

ENDSEM - WEB TECHNOLOGY**➤ MAY/JUN 2022 PYQ [unit -3]****Q1)**

a) What is difference between server side scripting language and client side scripting language. [5]

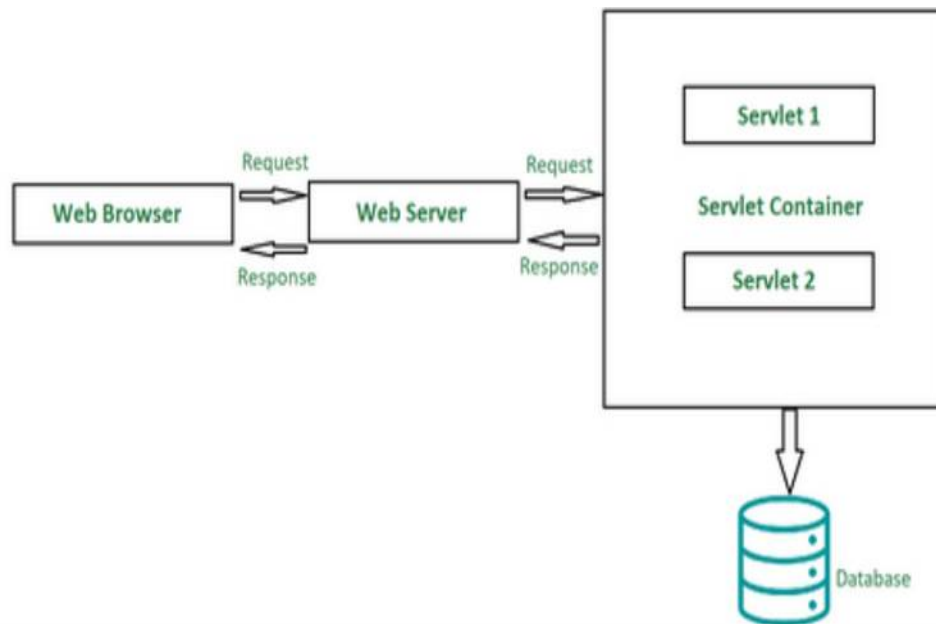
Client-side scripting	Server-side scripting
Source code is visible to the user.	Source code is not visible to the user because its output of server-side is an HTML page.
Its main function is to provide the requested output to the end user.	Its primary function is to manipulate and provide access to the respective database as per the request.
It usually depends on the browser and its version.	In this any server-side technology can be used and it does not depend on the client.
It runs on the user's computer.	It runs on the webserver.
There are many advantages linked with this like faster response times, a more interactive application.	The primary advantage is its ability to highly customize, response requirements, access rights based on user.
It does not provide security for data.	It provides more security for data.
It is a technique used in web	It is a technique that uses scripts on the

Client-side scripting	Server-side scripting
development in which scripts run on the client's browser.	webserver to produce a response that is customized for each client's request.
HTML, CSS, and javascript are used.	PHP, Python, Java, Ruby are used.
No need of interaction with the server.	It is all about interacting with the servers.
It reduces load on processing unit of the server.	It surge the processing load on the server.

b) Describe servlet architecture in detail.**[4]**

- i. Java Servlets are programs that run on a Web or Application server and act as a middle layer between a requests coming from a Web browser or other HTTP client and databases or applications on the HTTP server.
- ii. Servlets are robust, well scalable, and are primarily used in developing server-side applications

Components of Servlet Architecture :



1. Client

The client shown in the architecture above is the web browser and it primarily works as a medium that sends out HTTP requests over to the web server and the web server generates a response based on some processing in the servlet and the client further processes the response.

2. Web Server

Primary job of a web server is to process the requests and responses that a user sends over time and maintain how a web user would be able to access the files that has been hosted over the server.

3. Web Container

Web container is another typical component in servlet architecture which is responsible for communicating with the servlets. Two prime tasks of a web container are:

- Managing the servlet lifecycle
- URL mapping

Advantages of Servlets:

1. Servlets are platform-independent and follow Java's portability.
2. They are efficient and use multithreading to handle multiple requests.

