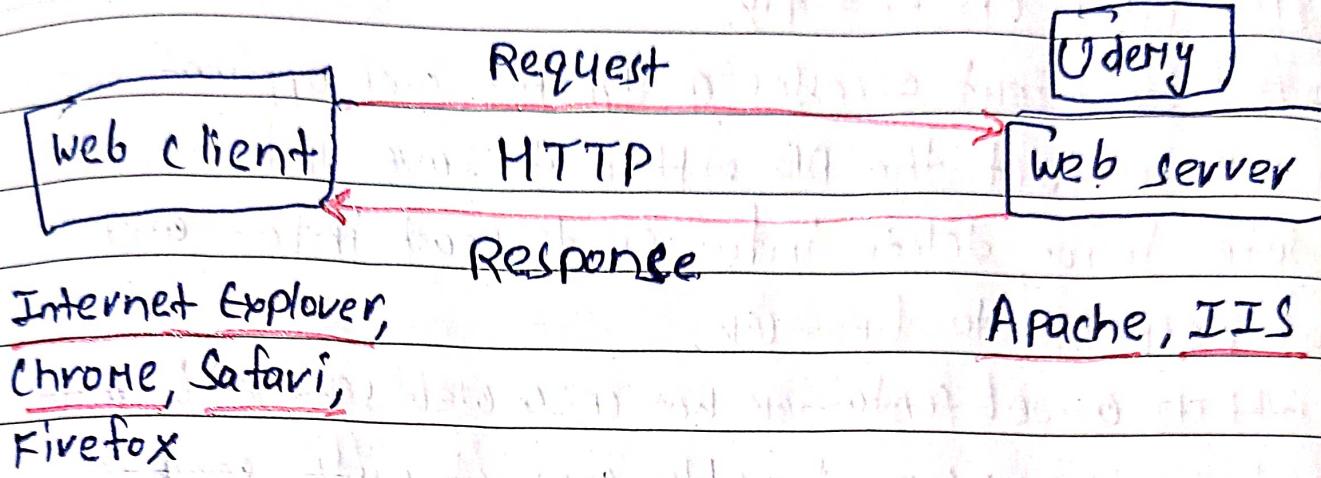


Java Web Development

Fri 11

* Web Application →

It is a software application whose services can be accessed via the web.

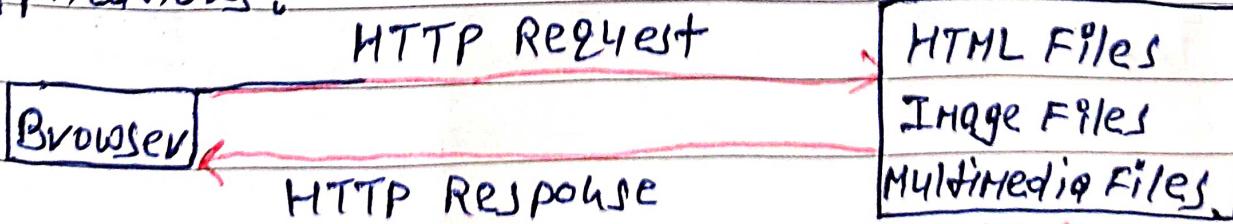


Google, Facebook, Twitter → Web Application

① Static Web Application →

It is a non-interactive web-application which provides some generic information no matter which client is accessing it, there is nothing dynamic or custom happening here for a particular client.

There is no DB and so there is no server side programming involved when we develop static web applications.



