

**CS8651 – INTERNET PROGRAMMING**

**CS8651****INTERNET PROGRAMMING****L T P C****3 0 0 3****OBJECTIVES:***The student should be made to:*

- To understand different Internet Technologies.
- To learn java-specific web services architecture

**UNIT I WEBSITE BASICS, HTML 5, CSS 3, WEB 2.0 9**

Web Essentials: Clients, Servers and Communication – The Internet – Basic Internet protocols – World wide web – HTTP Request Message – HTTP Response Message – Web Clients – Web Servers – HTML5 – Tables – Lists – Image – HTML5 control elements – Semantic elements – Drag and Drop – Audio – Video controls – CSS3 – Inline, embedded and external style sheets – Rule cascading – Inheritance – Backgrounds – Border Images – Colors – Shadows – Text – Transformations – Transitions – Animations.

**UNIT II CLIENT SIDE PROGRAMMING 9**

Java Script: An introduction to JavaScript–JavaScript DOM Model–Date and Objects,–Regular Expressions– Exception Handling–Validation–Built-in objects–Event Handling– DHTML with JavaScript– JSON introduction – Syntax – Function Files – Http Request – SQL.

**UNIT III SERVER SIDE PROGRAMMING 9**

Servlets: Java Servlet Architecture- Servlet Life Cycle- Form GET and POST actions- Session Handling- Understanding Cookies- Installing and Configuring Apache Tomcat Web Server- DATABASE CONNECTIVITY: JDBC perspectives, JDBC program example – JSP: Understanding Java Server Pages-JSP Standard Tag Library (JSTL)-Creating HTML forms by embedding JSP code.

**UNIT IV PHP and XML 9**

An introduction to PHP: PHP- Using PHP- Variables- Program control- Built-in functions- Form Validation- Regular Expressions – File handling – Cookies – Connecting to Database. XML: Basic XML- Document Type Definition- XML Schema DOM and Presenting XML, XML Parsers and Validation, XSL and XSLT Transformation, News Feed (RSS and ATOM).

**UNIT V INTRODUCTION TO AJAX and WEB SERVICES 9**

AJAX: Ajax Client Server Architecture-XML Http Request Object-Call Back Methods; Web Services: Introduction- Java web services Basics – Creating, Publishing, Testing and Describing a Web services (WSDL)-Consuming a web service, Database Driven web service from an application –SOAP.

**OUTCOMES:**

*At the end of the course, the students should be able to:*

- Construct a basic website using HTML and Cascading Style Sheets.
- Build dynamic web page with validation using Java Script objects and by applying different event handling mechanisms.
- Develop server side programs using Servlets and JSP.
- Develop server side programs using JSP.
- Construct simple web pages in PHP and to represent data in XML format.
- Use AJAX and web services to develop interactive web applications

**TEXT BOOK:**

1. Deitel and Deitel and Nieto, —Internet and World Wide Web – How to Programl, Prentice Hall, 5th Edition, 2011.

**REFERENCES:**

1. Stephen Wynkoop and John Burke —Running a Perfect Website, QUE, 2nd Edition,1999.
2. Chris Bates, Web Programming – Building Intranet Applications, 3rd Edition, Wiley Publications, 2009.
3. Jeffrey C and Jackson, —Web Technologies A Computer Science Perspectivel, Pearson Education, 2011.
4. Gopalan N.P. and Akilandeswari J., —Web Technologyl, Prentice Hall of India, 2011.
5. UttamK.Roy, —Web Technologiesl, Oxford University Press, 2011.

**TOTAL: 45 PERIODS**

**COURSE OUTCOME**

|     |   |
|-----|---|
| CO1 | Construct a basic website using HTML and Cascading Style Sheets.  |
| CO2 | Build dynamic web page with validation using Java Script objects and by applying different event handling mechanisms. |
| CO3 | Develop server side programs using Servlets and JSP.  |
| CO4 | Develop server side programs using JSP.   |
| CO5 | Construct simple web pages in PHP and to represent data in XML format.  |
| CO6 | Use AJAX and web services to develop interactive web applications   |

**UNIT I****WEBSITE BASICS, HTML 5, CSS 3, WEB 2.0**

Web Essentials: Clients, Servers and Communication – The Internet – Basic Internet protocols – World wide web – HTTP Request Message – HTTP Response Message – Web Clients – Web Servers – HTML5 – Tables – Lists – Image – HTML5 control elements – Semantic elements – Drag and Drop – Audio – Video controls – CSS3 – Inline, embedded and external style sheets – Rule cascading – Inheritance – Backgrounds – Border Images – Colors – Shadows – Text – Transformations – Transitions – Animations.

**PART – A****1. What are Clients and Servers? (U)**

The client machine (or the client process) makes the request for some resource or service, and the server machine (the server process) handles the request and sends the response (result) back to the client.

**2. Mention the characteristics of RIA.(R)**

- ✓ Responsiveness
- ✓ Rich User Interface
- ✓ Direct interaction
- ✓ Partial-page updating
- ✓ Better feedback
- ✓ Consistency of look and feel
- ✓ Offline use
- ✓ Performance impact

**3. List few technologies of RIA. (R)**

- ✓ HTML5 + CSS3 + JavaScript + JavaScript frameworks
- ✓ Ajax
- ✓ Dojo
- ✓ Flex
- ✓ Silverlight
- ✓ JavaFX
- ✓ Ruby on Rails
- ✓ JavaServer Faces
- ✓ ASP.NET Ajax

**4. What are the benefits of RIA? (U)**

1. Increased productivity
2. Improving customer loyalty
3. Increase in new customers

- 4.Reduced operational costs
- 5.Installation is not required
- 6.Easy to upgrade
- 7.Easily made available over internet / intranet
- 8.Richer UI
- 9.More responsive UI
10. Client / Server Balance
11. Asynchronous communication
12. Network efficiency

**5. Define Internet.(R)**

Internet is the network of networks connected via the public backbone and communicating using TCP/IP communication protocol. Internet is a network of interconnected computers that is now global.

**6. What is meant by website? (U)**

A website is a set of pages of information on the internet about a particular subject, published by a single person or organization.

**7. What is meant by web server? (U) (APRIL/MAY 2021, 2022)**

A web server is a computer system or program that sends websites and information to internet users.

Various functions of web server are:-

1. The web server accepts the requests from the web browsers.
2. The user request is processed by the web server.
3. The web server respond to the users by providing the services requested.

**8. What is meant by web client? (U) (APRIL/MAY 2022)**

A web client is an application that communicates with a web server, using Hypertext Transfer Protocol (HTTP). The web browser is an example of a web client.

**9. Write the function of a Web Server. (U) (MAY / JUNE 2014)**

Various functions of web server are:-

1. The web server accepts the requests from the web browsers.
2. The user request is processed by the web server.
3. The web server respond to the users by providing the services requested.

**10. List any four common browsers. (R) (NOV / DEC 2011)**

- (i) Internet Explorer
- (ii) Mozilla Firefox
- (iii) Opera

(iv) Google Chrome

**11. Differentiate websites and web server. (AN)**

| Website  | Web server  |
|--|---|
| A website is a set of linked documents associated with a particular person, organization or topic that is held on a computer system and can be accessed as part of the world wide web. | The web server on the other side is a computer program, which delivers content, such as websites or web pages, for example, over the world wide web from a web server to your computer. |

**12. Define Protocol and list various types of protocols. (R) (MAY/JUNE 2013)**

Protocol is a set of rules and regulations which should be obeyed by both sender and receiver for data communication.

- ✓ **TCP/IP** - Transmission Control Protocol/Internet Protocol – Connection Oriented Protocol
- ✓ **HTTP** – Hypertext Transfer Protocol – Connection Oriented Protocol
- ✓ **FTP** – File Transfer Protocol - Connection Oriented Protocol
- ✓ **POP3** – Post Office Protocol – Connection Oriented Protocol
- ✓ **SMTP** – Simple Mail Transfer Protocol –
- ✓ **Telnet Protocol** – Used in Remote Login System
- ✓ **UDP** – User Datagram Protocol- Connectionless Protocol

Note: **POP3 and SMTP** protocols are used only in **email-generation**.

**13. Give the difference between Internet and Intranet. (AN)**

(NOV/DEC 2015, NOV/DEC 2016)

| Internet  | Intranet   |
|---|--|
| Internet is network of Computers which is open for all. | Intranet is network of Computers designed for a specific group of users. |
| Internet itself contains a large number of intranets.   | Intranet can be accessed from Internet but with restrictions.            |
| Unlimited number of users.                              | Limited number of Users.   |
| Visitors traffic is unlimited.                          | Limited visitors traffic.  |
| Contains unlimited source of information.               | Contains only specific group purpose information.                        |
| Collection of various LANs, WANs and MANs.              | Mostly any of LAN or MAN or WAN.   |

**14. Define WWW. (R)**

- WWW means World Wide Web – It is a collection of software and corresponding protocols used to access the resources over the network. It is used to connect wide area network.

- WWW contains huge amount of documents, images and other resources which can be accessed using the hyperlinks. Thus the internet can be used through the web.

**15. What is meant by Domain Name Service? (U)**

*Domain Name System* (or *Service* or *Server*), is an Internet service that translates *domain names* into IP addresses. Because domain names are alphabetic, they're easier to remember. The Internet however, is really based on IP addresses. Every time you use a domain name, therefore, a DNS service must translate the name into the corresponding IP address. For example, the domain name *www.example.com* might translate to *198.105.232.4*.

**16. State the function of DNS. (R) (NOV / DEC 2012)**

Domain Name Service (DNS) is the service used to convert human readable names of hosts to IP addresses. The main function of DNS is name resolution. Name resolution is the process of mapping a domain name to an IP address.

**17. Define URI. (R) (APRIL / MAY 2011)**

The Uniform Resource Identifier (URI) is a string used to identify the resources on the internet. The URI is a combination of URL and URN.

Syntax:

**Scheme: //authority/path?query**

**18. What is URL? Give its syntax. (U)**

URL means Uniform Resource Locator. It is used to identify the resource with its location. The resource can be a file, directory, HTML page, image, program, and so on. The URL consists a number of pieces like protocol, hostname, path, query string, etc.

The syntax of URL is

**Protocol: //username@hostname:port/path/filename#fragment?query**

Example: **http://www.example.org:5678/a/b/c.txt?t=win&s=chess#para5**

**19. What is meant by absolute URLs? (U)**

An *absolute URL* contains all the information necessary to locate a resource. An *absolute URL* uses the following format:

***scheme://server/path/resource***

*scheme* – Specifies how the *resource* is to be accessed.

*server* - Specifies the name of the computer where the *resource* is located.

*path* - Specifies the sequence of directories leading to the target. If *resource* is omitted, the target is the last directory in *path*.

*resource* - If included, *resource* is the target, and is typically the name of a file. It may be a *simple file*, containing a single binary stream of bytes, or a *structured document*, containing one or more storages and binary streams of bytes.

**20. What do you mean by Relative URLs? (May / June 2014)(U)**

A *relative URL* locates a resource using an absolute URL as a starting point. In effect, the "complete URL" of the target is specified by concatenating the absolute and relative URLs. A relative URL typically consists only of the *path*, and optionally, the *resource*, but no *scheme* or *server* i.e. Relative URLs are a way to identify a resource relative to their context.

Example:

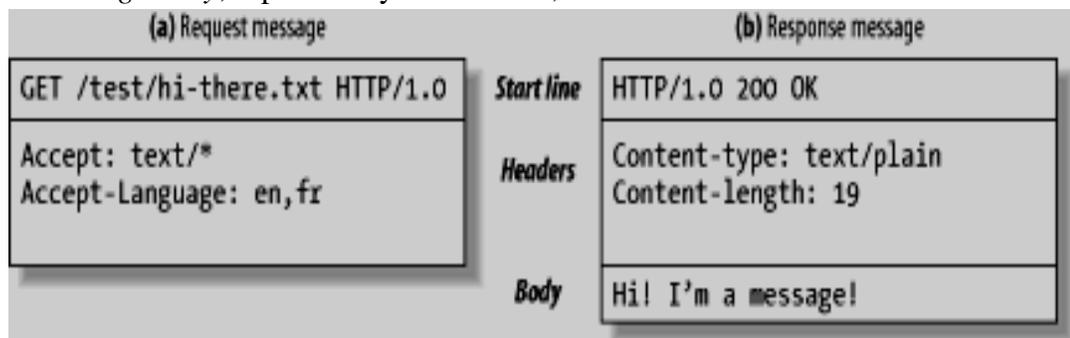
| Relative URL | Absolute URL  |
|--------------|---|
| d/e.html     | <a href="http://www.example.org/a/b/d/e.html">http://www.example.org/a/b/d/e.html</a> |

## 21. What do you mean by HyperText Transfer Protocol (HTTP)? (U)

HTTP is an asymmetric request-response client-server protocol. An HTTP client sends a request message to an HTTP server. The server, in turn, returns a response message. HTTP is a stateless protocol. In other words, the current request does not know what has been done in the previous requests.

## 22. What do you mean by HTTP Message? (U)

HTTP messages are how data is exchanged between a server and a client. There are two types of messages: *requests* sent by the client to trigger an action on the server, and *responses*, the answer from the server. HTTP messages are composed of textual information encoded in ASCII, and span over multiple lines. An HTTP message consists of a *message header* and an optional *message body*, separated by a *blank line*, as illustrated below:



## 23. Write the format of HTTP Request Message. (R)

The format of an HTTP request message is as follow:

### Request Line

The first line of the header is called the *request line*, followed by optional *request headers*.

The request line has the following syntax:

*request-method-name request-URI HTTP-version*

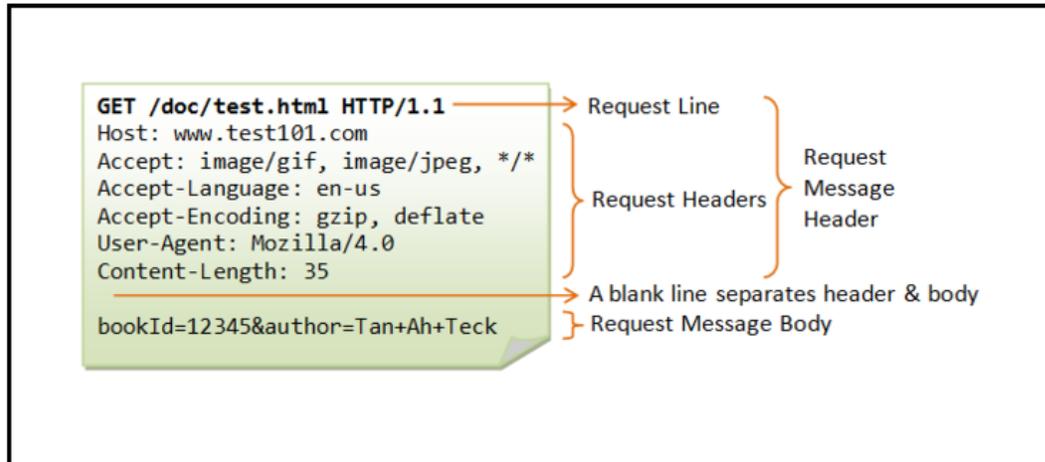
- *request-method-name*: HTTP protocol defines a set of request methods, e.g., GET, POST, HEAD, and OPTIONS. The client can use one of these methods to send a request to the server.
- *request-URI*: specifies the resource requested.