Disadvantages of Servlets:

1. Complex to write and maintain compared to scripting languages like PHP.
2. Requires Java knowledge and a servlet container (like Tomcat) to run.

c) Explain DTD in XML with schemes, elements & attributes.**[9]**

- i. DTD is a document-type definition.
- ii. A Document Type Definition (DTD) is a set of rules that defines the structure and content of an XML document.
- iii. It ensures that the XML data conforms to a predefined format, thus validating the structure and relationships of elements and attributes within the document.
- iv. DTD can be defined either inline (within the XML file itself) or externally (in a separate file).

Schemes:

- In DTD, schemes define how data types, elements, and attributes are validated, but this term is often more common in XML Schema.
- Example of a DTD scheme for an element:
`<!ELEMENT book (title, author, year)>`

Elements:

- Defines the building blocks of the XML document.
- Specifies what elements can appear in the document and their hierarchy.
- Example:
`<!ELEMENT book (title, author, year)>`

This defines a book element that contains title, author, and year as child elements.

Attributes:

- Specifies the properties of elements.
- Describes what attributes an element can have, along with their data types and whether they are required or optional.
- Example:
`<!ATTLIST book id ID #REQUIRED>`

This defines an id attribute for the book element that must be of type ID and is required.

Types of DTDs:

1. **Internal DTD:** Defined within the XML document itself.
2. **External DTD:** Defined in a separate file and linked to the XML document using the DOCTYPE declaration.

Advantages of DTD:

1. **Standardization:** DTD ensures that XML documents follow a standard structure.
2. **Validation:** It helps in validating the XML document against the predefined rules.

Limitations of DTD:

1. **Lack of Data Types:** DTD does not support complex data types like integers or dates.
2. **Limited Validation:** DTD provides only basic validation compared to XML Schema.

Q2)**a) What is session? How cookies & URL rewriting for session management in servlet [9]**

A **session** is used to store user-specific data across multiple HTTP requests. HTTP is stateless, so the session allows the server to remember user information between requests. Each session is identified by a unique **session ID**.

Cookies for Session Management:

- A **cookie** stores the session ID on the client-side. The browser sends it back to the server with each subsequent request.
- **How Cookies Work:**
 - The server generates a session ID and stores it in a cookie.
 - The browser sends this cookie back to the server with each request.
- **Cookie Example:**

```
java
Cookie sessionCookie = new Cookie("JSESSIONID", session.getId());
response.addCookie(sessionCookie);
```

Advantages:

- Automatically handled by the browser.
- Simplifies session management.

Disadvantages:

- Can be disabled by the user.
- Limited size (4 KB)

URL Rewriting for Session Management:

- When cookies are disabled, **URL rewriting** appends the session ID to the URL (e.g., sessionId=12345).
- **How URL Rewriting Works:**
 - The session ID is included in the URL of every response.
- **URL Rewriting Example:**

java

Copy

Edit

```
String urlWithSessionId = response.encodeURL("nextPage.jsp");  
response.sendRedirect(urlWithSessionId);
```

Advantages:

- Works when cookies are disabled.
- No data stored on the client.

Disadvantages:

- Session ID is visible in the URL.
- Makes the URL cluttered

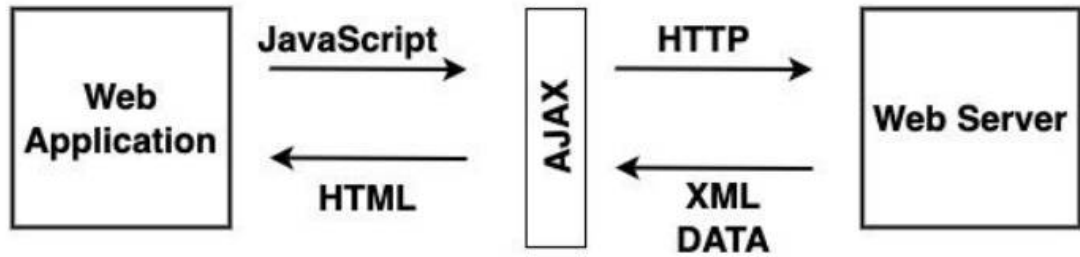
b) Write short note on : [9]

i) AJAX

ii) XML transformation

i)AJAX

- i. AJAX stands for asynchronous Javascript and XML.
- ii. AJAX is not a programming language or technology, but it is a combination of multiple web-related technologies like HTML, XHTML, CSS, JavaScript, DOM, XML, XSLT and XMLHttpRequest object.
- iii. The AJAX model allows web developers to create web applications that are able to dynamically interact with the user.



