

Date : __ / __ / __

UNIT-05 (LEC-01)

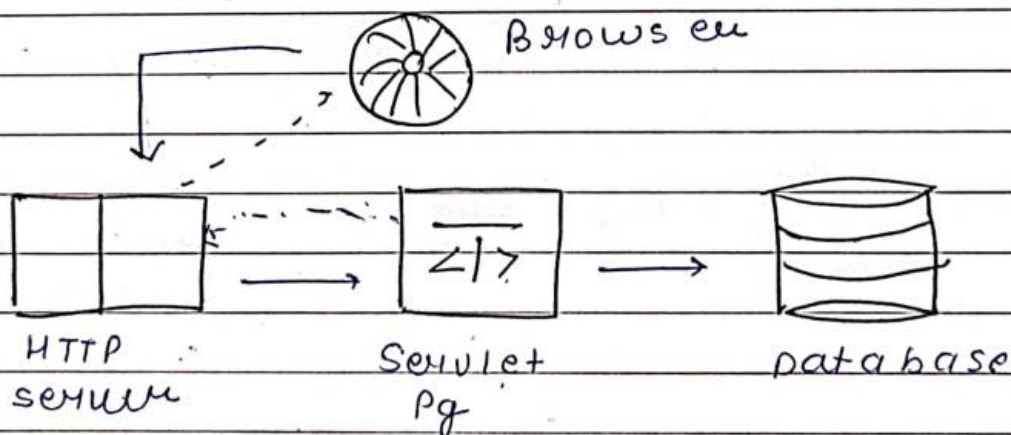
- Servlet [AKTU-20-21, 21-22, 22-23]
- Architecture of Servlet
- Life Cycle
- HTTP Reqⁿ
- Session Tracking
- Cookies



Servlets with its Lifecycle

[AKTU-20-21, 21-22, 22-23]

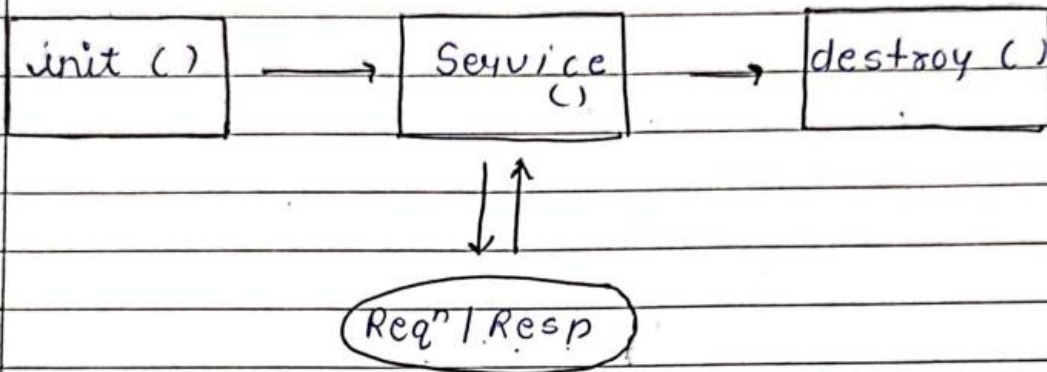
- Servlets are the Java Program that run on the Java-enabled web server or application server.
- They are used to handle the request obtained from the web-server, process the reqⁿ, produce the response, and then send a response to the web server.
- Servlets work on the server side.



- Servlets are help to create dynamic web content. and it is more efficient than older CGI Scripts.

Lifecycle

- `init()`
- `Service()`
- `destroy()`

1. init()

- The `init()` method is called only once
- Created only when the Servlet created
- The `init()` method must complete successfully before the servlet can receive any request and response
- when the servlet is first requested, the container creates an instance of servlet class.
- After creation, the servlet container calls the `init()` method to initialize the servlet.
- This phase is used to set up resources like database connections etc

2. Service ()

- The Service () method is called only after the init
- This is the main method to perform the actual task
- It check the HTTP request type
- This phase is repeated for every request

3. destroy ()

- The destroy () method is called only once at the end of the life cycle of a Servlet
- This method is used to release resources like closing database connection

HTTP Methods :-

1. **Get** :- To receive data.
2. **Post** :- To send or submit data to the server, usually to create something new.
3. **Patch** :- To partially update data to the server.
4. **Put** :- To update existing data on the server.

5. **Remove**: To delete data from the Server.

Redirecting Request to other Resources

- Redirecting a request in Servlet means sending the user request to another resource to handle it.