- *HTTP-version*: Two versions are currently in use: HTTP/1.0 and HTTP/1.1.

### Request Headers

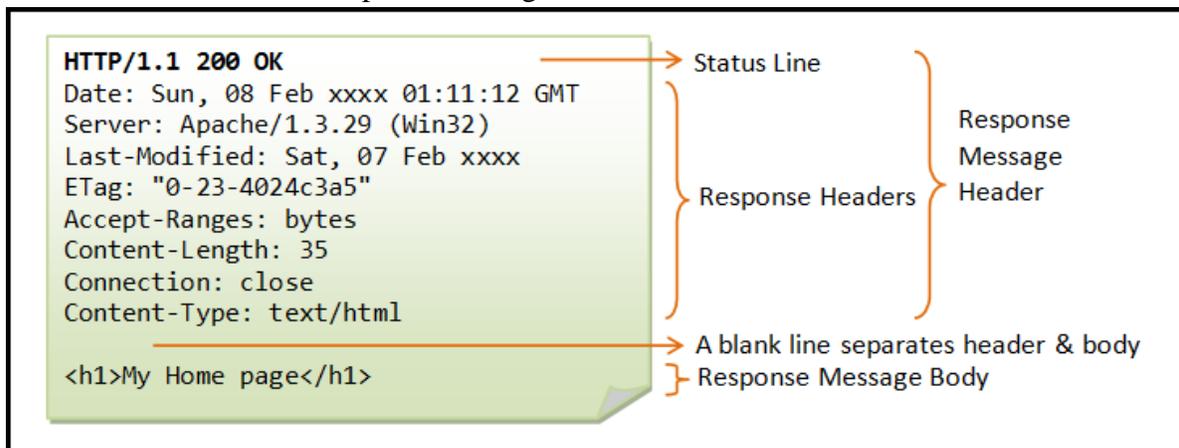
The request headers are in the form of name:value pairs. Multiple values, separated by commas, can be specified.

*request-header-name: request-header-value1, request-header-value2, ...*



### 24. Write the format of HTTP Response Message. (R)

The format of the HTTP response message is as follows:



### Status Line

The first line is called the *status line*, followed by optional response header(s).

The status line has the following syntax:

*HTTP-version status-code reason-phrase*

- *HTTP-version*: The HTTP version used in this session. Either HTTP/1.0 and HTTP/1.1.
- *status-code*: a 3-digit number generated by the server to reflect the outcome of the request.
- *reason-phrase*: gives a short explanation to the status code.
- Common status code and reason phrase are "200 OK", "404 Not Found", "403 Forbidden", "500 Internal Server Error".

**Response Headers**

The response headers are in the form name:value pairs:

*response-header-name: response-header-value1, response-header-value2, ...*

**25. What is HTML?(U)**

HTML is a **markup** language for **describing** web documents (web pages).

- HTML stands for **H**yper **T**ext **M**arkup **L**anguage
- A markup language is a set of **markup tags**
- HTML documents are described by **HTML tags**
- Each HTML tag **describes** different document content

**26. What is HTML5?(U)**

HTML5 is a next version of HTML that provides new features which will make HTML much easier. These new introducing features make your website layout clearer to both website designers and users. There are some elements like <header>, <footer>, <nav> and <article> that define the layout of a website.

**27. Mention some advantages of using HTML5.(AN)**

- It is enriched with advance features which makes it easy and interactive for designer/developer and users.
- It allows the user to play a video and audio file.
- It allows us to draw on a canvas.
- It facilitates to design better forms and build web applications that work offline.
- It provides advance features for that we would normally have to write JavaScript to do.

**28. List some of the newly added tags of HTML5.(R)**

| Tag          | Description  |
|--------------|--|
| <audio>      | It is used to play audio file in HTML.                           |
| <canvas>     | It is used to draw canvas.                                       |
| <data>       | It provides machine readable version of its data.                |
| <dialog>     | It defines a window or a dialog box.                             |
| <figcaption> | It is used to define a caption for a <figure> element.           |
| <figure>     | It defines a self-contained content like photos, diagrams etc.   |
| <footer>     | It defines a footer for a section.                               |
| <header>     | It defines a header for a section.                               |
| <main>       | It defines the main content of a document.                       |
| <mark>       | It specifies the marked or highlighted content.                  |
| <menuitem>   | It defines a command that the user can invoke from a popup menu. |

|            |   |
|------------|---|
| <nav>      | It is used to define the navigation link in the document. |
| <progress> | It specifies the progress of the task.                    |
| <section>  | It defines a section in the document.                     |
| <summary>  | It specifies a visible heading for <detailed> element.    |
| <time>     | It is used to define a date/time.                         |
| <video>    | It is used to play video file in HTML.                    |
| <wbr>      | It defines a possible line break.                         |

### 29. List the types of Lists in HTML. (R)

- **Unordered:** A bullet list
- **Ordered:** A numbered list

### 30. What are Semantic Elements in HTML5? (U)

A semantic element clearly describes its meaning to both the browser and the developer.

Examples of **non-semantic elements**: <div> and <span> - Tells nothing about its content.

Examples of **semantic elements**: <form>, <table>, and <article> - Clearly defines its content.

### 31. How will you add image in HTML 5? (A)

The <img> tag defines an image in an HTML page.

The <img> tag has two required attributes: **src** and **alt**.

The following attributes: align, border, hspace, and vspace are not supported in HTML5.

```

```

### 32. How will you create lists in HTML 5? (A)

1. Designate the beginning of the list with <ul> or <ol>.

The <ul> tag indicates an unordered (bullet) list, and the <ol> tag describes an ordered (numbered) list.

2. Surround each item in the list with an <li></li> pair.

**Example:**

```
<ol>
  <li>one</li>
  <li>two</li>
  <li>three</li>
</ol>
```

### 33. What is the use of Drag and drop property in HTML5? (U)

Drag and Drop (DnD) is powerful User Interface concept which makes it easy to copy, reorder and deletion of items with the help of mouse clicks. This allows the user to click and hold the mouse button down over an element, drag it to another location, and release the mouse button to drop the element there.

### 34. How to insert Audio & Video in HTML5? (A)

The HTML5 <audio> and <video> tags make it simple to add media to a website. Its need to set src attribute to identify the media source and include a controls attribute so the user can play and pause the media.

#### Embedding Video

Most commonly used video formats are –

- **Ogg** – Ogg files with Theora video codec and Vorbis audio codec.
- **mpeg4** – MPEG4 files with H.264 video codec and AAC audio codec.

<source> tag is used to specify media along with media type and many other attributes. A video element allows multiple source elements and browser will use the first recognized format.

```
<video width = "300" height = "200" controls autoplay>
  <source src = "/html5/foo.ogg" type = "video/ogg" />
  <source src = "/html5/foo.mp4" type = "video/mp4" />
  Your browser does not support the <video> element.
</video>
```

#### Embedding Audio

Most commonly used audio formats are **ogg, mp3 and wav**.

We can use <source> tag to specify media along with media type and many other attributes. An audio element allows multiple source elements and browser will use the first recognized format.

```
<audio controls autoplay>
  <source src = "/html5/audio.ogg" type = "audio/ogg" />
  <source src = "/html5/audio.wav" type = "audio/wav" />
  Your browser does not support the <audio> element.
</audio>
```

### 35. List out the Video/Audio Attributes in HTML5. (R)

#### Video Attributes & Description

- ✓ **autoplay** - This Boolean attribute if specified, the video will automatically begin to play back as soon as it can do so without stopping to finish loading the data.
- ✓ **autobuffer** - This Boolean attribute if specified, the video will automatically begin buffering even if it's not set to automatically play.
- ✓ **Controls** - If this attribute is present, it will allow the user to control video playback, including volume, seeking, and pause/resume playback.
- ✓ **Height** - This attribute specifies the height of the video's display area, in CSS pixels.
- ✓ **Loop** - This Boolean attribute if specified, will allow video automatically seek back to the start after reaching at the end.
- ✓ **Preload** - This attribute specifies that the video will be loaded at page load, and ready to run. Ignored if autoplay is present.
- ✓ **Poster** - This is a URL of an image to show until the user plays or seeks.

- ✓ **Src** - The URL of the video to embed. This is optional; you may instead use the <source> element within the video block to specify the video to embed.
- ✓ **Width** - This attribute specifies the width of the video's display area, in CSS pixels.

#### Audio Attributes & Description

- ✓ **Autoplay** - This Boolean attribute if specified, the audio will automatically begin to play back as soon as it can do so without stopping to finish loading the data.
- ✓ **Autobuffer** - This Boolean attribute if specified, the audio will automatically begin buffering even if it's not set to automatically play.
- ✓ **Controls** - If this attribute is present, it will allow the user to control audio playback, including volume, seeking, and pause/resume playback.
- ✓ **Loop** - This Boolean attribute if specified, will allow audio automatically seek back to the start after reaching at the end.
- ✓ **Preload** - This attribute specifies that the audio will be loaded at page load, and ready to run. Ignored if autoplay is present.
- ✓ **Src** - The URL of the audio to embed. This is optional; you may instead use the <source> element within the video block to specify the video to embed.

### 36. What is XHTML? (U)

XHTML stands for EXtensible HyperText Markup Language which is almost identical to HTML and is stricter than HTML.

### 37. Differentiate HTML and XHTML. (AN)

|                     | HTML   | XHTML   |
|---------------------|--|---|
| Introduction        | HTML or HyperText Markup Language is the main markup language for creating web pages and other information that can be displayed in a web browser. | XHTML (Extensible HyperText Markup Language) is a family of XML markup languages that mirror or extend versions of the widely used Hypertext Markup Language (HTML), the language in which web pages are written. |
| Filename extension  | .html, .htm  | .xhtml, .xht, .xml, .html, .htm   |
| Internet media type | text/html  | application/xhtml+xml   |
| Developed by        | W3C & WHATWG   | World Wide Web Consortium   |
| Type of format      | Document file format   | Markup language   |
| Extended from       | SGML   | XML, HTML   |
| Stands for          | HyperText Markup Language  | Extensible HyperText Markup Language  |
| Application         | Application of Standard Generalized Markup Language  | Application of XML  |

|          |  |   |
|----------|--|---|
|          | (SGML).  |   |
| Function | Web pages are written in HTML.                             | Extended version of HTML that is stricter and XML-based.                    |
| Nature   | Flexible framework requiring lenient HTML specific parser. | Restrictive subset of XML and needs to be parsed with standard XML parsers. |
| Origin   | Proposed by Tim Berners-Lee in 1987.                       | World Wide Web Consortium Recommendation in 2000.                           |
| Versions | HTML 2, HTML 3.2, HTML 4.0, HTML 5.                        | XHTML 1, XHTML 1.1, XHTML 2, XHTML 5.                                       |

**38. Define Cascading Style sheet? (R) (MAY/JUNE 2013)**

The Cascading Style Sheet is a markup language used in the web document for presentation purpose. The primary intension of CSS is to separate out the web content from the presentation. This separation of structure from presentation simplifies maintaining and modifying a document’s layout.

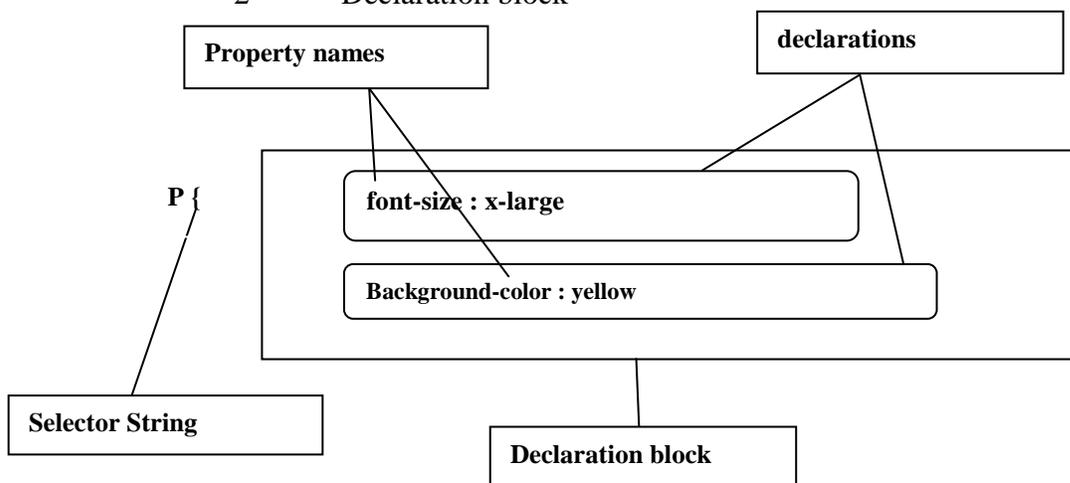
**39. Mention the need for Cascading Style Sheets. (U) (APRIL/ MAY 2011)  
Give some advantages of using CSS. (NOV/DEC 2013)**

- (1) CSS allows the separation between the information contained in a document and its presentation.
- (2) Any change in the presentation can be made without disturbing the information of the document.
- (3) CSS allows developers to give the consistent appearance to all elements of the web page.

**40. Give the syntax of a CSS rule. (R) (NOV/DEC 2011, MAY/JUNE 2012)**

A CSS style sheet consists of one or more style rules called rulesets.  
Each rule set consists of two parts:

- 1 Selector string
- 2 Declaration block



**41. What is Selector Strings? Explain its types.(U)**

The **selector string** indicates the elements to which the rule should apply, and each declaration within the declaration block specifies a value for one style property of those elements.

