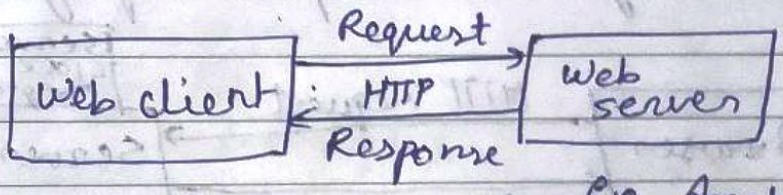
Java web development

Web application any software application whose services are accessed via the ~~used~~ web e.g - Udemy, Google, Fb



E.g chrome, safari, firefox etc

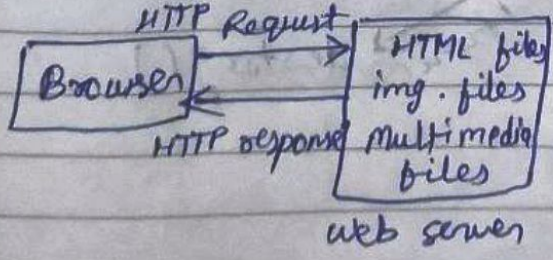
E.g Apache, IIS

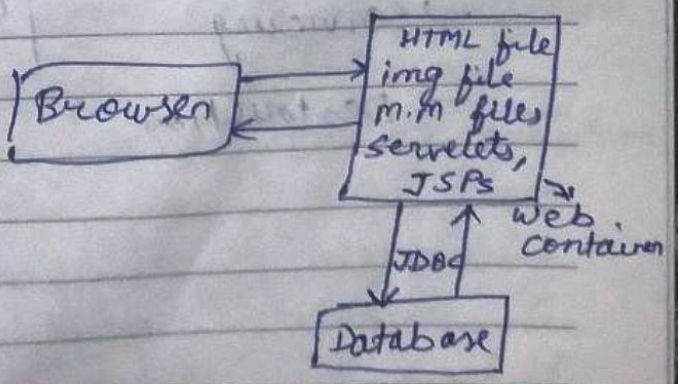
URL - http:// www.udemy.com : 80
protocol domainname Port no

Web application

Static (Non interactive web app)

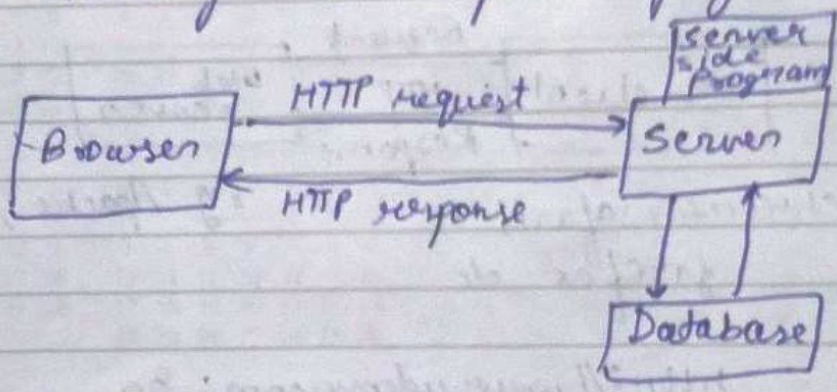
Dynamic (interactive web app)





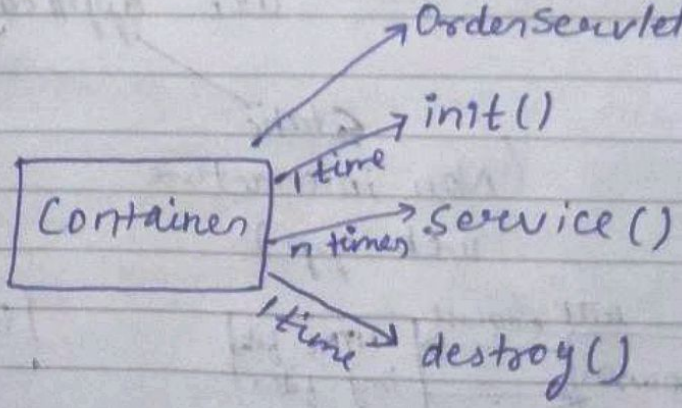
Server Side programming

→ Capturing the user i/p
→ Communicate with the database
→ Processing of data
→ Produce the response page
→ Handing the response page to the server