Advantages of AJAX

- It creates responsive and interactive web applications.
- It supports the development of patterns and frameworks that decrease the development time.

II) XML transformation

- **XML Transformation** is the process of converting one XML document into another format (like HTML, XML, or plain text).
- It is commonly done using **XSLT** (Extensible Stylesheet Language Transformations).

Why it's used:

- To display XML data in a readable format (like converting XML to HTML for webpages).
- To restructure XML content as per application needs.

Steps Involved:

1. XML data is loaded.
2. XSLT stylesheet defines transformation rules.
3. A processor applies the stylesheet to the XML and generates the output.

➤ NOV/DEC 2022 PYQ [unit – 3]

Q1)

a) Explain the following:

- i) Process of transforming XML document.
- ii) HTTP session

i)Process of transforming XML document.

XML Transformation is the process of converting XML data into another format (like HTML, plain text, or another XML) using XSLT (Extensible Stylesheet Language Transformations).

Steps in XML Transformation:

1. XML Document:

Contains raw structured data.

2. XSLT Stylesheet:

Defines the transformation rules – how the XML elements should be displayed or restructured.

3. XSLT Processor:

Applies the XSLT stylesheet to the XML document and generates the desired output.

4. Output Document:

The final result, which can be HTML, plain text, or transformed XML.

Example Workflow:

(XML + XSLT) → [XSLT Processor] → Transformed Output (HTML/XML/Text)

Small Example:

XML:

```
<book>
  <title>XML Basics</title>
</book>
```

XSLT:

```
<xsl:template match="/book">
  <h1><xsl:value-of select="title"/></h1>
</xsl:template>
```

Output:

```
<h1>XML Basics</h1>
```


ii) HTTP Session

An HTTP session is a way to store and manage user-specific data on the server across multiple requests made by the same user.

Since HTTP is stateless, the server doesn't remember previous interactions. A session helps maintain continuity by assigning a unique session ID to each user.

Key Features:

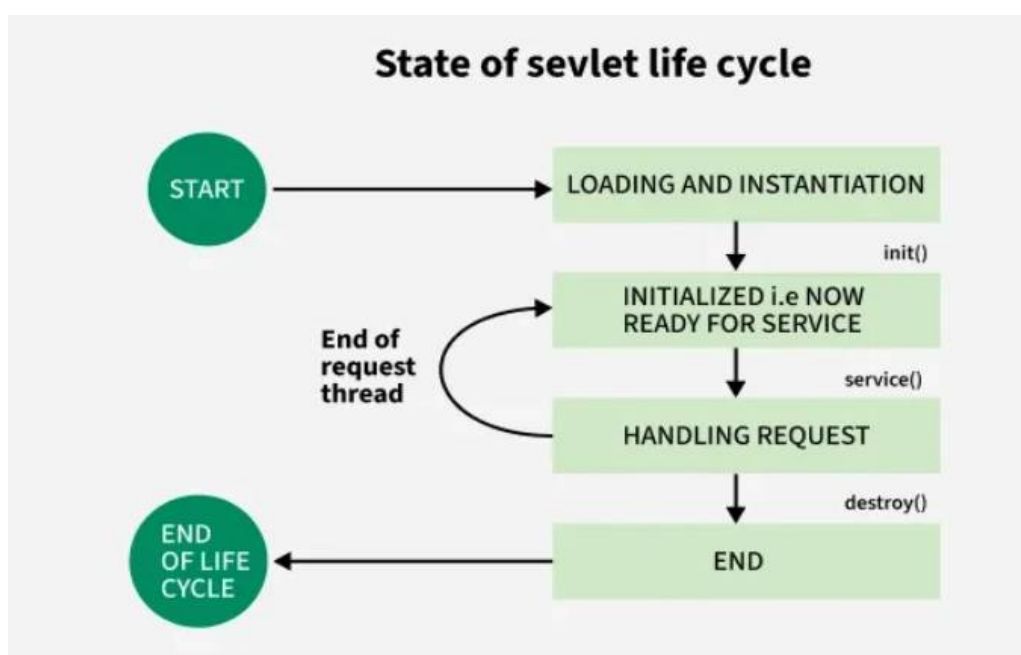
- Stores data like login info, shopping cart items, etc.
- Each session is identified by a unique JSESSIONID.
- Managed using HttpSession object in Java servlets.

b) What is Servlet? Explain the life cycle of servlet. Illustrate with example.