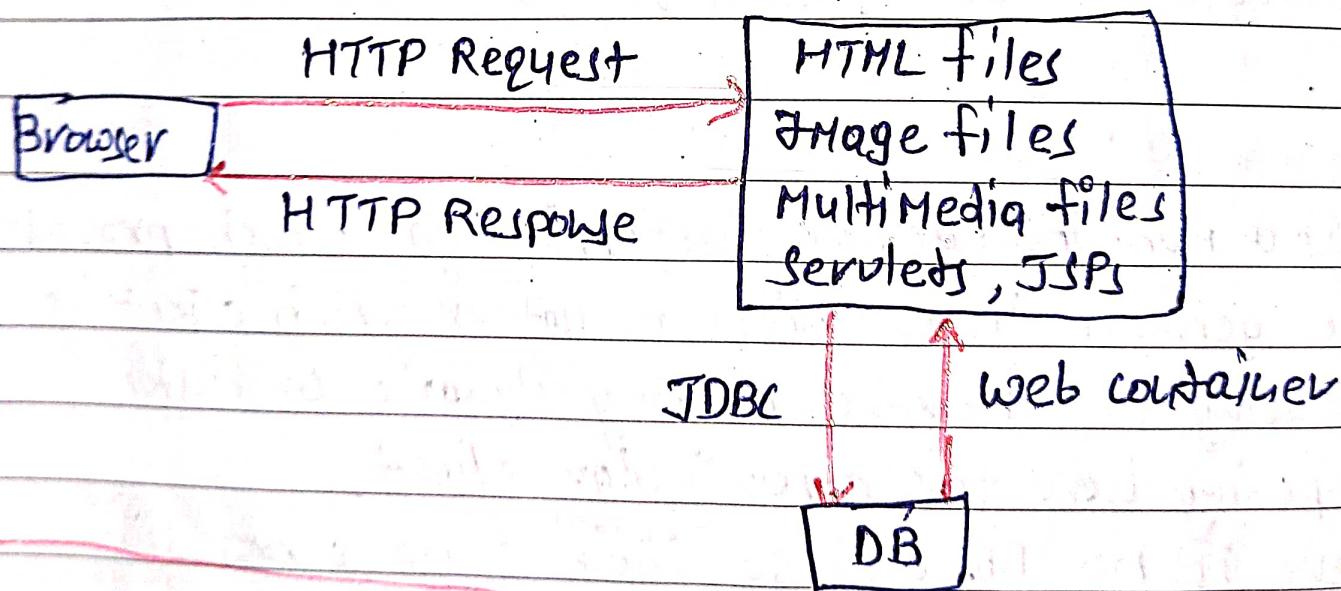
Haq, ek behtar zindagi

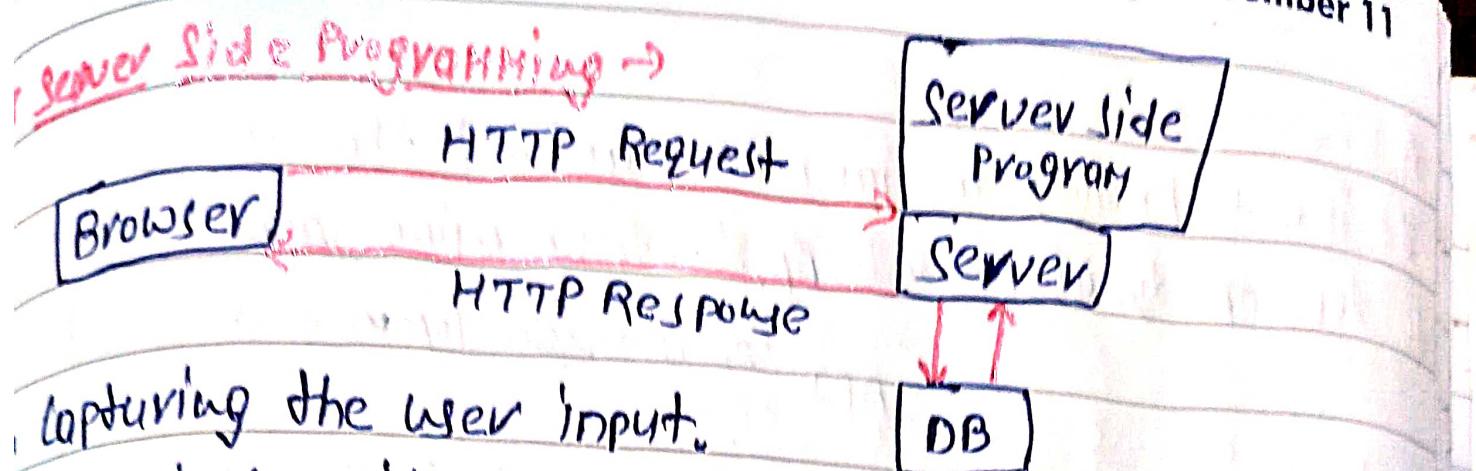
(2) Dynamic Web Application →

It comprises of servlets, JSP's and several other server side components written in different programming languages which can generate these HTML files on the fly.

A web client accesses a dynamic web application, it will contact the DB either to save info. or to update info., delete info. or to read info. and send response to browser.