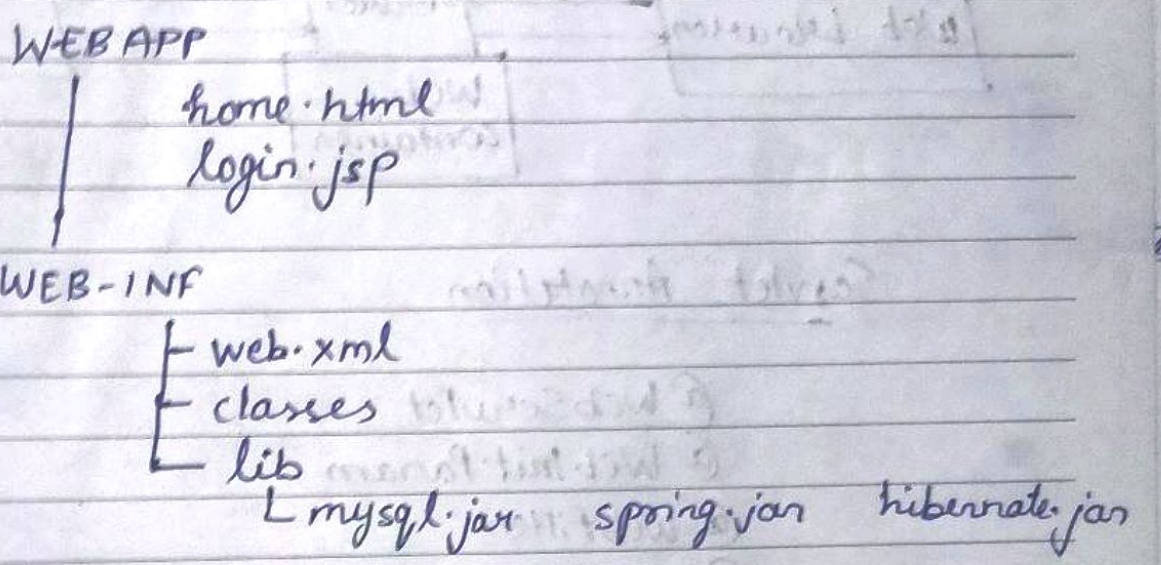
Servelet Life cycle method & phases

phases
→ instantiation
→ initialization
→ servicing
→ Destruction

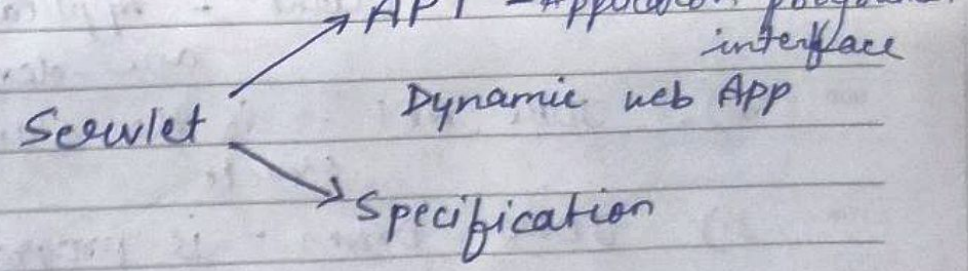


10 Sunday

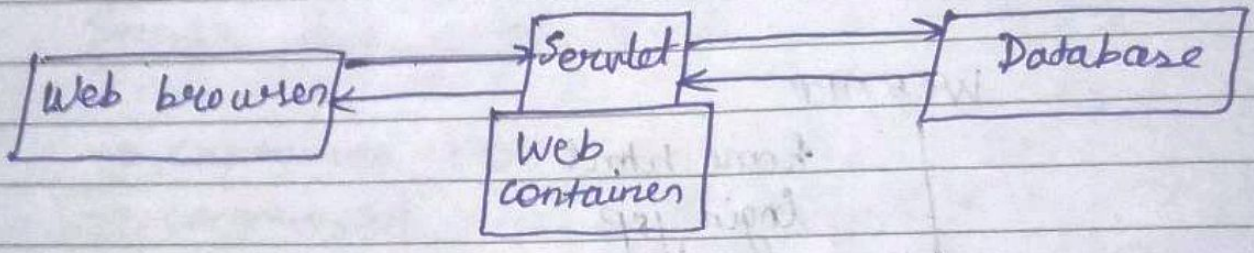
Web-App folder Structure



Servlets - Technology in a java EE standard that allows us to develop dynamic web application in java



→ Servlet on the other hand is a java program that runs on the server or the web container. It can perform following operation i.e it can capture the request come from web browser, processes it & depending on request it perform operation & take response from database & send it back to web browser.



Servlet Annotation

@ WebServlet
@ WebInitParam
@ Webfilter
@ Weblistner

JDBC Architecture It has 4 components

1.) JDBC client - Application code which we are developing

2.) JDBC API - is the standard API from oracle

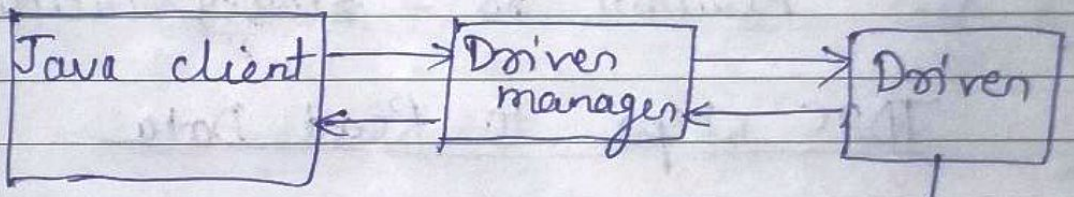
3.) JDBC Driver - is program which is an interface b/w our JDBC client & underlying database

4.) Driver Manager - is helper class which finds driver & establishes connection to database