- **Single element selector strings**
  - It contains a single selector string eg.
  - **h1 {text-align:center}**
- **Multiple selector strings**
  - **h1,h2,h3,h4,h5,h6 {text-align:center}**
  - Where h1 to h6 gives same output.
- **Universal selector**
  - **\*{font-weight:bold}**
  - **Universal selector applies** to whole document by default
- **Specific elements by id**
  - **#id1,#id2 {font-style: italic}**
  - This can be used as follows in html program
  - **<h1 id="id1"> About Panimalar </h1>**
  - **<h2 id="id2"> Welcome </h2>**
- **Specific element by class selector**
  - **/\* class Selector \*/**
  - **.class1 {font-variant: Big-caps}**

This can be used as follows in html program

- `<p id="id1" class ="class1"> Panimalar Engineering College </p>`

**42. What are the different types of style sheets? (U)**

1. Inline style sheet
2. External style sheet
3. Embedded style sheet

**43. List the different <style> tag Font Attributes. (R)****FONT ATTRIBUTES:**

| Attribute Name     | Values   |
|--------------------|--|
| <b>Font-family</b> | A comma-delimited sequence of font family names (Serif, sanserif, cursive)   |
| <b>Font-style</b>  | Normal, italic or oblique.   |
| <b>Font-size</b>   | A term that denotes absolute size (xx-small, x-small, small, medium, large, x-large, xx-large), relative size (larger, smaller), a number (of pixels), percentage (of the parent element's size) |
| <b>Font-weight</b> | Normal, bold, lighter, or one of the nine numerical values (100,200 to 900)  |

## 44. List the different &lt;style&gt; tag color and background attributes. (R)

| Attributes               | Values  |
|--------------------------|---|
| <b>Color</b>             | Sets an element's text-color- a color name or a color code  |
| <b>Background-color</b>  | Specifies the color in an element's background. A color name or a color code.   |
| <b>Background-image</b>  | Sets the background image. A URL or none.   |
| <b>Background-repeat</b> | Within the background image specified, sets up how the image repeats throughout the page. Repeat-x (repeats horizontally), repeat-y(repeats vertically), repeat(both), no-repeat. |

## 45. List the different &lt;style&gt; tag text attributes. (R)

| Attributes             | Values  |
|------------------------|---|
| <b>Text-decoration</b> | Adds decoration to an element's text- none, underline, overline, line-through, blink  |
| <b>vertical-align</b>  | Determines an element's vertical position. Baseline, sub, super, top, text-top, middle, bottom, text-bottom, also percentage of the element's length. |
| <b>Text-transform</b>  | Applies a transformation to the text. Capitalize (puts the text into initial caps), uppercase, lowercase or none.                                     |
| <b>text-align</b>      | Aligns text with an element. Left, right, center, or justify can be given.  |
| <b>text-indent</b>     | Indents the first line of text. A percentage of the element's width or a length.  |

## 46. List the different &lt;style&gt; tag Border attributes.(R)

| Attributes                 | Values                                      |
|----------------------------|---|
| <b>Border-style</b>        | Solid, double, groove, ridge, inset, outset |
| <b>Border-color</b>        | A color name or color code                  |
| <b>Border-width</b>        | Thin, medium thick or length                |
| <b>Border-top-width</b>    | Thin, medium thick or length                |
| <b>Border-bottom-width</b> | Thin, medium thick or length                |
| <b>Border-left-width</b>   | Thin, medium thick or length                |
| <b>Border-right-width</b>  | Thin, medium thick or length                |
| <b>Border-top</b>          | Specifies width, color and style            |
| <b>Border-bottom</b>       | Specifies width, color and style            |
| <b>Border-left</b>         | Specifies width, color and style            |
| <b>Border-right</b>        | Specifies width, color and style            |
| <b>Border</b>              | Sets all the properties at once             |

**47. List the different <style> tag Margin attributes.(R)**

| Attributes           | Values                  |
|----------------------|-------------------------|
| <b>Margin-top</b>    | Percent, length or auto |
| <b>Margin-bottom</b> | Percent, length or auto |
| <b>Margin-left</b>   | Percent, length or auto |
| <b>Margin-right</b>  | Percent, length or auto |
| <b>Margin</b>        | Percent, length or auto |

**48. What are the advantages of CSS3 over CSS2? (AN)**

1. CSS3 is divided into many different documents called Modules. Every module adds new capability or extends features defined in CSS2 over preserving backward compatibility. Work on CSS3 started around the time of publication of the original CSS2 recommendation.
2. Because of the modularization in CSS3, every modules has different stability and is in different status.
3. The CSS3 version supports many more browsers than CSS2.
4. CSS3 has other added features such as new combinator, new CSS selectors, new pseudo-elements and new style properties

**49. List out some of the new properties added in CSS3. (R)**

Following is a list of CSS properties that were not defined in the CSS2.1 specification.

- animation
- background-clip
- background-origin
- background-size
- border-radius
- border-image
- box-decoration-break
- box-shadow
- box-sizing
- columns
- clear-after
- flex
- font-stretch
- font-size-adjust
- font-synthesis
- font-kerning
- font-variant-caps
- hanging-punctuation
- hyphens
- icon
- image-resolution

- image-orientation
- line-break
- object-fit
- object-position
- opacity
- outline-offset
- overflow-wrap / word-wrap
- backface-visibility
- perspective
- perspective-origin
- pointer-events (for HTML)
- resize
- tab-size
- text-align-last
- text-decoration-line
- text-decoration-skip
- text-decoration-position
- text-decoration-style
- text-emphasis
- text-justify
- text-orientation
- text-overflow
- transform
- transform-style
- text-shadow
- transition
- word-break
- word-spacing
- writing-mode

**50. How does a Rich Internet Application (RIA) differ from a traditional web application?**

**(AN) (MAY/JUNE 2016)**

| <b>Features</b>                | <b>Web applications</b> | <b>RIAs</b> |
|--------------------------------|-------------------------|-------------|
| <b>Rich User Experience</b>    | N                       | Y           |
| <b>Interactive, Responsive</b> | N                       | Y           |
| <b>Low Maintenance</b>         | Y                       | Y           |

**51. Write appropriate inline CSS to show a section of the HTML document with a font size of 20. (AP) (MAY/JUNE 2016)**

An inline CSS is used to apply a unique style to a single HTML element.

An inline CSS uses the style attribute of an HTML element.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Inline CSS</title>
</head>
<body>
<p style="font-size:20px;">This is thick</p>
</body>
</html>
```

## 52. List out some of the shadow effects added in CSS3. (R)

CSS is used to add shadow to text and to elements. It has following properties:

- text-shadow
- box-shadow

### CSS Text Shadow

The CSS text-shadow property applies shadow to text.

In its simplest use, only specify the horizontal shadow (2px) and the vertical shadow (2px). To add more than one shadow to the text, you can add a comma-separated list of shadows.

### CSS Box Shadow

The box-shadow property attaches one or more shadows to an element.

## 53. List out some of the border images added in CSS3. (R)

CSS Border image property is used to add image border to some elements. A sample syntax of boarder image is as follows –

```
#borderimg {
    border: 10px solid transparent;
    padding: 15px;
}
```

The most commonly used values are shown below –

**border-image-source** - Used to set the image path

**border-image-slice** - Used to slice the boarder image

**border-image-width** - Used to set the boarder image width

**border-image-repeat** - Used to set the boarder image as rounded, repeated and stretched

## 54. What is meant by transitions in CSS? (U)

The CSS transitions are effects that are added to change the element gradually from one style to another, without using flash or JavaScript.

specify two things to create CSS transition.

- The CSS property on which you want to add an effect.
- The time duration of the effect.

It is used to add transition effect for more than one CSS property. If you want to add transition effect on more than one property, separate those properties with a comma.

### 55. How to create CSS Animation? (C)

**CSS Animation property** is used to create animation on the webpage. It can be used as a replacement of animation created by Flash and JavaScript. An animation makes an element change gradually from one style to another. The animation could be bound to the selector by specifying at least these two properties:

- The name of the animation
- The duration of the animation

#### CSS animation properties

**@keyframes** - It is used to specify the animation.

**Animation** - This is a shorthand property, used for setting all the properties, except the animation-play-state and the animation-fill-mode property.

**animation-delay** - It specifies when the animation will start.

**animation-direction** - It specifies if or not the animation should play in reverse on alternate cycle.

**animation-duration** - It specifies the time duration taken by the animation to complete one cycle.

**animation-fill-mode** - It specifies the static style of the element. (when the animation is not playing)

**animation-iteration-count** - It specifies the number of times the animation should be played.

**animation-play-state** - It specifies if the animation is running or paused.

**animation-name** - It specifies the name of @keyframes animation.

**animation-timing-function** - It specifies the speed curve of the animation.

### 56. Define Transformation in CSS. List out its types. (R)

CSS3 supports transform property. This transform property facilitates you to translate, rotate, scale, and skew elements. Transformation is an effect that is used to change shape, size and position. There are two types of transforms:

#### CSS 2D Transforms

The CSS 2D transforms are used to re-change the structure of the element as translate, rotate, scale and skew etc.

#### CSS 3D Transforms

The CSS 3D transforms facilitate you to move an element to X-axis, Y-axis and Z-axis. Following is a list of 3D transforms methods:

### 57. Define rich internet application.(NOV/DEC 2016) (R)

A rich Internet application (RIA) is a Web application designed to deliver the same features and functions normally associated with desktop applications. RIAs generally split the processing across the Internet/network divide by locating the user interface and related activity and

capability on the client side, and the data manipulation and operation on the application server side.

**58. Give an example for inline style sheet. (R)****(NOV/DEC 2018)**

An inline style may be used to apply a unique style for a single element. To use inline styles, add the style attribute to the relevant element. The style attribute can contain any CSS property.

**Example**

Inline styles are defined within the "style" attribute of the relevant element:

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h1 style="color:blue;text-align:center;">This is a heading</h1>
```

```
<p style="color:red;">This is a paragraph.</p>
```

```
</body>
```

```
</html>
```

**59. Write about CSS Inheritance. (R)**

The property value is inherited from the nearest ancestor element that has a value for the property. If no ancestor has a value (or the property does not inherit) then CSS defines an initial value that is used

Property values:

Specified

- Absolute
- Relative

Computed

Actual

**60. What is Web 2.0 and give its example? (R)****(APRIL/MAY 2021)**

A Web 2.0 website allows users to interact and collaborate with each other through social media dialogue as creators of user-generated content in a virtual community. This contrasts the first generation of Web 1.0-era websites where people were limited to viewing content in a passive manner.

**Examples of Web 2.0** features include social networking sites or social media sites (e.g., Facebook), blogs, wikis, folksonomies ("tagging" keywords on websites and links), video sharing sites (e.g., YouTube), image sharing sites (e.g., Flickr), hosted services, Web applications ("apps"), collaborative consumption platforms, and mashup applications.

**PART – B**

- 1) Explain the following with an example :
  - a. Basic Internet Protocols
  - b. World Wide Web (U)
- 2) Illustrate the techniques used to optimize website assets loading. (U) (APRIL/MAY 2022)
- 3) List and explain the various HTML elements in detail. (U)  
(APRIL / MAY 2011, NOV / DEC 2013)
- 4) State the types of lists supported by HTML and explain them in detail. (U)
- 5) How do you create frames? Why do we need them? Develop an application to explain the same. (C)
- 6) Explain the way in which data can be presented in a tabular form using HTML. (U)
- 7) Develop a simple HTML application using List, Tables, Frames, Images and Forms and explain in details of its every attributes of respectively. (C)
- 8) Create a HTML document for a company home page and explain. (C) (MAY / JUNE 2013)
- 9) Explain about the all CSS Styles and its Core Syntax with an example. (U)
- 10) Elaborate with example the usage of CSS in HTML. (U) (NOV/DEC 2015)
- 11) List and explain in detail the various selector strings. (U)  
(APRIL / MAY 2011, MAY/JUNE 2012)
- 12) Explain the various types of Cascading Style Sheets with example. (U)
- 13) List and explain the various positioning schemes in detail. (Nov / Dec 2013) (U)
- 14) Explain the seven values of position attributes that is used to position an HTML element. (U)  
(APRIL/MAY 2022)
- 15) Explain the CSS box model in detail. (U) (MAY/JUNE 2012, NOV/DEC 2013)
- 16) Discuss on (i) Web Servers (ii) XHTML. (U) (NOV/DEC 2015)
- 17) Explain about internet technologies in detail. (U) (NOV/DEC 2016)
- 18) Write a XHTML program to display data in a table. (AP) (NOV/DEC 2016)
- 19) Explain the three ways of inserting a CSS to a document with examples. (U) (NOV/DEC 2016)
- 20) Discuss the structure of the HTTP request message. (U) (NOV/DEC 2018)
- 21) Explain about the XHTML DTD with an example. (U) (NOV/DEC 2018)
- 22) Explain the significance of XHTML with the help of a real time application. Write necessary code snippets. (U) (NOV/DEC 2018)
- 23) Discuss the functionalities of Webserver and Web client in detail. (U) (NOV/DEC 2018)
- 24) Illustrate the different types of request methods of http. Compare any two http request methods.
- 25) What do you mean by status code in http? (U) (APRIL/MAY 2021)
- 26) Name all the modules which are used in the current version of CSS. Distinguish between CSS2 and CSS3. What are the uses of an embedded style sheet? (AN) (APRIL/MAY 2021)

**UNIT – II CLIENT SIDE PROGRAMMING**

Java Script: An introduction to JavaScript–JavaScript DOM Model-Date and Objects,-Regular Expressions- Exception Handling-Validation-Built-in objects-Event Handling- DHTML with JavaScript- JSON introduction – Syntax – Function Files – Http Request – SQL.

**PART – A****1. What is JavaScript? (U)**

JavaScript is a client-side scripting language which is to develop dynamic and interactive web pages. It facilitates a programmer to design programs that enhance the functionality and appearance of web pages. It is not an Object Oriented Language but it is an object based language.

**2. What is the purpose of Java Script? (U)****Two purpose of JavaScript:**

- It Introduces client-side scripting, which makes web pages more dynamic and interactive.
- It provides the programming foundation for more complex server side scripting.

**3. List out the some of the java script properties. (R)****JavaScript Properties are:**

- ◆ JavaScript code did not need to be compiled.
- ◆ Interpreted vs. compiled languages:
  - Advantage: simplicity
  - Disadvantage: efficiency
- ◆ JavaScript is a scripting language: designed to be executed within a larger software environment
- ◆ JavaScript can be run within a variety of environments like browser, web server, etc.
- ◆ All data in JavaScript is an object or a property of an object.