It runs on a web container not on a web server, because a web container can directly connect with servlets, JSPs and other files easily.





Capturing the user input.

Communicate with the DB.

Processing of data.

Produce the response page.

Handing the response page to the server.

* Life cycle Methods →

- `init()` → 1 time

- `service()` → n times

- `destroy()`

* Life cycle phases →

- instantiation

- initialization

- servicing

- destruction

Sunday, November 12

* Web App. Folder Structure →

WEBAPP

- HTML
- JSP

WEB-INF

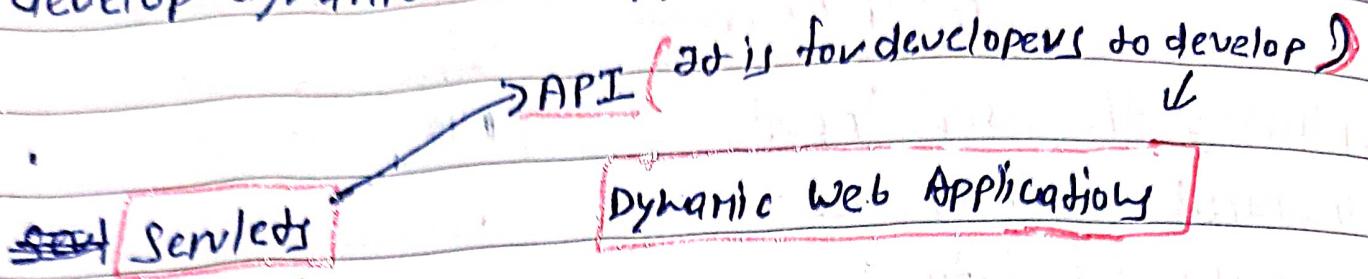
- XML
- classes

lib.jar

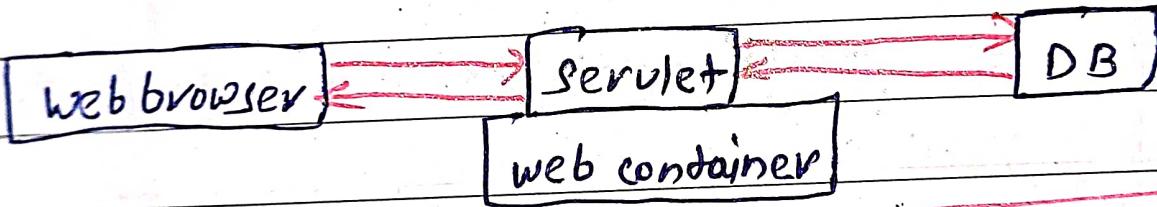
Haq, ek behtar zind

* Servlets →

It is a technology in Java EE standard that allows us to develop dynamic web applications using Java.



- It is a Java program that runs on the server or the web container, it can perform the following operations i.e.
- Specification (Set of rules written in plain English, it is for application server or web container developers like Apache, Tomcat.)



