment)

=> This scope will destroy at the time of context object destruction (during undeployment)

=> The data stored in the context object will be available for all the components of the application, irrespective of request and the user.

write a program to display hit count (number of requests) of a webapplication?

Eg: ScopeApp

Write a program to display all the attribute information present in application scope?

Along with the attributes data container will add some attributes for internal information

Eg: ScopeApp

Getting information from the url

1. getRequestURI()

2. getQueryString()

3. getServletPath()

4. getPathInfo()

5. getContextPath()

Eg: RequestAppInfo

Deployment

=============

Harddeployment

Creating an webapplication inside webapps folder of tomcat and starting the server manually is called "hard-deployment".

Smoothdeployment

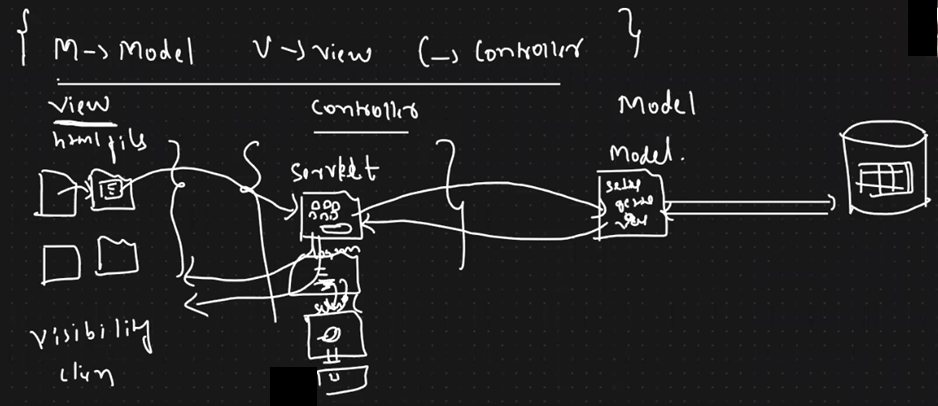
Creating an application outside the webapps folder of tomcat and starting the server through some additional set up is called "smooth-deployment".

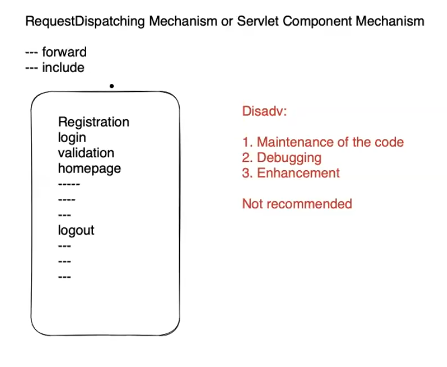
webapps(deployment folder)

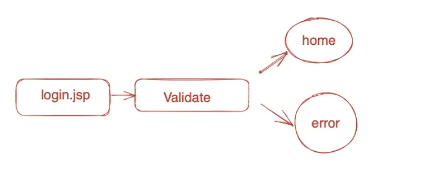
In case of eclipse integration with tomcat, internally eclipse uses "smooth" deployment through which it clones our project and perfoms deployment as shown in the following path

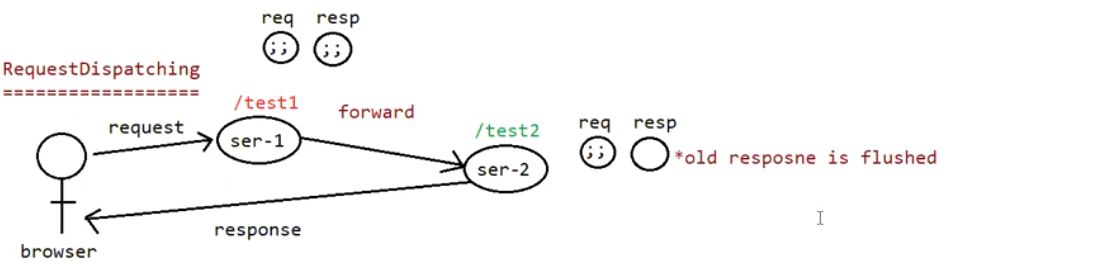
\*\.metadata\.plugins\org.eclipse.wst.server.core\tmp0\wtpwebapps