**4. Write a JavaScript program to display “welcome” message on the browser window. (C)**

```
<!DOCTYPE html>
<html>
<head>
<title>This is a JavaScript example</title>
<script language="JavaScript">
    document.write("Hello World!");
</script>
</head>
<body> Hello, User! </body>
</html>
```

**5. Write a JavaScript program to display “welcome” message in the alert box. (C)**

```
<!DOCTYPE html>
<html>
<head>
<title>This is a JavaScript example</title>
<script language="JavaScript">
    Window.alert("Hello World!");

</script>
</head>
<body> Hello, User! </body>
</html>
```

**6. Give the syntax to get input from the user in JavaScript. (R)**

**Prompt dialog** - which allows the user to input a value that the script can use.

**Syntax:**

```
Variable = window.prompt("message to be displayed","default value");
```

**Example:**

```
var name = window.prompt("Please Enter your name:","XXX");
```

**7. List out the java script built in Objects. (R)****(MAY / JUNE 2012)**

- ✓ The **global Object** (window)
- ✓ **String**
- ✓ **Number**
- ✓ **Boolean**
- ✓ **Date**
- ✓ **Math**
- ✓ **Array**

**8. What is Regular Expression in Java Script? How do you define RE? (U)****(APRIL/MAY 2022)**

A regular expression is a certain way of representing a set of strings. Regular expressions are frequently used to test that a string entered in an HTML form has a certain format or, in the terminology of regular expressions, belongs to the set of string that have the correct format.

**Syntax:**            **/pattern/modifiers;**

**Example:**        **var regexp=/^[0-9]+\$;**

**9. Give the syntax to handle exceptions in JavaScript. (R)**

```
<script type="text/javascript">
<!--
try {
```

```

// Code to run
[break;]
} catch ( e ) {
// Code to run if an exception occurs
[break;]
}finally {
// Code that is always executed regardless of
// an exception occurring
}}
//-->
</script>

```

**10. Define event programming. (R)****(NOV / DEC 2015)**

Event-driven programming is a programming paradigm in which the flow of the program is determined by events such as user actions (mouse clicks, key presses), or messages from other programs/threads.

**11. What is meant by intrinsic event handling? (U)****(MAY / JUNE 2012)**

Intrinsic event handling is a mechanism to attach specific scripts to your documents that are executed only when something happens to an element.

- **event** - an action performed by the user, browser, or script
- **event handler** - JavaScript statements that execute when the event occurs
- An event is an occurrence of something potentially interesting to a script:
  - Ex: mouseover and mouseout events

**12. List some Intrinsic Event Attributes. (R) (APRIL/MAY 2011, NOV/DEC 2013)**

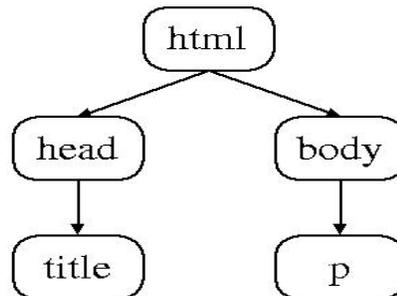
| Attribute   | When Called   |
|-------------|---|
| Onload      | The body of the document has just been <b>fully read and parsed by the</b> browser (this attribute pertains only do body and frameset)                |
| Onunload    | The browser is ready to load a <b>new document in place</b> of the current document (this attribute only pertains to body and frameset)               |
| Onclick     | The mouse button has been <b>clicked and released</b> over the element  |
| ondblclick  | The mouse button has been <b>double clicked</b> and released over the element   |
| onmousedown | The mouse has been <b>clicked</b> over the element  |
| onmouseup   | The mouse has been <b>released</b> over the element   |
| onmouseover | The mouse has just <b>moved</b> over the element  |
| onmousemove | The mouse has moved from one location to another over the element   |
| onmouseout  | The element has just moved away from the element.   |
| Onfocus     | The element has just to receive the keyboard focus (attribute pertains only to certain element including a label.input, select, text area and button) |
| Onblur      | The element has just lost the keyboard focus (attribute pertain only to the same  |

|            |   |
|------------|---|
|            | element as on focus   |
| onkeypress | This element has the focus and a key has been <b>present</b> and released   |
| onkeydown  | This element has the focus and a key has been <b>pressed</b> and released   |
| Onkeyup    | This element has the focus and a key has been <b>released</b> .   |
| Onsubmit   | This element is ready to be submitted (applies only to form elements)   |
| Onreset    | This element is ready to be reset (applies only to form elements)   |
| onselect   | Text in this element has been selected (highlighted) in preparation for editing (applies only to input and text area element) |
| Onchange   | The value of this element has changed (applies only to input, text area and select elements)                                  |

### 13. What do you mean by DOM Tree? Give an example. (U)

The documents in DOM are represented using tree like structure in which every element is represented as a node. Such tree is called as DOM tree.

Exmample:



**Fig. HTML DOM Node Tree**

### 23. What is JSON? Whether it is platform dependent? (U) (APRIL/MAY 2022)

A common use of JSON is to read data from a web server, and display the data in a web page.

- JSON stands for JavaScript Object Notation.
- JSON is lightweight data-interchange format.
- JSON is easy to read and write than XML.
- JSON is language independent.
- JSON supports array, object, string, number and values.

### 24. What are the features of JSON? (U)

1. Simplicity
2. Openness
3. Self-Describing
4. Internationalization
5. Extensibility

## 6. Interoperability

**25. What is the syntax of JSON? (U)**

JSON syntax is basically considered as a subset of JavaScript syntax; it includes the following:

- Data is represented in name/value pairs.
- Curly braces hold objects and each name is followed by ':'(colon), the name/value pairs are separated by , (comma).
- Square brackets hold arrays and values are separated by ,(comma).

**26. List out the JSON Data types. (R)**

In JSON, values must be one of the following data types:

- a string
- a number
- an object (JSON object)
- an array
- a boolean
- *null*

**27. JSON Function Files**

A common use of JSON is to read data from a web server, and display the data in a web page.

1: Create an array of objects.

2: Create a JavaScript function to display the array.

3: Use an array literal as the argument (instead of the array variable):

4: Put the function in an external js file

**28. What is the DOM used for? (U)****(APRIL/MAY 2021)**

**The Document Object Model (DOM)** is a programming interface for HTML and XML documents. It represents the page so that programs can change the document structure, style, and content. The DOM represents the document as nodes and objects. That way, programming languages can connect to the page.

**29. What do you mean by exception handling? (U)****(APRIL/MAY 2021)**

The **Exception Handling** in Java is one of the powerful mechanism to handle the runtime errors so that normal flow of the application can be maintained. Exception Handling is a mechanism to handle runtime errors such as `ClassNotFoundException`, `IOException`, `SQLException`, `RemoteException`, etc.

**PART – B**

1. Whether JavaScript a statically types or dynamically typed language. Justify your answer with examples. (AN) (APRIL/MAY 2022)
2. Write a JavaScript program to add two integers. Get the input from the user and display the result in pop-up box. (C)
3. Explain about DOM Nodes and Tress. (U)
4. Write JavaScript to find sum of first n even numbers and display the result. Get the value of 'n' from user. (C) (MAY/JUNE 2013)
5. Write JavaScript to find factorial of a given number. (C)
6. Describe how do you use regular expression on java script for form validation? Develop a complete application that would include functions to validate the user data. (C) (MAY/JUNE2014)
7. Explain in detail about how the exceptions can be handled in JavaScript. (U)
8. Explain about java script Built-in Objects. (U) (MAY / JUNE 2014)
9. Explain DOM Model and its properties. (U) (NOV/DEC 2015, APRIL/MAY 2022)
10. Explain Document Tree with an example. (U) (MAY/JUNE 2011, MAY/JUNE 2012)
11. Explain the DOM Event handling with suitable example. (U) (MAY/JUNE 2011, NOV/DEC 2011, MAY/JUNE 2012)
12. Use JavaScript and HTML to create a page with two panes. The first pane (on the left) should have a text area where HTML code can be typed by the user. The pane on the right side should display the preview of the HTML code typed by the user, as it would be seen on the browser. (C) (MAY/JUNE 2016)
13. Write a DHTML program to handle the user click event. (C) (NOV/DEC 2016)
14. Explain any two validation functions in Java script. (U) (NOV/DEC 2018)
15. Write a Java script program to print prime numbers from 1 to 100. (U) (NOV/DEC 2018)
16. Explain the concept of JSON with an example. (U) (NOV/DEC 2018)
17. What is the significance and what are the benefits of including 'Use strict' at the beginning of a JavaScript source file? (U) (APRIL/MAY 2021)
18. Demonstrate use of JSON. How to convert JavaScript objects to JSON ? List the benefits of JSON over XML. (A) (APRIL/MAY 2021)

**UNIT - III SERVER SIDE PROGRAMMING**

Servlets: Java Servlet Architecture- Servlet Life Cycle- Form GET and POST actions- Session Handling- Understanding Cookies- Installing and Configuring Apache Tomcat Web Server- DATABASE CONNECTIVITY: JDBC perspectives, JDBC program example – JSP: Understanding Java Server Pages-JSP Standard Tag Library (JSTL)-Creating HTML forms by embedding JSP code.

**PART - A****1. What is a Servlet? (R)**

Servlet is a server side programming language which is used for generating dynamic web pages. It generates web-page as a response of the request received from client (browser).

**2. List out the Advantages of Servlet. (R)**

The advantages of Servlet are as follows:

- i. **Better performance:** because it creates a thread for each request, not process.
- ii. **Portability:** because it uses Java language.
- iii. **Robust:** JVM manages Servlets, so we don't need to worry about the memory leak, garbage collection, etc.
- iv. **Secure:** because it uses java language.

**3. What is Static webpage and Dynamic webpage? (R)**

The webpages which are same for all the users are static webpages and the webpages that are dynamically generated based on the user's request (that may be different for each user depending on the request) are known as dynamic webpages. Servlet is mainly used for dynamic webpages.

**4. What is a web application? (R)**

A web application is an application accessible from the web. A web application is composed of web components like Servlet, JSP, Filter, etc. and other elements such as HTML, CSS, and JavaScript. The web components typically execute in Web Server and respond to the HTTP request.

**5. What is a web application? (R)**

Following the stages of servlet life cycle:

- Loading of Servlet class: The servlet container finds the servlet class mentioned in web.xml file and loads it.
- Servlet instantiation: The object of servlet class gets created in this phase.
- Initialization: Servlet initialization by calling init() method.
- Servicing the request: In this phase the servlet service the client request by calling the service() method.

- Destroy: Last phase of servlet life cycle. The destroy() method free up the servlet instance so that it can be garbage collected.

#### 6. Why Servlet is better than CGI? (U)

- 1) Servlet responses faster than CGI because it uses the multithreading concept to service each request. CGI performance is not that good as it creates a new object for each request while servlet allots a new thread for each request.
- 2) Learning and implementing servlet is quite easier compared to CGI.
- 3) Memory consumption is low in servlet compared to CGI.

#### 7. What are Servlets? List its life cycle methods. (U)

(MAY / JUNE 2012, MAY / JUNE 2013) (APRIL/MAY 2021)

Servlets are Java programs which run inside a Servlet Container. A Servlet Container is much like a Web Server which handles user requests and generates responses. Servlets extend the server's functionality by allowing it to generate dynamic content.

Life cycle methods of a Servlet:

- i. **Init()** – It performs servlet **initialization** tasks. This method is called when the servlet is loaded in the memory for the first time.
- ii. **Service()** – is a method actually called when an HTTP request is received; this method, in turn, calls doGet(). Normally, you should **not override service()**.
- iii. **Destroy()** – method will be called when the server is in the process of taking a servlet **out of service**, usually when the **server itself is shutting down**.

#### 8. Differentiate Client-Side Scripting and Sever-side Scripting. (AN)

| S.No | Server-Side Scripting  | Client-Side Scripting   |
|------|--|---|
| 1    | It is used to create dynamic pages based on browser request  | It is used when the browser already has all the code & the page is altered on the basis of user input |
| 2    | Web server executes the server side scripting  | Web browser executes the client side scripting  |
| 3    | It is used to connect to the databases on the web server   | It cannot be used to connect to the databases on the web server                                       |
| 4    | Responses from server side scripts are slower because the scripts are processed on remote computer | Response from client side scripts are faster because the scripts are processed on local computer      |
| 5    | Examples: Servlets, JSP, PHP, ASP, Ruby, Perl, Python  | Examples: JavaScript, VBScript  |

**9. Difference between GenericServlet and HttpServlet. (AN)**

- ✓ GenericServlet is an abstract class which implements Servlet interface while HttpServlet abstract class extends the GenericServlet class. In short: GenericServlet class is a parent class for HttpServlet.
- ✓ GenericServlet does not support any protocol. HTTPSeervlet support HTTP and HTTPS Protocol.
- ✓ GenericServlet cannot handle cookies and session while HttpServlet can handle them.**GenericServlet class**

**10. What are the common HTTP request methods used for request-response between browser and server? (U)**

| Method  | Description  |
|---------|--|
| GET     | Requests data from a specified resource                        |
| POST    | Submits data to be processed to a specified resource           |
| HEAD    | Same as GET but returns only HTTP headers and no document body |
| PUT     | Uploads a representation of the specified URI                  |
| DELETE  | Deletes the specified resource                                 |
| OPTIONS | Returns the HTTP methods that the server supports              |
| CONNECT | Converts the request connection to a transparent TCP/IP tunnel |

**11. Differentiate GET and POST method. (AN)**

| S.NO | GET   | POST  |
|------|---|---|
| 1    | Query String is part of URL   | Query String is sent as body of the HTTP request  |
| 2    | Length of the query string may be limited   | Length of the query string is unlimited   |
| 3    | Recommended when parameter data is not stored but used only to request information.<br><b>Eg:</b> Search Engine Query | Recommended if parameter data is intended to cause the server to update stored data.<br><b>Eg:</b> Online transaction Query |
| 4    | The URL can be bookmarked or emailed and the same data will be passed to the server when the URL is revisited         | Most browser will warn you if they are about to resubmit POST data to avoid duplicate update                                |
| 5    | <b>Eg:</b><br><form name="form1" method="GET">  | <b>Eg:</b><br><form name="form1" method="POST">   |
| 6    | <b>Servlet Method:</b><br>doGet(HttpServletRequest req,<br>HttpServletResponse res){ }                                | <b>Servlet Method:</b><br>doPost(HttpServletRequest req,<br>HttpServletResponse res){ }                                     |

**12. What is ServletConfig? (R)**

ServletConfig interface belongs to the package javax.servlet.ServletConfig. It is used for passing the configuration parameters to the servlet. Servlet container instantiate it implicitly.

**13. What is ServletContext?(R)**

Each web application has a common ServletContext. All the servlets in the web application can access the ServletContext. It has the web-application information & resources which are common and accessible to all the servlets present in the web application.