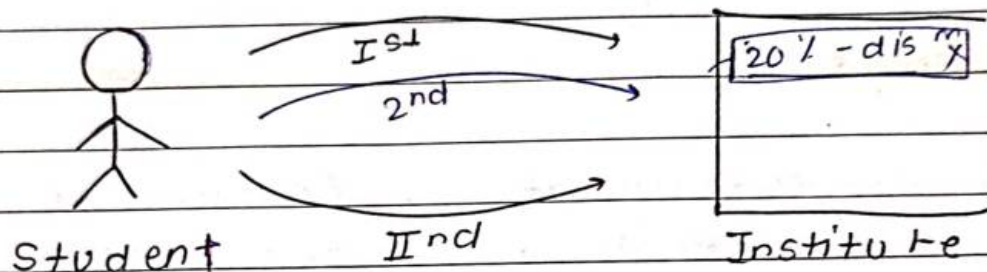
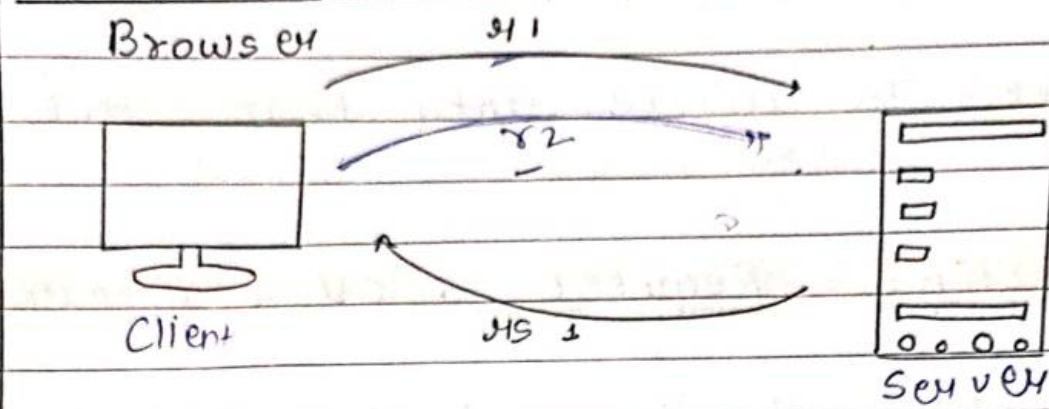
Two-ways :-

1. Using Request Dispatcher (Server-side)
2. Using sendRedirect (Client-side)

Session Tracking :-

- Session Tracking in Java Servlets is the mechanism to maintain the state of a user interacting with a web application. Since HTTP is a stateless
- Session Tracking is a way to maintain state of an user, It is also known as Session Management.
- HTTP is stateless that means each request is considered as the new reqⁿ.
- To recognize the user.

Date : __/__/__



Session Tracking Techniques

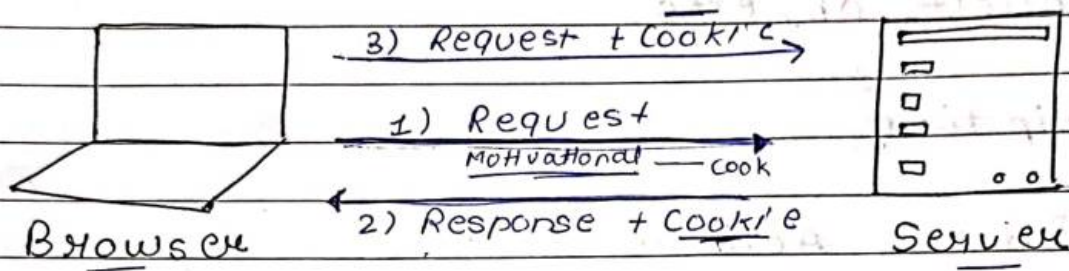
1. Cookies
2. Hidden Form Field
3. URL Rewriting
4. Http Session

Cookies

- A Cookies is a small piece of inform^m stored on the client.
- Servlets

Date : __/__/__


- A Cookie has a name, a single value and Optional attributes.
- In cookies technique, we add cookie with response from the Servlet.



Types of Cookie :-

1. Non - Persistent cookie → Valid for Single Session
2. Persistent cookie → Valid for multiple Session.

SUBSCRIBE 

JOIN TELEGRAM 


Date : __ / __ / __

UNIT-05 (LEC-2)

JSP [AKTU-20-21, 21-22, 22-23]

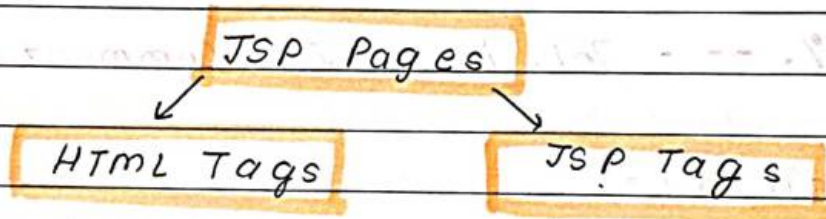
- Implicit objects
- Scripting
- Standard Actions
- Directives
- Custom tag Libraries

SUBSCRIBE 

JOIN TELEGRAM 

JSP (Java Server Page) [AKTU-20-21, 21-22, 22-23]

- JSP is a serverside technology that is used for creating web application. It is used to create dynamic web content.
- It supports both scripting and element-based dynamic content.
- JSP is an advanced version of Servlet Technology.