MVC

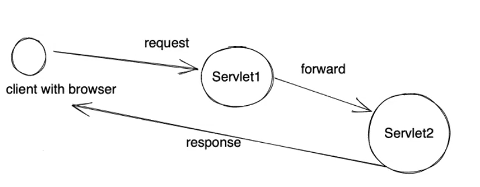




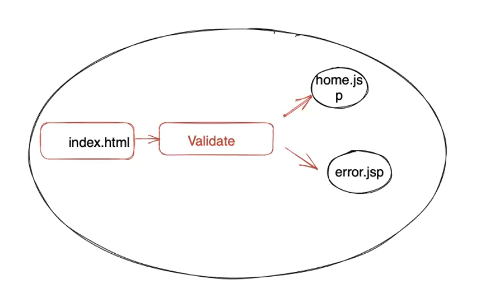




So data in servlet-1 println will not get printed.

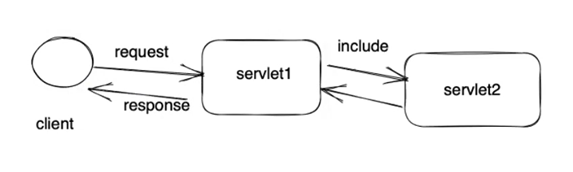


Eg: ServletDispatching

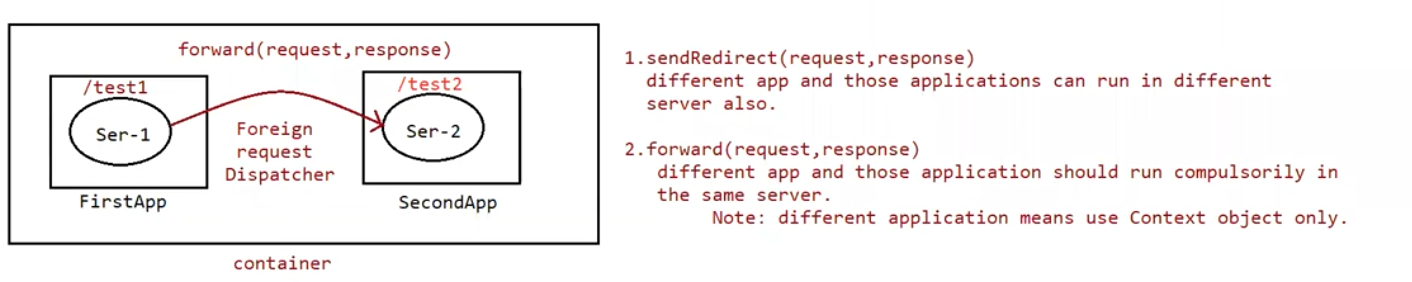


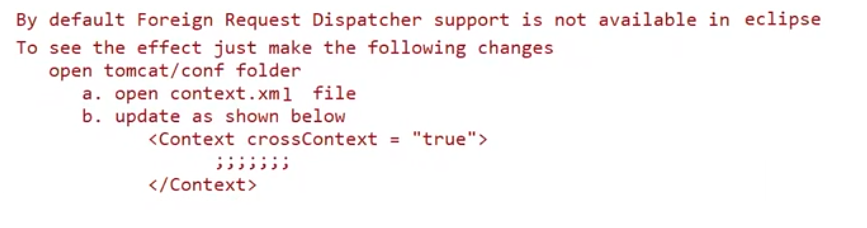
Eg: ServletRequestDispatch2

Eg: ServletDispatchAttribute



Eg: RequestDispatchInclude

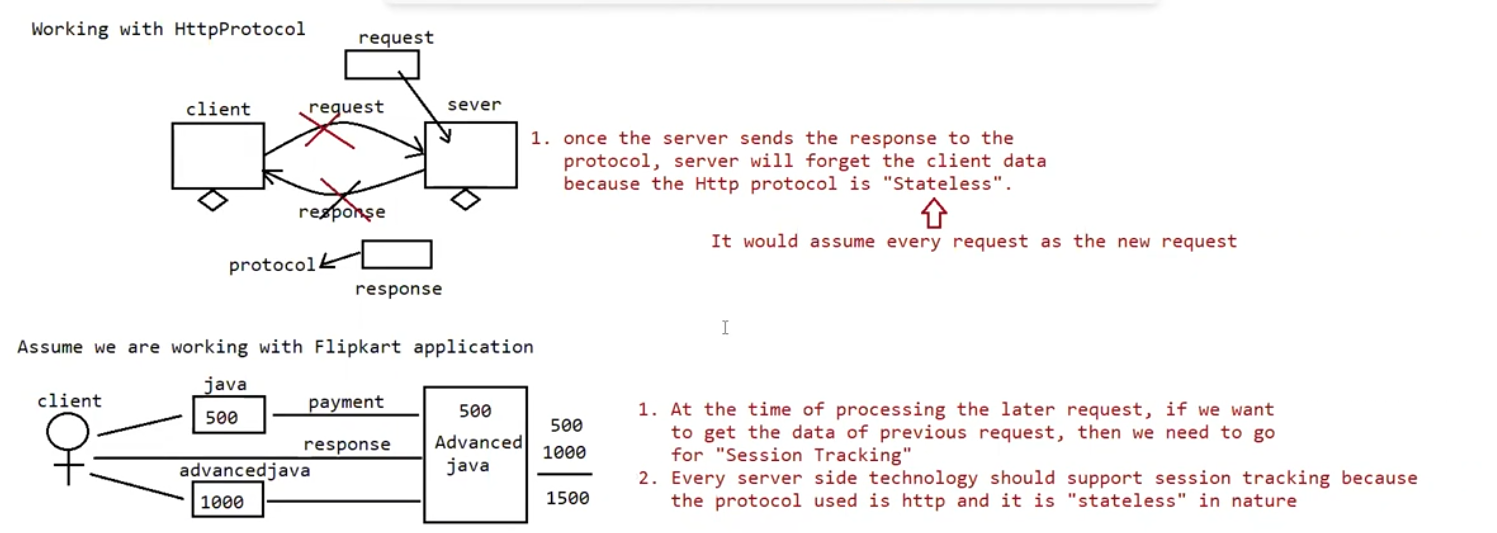