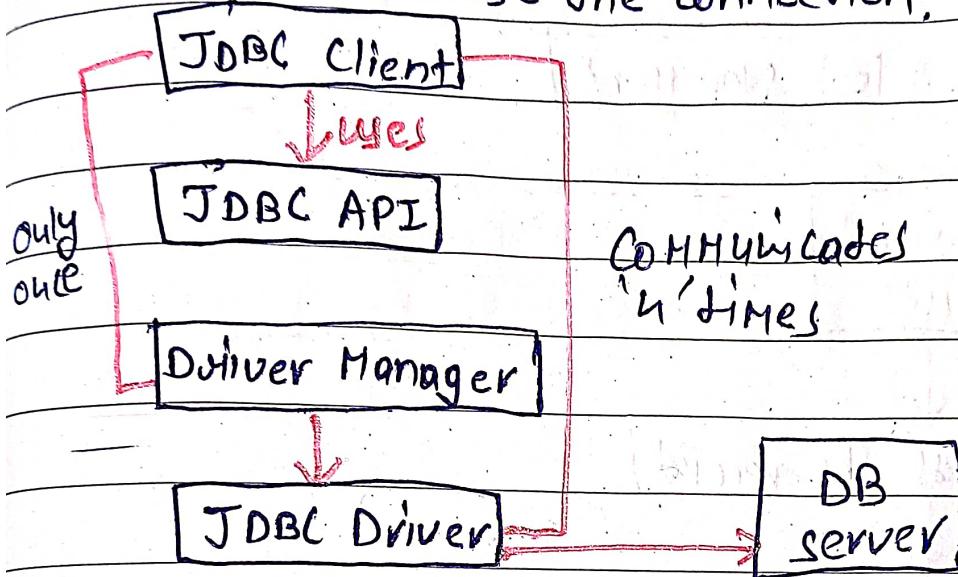
* JDBC Architecture →

- ① JDBC API → std. API from Oracle, which we learn as programmers.
- ② JDBC Drivers → A program which is an interface b/w our JDBC client and DB.
- ③ JDBC Client → APP code which we are developing.
- ④ JDBC Driver Manager → Helper class which finds a driver and establishes a connection to the DB.

, ek behtar zindagi ka.

JDBC Client →

- Connect to the DB,
- Perform CRUD,
- Process the response,
- Handle the exception.
- Do transaction Management.
- Close the connection.



* Steps to perform CRUD (Create, read, update and delete) →

① Establish the connection

② Create the statement object

③ Submit the SQL query to DBMS

④ Close the statement

⑤ Close the connection.

Haq, ek zikr

* JDBC steps to read data →

Establishing the connection

Creating Statement object

Submit the select statement

Record
exists

Process the record

Yes

More
Records

No

Close the Result Set, Statement, Connection

* Init Parameters →

It refers to the initialization parameters of a servlet or filter.

<init-param> attribute is used to define a init parameter.

<init-param> attribute has 2 main sub attributes <param-name> and <param-value>.

1, ek behtar zindagi ka.

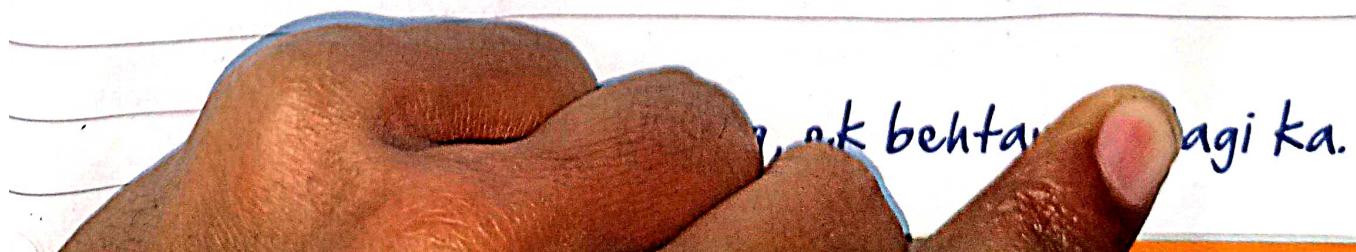
Servlet context →

It is a library interface in the servlet API and it lives in `java.x.servlet.ServletContext` package

An objective of it is created by the web container at time of deploying the project. This object can be used to get configuration info. from `web.xml` file. There is only one `ServletContext` object per web application.

If any info. is shared to many servlet, it is better to provide it from the `web.xml` file using the `<context-param>` element.

- Use → ① The object of `ServletContext` provides an interface b/w the container and servlet.
- ② The `ServletContext` object can be used to get configuration info. from the `web.xml` file.
- ③ The `ServletContext` object can be used to set, get or remove attribute from the `web.xml` file.
- ④ The `ServletContext` object can be used to provide inter-application communication.



Friday, November 17

2-3-22

* Prepared Statement interface →

This interface is a sub-interface of statement. It is used to execute parameterized query.

Ex- String sql = "insert into emp value(?, ?, ?);"

- We are passing parameter (?) for the values. Its value will be set by calling the setter methods of Prepared Statement.

- The performance of the application will be faster if you use Prepared Statement interface because query is compiled only once.

* Inter-servlet communication →

- A process where 2 or more servlets communicates with each other to process the client request.

- A servlet can forward the request to another servlet to process the client request.

- A servlet can include the output of another servlet to process the client request.

2 ways →

- `ServletContext.getRequestDispatcher(String resource)`
- `ServletRequest.getRequestDispatcher(String resource)`

ek behtar zindagi ka.

* Pre-initialization of servlets →

① Lazy initialization →

If a servlet container initializes the servlet only ^{when} the first web client ~~reg~~ request comes in than it is called lazy initialization. By default all the servlets are lazily initialized.