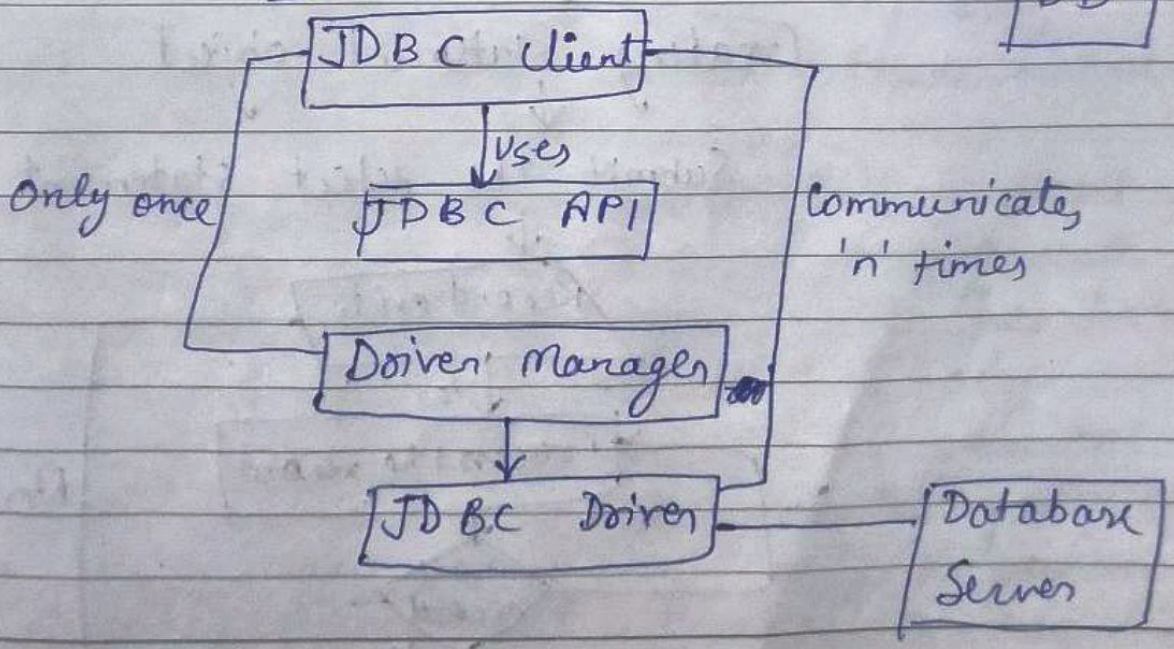
JDBC client → performs following

1.) Connects to database
2.) Perform CRUD (Create Read Update delete)
3.) Process the response
4.) Handle the exceptions
5.) DO transaction managment
6.) Close the connection.

Driver Manager



JDBC Architecture



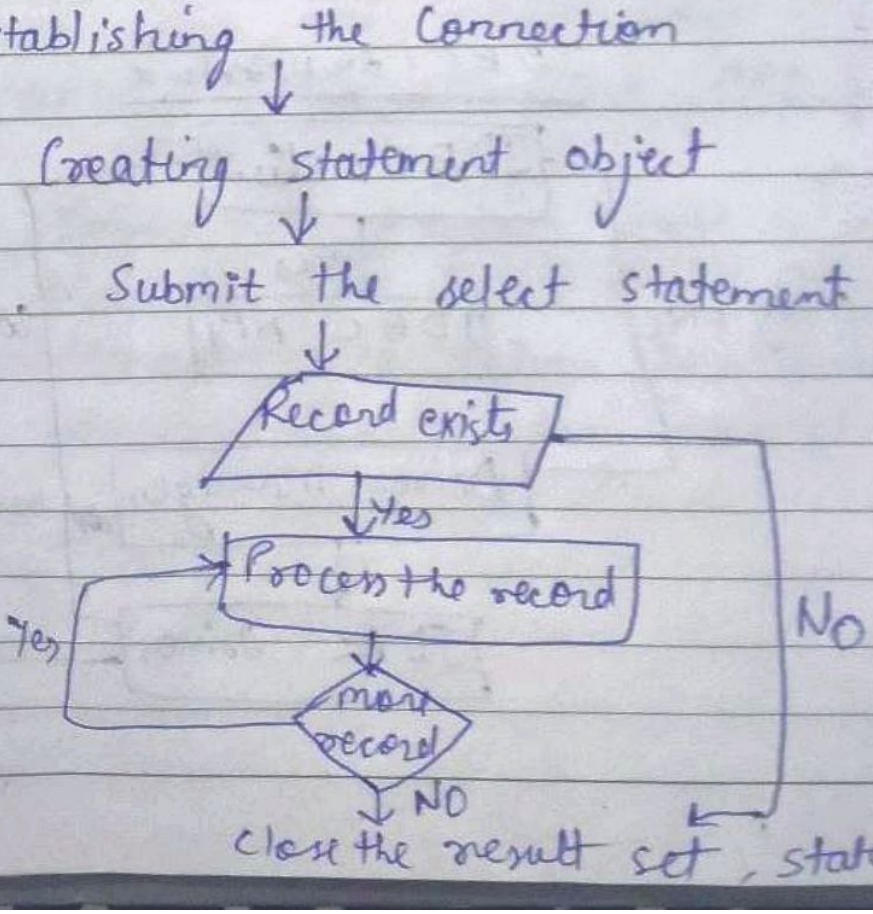
Steps to perform CRUD

S-(i) Establish the connection
S-(ii) Create the statement object
S(iii) Submit the SQL query to DBMS
S(iv) Close the statement
S(v) Close the connection

Statement interface

```
Statement stmt = con.createStatement();
int result = stmt.executeUpdate("");
ResultSet rs = stmt.executeQuery("");
```

JDBC steps to Read Data



ResultSet interface in JDBC API is use to handle the data that comes back when we execute a select query.