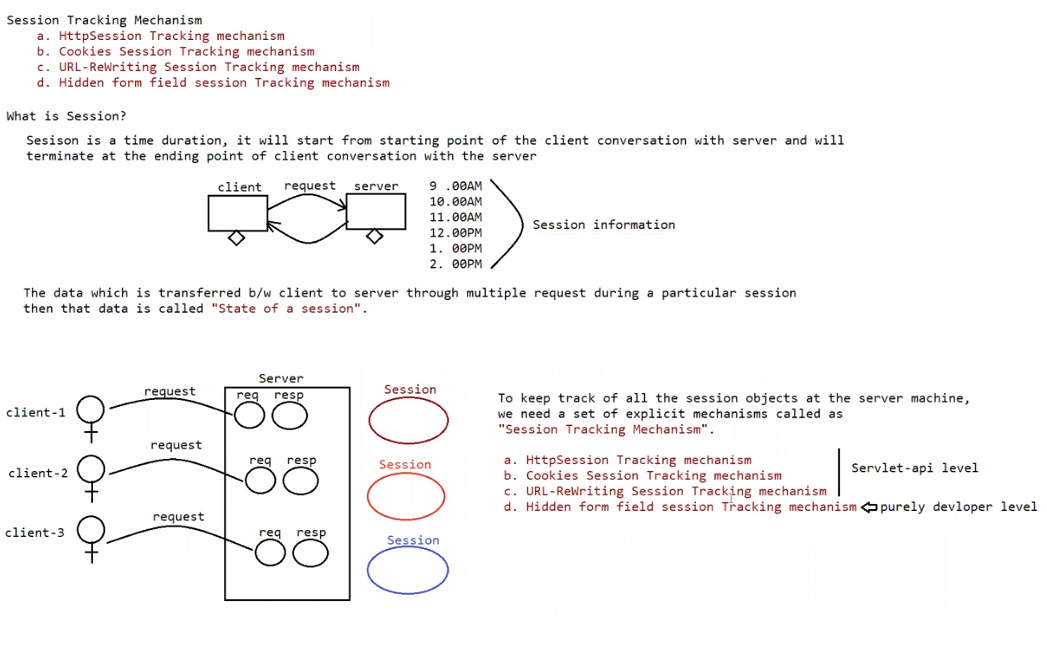


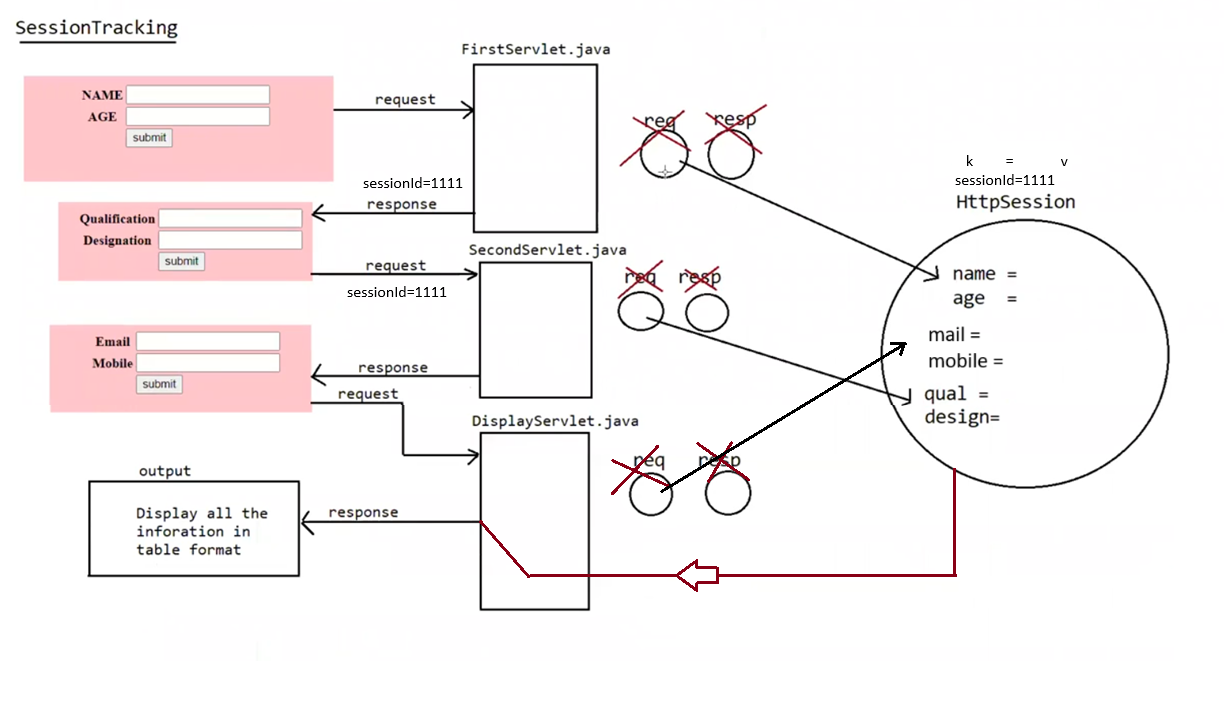
Eg: ForeignRequestDispatchApp1

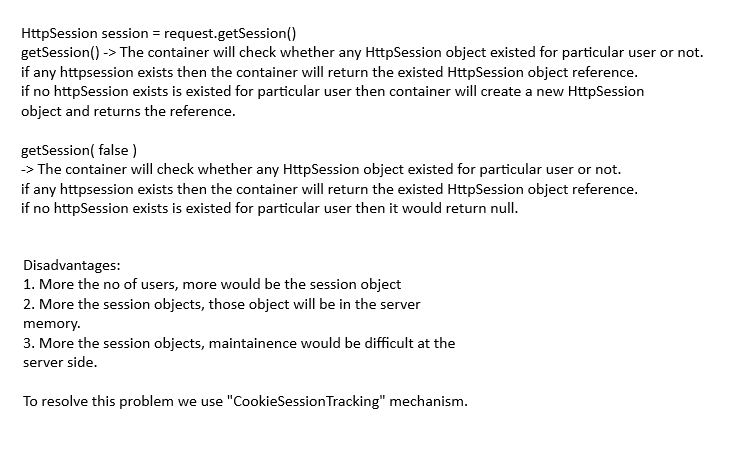
Eg: ForeignRequestDispatchApp2