② Pre-initialization →

But if the container initializes the servlet even before a client ~~reg~~ request comes in that is called pre-initialization. We can request the container to do pre-initialization by using the load-on-startup element in the web.xml file.

```
[< servlet >
    < servlet-name > ----- </ servlet-name >
    < servlet-class > ----- </ servlet-class >
    < load-on-startup > 1 < /load-on-startup >
< /servlet >]
```

* servlet listeners →

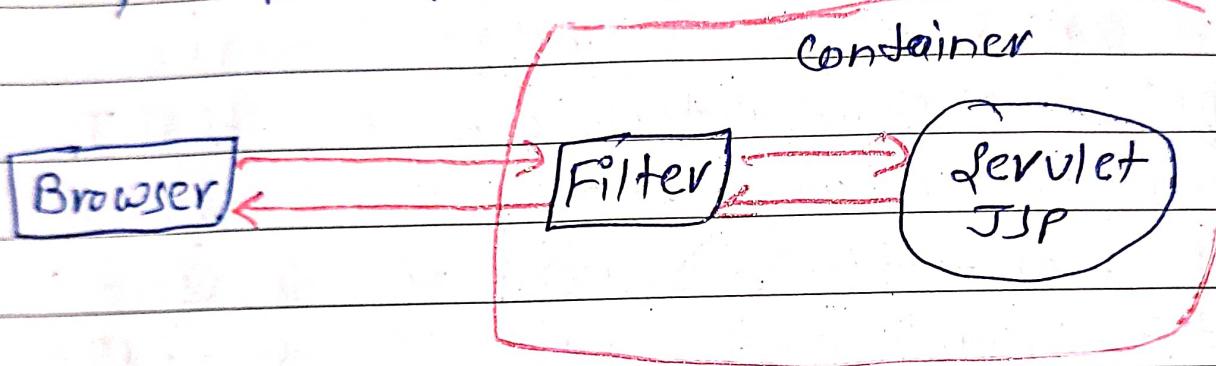
They enables our applications to react to Sunday, November 19 setting the events that happen in the web container. So, our web application can handle those events can take appropriate action if required. These events could be -

- Request • Context
- Session • Async

Haa, ek behtar zindagi!

★ Servlet Filter →

A filter is a java class that can intercept the request and response cycles of a servlet, any logic that we want to execute before the servlet request is processed by the servlet can be put into the filter, i.e., the pre-processing logic and any logic that we want to execute before the servlet's response goes back to the web-client can also be put into the filter, i.e., post-processing logic. Filters are pre-initialized by default.



Injection Attack Filter

★ Session Management →

- Session simply means a particular interval of time.
- It is a way to maintain state (data) of an user.
- HTTP is a stateless protocol, if a protocol is stateless, it means the server does not maintain a continuous connection once the server handles the request and sends back the response. There is no locked connection alive, it is destroyed every time a new request comes. ek behtar zindagi ka. the server sends a response back.

Tuesday, November 21

A. Techniques →

- ① URL rewriting
- ② Cookies

- ③ Hidden form fields
- ④ HTTPS and SSL

Request

Response

Request

Response

web client

web server

- Advantages -

- ① Performance
- ② Scalability.

- Steps for Session Management -

- ① Step 1: Create the session

HttpSession session = request.getSession();

- ② Step 2: Maintain the data using the four attribute methods on HttpSession.

- ③ Step 3: End the session.

- Cookies →

Http cookies are name-value pairs of textual info. that can be used to exchange data b/w the web server and the web client or the web browser as a part of the http headers. we can pass any kind of info. which could be anything, which are part of cookies, but usually cookies are used to maintain sessions b/w a web container and a web client. Haq, ek behtar zindagi ka