- JSP tags are reusable then pages run faster.
- JSP and CGI have the same purpose but JSP offers several advantages in comparison with CGI.
- Java code can be inserted in HTML pages on both. JSP allows embedding Java code in HTML pages.

Features in JSP :

- Coding in JSP is Easy
- Easy to Use and Learn
- It Does Not Require Advanced knowledge of Java

Tags and Elements in JSP :-

1. JSP Comment :-

- JSP Comment is to document the code
- JSP comment is used to note some parts of JSP pages to make it clearer and easier to maintain

`<% -- - This is a JSP Comment --- %>`

2. Expression :-

- Basic Scripting elements in JSP
- expression is used to insert value directly to the output

`<% = Expression %>`

3. Scriptlet tag :-

- Insert any plain Java code inside a scriptlet

`<% Java-code %>`

4. Declaration tag :-

We can declare static member, instance variable and methods inside declaration tag.

`<% ! declaration %>`



Ques JSP is an extension of Servlets not replacement. Justify? What problems of Servlets technology can JSP solve? [AKTU-20-21]

Sol. JSP is not a replacement of Servlets but extension of Servlet, as coding decreases more than half.

- In JSP, static code and dynamic code are separated.
- JSP needs no compilation by the programmer.

Problem of Servlet technology Solved by JSP:

- Difficult to code
- It cannot be integrated with
- It does not manage Cookies
- Do not all reading, and sending HTML headers.

Implicit Object:

- Implicit Objects are a set of Java Objects that the JSP Container make available to developers on each Pages
- The JSP Request is an implicit Object which is provided by `HttpServletRequest`.
- JSP Request Object is created by the web container for each Request of the client.

Total Nine Implicit objects as follows:

1. Request
2. Response
3. Config

- 4 application
- 5 Session
- 6 Page Context
- 7 Page Object
- 8 Exception
- 9 out

Standard Actions :-

- In JSP, Standard actions are special JSP tags that are used to perform common tasks like forwarding request, including files or interacting with JavaBeans.

1. <jsp:include>

- To include another JSP

Ex :- `<jsp:include page = "header.jsp" />`

2. <jsp:forward>

- Forward the request to another JSP

Ex :- `<jsp:forward page = "next Page.jsp" />`

3. <jsp:useBean>

- Create or retrieves a JavaBean object.

Ex :- `<jsp:useBean id = " " class = " " Scope = " " />`

4. <jsp:setProperty>

→ sets Properties of a JavaBean

Ex:- `<jsp:setProperty name=" " Prop=" " Value=" " />`

5. <jsp:getProperty>

→ Retrieves a Property Value from a Java Bean

Ex:- `<jsp:getProperty name=" " Property=" " />`

6. <jsp:param>

→ Passes parameters to an included page or a Forwarded resource.

Ex:- `<jsp:param name=" " Value=" " />`

Directives :-

- JSP Directives are instructions to the JSP Container that affect the entire JSP page, like how it is translated into a Servlet.
- These directives are not used.

directly in Servlets because Servlets do not go through a translation process.

3-types of directives :

1. Page directive `<% @ Page attribute = " " %>`
2. Include directive `<% @ Include File = " " %>`
3. Taglib directive `<% @ taglib uri = " " %>`

Custom Tag Libraries :

- Custom Tag Libraries are primarily a feature of JSP and are not directly used in Servlets.
- We are going to create a custom tag that prints the current date and time.
- Custom tag Libraries are collection of user-defined tags encapsulating reusable functionality.

Ex : ~~<hr~~ `<custom: welcome name = " " />`

How Tag Libraries Work :

1. Tag Handler Classes :-

Java classes that define the behaviour of custom tags

2. Tag Library Description :-

XML Files that maps custom tags to their handler classes

3. Usage in JSP :-

Custom tags are used in JSP with `<%@ taglib %>` directives

Custom Tags In JSP Imp^m Points :

- Declarative like HTML
- High , encapsulated in tag
- Writing plain Java code

SUBSCRIBE 

JOIN TELEGRAM 