Session Tracking:





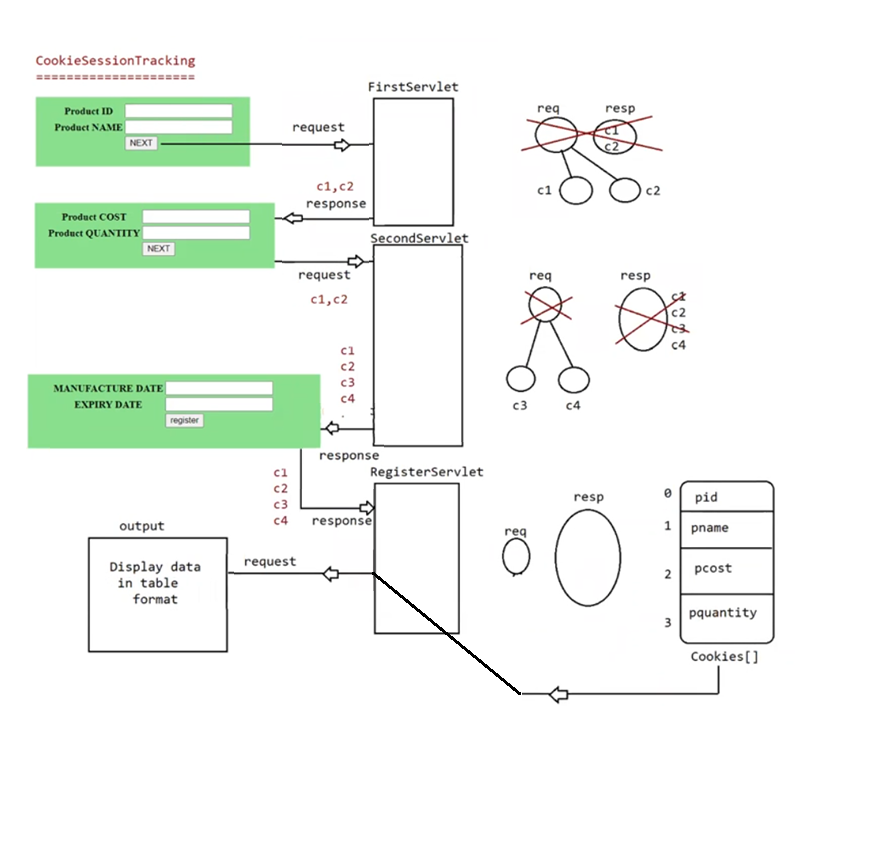




Eg: HttpSessionTracking

Note: In case of HttpSessionTracking mechanism, if the client disables cookies then HttpSessionTracking mechanism won't work.