**14. What is meant by a Session? (U)**

A collection of HTTP requests to the server, all associated with a single session ID is known as a Session. i.e Session means to time duration of the conversation held between the client and the server. Session provides a way to identify a user across more than one page request or visit to a Web site and to store information about that user. The session persists for a specified time period, across more than one connection or page request from the user.

**15. What is meant by Session Tracking or Session Handling? (U) (APRIL/MAY 2021, 2022)**

**Session Tracking** is the capability of a server to maintain the current state of a single client's sequential requests. Using session tracking, we can keep track of previous session (conversations) held between server and the browser by using session ID.

**16. What is Session ID? (U)**

Session ID is an unique identification number assigned by the server to each new HTTP request for identifying them in subsequent conversations. It is used for sending all state information to and fro between the browser and server.

**17. What are the different techniques used for handling sessions? (R)**

- ✓ Cookies
- ✓ Hidden Form Field
- ✓ URL Rewriting
- ✓ HttpSession

**18. What are cookies? Give its uses. (U)****(NOV/DEC 2015)**

A **Cookie** is a name-value pair of information that a web server sends to a client machine as a part of an HTTP response. The browser then returns the cookie unchanged to the server to indicate the state of the conversation.

**19. What is Servlet chaining? (R)**

Servlet chaining is a concept where the request is processed in a chain of servlets. First Servlet processes the request partially and passes to the second one, then second servlet process it and passes to third one and so on. The last servlet returns the response to the client (browser).

**13. How Servlet maintains session using cookies?(U)**

Cookie is a small piece of information, which is sent by a servlet to the Web browser. Cookies gets stored in the browser and returned back to the server when needed. A cookie has a name, a single value, and few other attributes.

**14. What are Cookies?(R)**

A cookie is a small text file that is stored in the client's machine, which will be send to the server on each request. You can put your session data inside the cookie. The biggest problem **in using cookie is clients may disable the cookie.**

**15. How to create a cookie using servlet?(U)**

Setting cookies with servlet involves three steps:

(1) Creating a Cookie object:

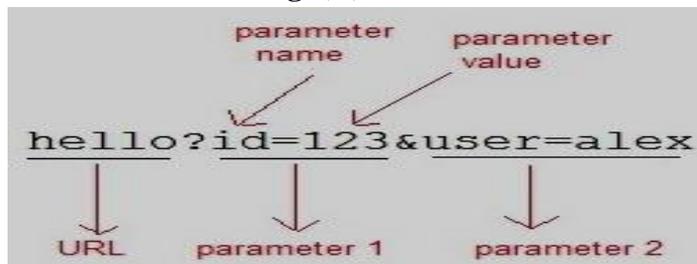
```
Cookie cookie = new Cookie("key","value");
```

(2) Setting the maximum age:

```
cookie.setMaxAge(60*60*24);
```

(3) Sending the Cookie into the HTTP response headers:

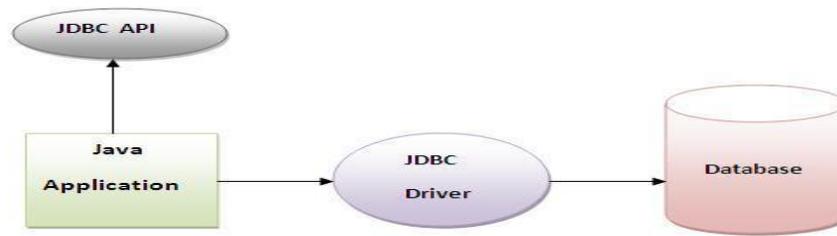
```
response.addCookie(cookie);
```

**16. What is URL rewriting?(R)**

In URL rewriting, a token(parameter) is added at the end of the URL. The token consist of name/value pair seperated by an equal(=) sign.

**17. What is JDBC? (U)**

**Java Database Connectivity(JDBC)** is an **Application Programming Interface(API)** used to connect Java application with Database. JDBC is used to interact with various type of Database such as Oracle, MS Access, My SQL and SQL Server. JDBC can also be defined as the platform-independent interface between a relational database and Java programming. It allows java program to execute SQL statement and retrieve result from database.



**18. What is the use of JDBC? (U)**

Before JDBC, ODBC API was the database API to connect and execute query with the database. But, ODBC API uses ODBC driver which is written in C language (i.e. platform dependent and unsecured). That is why Java has defined its own API (JDBC API) that uses JDBC drivers (written in Java language).

**19. Define JDBC Driver and List its type.(R)**

JDBC Driver is a software component that enables java application to interact with the database. JDBC Driver is required to process SQL requests and generate result. The following are the different types of driver available in JDBC.

- **Type-1 Driver** or **JDBC-ODBC bridge**
- **Type-2 Driver** or **Native API Partly Java Driver**
- **Type-3 Driver** or **Network Protocol Driver (fully Java Driver)**
- **Type-4 Driver** or **Thin Driver (fully Java Driver)**

**20. List out the steps to connect a Java application with DB.(R)**

There are 5 steps to connect any java application with the database in java using JDBC.

They are as follows:

1. Register the driver class
2. Creating connection
3. Creating statement
4. Executing queries
5. Closing connection

**21. What is JSP? (U)**

**(MAY/JUNE 2013)**

Java Server Pages (JSP) is a technology for developing web pages that support dynamic content which helps developers insert java code in HTML pages by making use of special JSP tags.

**22. List some advantages of JSP. (AN)**

1. Extension to Servlet: JSP technology is the extension to servlet technology. We can use all the features of servlet in JSP
2. Dynamic contents can be handled using JSP because JSP allows scripting and element based programming.
3. JSP is useful for server side programming.
4. JSP programming is easy to learn and easy to implement.
5. JSP programming environment provides the separation between presentation logic and business logic.
6. JSP provides optional mechanism in configuring web application file (web.xml).
7. JSP allows creating and using our own custom tag libraries. Hence any application specific Requirements can be satisfied using custom tag libraries. This helps the developer to develop any kind of application.
8. Fast Development: No need to recompile and redeploy If JSP page is modified, we don't need to recompile and redeploy the project.

**23. Differentiate JSP and Servlet.(AN)**

| S.No | JSP  | Servlets   |
|------|--|--|
| 1.   | JSP is a webpage scripting language that can generate dynamic content.   | Servlets are Java programs that are already compiled which also creates dynamic web content.                             |
| 2.   | In MVC, JSP act as a view.   | In MVC, servlet act as a controller.   |
| 3.   | It's easier to code in JSP than in Java Servlets.  | Its little much code to write here.  |
| 4.   | JSP are generally preferred when there is not much processing of data required.  | Servlets are best for use when there is more processing and manipulation involved.                                       |
| 5.   | JSP run slower compared to Servlet as it takes compilation time to convert into Java Servlets.   | Servlets run faster compared to JSP.   |
| 6.   | The advantage of JSP programming over servlets is that we can build custom tags which can directly call Java beans.  | There is no such custom tag facility in servlets.  |
| 7.   | JSP is almost a replacement of Servlets, (by large, the better word is extension of Servlets), where coding decreases more than half   | Servlets are difficult to code   |
| 8.   | In JSP, static code and dynamic code are separated.<br>Example:<br>Hello Mr. // static content<br><%= str %> // dynamic content as JSP code<br>you are great man // static content | Both static code and dynamic code are put together<br>Example:<br>out.println("Hello Mr." + str + " you are great man"); |

|     |   |   |
|-----|---|---|
| 9.  | In JSP, both presentation layer and business logic layer can be separated with the usage of JavaBeans.    | Both presentation layer and business logic layer put together in Servlets                         |
| 10. | Writing alias name in <url-pattern> tag of web.xml is optional in JSP.                                    | Writing alias name in <url-pattern> tag of web.xml is mandatory in Servlets                       |
| 11. | JSP needs no compilation by the Programmer. Programmer deploys directly a JSP source code file in server. | In Servlets, the Programmer compiles manually a Servlet file and deploys a .class file in server. |

**24. State the information in a JSP Documents. (R)**

JSP page is a simple web page which contains the three types of information:

1. JSP Markup
2. JSP Expression Language expressions
3. Template data

**25. What is a Scriptlet? (U)**

**A scriptlet is a fragment of Java code that is embedded within the JSP document.**

A scriptlet can contain any number of JAVA language statements, variable or method declarations, or expressions that are valid in the page scripting language.

**Syntax:** `<% java source code %>`

**26. What is meant by JSTL? (U)**

The Java Server Pages Standard Tag Library (JSTL) is a collection of useful JSP tags which encapsulates core functionality common to many JSP applications.

**27. List out the various JSTL tags.(R)**

- Core Tags
- Formatting tags
- SQL tags
- XML tags
- JSTL Functions

**28. Write appropriate javascript code to remove an element(current element) from a DOM. (C) (MAY/JUNE 2016)**

Navigate to the node to be removed then use parentNode property and removeChild ()

```

<html>
  <head><script src="loadxml.js" </script></head>
  <body><script>
    xmlDoc=loadXMLDoc("books.xml");
    x=xmlDoc.getElementsByTagName("book")[0];
    x.parentNode.removeChild(x);
  </script></body>
</html>

```

**29. Write the code segment to store current server time in session using java servlet API. (C) MAY/JUNE 2016)**

```

protected void doGet(HttpServletRequest req, HttpServletResponse resp)
    throws ServletException, IOException {
    resp.setContentType("text/plain");
    resp.setCharacterEncoding("UTF-8");
    PrintWriter pw = resp.getWriter();

    HttpSession session = req.getSession(true);
    session.invalidate();

    boolean result;
    try {
        session.getCreationTime();
        result = false;
    } catch (IllegalStateException ise) {
        result = true;
    }

    if (result) {
        pw.print("PASS");
    } else {
        pw.print("FAIL");
    }
}

```

**32. List any four methods of Date object. (R) (NOV/DEC 2016)**

The Date object is a datatype built into the JavaScript language.

| Method    | Description  |
|-----------|--|
| Date()    | Returns today's date and time                                    |
| getDate() | Returns the day of the month for the specified date according to |

|               |   |
|---------------|---|
|               | local time.   |
| getDay()      | Returns the day of the week for the specified date according to local time. |
| getFullYear() | Returns the year of the specified date according to local time.             |

**33. Explain the servlet interface and its methods. (U) (NOV/DEC 2016)**

**Servlet interface** provides common behaviour to all the servlets. Servlet interface needs to be implemented for creating any servlet (either directly or indirectly). It provides 3 life cycle methods that are used to initialize the servlet, to service the requests, and to destroy the servlet and 2 non-life cycle methods.

**Methods of Servlet interface**

There are 5 methods in Servlet interface. The init, service and destroy are the life cycle methods of servlet. These are invoked by the web container.

| Method   | Description   |
|--|---|
| <b>public void init(ServletConfig config)</b>                                | initializes the servlet. It is the life cycle method of servlet and invoked by the web container only once. |
| <b>public void service(ServletRequest request, ServletResponse response)</b> | provides response for the incoming request. It is invoked at each request by the web container.             |
| <b>public void destroy()</b>   | is invoked only once and indicates that servlet is being destroyed.   |
| <b>public ServletConfig getServletConfig()</b>                               | returns the object of ServletConfig.  |
| <b>public String getServletInfo()</b>  | returns information about servlet such as writer, copyright, version etc.                                   |

**PART B**

1. What are the different phases of Java Servlet Life Cycle? Illustrate the different functionalities of servlet container. (U) (APRIL/MAY 2021)
2. Discuss about the architecture and life cycle of a servlet with example.  
Explain Single-Threaded Model in Servlets. How does Servlets Collaboration take place? (U) (NOV/DEC 2012, MAY/JUNE 2013, NOV/DEC 2013. APRIL/MAY 2022)
3. Explain the Servlet operation in detail with sample servlet program. (U) (MAY / JUNE 2011)
4. Explain in detail with an example the dynamic content generation by a servlet. (U) (MAY/JUNE 2012, MAY/JUNE 2014)
5. What is Session? Explain how client state is maintained using session and also

- Explain about session tracking and session management using an example. (U)  
(NOV / DEC 2011)
6. Write a servlet to illustrate the principles of Cookies and explain. (C)  
(MAY/JUNE 2013, MAY/JUNE 2014)
7. Explain how java servlets perform session handling. (U) (MAY / JUNE 2012)
8. Write a servlet to illustrate the principles of Cookies and explain. (A)  
(MAY/JUNE 2013, MAY/JUNE 2014)
9. Explain the use of cookies for tracking requests with a program. (C) (NOV/DEC 2016)
10. Write a Java servlet to display net salary of employee. Use JDBC connectivity to get employee details from data base. (A)  
(MAY/JUNE 2013)
11. Explain the JDBC and its components. (U) (APRIL/MAY 2022)
12. What is JDBC? What is the use of it? Write and explain steps to connect a Java application to the DB with sample program. (U)  
(NOV/DEC 2015)
13. Elaborate briefly about JDBC database connection pool. How to setup in Java? Explain different types of JDBC drivers in Java? (U) (APRIL/MAY 2021, 2022)
14. Explain the following with an example. (U)  
a. JSP and Servlet                      b. Running JSP Application.
15. Write a client server JSP program to find simple interest and display the result in the client.  
(C) (MAY / JUNE 2013)
16. Explain the JSP scripting components in detail with examples. (U) (NOV/DEC 2016)
17. Explain in detail about JSTL with necessary examples. (U)
18. Using HTML and JSP, design a scientific calculator. (C) (NOV/DEC 2015)
19. Assume that a database has a table Employee with two columns EmployeeID and Name. Assume that the administrator user id and password to access the database table are, scott and tiger. Write a JDBC program that can query and print all the entries in the table Employee. Make the database connection using a type2 driver database.driver and connection string jdbc:db:oci. (C)  
(MAY/JUNE 2016)

**UNIT IV - PHP and XML**

An introduction to PHP: PHP- Using PHP- Variables- Program control- Built-in functions- Form Validation- Regular Expressions – File handling – Cookies – Connecting to Database. XML: Basic XML- Document Type Definition- XML Schema DOM and Presenting XML, XML Parsers and Validation, XSL and XSLT Transformation, News Feed (RSS and ATOM).

**PART - A****1. What is PHP? (NOV/DEC 2016) (U)**

**Hypertext Preprocessor** is an acronym for PHP. PHP is an open-source server scripting language, and a powerful tool for creating dynamic and interactive Web pages. PHP is a widely-used, free, and efficient alternative to competitors such as Microsoft's ASP.