[9]

Java Servlets are programs that run on a Web or Application server and act as a middle layer between a requests coming from a Web browser or other HTTP client and databases or applications on the HTTP server.

Servlets are robust, well scalable, and are primarily used in developing server-side applications



The Servlet life cycle mainly goes through four stages, which is explained below:

1. Loading a Servlet

The first stage of the Servlet lifecycle involves loading and initializing the Servlet. The Servlet container performs the following operations:

- **Loading:** The Servlet container loads the Servlet class into memory.
- **Instantiation:** The container creates an instance of the Servlet using the no-argument constructor.

The Servlet container can load the Servlet at one of the following times:

- During the initialization of the web application (if the Servlet is configured with a zero or positive integer value in the deployment descriptor).
- When the Servlet is first requested by a client (if lazy loading is enabled).

2. Initializing a Servlet

After the Servlet is instantiated, the Servlet container initializes it by calling the `init` (`ServletConfig config`) method. This method is called only once during the Servlet's life cycle.

3. Handling request

Once the Servlet is initialized, it is ready to handle client requests. The Servlet container performs the following steps for each request:

- **Create Request and Response Objects**
 - The container creates `ServletRequest` and `ServletResponse` objects.
 - For HTTP requests, it creates `HttpServletRequest` and `HttpServletResponse` objects.
- **Invoke the `service()` Method**
 - The container calls the `service(ServletRequest req, ServletResponse res)` method.
 - The `service()` method determines the type of HTTP request (GET, POST, PUT, DELETE, etc.) and delegates the request to the appropriate method (`doGet()`, `doPost()`, etc.).

4. Destroying a Servlet

When the Servlet container decides to remove the Servlet, it follows these steps which are listed below

- **Allow Active Threads to Complete:** The container ensures that all threads executing the `service()` method complete their tasks.

- **Invoke the destroy() Method:** The container calls the destroy() method to allow the Servlet to release resources (e.g., closing database connections, freeing memory).
- **Release Servlet Instance:** After the destroy() method is executed, the Servlet container releases all references to the Servlet instance, making it eligible for garbage collection.

Q2)

a) Compare doGet and doPost methods in servlet.

[9]

In Java Servlets, both doGet() and doPost() methods are defined in the HttpServlet class. They are used to handle HTTP GET and POST requests from the client to the server.

doGet() is used when the client sends a request to retrieve information from the server.

doPost() is used when the client sends data to the server, typically from a form submission.

Feature	doGet()	doPost()
Request Method	Handles HTTP GET requests.	Handles HTTP POST requests.
Parameter Visibility	Parameters are appended in the URL and are visible to the user.	Parameters are sent in the body of the request and are hidden from the user.
Security	Less secure due to visibility of data in URL.	More secure, suitable for sending sensitive data like passwords.
Data Length Limit	Limited data can be sent (depends on URL length limit - ~2KB in browsers).	Can send large amounts of data, no practical limit in most servers.
Caching	Can be cached and bookmarked.	Cannot be cached or bookmarked.
Idempotency	Usually idempotent (multiple identical requests have same effect).	Not idempotent (repeated submission may cause duplicate actions like payments).
Form Submission	Used for sending non-sensitive form data (e.g., search query).	Used for submitting sensitive or large form data (e.g., login, registration).
Bookmarking	URL with parameters can be	Cannot be bookmarked as data is not

	bookmarked.	in URL.
Use Cases	Search engines, filters, displaying content.	Form submissions, login forms, updating records.

b) Explain XML with respect to structure, declaration syntax, namespace. [8]

1. XML (eXtensible Markup Language) is a markup language used to store and transport data.
2. It uses **custom tags** to define data in a hierarchical and human-readable format.

1. XML Structure:

Structure includes:

- **Prolog** (XML declaration)
- **Root Element**
- **Child Elements**
- **Attributes (optional)**

Example:

```
<student>
  <name> Ayaan </name>
  <rollno> 45 </rollno>
</student>
```

2. XML Declaration Syntax:

- The XML declaration appears at the top of the file.
- It defines the version, encoding, and standalone status.