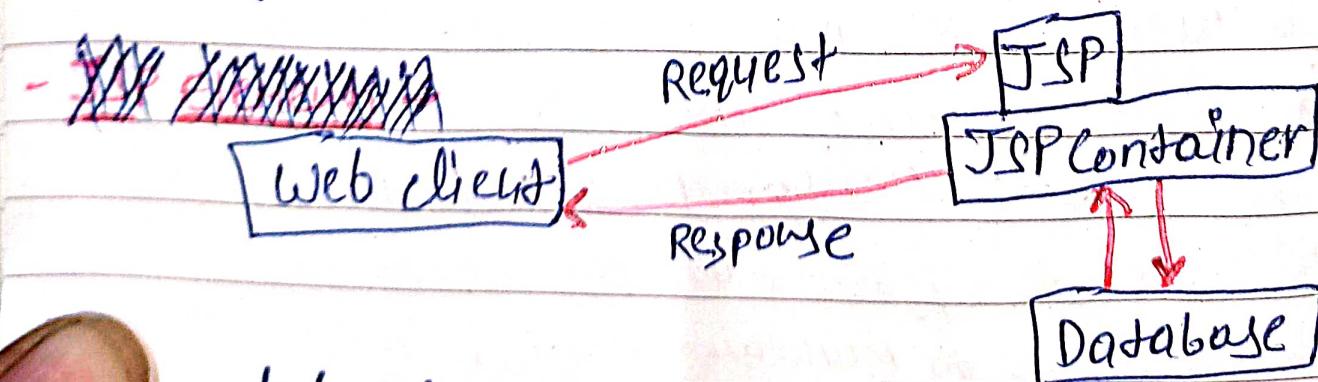
```
Cookie c = new Cookie("jsessionid", "1234");
response.addCookie(c); / cookie created
Cookie[] cookies = request.getCookies(); / cookie retrieved
cookies[0].getName(); / get name of cookie.
cookies[0].getValue(); / get value of cookie.
```

* JSP Basics → Java Server Pages

- It is a technology in Java EE space that comes with API and a specification. The API is for the application developers. The specification itself is a set of rules that should be followed by the JSP container creators.

A JSP can perform several different operations, it runs on a JSP container like Tomcat.

- It can take request from a web client, it can process that request, it can make calls to database, receive the response, process the response and send a proper HTML response back to the web browser. JSP can do everything a servlet can do while overcoming the limitations of a servlet.



q, ek behtar zindagi ka.

JSP Elements →

① Scripting elements -

To embed java code into a JSP page.

② Directives -

Translation time instructions to the container.

③ Actions -

Runtime instructions to the JSP container.

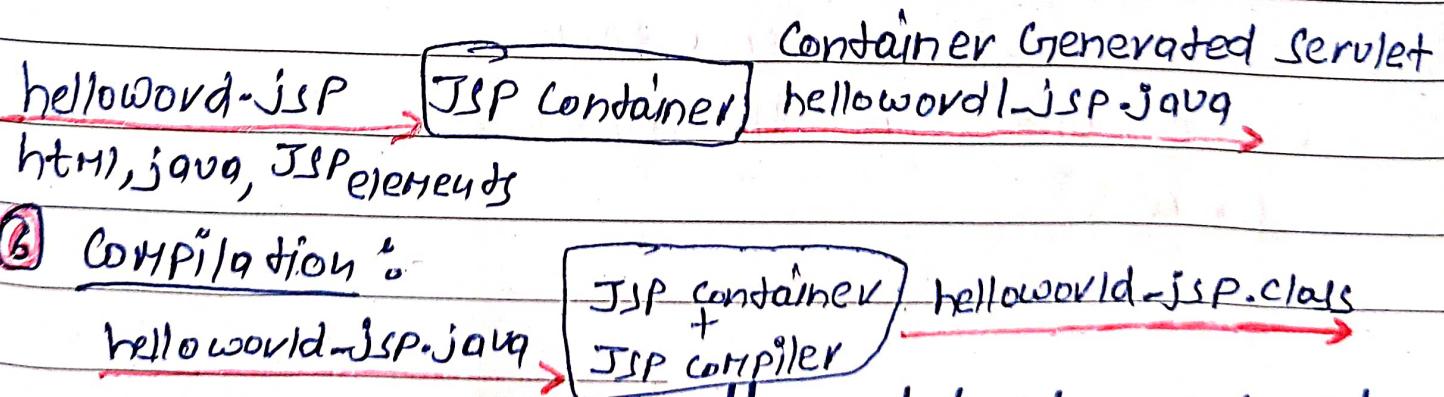
- JSP life cycle Methods →

- ① jspInit()
- ② jspService()
- ③ jspDestroy()

- JSP life cycle phases →

- ① Instantiation
- ② Initialization
- ③ Servicing
- ④ Destruction
- ⑤ Translation:

First Client Request



Haq, ek behtar zindagi ka.