**2. What is a PHP file? (U)**

- PHP files can contain text, HTML, CSS, JavaScript, and PHP code
- PHP code are executed on the server, and the result is returned to the browser as plain HTML
- PHP files have extension ".php"

**3. List out the features of PHP. (R)**

- PHP can generate dynamic page content
- PHP can create, open, read, write, delete, and close files on the server
- PHP can collect form data
- PHP can send and receive cookies
- PHP can add, delete, modify data in your database
- PHP can be used to control user-access
- PHP can encrypt data
- With PHP we are not limited to output HTML. we can output images, PDF files, and even Flash movies. we can also output any text, such as XHTML and XML.

**4. What are advantages of PHP? (AN)**

- Used for creating dynamic and interactive web pages
- PHP runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)
- PHP is compatible with almost all servers used today (Apache, IIS, etc.)
- PHP supports a wide range of databases
- PHP is free. Download it from the official PHP resource
- PHP is easy to learn and runs efficiently on the server side

**5. Write the syntax for writing PHP scripts. (R)**

A PHP script can be placed anywhere in the document.

A PHP script starts with **<?php** and ends with **?>**:

```
<?php  
    // PHP code goes here  
?>
```

The default file extension for PHP files is ".php".

A PHP file normally contains HTML tags, and some PHP scripting code.

**6. Write the PHP script to print "Hello" message on the screen. (C)**

```
<!DOCTYPE html>  
<html>  
<body>  
<h1>My first PHP page</h1>  
  
<?php  
    echo "Hello World!";  
?>  
  
</body>  
</html>
```

**7. How will you create variables in PHP? (AN)**

Variables are "containers" for storing information.

In PHP, a variable starts with the \$ sign, followed by the name of the variable:

*Example:*

```
<?php  
$txt = "Hello world!";  
$x = 5;  
$y = 10.5;  
?>
```

**8. List the rules for creating variables in PHP.(R)**

A variable can have a short name (like x and y) or a more descriptive name (age, carname, total\_volume).

**Rules for PHP variables:**

- A variable starts with the \$ sign, followed by the name of the variable
- A variable name must start with a letter or the underscore character
- A variable name cannot start with a number
- A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and \_)
- Variable names are case-sensitive (\$age and \$AGE are two different variables)

**9. What is meant by variable scope?(U)**

The lifetime of a variable during the program execution is called the scope of a variable and it is the part of the script where the variable can be referenced/used.

PHP has three different variable scopes:

- Local - A variable declared **within** a function has a LOCAL SCOPE and can only be accessed within that function
- global - A variable declared **outside** a function has a GLOBAL SCOPE and can only be accessed outside a function
- static – A variable declared inside the function with the keyword “static” and it helps to preserve the variable even after the function execution has completed.

**10. List the various data types supported in PHP.(R)**

PHP variables are loosely typed – they can contain different types of data at different times.

The various data types supported in PHP are:

| Type                       | Description  |
|----------------------------|--|
| <b>int, integer</b>        | Whole numbers (i.e., numbers without a decimal point)      |
| <b>float, double, real</b> | Real numbers (i.e., numbers containing a decimal point)    |
| <b>String</b>              | Text enclosed in either single(‘ ’) or double (“”) quotes. |
| <b>bool, boolean</b>       | True or false  |
| <b>Array</b>               | Group of elements  |
| <b>Object</b>              | Group of associated data and methods                       |
| <b>Resource</b>            | An external resource – usually information from database   |
| <b>NULL</b>                | No Value   |

**11. Name some built-in functions in PHP. (NOV / DEC 2015)(R)**

- ✓ Array Functions
- ✓ Class/Object Functions
- ✓ Character Functions
- ✓ Date & Time Functions
- ✓ Error Handling Functions
- ✓ MySQL Functions
- ✓ ODBC Functions
- ✓ String Functions
- ✓ XML Parsing Functions

**12. How will create user-defined function in PHP? (AN)**

A function is a block of statements that can be used repeatedly in a program.

A function will not execute immediately when a page loads.

A function will be executed by a call to the function.

**Creating User Defined Function in PHP**

A user defined function declaration starts with the word "function":

**Syntax**

```
function functionName()
{
    code to be executed;
}
```

**13. What is form validation PHP? (U)****(APRIL/MAY 2021, 2022)**

Form validation is a “technical process where a web-form checks if the information provided by a user is correct.” The form will either alert the user that they messed up and need to fix something to proceed, or the form will be validated and the user will be able to continue with their registration process.

There are two types of validation are available in PHP. They are as follows –

- Client-Side Validation – Validation is performed on the client machine web browsers.
- Server Side Validation – After submitted by data, The data has sent to a server and perform validation checks in server machine.

Some of Validation rules for field:

| <b>Field</b>   | <b>Validation Rules</b>                 |
|----------------|---|
| Name           | Should require letters and white-spaces |
| Email          | Should require @ and .                  |
| Website        | Should require a valid URL              |
| Radio          | Must be selectable at least once        |
| Check Box      | Must be checkable at least once         |
| Drop Down menu | Must be selectable at least once        |

**14. Write a PHP program to add two numbers.(C)**

```
<html>
<head>
<title>Writing PHP Function to add Two Numbers</title>
</head>
<body>

<?php
    function addFunction($num1, $num2)
    {
        $sum = $num1 + $num2;
        echo "Sum of the two numbers is : $sum";
    }
    addFunction(10, 20);
?>
</body>
</html>
```

**Output:**

**Sum of the two numbers is : 30**

**15. Write the steps to connect a PHP program with a database.(R)**

- i. Open a connection to MySQL itself - **mysql\_connect()**
- ii. Specify the database we want to open - **mysql\_select\_db()**
- iii. Close the connection - **mysql\_close()**

**16. How to set cookies in PHP? (AN)**

PHP provided **setcookie()** function to set a cookie. This function requires upto six arguments and should be called before <html> tag. For each cookie this function has to be called separately.

**Syntax:** setcookie(name, value, expire, path, domain, security);

**Example:**

```
<?php
    setcookie("name", "John Watkin", time()+3600, "/", "", 0);
    setcookie("age", "36", time()+3600, "/", "", 0);
?>
```

**17. What is meant by Regular Expression? (U)**

Regular expressions are nothing more than a sequence or pattern of characters itself. They provide the foundation for pattern-matching functionality.

Using regular expression we can search a particular string inside another string, we can replace one string by another string and you can split a string into many chunks.

**18. What are the two sets of regular expressions offered by PHP?(U)**

PHP offers functions specific to two sets of regular expression functions:

- POSIX extended Regular Expressions
- PERL compatible Regular Expressions

**19. List some POSIX regular expression functions.(R)**

| Function              | Description  |
|-----------------------|--|
| <u>ereg()</u>         | The <u>ereg()</u> function searches a string specified by string for a string specified by pattern, returning true if the pattern is found, and false otherwise. |
| <u>ereg_replace()</u> | The <u>ereg_replace()</u> function searches for string specified by pattern and replaces pattern with replacement if found.                                      |

|                        |  |
|------------------------|--|
| <u>eregi()</u>         | The eregi() function searches throughout a string specified by pattern for a string specified by string. The search is not case sensitive.   |
| <u>eregi_replace()</u> | The eregi_replace() function operates exactly like ereg_replace(), except that the search for pattern in string is not case sensitive.   |
| <u>split()</u>         | The split() function will divide a string into various elements, the boundaries of each element based on the occurrence of pattern in string.                                      |
| <u>spliti()</u>        | The spliti() function operates exactly in the same manner as its sibling split(), except that it is not case sensitive.  |
| <u>sql_regcase()</u>   | The sql_regcase() function can be thought of as a utility function, converting each character in the input parameter string into a bracketed expression containing two characters. |

## 20. List some Perl compatible regular expression functions. (R)

| Function                | Description   |
|-------------------------|---|
| <u>preg_match()</u>     | The preg_match() function searches string for pattern, returning true if pattern exists, and false otherwise.   |
| <u>preg_match_all()</u> | The preg_match_all() function matches all occurrences of pattern in string.   |
| <u>preg_replace()</u>   | The preg_replace() function operates just like ereg_replace(), except that regular expressions can be used in the pattern and replacement input parameters. |
| <u>preg_split()</u>     | The preg_split() function operates exactly like split(), except that regular expressions are accepted as input parameters for pattern.                      |
| <u>preg_grep()</u>      | The preg_grep() function searches all elements of input_array, returning all elements matching the regexp pattern.  |
| <u>preg_quote()</u>     | Quote regular expression characters   |

## 21. What is XML? (U)

XML stands for **Extensible Markup Language**. It is a text-based markup language for describing the format for data exchanged between applications over the internet. XML permits document authors to create new markup languages for describing any type of data.

## 22. List the important characteristics of XML.(R)

- ✓ **XML is extensible:** XML allows you to create your own self-descriptive tags, or language, that suits your application.

- ✓ **XML carries the data, does not present it:** XML allows you to store the data irrespective of how it will be presented.
- ✓ **XML is a public standard:** XML was developed by an organization called the World Wide Web Consortium (W3C) and is available as an open standard.

**23. Mention the uses (advantages) of XML.(AN)**

- ✓ XML can work behind the scene to simplify the creation of HTML documents for large web sites.
- ✓ XML can be used to exchange the information between organizations and systems.
- ✓ XML can be used for offloading and reloading of databases.
- ✓ XML can be used to store and arrange the data, which can customize your data handling needs.
- ✓ XML can easily be merged with style sheets to create almost any desired output.
- ✓ Virtually, any type of data can be expressed as an XML document.

**24. What are the XML Syntax Rules?(U)**

1. The XML declaration is case sensitive and must begin with "<?xml>" where "xml" is written in lower-case.
2. If document contains XML declaration, then it strictly needs to be the first statement of the XML document.
3. Attribute names in XML (unlike HTML) are case sensitive. That is, *HREF* and *href* are considered two different XML attributes.
4. Same attribute cannot have two values in a syntax. The following example shows incorrect syntax because the attribute *b* is specified twice:
5. `<a b="x" c="y" b="z">...</a>`
6. Attribute names are defined without quotation marks, whereas attribute values must always appear in quotation marks. Following example demonstrates incorrect xml syntax:
7. `<a b=x>...</a>`
8. In the above syntax, the attribute value is not defined in quotation marks.
9. The names of XML-elements and XML-attributes are case-sensitive, which means the name of start and end elements need to be written in the same case.
10. To avoid character encoding problems, all XML files should be saved as Unicode UTF-8 or UTF-16 files.
11. Whitespace characters like blanks, tabs and line-breaks between XML-elements and between the XML-attributes will be ignored.
12. Some characters are reserved by the XML syntax itself. Hence, they cannot be used directly.

**25. What do you mean by XML declaration? Give an example. (MAY/JUNE 2013)(U)**

The XML declaration is a *processing instruction* that identifies the document as being XML. All XML documents should begin with an XML declaration.

For example,

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
```

**26. What is meant by a XML namespace?(April / May 2011, 2014, Nov / Dec 2012, 2013,NOV/DEC 2016)(U)**

An XML namespace is a collection of element and attribute names. XML namespace provide a means for document authors to unambiguously refer to elements with the same name (i.e., prevent collisions).

**Example:**

```
<subject>Geometry</subject> // data for student from school
```

and

```
<subject>Cardiology</subject> // data for student from medicine
```

Both the markup uses the element “subject” to markup data. Namespace can differentiate these two “subject” elements:

```
<highschool:subject>Geometry</highschool:subject>
```

and

```
<medicalschoo:subject>Cardiology</medicalschoo:subject>
```

Both **highschool** and **medicalschoo** are namespace prefixes.

**27. How the namespace is declared?(R)**

A Namespace is declared using reserved attributes. Such an attribute name must either be **xmlns** or begin with **xmlns:** shown as below:

```
<element xmlns:name="URL">
```

**Syntax:**

- The Namespace starts with the keyword **xmlns**.
- The word **name** is the Namespace prefix.
- The **URL** is the Namespace identifier.

**Example:**

```
<root>
```

```
<highschool:subject xmlns:highschool="http://www.abcschool.edu/subjects">
```

```
  Geometry
```

```
</highschool:subject>
```

```
<medicalschoo:subject xmlns:medicalschoo="http://www.rmc.org/subjects">
```

```
  Cardiology
```

```
</medicalschoo:subject>
```

```
</root>
```

**28. What is meant by Document Type Definition?(U)**

Rules that define the legal elements and attributes for XML documents are called Document Type Definitions (DTD) or XML Schemas.

There are two different document type definitions that can be used with XML:

- DTD - The original Document Type Definition
- XML Schema - An XML-based alternative to DTD

**29. What is the purpose of DTD?(U)**

The purpose of a DTD is to define the structure of an XML document. It defines the structure with a list of legal elements:

**Example DTD :**

```
<!DOCTYPE note
[
<!ELEMENT note (to,from,heading,body)>
<!ELEMENT to (#PCDATA)>
<!ELEMENT from (#PCDATA)>
<!ELEMENT heading (#PCDATA)>
<!ELEMENT body (#PCDATA)>
]>

<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget to complete the work !</body>
</note>
```

The DTD above is interpreted like this:

- !DOCTYPE note defines that the root element of the document is note
- !ELEMENT note defines that the note element must contain four elements: "to, from, heading, body"
- !ELEMENT to defines the to element to be of type "#PCDATA"
- !ELEMENT from defines the from element to be of type "#PCDATA"
- !ELEMENT heading defines the heading element to be of type "#PCDATA"
- !ELEMENT body defines the body element to be of type "#PCDATA"

**30. What is meant by Internal DTD?(U)**

A DTD is referred to as an internal DTD if elements are declared within the XML files. To refer it as internal DTD, *standalone* attribute in XML declaration must be set to **yes**. This means, the declaration works independent of external source.

**Syntax**

The syntax of internal DTD is as shown:

```
<!DOCTYPE root-element [element-declarations]>
```

where *root-element* is the name of root element and *element-declarations* is where you declare the elements.

**Example**

Following is a simple example of internal DTD:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<!DOCTYPE address [
  <!ELEMENT address (name,company,phone)>
  <!ELEMENT name (#PCDATA)>
  <!ELEMENT company (#PCDATA)>
  <!ELEMENT phone (#PCDATA)>
]>
<address>
  <name>Tanmay Patil</name>
  <company>TutorialsPoint</company>
  <phone>(011) 123-4567</phone>
</address>
```

**31. Give an example for External DTD.(U)**

In external DTD elements are declared outside the XML file. They are accessed by specifying the system attributes. To refer it as external DTD, *standalone* attribute in the XML declaration must be set as **no**. This means, declaration includes information from the external source.

**Syntax:**

Following is the syntax for external DTD:

```
<!DOCTYPE root-element SYSTEM "file-name">
```

where *file-name* is the file with *.dtd* extension.