Syntax :

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
```

- version="1.0" → Specifies XML version.
- encoding="UTF-8" → Character encoding used.
- standalone="yes" → Specifies if the document relies on external DTD or not.

3. Namespaces in XML:

- Namespaces are used to **avoid element name conflicts** when combining XML documents from different sources.
- Defined using xmlns attribute

Syntax :

```
<books xmlns:bk="http://example.com/books">  
  <bk:title>Web Technology</bk:title>  
</books>
```

- bk is the prefix (alias), and http://example.com/books is the namespace URI.

➤ **MAY / JUN 2023 PYQ [UNIT – 3]**

Q1)

a) What are strengths of XML technology? Explain the need of XML. [5]

XML (eXtensible Markup Language) is a markup language used to store and transport data.

Strengths of XML Technology:

1. **Platform-Independent:** XML is a text-based format that can be used across different systems and platforms.
2. **Self-Descriptive:** XML data includes metadata, making it easy to understand without external documentation.
3. **Supports Hierarchical Data:** Ideal for representing complex data structures in a tree format.
4. **Extensible:** Users can define their own tags based on the application's needs.

Need for XML:

1. XML is needed for data transport, storage, and configuration in web applications.
2. It allows for standardized communication between heterogeneous systems.
3. XML is also essential for technologies like SOAP, RSS, and XHTML, and is widely used in Web Services and APIs.

b) What are DTD's? Explain how do they work? [5]

- **DTD (Document Type Definition)** defines the **structure and rules** for an XML document.
- It ensures that the XML data is **valid**, meaning it follows a **defined format and sequence of elements**.

How DTD Works:

- i. It defines rules like allowed tags, order of elements, nesting, and attribute types.
- ii. When an XML file is parsed, the parser checks it against the DTD to ensure it follows the defined structure.
- iii. If the XML data follows all rules, it is considered **valid XML**.

Example (Internal DTD):

```
<?xml version="1.0"?>
<!DOCTYPE student [
  <!ELEMENT student (name, age)>
  <!ELEMENT name (#PCDATA)>
  <!ELEMENT age (#PCDATA)>
]>
<student>
  <name>Himanshu</name>
  <age>21</age>
</student>
```

c) Explain URL writing and cookies in servlet with example.**[8]**

In web applications, the HTTP protocol is stateless, meaning each request is independent. To maintain user data across multiple requests, session tracking techniques like URL Rewriting and Cookies are used in servlets.

1. URL Rewriting

- URL rewriting is a session tracking technique where session information is appended to the URL as a query string.
- Used when cookies are disabled in the browser.

Advantages :

- Works even if the user has disabled cookies.
- Simple to implement.

Syntax Example:

java

```
String encodedURL = response.encodeURL("dashboard.jsp?user=Himanshu");  
out.println("<a href='" + encodedURL + "'>Go to Dashboard</a>");
```

Explanation:

- encodeURL() ensures the session ID is included if cookies are not supported.
- The URL may look like: dashboard.jsp?user=Himanshu&jsessionId=1234

2. Cookies:

- A cookie is a small piece of data stored in the client's browser, used to maintain session info across requests.
- Stored as a name–value pair.

Advantages:

- Hidden from user view.
- Automatically sent with every HTTP request.

Set Cookie Example (Servlet):

java

```
Cookie ck = new Cookie("user", "Himanshu");  
response.addCookie(ck);
```

Get Cookie Example:

java