Cookies Session Tracking:



Eg: CookiesSessionTrackingApp

Disadvantage

============

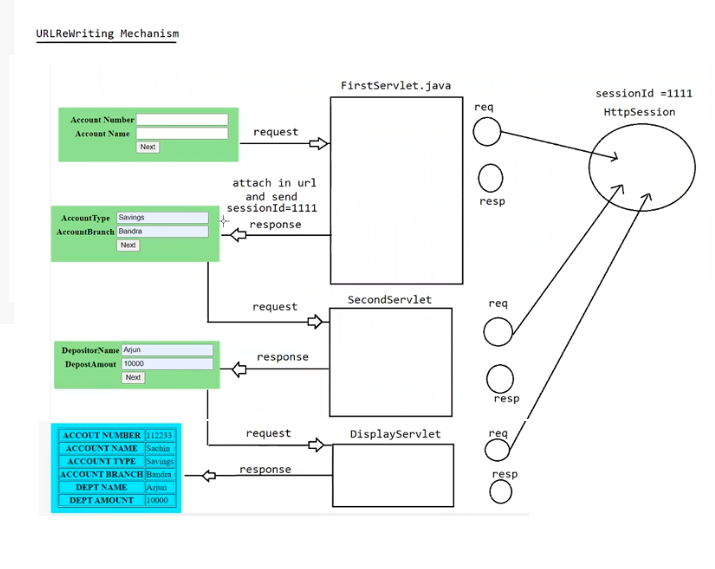
1. Maintainece of client data is happening at the broswer

2. If client disables cookies, then SessionTracking won't happen.

3. Since cookie is exposed to the client, it is not a safe approach.

To overcome this limitation we need to use "URLReWriting".

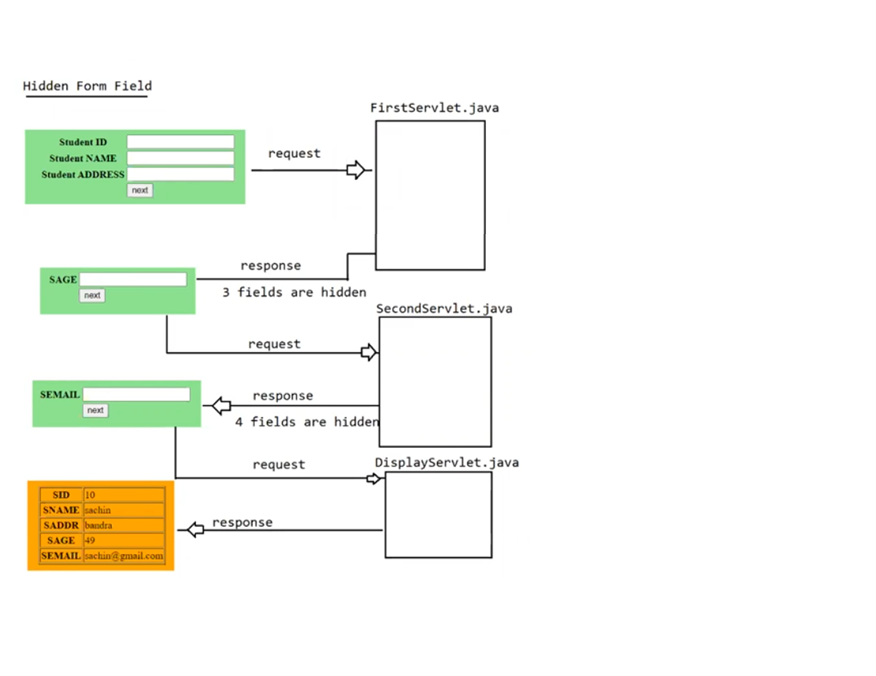
URLRewritingSessionTracking:



Eg: URLRewritingApp

Hidden Form Field Session Tracking:

=============================



Eg: HiddenFormFieldApp