A result set is a object oriented representation of table of record retrive from database.

Get & Post

Http protocol support several method using which a web client can access a server two of popular methods are Get & post.

Get V/s Post

Get	Post
- Get is use by default	- Post has to be mentioned explicitly <form method = "Post"...
- Get is used in 3 cases i) User specified URL ii) User click on a hyperlink iii) User submits the form	- It is meant for posting data
- It is meant for getting data	- It has both header & body
- It only has header field doesn't have body	- The goes to server as the part of http body or payload.
- All the data that we submit from browser goes as query string	- We can use this for sensitive data
- We should'nt use to send sensitive data	

→ The query string has restriction on length so we have restriction on amount of data that we sent using get method

→ Get is idempotent. means we can execute get any no. of times but it will not affect state of application.

→ we can send any amount of data.

→ It is non idempotent

Init Params are the name value pairs of textual information that can be passed to a servlet through the web.xml. You can configure the init params in the web.xml right where we declare the servelet

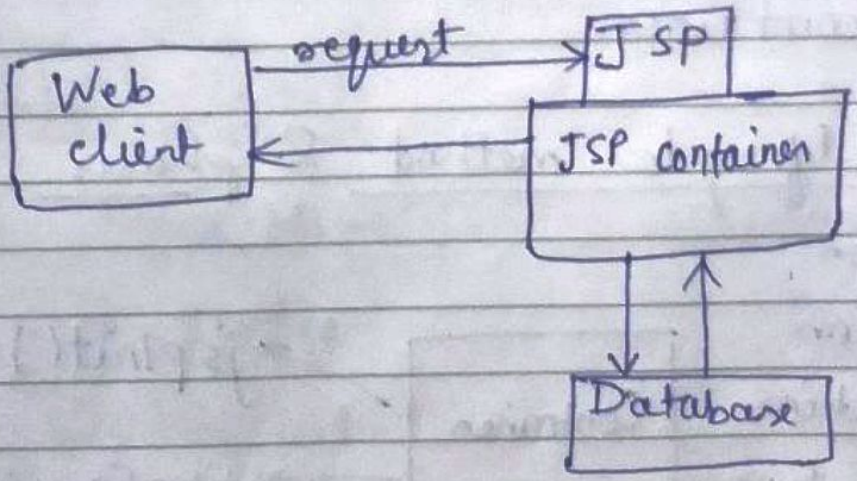
17 Sunday

```
<servlet>
<servlet-name> ...
<servlet-class> ...
<init Param>
<param-name> db user </param-name>
<param-value> root </param-value>
</ init param>
```

Notes

JSP (Java Server Pages) is a tech in Java EE space that comes with API & specification.

JSP
- API (it is for app developers)
- Specification (set of rules that should be followed by JSP container creator)



→ It overcome limitation of Servlet

JSP Elements

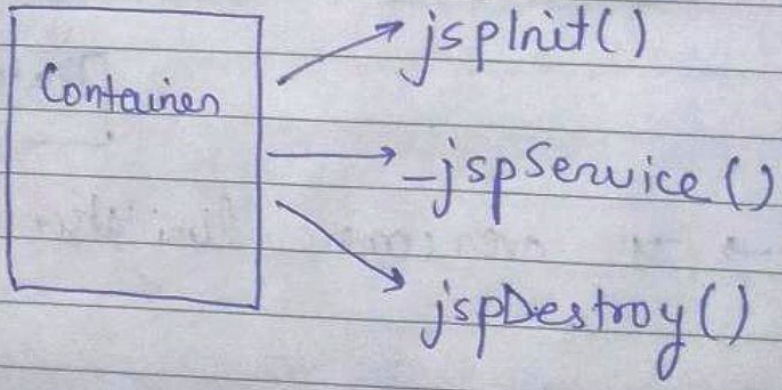
1.) scripting Elements
→ To embed java code in to a JSP Page

2.) Directives
→ Translation time instruction to the container which will allow us to import java other class lib, or files etc.