Once - Only if the code is modified

* Implicit objects →

Object name

- ① config
- ② request
- ③ response
- ④ session
- ⑤ application
- ⑥ page
- ⑦ pageContext
- ⑧ exception
- ⑨ out

Type

- | |
|-------------------------------|
| ServletConfig |
| HttpServletRequest |
| HttpServletResponse |
| HttpSession |
| ServletContext |
| java.lang.Object |
| javax.servlet.jsp.PageContext |
| java.lang.Throwable |
| javax.servlet.jsp.JSPWriter |

* JSP Scripting elements -

① Declaration :- <% !

```
int x;  
int y;  
void method()  
{  
    ...  
}
```

/ anything inside it

[Member of the servlet.]

② Expression :- <% = atb % >

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

[Evaluates and sends the response]

[- JSP Service]

Waq, ek behtar zindagi ka.

① Scriptlet :- <% %>

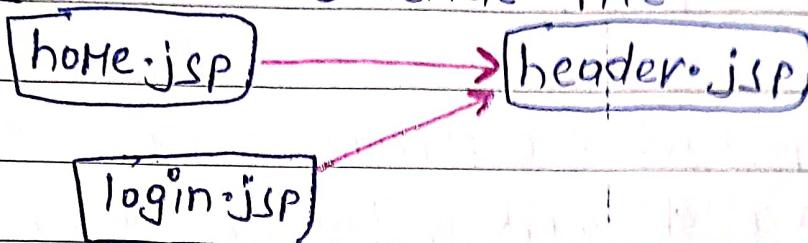
```
String num1 = request.getParameter("num1");
String num2 = request.getParameter("num2");
int result = num1 + num2;
```

%>

[jspService()]

* JSP Directives →

① Include → <%@include file = ""%>



② Page →

- import
- session
- Pageencoding
- errorpage
- extend
- buffer
- iserrorpage
- isEligible
- isThreadSafe
- language
- autoflush
- contentType
- info

Sunday, November 26

③ taglib → Set of Tags

* Scopes JSP →

- | | | |
|------------------------|----------------------------|----------------------------|
| ① <u>Page scope</u> | ③ <u>Session scope</u> | ④ <u>Application scope</u> |
| ② <u>Request scope</u> | Haq, ek behtar zindani ka. | |

Monday, November 27

★ JSP Action →

JSP actions also known as JSP tags are runtime instructions to the JSP container.

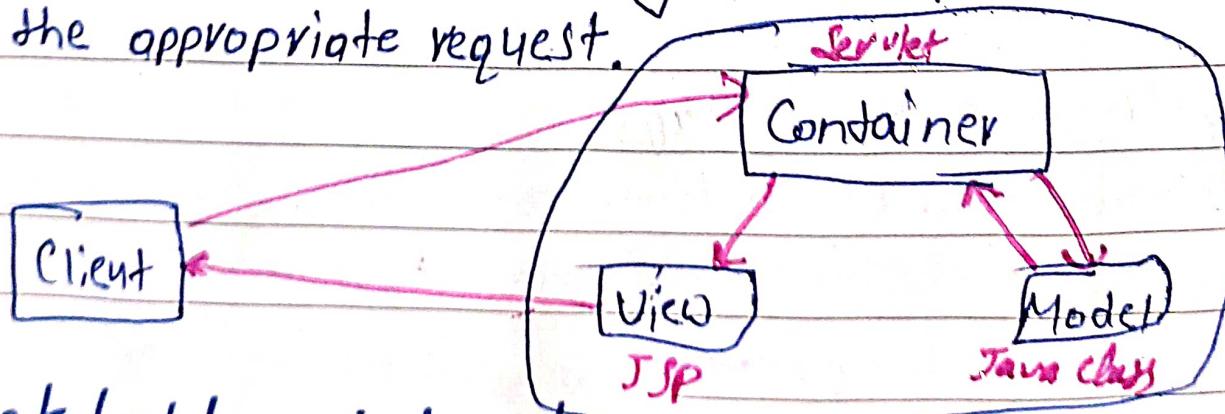
Type →

- ① Predefined → include, forward, param, usebean, getProperty, setProperty.
- ② Custom tags

★ MVC Design pattern → (Model View Controller)

It is a design pattern or a framework that splits the web layer into 3 parts.

- The Model represents the current state of the application and does most of the work or business logic. It can call into other db classes or connect to db and do lot of work.
- The view is responsible for displaying the current Model to the end user. So, it will print or generate a HTML output and send back to browser.
- The controller acts as a glue b/w view and Model, so, it is responsible for selecting the appropriate Model to serve the appropriate request.