```
Cookie[] cookies = request.getCookies();  
for(Cookie ck : cookies) {  
    if(ck.getName().equals("user")) {  
        out.print("Welcome " + ck.getValue());  
    }  
}
```

Q2

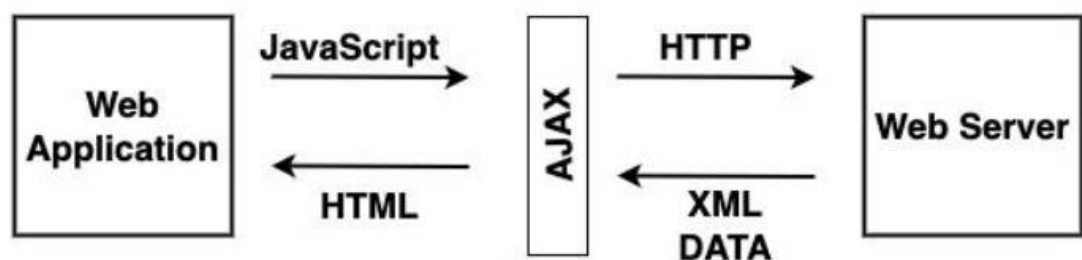
a) Describe servlet architecture in detail.

----> May / Jun 2022 Q1 b)

b) What do you understand by AJAX?. Explain it.

[5]

- AJAX stands for *Asynchronous JavaScript and XML*. It is not a programming language or a standalone technology, but rather a combination of several web technologies such as HTML, XHTML, CSS, JavaScript, DOM, XML, XSLT, and the XMLHttpRequest object.
- These technologies work together to create faster, more interactive, and dynamic web applications.
- The AJAX model allows web developers to send and receive data asynchronously without reloading the entire web page.
- This means parts of a web page can be updated independently, leading to a smoother and more seamless user experience



Advantages of AJAX:

- It creates responsive and interactive web applications.
- It reduces server traffic and load time since only necessary data is exchanged and not the whole page.

C) Write a servlet which accepts two numbers using POST method and display the maximum number

[5]

Java Servlets are programs that run on a Web or Application server and act as a middle layer between a requests coming from a Web browser or other HTTP client and databases or applications on the HTTP server.

✓ Servlet Code (Using `doPost` Method)

```

java                                                                    Copy Edit

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class MaxNumberServlet extends HttpServlet {
    public void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        // Fetch numbers from request
        int num1 = Integer.parseInt(request.getParameter("number1"));
        int num2 = Integer.parseInt(request.getParameter("number2"));

        // Calculate maximum
        int max = (num1 > num2) ? num1 : num2;

        // Display result
        out.println("<html><body>");
        out.println("<h3>The maximum of " + num1 + " and " + num2 + " is: " + max + "</h3>");
        out.println("</body></html>");

    }
}

```

➤ NOV / DEC 2023 PYQ [UNIT – 3]

Q1) a) Explain the Servlet architecture with diagram and explain servlet lifecycle. [9]

---> Refer may / jun 2022 Q1 b)

Refer nov / dec 2022 Q1 b)

b) Explain the concept of DTDs with example. Differentiate DTD Vs XML Schema (Min 4). [9]

A **Document Type Definition (DTD)** defines the **structure** and **legal elements and attributes** of an XML document. It ensures the XML file follows a specific format by validating element names, nesting, and order.

How DTD Works:

- DTD can be either **internal** (within XML file) or **external** (as a separate file).
- XML parser checks the document against the DTD rules.
- If the document conforms to the rules, it is considered **valid**.

✅ Example of Internal DTD:

```
xml

<?xml version="1.0"?>
<!DOCTYPE student [
  <!ELEMENT student (name, rollno)>
  <!ELEMENT name (#PCDATA)>
  <!ELEMENT rollno (#PCDATA)>
]>
<student>
  <name>Himanshu</name>
  <rollno>34</rollno>
</student>
```

Difference between DTD and XML Schema:

Feature	DTD (Document Type Definition)	XML Schema (XSD - XML Schema Definition)
1. Syntax	Uses its own non-XML syntax.	Written in XML syntax itself.
2. Data Type Support	Supports only basic data types like #PCDATA, CDATA, etc.	Supports a wide range of data types (string, int, float, date, boolean, etc.).
3. Namespace Support	Does not support XML namespaces.	Fully supports XML namespaces.
4. Extensibility	Not extensible or customizable.	Highly extensible and supports user-defined types.
5. Readability	Simple but less powerful.	Slightly complex but more powerful and expressive.
6. Default	Can assign default values to	Also allows default and fixed values

Values	elements/attributes.	with more validation rules.
7. Industry Preference	Older standard; less preferred today.	Widely used in modern XML-based web services and applications.

Q2) a) Explain the Servlet and MySQL database connectivity with example code to display data from employee(emp_id, emp_name, emp_dept) table. [9]

----> don't know (do by yourself)

**b) Write note on: i) AJAX
ii) XML**

----> May / Jun 2022 Q2 b)

➤ **MAY / JUN 2024 PYQ [UNIT -3]**

Q1)

a) Explain doGet() & doPost() methods of servlet. Differentiate do Get Vs do Post (Min 04). [9]

In Java Servlets, doGet() and doPost() are two important methods defined in the HttpServlet class used to handle HTTP requests. These methods process requests from clients (usually web browsers) and generate appropriate responses.

◆ `doGet()` Method:

- It handles **HTTP GET requests**.
- Commonly used when the client is requesting data, like fetching records.
- Parameters are appended in the URL (query string).
- It's **not secure** for sensitive data (e.g., passwords).

Syntax:

```
java                                                                    Copy Edit

protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    // Code to handle GET request
}
```

◆ `doPost()` Method:

- It handles **HTTP POST requests**.
- Used for **submitting form data** or sending sensitive information.
- Parameters are sent in the **request body**, making it more secure.
- Suitable for sending **large data**.

Syntax:

```
java                                                                    Copy Edit

protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    // Code to handle POST request
}
```

Feature	<code>doGet()</code>	<code>doPost()</code>
Request Method	Handles HTTP GET requests.	Handles HTTP POST requests.
Parameter Visibility	Parameters are appended in the URL and are visible to the user.	Parameters are sent in the body of the request and are hidden from the user.
Security	Less secure due to visibility of data in URL.	More secure, suitable for sending sensitive data like passwords.
Data Length	Limited data can be sent (depends	Can send large amounts of data, no

Feature	doGet()	doPost()
Limit	on URL length limit - ~2KB in browsers).	practical limit in most servers.
Caching	Can be cached and bookmarked.	Cannot be cached or bookmarked.
Idempotency	Usually idempotent (multiple identical requests have same effect).	Not idempotent (repeated submission may cause duplicate actions like payments).
Form Submission	Used for sending non-sensitive form data (e.g., search query).	Used for submitting sensitive or large form data (e.g., login, registration).
Bookmarking	URL with parameters can be bookmarked.	Cannot be bookmarked as data is not in URL.
Use Cases	Search engines, filters, displaying content.	Form submissions, login forms, updating records.

b) What is XML DTDs? Explain with example. Differentiate XML DTDs Vs XML schema (Min.04). [9]

DTD (Document Type Definition) defines the **structure and legal elements and attributes** of an XML document. It ensures the XML document follows specific formatting rules by validating element names, nesting, attributes, and their order.

How DTD Works:

- DTD can be **internal** (within the XML file) or **external** (referenced as a separate file).
- An **XML parser** checks the XML document against the DTD.
- If the document conforms to the rules, it is considered **valid**

✓ Example of Internal DTD:

```
xml

<!DOCTYPE employee [
  <!ELEMENT employee (emp_id, emp_name, emp_dept)>
  <!ELEMENT emp_id (#PCDATA)>
  <!ELEMENT emp_name (#PCDATA)>
  <!ELEMENT emp_dept (#PCDATA)>
]>
<employee>
  <emp_id>101</emp_id>
  <emp_name>John</emp_name>
  <emp_dept>IT</emp_dept>
</employee>
```

Feature	XML DTD	XML Schema
1. Syntax	Uses its own unique syntax	Uses XML syntax
2. Data Types	Limited (only parsed character data, etc.)	Rich data types (string, int, boolean, etc.)
3. Namespaces Support	Does not support namespaces	Fully supports namespaces
4. Extensibility	Not extensible	Highly extensible
5. Validation Accuracy	Less precise	More accurate and robust validation

Q2

- c) Explain the servlet lifecycle. Explain session management using cookies and URL Rewriting.

-----> Nov / Dec 2022 Q1 a)
May / Jun 2023 Q1 c)

- d) Write note on AJAX

-----> May / Jun 2022 Q2 b)

NOV / DEC 2024 UNIT – 3 PYQ

Q1)

a) Explain Http Servlet class with example. Explain the servlet Life cycle method. [10]

--> Already done !! (modify according to marks !)

b) Explain XML using DTDS & XML using XML Schema with example.[8]

--> already done !!

Q2)

a) Explain servlet session management & tracking methods. [9]

--> already done !!

b) Write note on : i) AJAX [5] ii) do Get Vs dopost [4]

--> already done !!