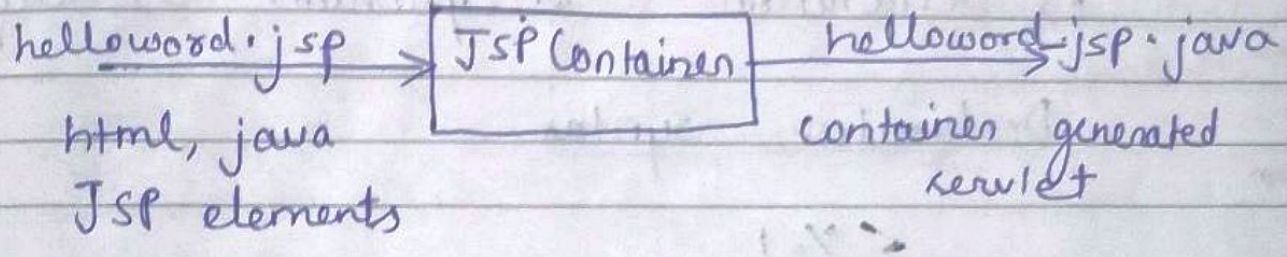
3.) Actions
Runtime instructions to the JSP container.

JSP life cycle method & phases

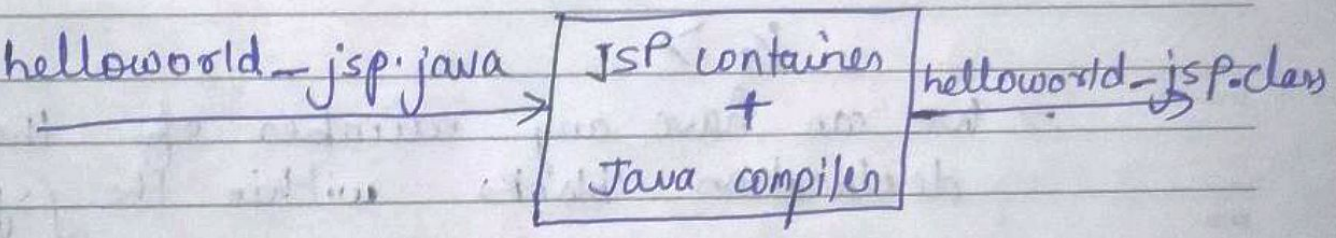
Translation
Compilation
Instantiation
Initialization
Servicing
Destruction



Translation



Compilation



Implicit Objects

Object Name	Type
i) config	ServletConfig
ii) request	HttpServletRequest
iii) response	HttpServletResponse
iv) session	HttpSession
v) application	ServletContext
vi) page	java.lang.Object
vii) pageContext	javax.servlet.jsp.PageContext
viii) exception	java.lang.Throwable
ix) out	javax.servlet.jsp.JSPWriter

Scripting Elements

1) Declaration 2.) Expression 3.) Scriptlet

Declaration syntax

```
<%!
    Any java code here
%>
```

→ We can have any number of these declaration blocks within the jsp page

Expression syntax

```
<%= Any java expression that evaluates a value. %>
```

E.g `<%= user.getName() %>`

Evaluates & send the response

Scriptlet

```
<%
    Any no. of java statements
%>
```

E.g

```
<%
String num1 = request.getParameter("num1");
String num2 = request.getParameter("num2");
int result = num1 + num2;
%>
```