Haq, ek behtar zindagi ka.

- Advantages →

- ① Maintenance.
- ② Parallel development

* Custom tags →

Custom tag creation is a 2 step process -

- Create the Tag Handler Class
- Create the Tag Lib Descriptor (TLD File)

* JSTL → (Java Server pages Standard Tag Library)

- These are the set of predefined tags from oracle
- These represents a set of tags to simplify the JSP development.

- c:set → `<c:set var="score" scope="session" value="${expr}">`
Let the result of an expression evaluation in a 'scope'.

- c:remove →

Removes a scoped variable (from a particular scope, if specified).

- c:if →

Simple conditional tag, which evaluates its body if the supplied condition is true and optionally exposes a Boolean scripting variable representing the evaluation of this condition.

Haq, ek behtar zindagi ka.

- c:choose →

Simple conditional tag that establishes a context for mutually exclusive conditional operations, marked by <when> and <otherwise>.

- c:forEach →

The basic iteration tag, accepting many diff. collection types and supporting subsetting and other functionality.

- c:url →

Creates a URL with optional query parameters.

• Formatting → [<fmt:]]

- parseNumber
- formatNumber
- parseDate
- message
- formatDate

★ Connection Pooling →

It allows us to request the container to create a set of JDBC connections right when it starts up. These connections can then be used by servlets or JSPs or any other web resource to do what they have to do and send the connection back once they are done with their work.

- Advantages →

- ① Performance
- ② Reuse

- Connection pooling configuration →

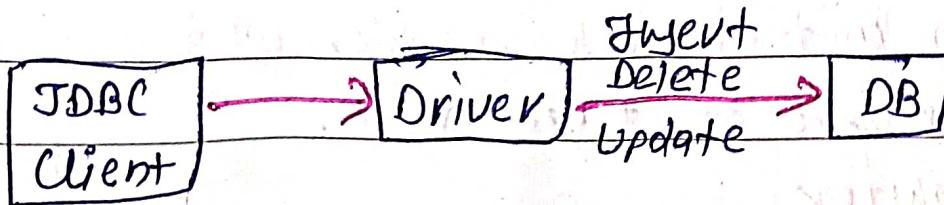
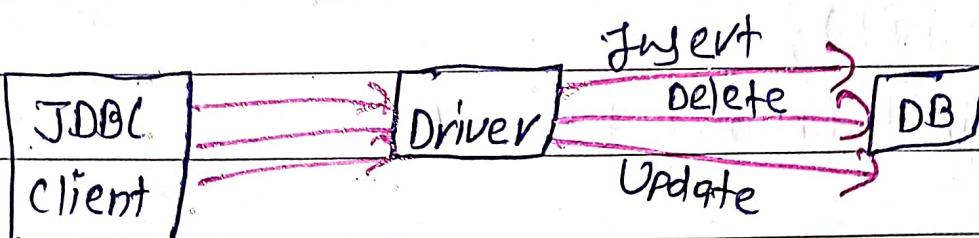
- ① Copy driver jar
- ② Configure resource element in context.xml,

- Acquiring the connection →

- ① Connect to the naming service.
- ② Look up for the DataSource.
- ③ Get the connection.

* Advanced JDBC →

- JDBC batch updates →



Friday, December 01

- Steps for batching →

① Add the dml statement

```
stmt.addBatch(String dml);
```

② Execute the batch

```
int result[] = stmt.executeBatch();
```

* ResultSet →

It is an interface in the JDBC API using which we can move through the rows of data that comes back when we execute a select query and deal with each row and the data in each column one at a time.

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

```
ResultSetMetaData rsmd = rs.getMetaData();
```

```
int n = rsmd.getColumnCount();
```

```
for (int i=1; i<n; i++) {
```

```
System.out.println(rsmd.getColumnName(i));
```

* Transaction Management →

A transaction is a logical unit of work in which all the operations get done or none of it gets done.

Ex- Online Money transfer.

"Ag, ek behtar zindagi ka."

Saturday, December 05

- Properties →

- ① Automicity
- ② consistency
- ③ Isolation
- ④ Durability

- Controlling transactions →

- ① COMMIT
- ② Rollback
- ③ Savepoint

Sunday, December 03