**Example:**

The following example shows external DTD usage:

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<!DOCTYPE address SYSTEM "address.dtd">
<address>
  <name>Tanmay Patil</name>
  <company>TutorialsPoint</company>
  <phone>(011) 123-4567</phone>
</address>
```

The content of the DTD file **address.dtd** are as shown:

```
<!ELEMENT address (name,company,phone)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT company (#PCDATA)>
<!ELEMENT phone (#PCDATA)>
```

### 32. What is XML Schema?(U) (April/May 2021)

XML Schema is commonly known as XML Schema Definition (XSD). It is used to describe and validate the structure and the content of XML data. XML schema defines the elements, attributes and data types. Schema element supports Namespaces. It is similar to a database schema that describes the data in a database.

### 33. Give the syntax to declare XML schema.(R)

Syntax:

The common syntax to declare a schema in our XML document as follows:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

**Example:** The following example shows how to use schema:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="contact">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="name" type="xs:string" />
      <xs:element name="company" type="xs:string" />
      <xs:element name="phone" type="xs:int" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>
```

The basic idea behind XML Schemas is that they describe the legitimate format that an XML document can take.

### 34. What are the data types supported by XML Schema?(R)

| Type           | Description             | Example             |
|----------------|-------------------------|---------------------|
| <b>String</b>  | A character string      | “hello”             |
| <b>Boolean</b> | True or False           | true                |
| <b>Decimal</b> | A decimal numeral       | 5, -12, -45.78      |
| <b>Float</b>   | A floating-point number | 0, 12, -109.75, NaN |

|               |   |                         |
|---------------|---|-------------------------|
| <b>Double</b> | A floating-point number                         | 0, 12, -109.75, NaN     |
| <b>Long</b>   | A whole number                                  | 1234567890, -1234567890 |
| <b>Int</b>    | A whole number                                  | 12345, -12345           |
| <b>Short</b>  | A whole number                                  | 12, -345                |
| <b>Date</b>   | A date consisting of year, month and day        | 2005-05-10              |
| <b>Time</b>   | A time consisting of hours, minutes and seconds | 16:30:25-05:00          |

### 35. What is meant by XML DOM?(U)

The W3C Document Object Model (DOM) is a platform and language-neutral Application Programming Interface (API) that allows programs and scripts to dynamically access and update the content, structure, and style of a HTML, XHTML & XML document.

### 36. Define DOM tree.(R)

In XML document the information is maintained in hierarchical structure, this hierarchical structure is referred as *Node Tree* or *DOM Tree*.

The structure of the node tree begins with the root element and spreads out to the child elements till the lowest level.

The nodes in the node tree have a hierarchical relationship to each other. The terms parent, child, and sibling are used to describe the relationships. Parent nodes have children. Children on the same level are called siblings (brothers or sisters).

- In a node tree, the top node is called the root
- Every node, except the root, has exactly one parent node
- A node can have any number of children
- A leaf is a node with no children
- Siblings are nodes with the same parent

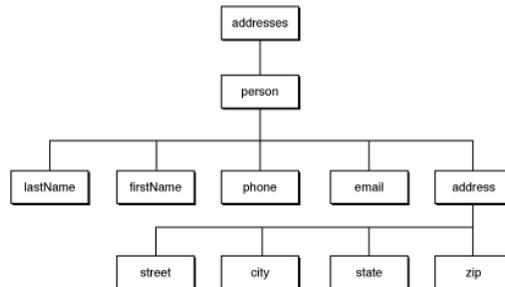
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE addresses SYSTEM "addresses.dtd">
<addresses>
  <person idnum="0123">
    <lastName>Doe</lastName>
    <firstName>John</firstName>
    <phone location="mobile">(201) 345-6789</phone>
    <email>jdoe@foo.com</email>
    <address>
      <street>100 Main Street</street>
      <city>Somewhere</city>
      <state>New Jersey</state>
      <zip>07670</zip>
    </address>
  </person>
</addresses>
```

```

</address>
</person>
</addresses>

```

The following tree of element nodes represents this document:



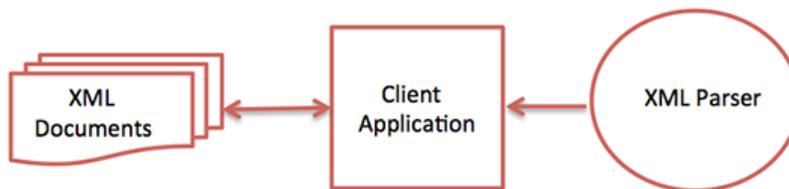
### 37. What is meant by DOM Parser?(U)

An XML parser that creates the DOM tree structure is known as **DOM parser**.

### 38. What is an XML Parser? (U)

XML parser is a software library or a package that provides interface for client applications to work with XML documents. It checks for proper format of the XML document and may also validate the XML documents. Modern day browsers have built-in XML parsers.

Following diagram shows how XML parser interacts with XML document:



The goal of a parser is to transform XML into a readable code.

### 39. Explain the use of XML Parser. (U)

(APRIL/MAY 2022)

- ✓ It checks for proper format of the XML document (Well Formed Document) and may also validate the XML documents (Valid XML Document).
- ✓ XML parser validates the document and check that the document is well formatted.

### 40. What is meant by XML validation? (U)

**XML validation** is the process of checking a document written in XML (eXtensible Markup Language) against its specified DTD to confirm that it is both well-formed and also

"valid". When a document fails to conform to a DTD or a schema, the validator displays an error message.

#### 41. What is meant by well-formed XML document?(U)

An XML document with correct syntax is "Well Formed". A well-formed XML document must follow the following XML syntax rules:

- XML documents must have a root element
- XML elements must have a closing tag
- XML tags are case sensitive
- XML elements must be properly nested
- XML attribute values must be quote.

Example well-formed XML document:

```
<?xml version="1.0" encoding="UTF-8"?>
<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget to complete the work!</body>
</note>
```

#### 42. What is meant by "Valid XML document"?(U)

An XML document that contains the proper elements with the proper attributes in the proper sequence according to the DTD specification is called a "Valid" XML document. An XML document cannot be valid unless it is well-formed.

#### 43. What are the two types of XML parsers? And differentiate them.(U/AN)

Two types of XML Parsers:

1. Non-Validating XML Parsers
2. Validating XML Parsers

| Non-Validating XML Parsers   | Validating XML Parsers   |
|--|--|
| A non-validating parser checks if a document follows the XML syntax rules. It builds a tree structure from the tags used in XML document and return an error only when there is a problem with the syntax of the document. | A Validating parser checks the syntax, builds the tree structure, and compares the structure of the XML document with the structure specified in the DTD associated with the document. |
| Non validating parsers process a document faster because they do not have to check every element against a DTD. In other words, these parsers check whether an XML document  | Validating parsers are slower because, in addition to checking whether an XML document is well formed; validating parsers also check whether the XML document                          |

|  |  |
|--|--|
| adheres to the rules of well-formed document.            | adheres to the rules in the DTD used by the XML document.    |
| The Expat parser is an example of non-validating parser. | Microsoft MSXML parser is an example of a validating parser. |

**44. Define XSL.(R)**

The **Extensible Style sheet Language (XSL)** is an XML vocabulary that decides how an XML document data should look on the web browser. XSL is group of three technoliges:

1. **XSL Transformations (XSLT):** is technology for transforming the structure of the XML document data into another structure.
2. **XML Path Language (XPath):** is a string-based language of expressions which defines the syntax and semantics for efficiently locating elements and attributes in XML documents.
3. **XSL Formatting Objects (XSL-FO):** It is a separate XML vocabulary for defining document style properties of an XML document .(print-oriented)

**45. Explain two types of XSL information.(U)**

An XSL document normally contains **two types of information:**

- ✓ **Template data** - Which is text that is copied to the output XML text document with little or no change? i.e. Everything in the body of the document that is not XSL markup is *template data* And
- ✓ **XSL markup** - which controls the transformation process always **start with “xsl:”**.

**46. Name any four XSL elements .Mentions its use.(R)**

- a. The **<xsl:variable>** XSL element creates a named variable.
- b. The **<xsl:value-of>** XSL element processes a value.
- c. The **<xsl:if>** XSL element invokes the enclosed code if the condition specified by the test attribute is true.
- d. The **<xsl:sort>** XSL element sorts a list of elements, such as within **<xsl:for-each>**.
- e. The **<xsl:template>** XSL element encapsulates a segment of XSL code, similar to a method, procedure, or function in other languages.

**47. What is XSLT?(U)**

XSLT stands for XSL Transformations. XSLT is a language for transforming XML documents into XHTML documents or to other XML documents. XSLT uses XPath to navigate in XML documents. With XSLT user can add/remove elements and attributes to or from the output file.

**48. What is XPath?(U)**

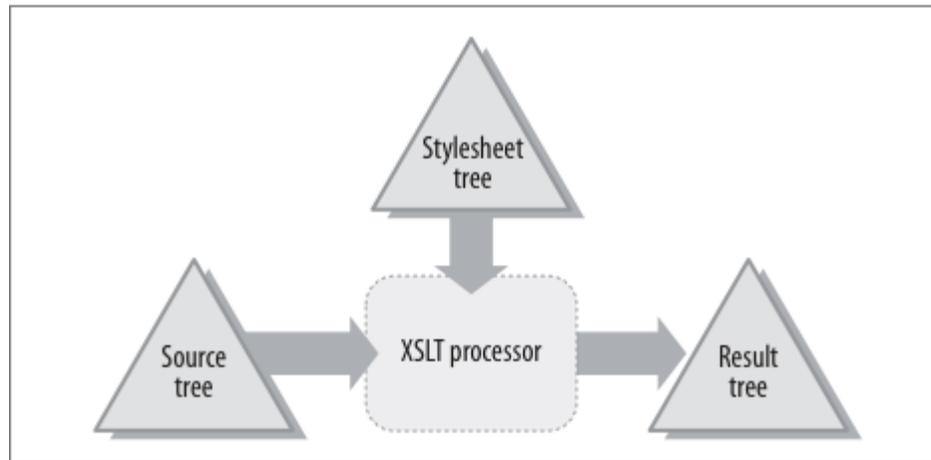
XPath is a language used to navigate through elements and attributes in an XML document.

- XPath is a syntax for defining parts of an XML document
- XPath uses path expressions to navigate in XML documents
- XPath contains a library of standard functions

- XPath is a major element in XSLT
- XPath is a W3C recommendation

**49. How does XSLT works?(AN)**

In the transformation process, XSLT uses XPath to define parts of the source document that should match one or more predefined templates. When a match is found, XSLT will transform the matching part of the source document into the result document.

**50. What is meant by RSS?(U)**

RSS (**Rich Site Summary** or **Really Simple Syndication**) is an XML format for delivering regularly changing web contents (i.e., Used to syndicate website contents). It is a mean of transmitting and updating news in an automated way.

**51. What is the use of RSS?(U)**

RSS was designed to show selected data.

Without RSS, users will have to check your site daily for new updates. This may be too time-consuming for many users.

With an RSS feed (RSS is often called a News feed or RSS feed) they can check your site faster using an RSS aggregator (a site or program that gathers and sorts out RSS feeds).

Since RSS data is small and fast-loading, it can easily be used with services like cell phones or PDA's.

**52. Why is XSLT an important tool in development of web applications? (AN)**

(MAY/JUNE 2016)

XSLT is the most important part of XSL.

XSLT is used to transform an XML document into another XML document, or another type of document that is recognized by a browser, like HTML and XHTML. Normally XSLT does this by transforming each XML element into an (X)HTML element.

With XSLT you can add/remove elements and attributes to or from the output file. You can also rearrange and sort elements, perform tests and make decisions about which elements to hide and display, and a lot more.

A common way to describe the transformation process is to say that **XSLT transforms an XML source-tree into an XML result-tree.**

**53. When should the super global arrays in PHP be used? Which super global array in PHP would contain a HTML form's POST data? (MAY/JUNE 2016)(AN)**

PHP super global variable is used to access global variables from anywhere in the PHP script.

**54. List the important characteristics of PHP. (NOV/DEC 2018) (R)**

- Familiarity
- Simplicity
- Efficiency
- Security
- Flexibility
- Open source
- Object Oriented

**55. Explain DTD for XML Schemas. (U) (NOV/DEC 2018)**

A DTD is a Document Type Definition. A DTD defines the structure and the legal elements and attributes of an XML document. An XML Schema describes the structure of an XML document. XML document has a reference to a DTD.

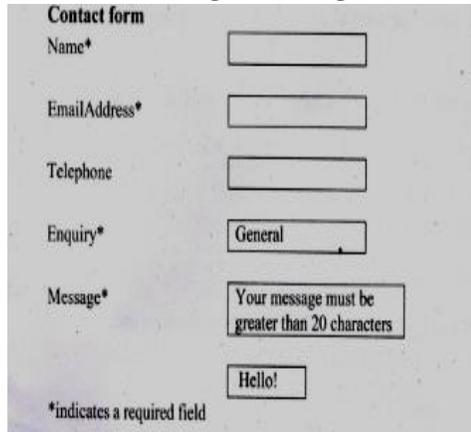
**56. What is meant by form validation? (U) (APRIL/MAY 2021)**

**PART – B**

1. Discuss in detail about how to create and use variables in PHP with example program. (C)
2. What are the different data types available in PHP? Explain about converting between different data types with example program. (U)
3. Write a PHP program using arithmetic operator. (C)
4. Explain the features of built-in functions in PHP with examples. (U) (APRIL/MAY 2022)
5. Write a PHP program to do string manipulation. (NOV / DEC 2015) (C)
6. Explain how user-defined functions are created in PHP. (U)
7. Explain how input from an XHTML form is received in a PHP program. (U)
8. How will you connect a PHP program with database? Explain with example application. (U)
9. Explain how cookies are handled in PHP. (U)
10. Explain in detail about using Regular Expressions in PHP. (U)
11. Explain XML DTD. (U) (APRIL/MAY 2022)

12. What are different types of DTD? Explain the same with example. (U)
13. Explain the role of XML name spaces with examples. (U) (MAY/JUNE 2012)
14. What is XML Schema? Explain how to create and use XML Schema document. (C) (NOV/DEC 2015)
15. Explain how a XML document can be displayed on a browser. (U) (APRIL/MAY 2011)
16. Explain about DOM. (U) (MAY/JUNE 2014)
17. Explain Document Tree with an example. (U) (MAY/JUNE 2011, MAY/JUNE 2012)
18. Explain in detail about XSL. (U) (NOV/ EC 2013)
19. Give an XSLT document and a source XML document explain the XSLT transformation process that produces a single result XML document. (U) (NOV/DEC 2012)
20. Explain in detail about XML parsers and validation. (U)(NOV/DEC 2015, MAY/JUNE 2016)
21. Elaborate on RSS. (U)
22. Create a webserver based chat application using PHP. The application should provide the function functions. (C)
  - Login
  - Send message (to one or more contacts)
  - Receive messages(from one or more contacts)
  - Add/delete /modify contact list of the user
23. Discuss the application's user interface and use comments in PHP to explain the code clearly. (U) (MAY/JUNE 2016)
24. List at least five significant differences between DID and XML schema for defining XML document structures with appropriate examples. (AN) (MAY/JUNE 2016)
25. Explain the string comparison capability of PHP using regular expressions with an example. (U) (NOV/DEC2016)
26. Explain the steps in connecting a PHP code to a database. (U) (NOV/DEC2016)
27. Explain in detail the XML schema, built in and user defined data type in detail. (U) (NOV/DEC2016)
28. Design simple calculator using PHP. (C) (NOV/DEC2018)
29. Explain about the controls statements in PHP with example. (U) (NOV/DEC 2018)
30. Explain about DOM with the XML data processing. (U) (NOV/DEC2018)
31. Design a PHP application for College Management System with appropriate built-in functions and database. (C) (NOV/DEC 2018)
32. What is spring framework ? What is use of it ? What are the advantages of Spring Framework? (U) (APRIL/MAY 2021)
33. Using XSLT, how would you extract a specific attribute from an element in an Xml document? When constructing an XML DTD, how do you create an external entity reference in an attribute value ? (A) (APRIL/MAY 2021)
34. Write the HTML code for creating a feedback form as shown below. Include comments in code to highlight the markup elements and their purpose. The HTML form should use POST for submitting the form to a program ProcessContactForm.PHP. (C) (MAY/JUNE 2016)

**Internet Programming**



**UNIT – V**

**UNIT-V INTRODUCTION TO AJAX AND WEB SERVICES**

AJAX: Ajax Client Server Architecture – XML HttpRequest Object – Call Back Methods; Web Services: Introduction – Java web services basics – Creating, Publishing, Testing and Describing a web services (WSDL) – Consuming a web service – Database driven web services from an application – SOAP.

**PART - A**

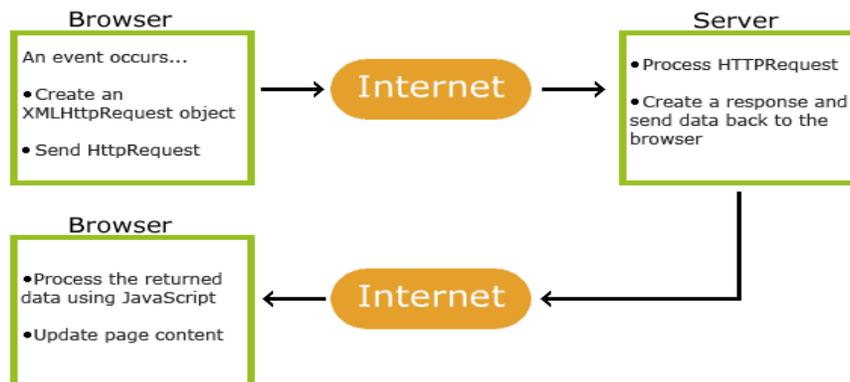
**1. What is AJAX?(U)**

AJAX = Asynchronous JavaScript and XML.

AJAX is a technique for creating fast and dynamic web pages.

AJAX allows web pages to be updated asynchronously by exchanging small amounts of data with the server behind the scenes. This means that it is possible to update parts of a web page, without reloading the whole page.

**2. Draw the Ajax Client-Server architecture? (R)**



**3. Give some example applications that use AJAX. (R)**

Examples of applications using AJAX:

Google Maps  
Gmail  
YouTube  
Facebook tabs.

**4. Define HTTP Request Object. (R) (APRIL/MAY 2022)**

The XMLHttpRequest Object is the layer between the client and the server that creates and manages asynchronous requests in Ajax applications. All modern browsers support the XMLHttpRequest object (IE5 and IE6 use an ActiveXObject).

**5. What is the use of XMLHttpRequest Object?(U)**

The **XMLHttpRequest** object is used to exchange data with a server behind the scenes. This means that it is possible to update parts of a web page, without reloading the whole page. All modern browsers support the **XMLHttpRequest** object (IE5 and IE6 use an ActiveXObject).

**6. Give the syntax to create XMLHttpRequest object.(R)**

To handle all modern browsers, including IE5 and IE6, check if the browser supports the XMLHttpRequest object. If it does, create an XMLHttpRequest object, if not, create an ActiveXObject:

Example:

```
var xmlhttp;
if (window.XMLHttpRequest)
    { // code for IE7+, Firefox, Chrome, Opera, Safari
    xmlhttp=new XMLHttpRequest();
    }
else
    { // code for IE6, IE5
    xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
    }
```

**7. Write the syntax to send a request to the server using XMLHttpRequest object.(R)**

To send a request to a server, we use the open() and send() methods of the XMLHttpRequest object:

```
xmlhttp.open("GET","ajax_info.txt",true);
xmlhttp.send();
```

| Method                              | Description   |
|-------------------------------------|---|
| <code>open(method,url,async)</code> | Specifies the type of request, the URL, and if the request should be handled asynchronously or not.<br><i>method</i> : the type of request: GET or POST<br><i>url</i> : the location of the file on the server<br><i>async</i> : true (asynchronous) or false (synchronous) |
| <code>send(string)</code>           | Sends the request off to the server.<br><i>string</i> : Only used for POST requests   |

### 8. How to get the response from the server?(R)

To get the response from a server, use the **responseText** or **responseXML** property of the XMLHttpRequest object.

| Property     | Description                       |
|--------------|-----------------------------------|
| responseText | get the response data as a string |
| responseXML  | get the response data as XML data |

### 9. Mention the three important properties of XMLHttpRequest object.(R)

Three important properties of the XMLHttpRequest object:

| Property           | Description   |
|--------------------|---|
| onreadystatechange | Stores a function (or the name of a function) to be called automatically each time the readyState property changes  |
| readyState         | Holds the status of the XMLHttpRequest. Changes from 0 to 4:<br>0: request not initialized<br>1: server connection established<br>2: request received<br>3: processing request<br>4: request finished and response is ready |
| Status             | 200: "OK"<br>404: Page not found  |

### 10. What is meant by callback function?(U)

A callback function is a function passed as a parameter to another function.

If you have more than one AJAX task on your website, you should create ONE standard function for creating the XMLHttpRequest object, and call this for each AJAX task.

The function call should contain the URL and what to do on onreadystatechange (which is probably different for each call):

**Example**

```
function myFunction()
{
loadXMLDoc("ajax_info.txt",function()
{
if (xmlhttp.readyState==4 && xmlhttp.status==200)
{
document.getElementById("myDiv").innerHTML=xmlhttp.responseText;
}
});
}
```

**11. List any four merits for usage of AJAX in web service. (NOV / DEC 2015,NOV/DEC 2016)(AN)**

1. Reduce the traffic travels between the client and the server.
2. Response time is faster so increases performance and speed.
3. AJAX communicates over HTTP protocol.
4. Asynchronous calls to a web server allows client need not to wait for the arrival of all data before start rendering.
5. If a section of a page encounters any error, other sections do not get affected and the data entered by the user is also not lost.

**12. What is meant by Web Services? Give any four examples. (U)**

(NOV/DEC 2015, APRIL/MAY 2022)

A web service is a software component stored on one computer that can be accessed via method calls by an application ( or other software component) on another computer over a network. These software components are used by some software applications rather than by end-users directly.

Example:

- Weather forecast system.
- Currency converters.
- Credit card validation system.
- Mashups

**13. What are the Components of Web Services?(U)**

The basic Web services platform is XML + HTTP. All the standard Web Services works using following components

- SOAP (Simple Object Access Protocol)
- UDDI (Universal Description, Discovery and Integration)
- WSDL (Web Services Description Language)

**14. List some examples of web services. (May / June 2012)(R)**

- (1) Weather forecast system
- (2) Currency converter
- (3) Electronic payment processing system
- (4) Credit card validation system
- (5) Resource management system

**15. What is meant by WSDL? (May / June 2011, NOV/DEC 2016)(U)**

WSDL (Web Services Description Language) is an XML-based language for describing Web services and how to access them. Web Services Description Language is the standard format for describing a web service in XML format.

**16. Define WSDL Specification. List out its six major elements .(R)**

WSDL is an XML grammar for describing web services. The specification itself is divided into **six major elements**:

- ✓ **definitions** - The definitions element must be the root element of all WSDL documents.
- ✓ **types** -The types element describes all the data types used between the client and server.
- ✓ **message** -The message element describes a one-way message, whether it is a single message request or a single message response.
- ✓ **portType** - It defines a **web service**, the **operations** that can be performed, and the **messages** that are involved.
- ✓ **binding** - The binding element describes the concrete specifics of how the service will be implemented on the wire.
- ✓ **Service** - The service element defines the address for invoking the specified service. Most commonly, this includes a URL for invoking the SOAP service.

**17. What is UDDI? (Nov / Dec 2011)(U)**

UDDI stands for Universal Description, Discovery and Integration. It is an XML based specification for registry of web services. It is platform-independent frame work.

**18. What is meant by SOAP? Mention its use. (MAY/JUNE 2013)(U)**

SOAP is a simple XML-based protocol to let applications exchange information over HTTP, because HTTP is supported by all Internet browsers and servers.

SOAP provides a way to communicate between applications running on different operating systems, with different technologies and programming languages.

**Use :**

- Communicating Object Data
- exchange information over HTTP

**19. What are the SOAP Elements? Mention its use .(R)**

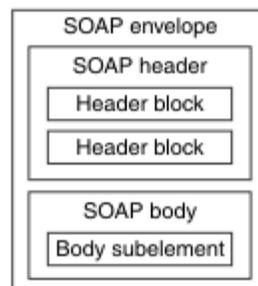
- a. An **Envelope element** that identifies the XML document as a SOAP message
- b. A **Header element** that contains header information

- c. A **Body element** that contains call and response information
- d. A **Fault element** containing errors and status information

**20. Mention syntax rules for SOAP.**

- a. A SOAP message **MUST** be encoded using XML
- b. A SOAP message **MUST** use the SOAP Envelope namespace
- c. A SOAP message **MUST** use the SOAP Encoding namespace
- d. A SOAP message must **NOT** contain a DTD reference
- e. A SOAP message must **NOT** contain XML Processing Instructions

**21. Draw the structure of a SOAP message. (C)**



**22. What is the role of a callback function in performing a partial page update in an AJAX application ? (U)**

The callback function uses partial page updates to display the data in the existing web page without reloading the entire page. At the same time, the server may be responding to the second request and the client-side may be starting to do another partial page update. The callback function updates only a designated part of the page. Such partial page updates help make web applications more responsive, making them feel more like desktop applications. The web application does not load a new page while the user interacts with it.

**23. Describe AJAX control Extender Toolkit.(U) (NOV/DEC 2018)**

AJAX Control Toolkit is a set of extenders that are used to extend the functionalities of the ASP.NET controls. The extenders use a block of JavaScript code to add new and enhanced capabilities to the ASP.NET controls. AJAX Control Toolkit is a free download available on the Microsoft site. You need to install this toolkit on your system before using extenders.

**24. State the significance of a WSDL document.(R) (NOV/DEC 2018)**

- WSDL is an XML-based protocol for information exchange in decentralized and distributed environments.
- WSDL definitions describe how to access a web service and what operations it will perform.
- WSDL is a language for describing how to interface with XML-based services.

- WSDL is an integral part of Universal Description, Discovery, and Integration (UDDI), an XML-based worldwide business registry.
- WSDL is the language that UDDI uses.
- WSDL is pronounced as 'wiz-dull' and spelled out as 'W-S-D-L'.

**25. Define WSDL documents. (April/May 2021)**

A WSDL document uses the following elements in the definition of network services:

- Types– a container for data type definitions using some type system (such as XSD).
- Message– an abstract, typed definition of the data being communicated.
- Operation– an abstract description of an action supported by the service.
- Port Type–an abstract set of operations supported by one or more endpoints.
- Binding– a concrete protocol and data format specification for a particular port type.
- Port– a single endpoint defined as a combination of a binding and a network address.
- Service– a collection of related endpoints.

**PART – B**

1. What do you mean by AJAX? Write the advantages of AJAX. (U) (MAY/JUNE 2014)
2. Explain the AJAX client-server architecture with neat diagram. (U)  
(NOV/DEC 2015, NOV/DEC 2016)
3. Explain about the object that helps AJAX reload parts of a web page without reloading the whole page. (U)  
(NOV/DEC 2011, NOV/DEC 2018)  
Illustrate the process of handling concurrent AJAX request. (U) (APRIL/MAY 2022)
4. Differentiate between synchronous and asynchronous Ajax requests. How to send an Ajax request in JavaScript ? (April/May 2021)
5. Explain in detail about the XML HttpRequest object with example program. (U)  
(NOV/DEC 2016)
6. Explain the steps for writing a Java Web Service server and a Java Web Service Client with an example. (U)  
(MAY/JUNE 2012, NOV/ EC 2013)
7. Explain in detail about how to consume a web service with example. (U)
8. Develop a Java Web Service that would do arithmetic operations. (C) (MAY/JUNE 2014)
9. Describe the significance and working of WSDL with an example. (U) (NOV/DEC 2011)
10. Explain how an application can be connected with databases through web services with neat example. (U)
11. Explain the SOAP elements in detail. (U)(NOV/DEC 2011, 2012, 2013, MAY/JUNE 2014)
12. Write short notes on: (i) SOAP (ii) WSDL. (U) (NOV / DEC 2015)
13. Compare and contrast the traditional web application architecture and AJAX based web application architecture. (AN) (MAY/JUNE 2016)
14. Describe the structure of a WSDL document , its elements and their purposes with appropriate examples. (U) (MAY/JUNE 2016)

15. Explain in detail the steps in the creation, publishing and testing of a web services in detail. (U) (NOV/DEC 2016)
16. Define SOAP and explain the building blocks of a SOAP message in detail. (U) (NOV/DEC 2016)
17. Describe SOAP, UDDI and WSDL in detail. (U) (NOV/DEC 2018)
18. What are the Web service features and explain the components of a web service? How does a web service work? (U) (APRIL/MAY 2021)
19. What are the two methods thread can be created in Java? Explain them each with an example code. Mention the different thread methods. (U) (APRIL/MAY 2021)
20. Explain the following with proper example : (U) (APRIL/MAY 2021)
- i) Exception handling
  - ii) Event handling
  - iii) Call back methods
  - iv) XML validation