Q.1 Define WWW.

⇒ World wide Web

⇒ The world wide web, or simply web, is a way of accessing information over the medium of the internet. It is an information-sharing model that is built on the top of the internet.

⇒ The web uses the HTTP Protocol, only one of the language spoken over the internet, to transmit data.

⇒ The web also utilizes browsers, such as internet Explorer or Firefox, to access web documents called web pages that are linked to each other via hyperlinks. Web documents also contain graphics, sounds, text and video.

Q.3. Differentiate GET and POST methods.

HTTP GET	HTTP POST
• In GET method we can send large amount of data rather limited data is sent because the request parameter is appended into the URL.	• In Post method large amount of data can be sent because the request parameter is appended into the body.
• GET request is comparatively better than Post so it is used more than the Post request.	• Post request is comparatively less better than Get so it is used less than the Get request.
• GET request is Comparatively less secure because the data is exposed in the URL bar.	• POST request is more secure because the data is not exposed in the URL bar

• Request made through GET method are stored in Browser history.	• Request made through Post method is not stored in Browser history.
• GET method request can be saved as bookmark in browser.	• Post method request can not be saved as bookmark in browser.
• ~~request made~~ • Data passed through GET method can be easily stolen by attackers.	• Data passed through POST method can no be easily stolen by attackers.

Q.4 What is HTTP ? How do browser and server communicate using HTTP request and response ? Explain with example.

⇒ HTTP stands for Hypertext Transfer Protocol.

⇒ It is the data communication protocol used to establish communication between client and server.

⇒ HTTP Request / Response :

⇒ Communication between clients and servers is done by requests and responses :

Steps:

1. A client (a browser) sends an HTTP request to the web.
2. A web server receive the request.
3. The server runs an application to process the request.
4. The server return an HTTP response (output) to the browser
5. The client (the browser) receives the response

⇒ Overview of how this communication works:

1. Client sends an HTTP request:

⇒ A user initiate an action in their web browser, such as entering a URL, clicking a link, or submitting a form.

⇒ The browser creates an HTTP request that contains information about the action, including the requested URL, HTTP method (GET, POST, PUT, DELETE etc), headers, and potentially data.

2. Request Routing:

⇒ The browser sends the HTTP request to the appropriate server based on the URL's domain name.

⇒ DNS (Domain Name System) may be reached

to resolve the domain name to an IP address is necessary.

3. Server Processes the Request:

⇒ The web server at the specified IP address receives the request.

⇒ The server processes the request based on the URL and HTTP method, possibly executing server-side code or retrieving data from a database.

4. Server Sends an HTTP response:

⇒ The server generates an HTTP response, which includes a response status code (indicating success, error or redirection), headers (metadata about the response), and optionally a response body.

⇒ The response is often dynamically generated based on the request.

5. Response Sent to Browser:

⇒ The server sends the HTTP response back to the browser using the same connection established for the request.

6. Browser Renders the Response:

⇒ The browser receives the HTTP response and process it.

⇒ If the response contains HTML, CSS and JavaScript, the browser renders the web page, style it and executes any scripts as necessary.

⇒ The user sees the resulting web page or content in the browser window.

7. Closing the connection:

⇒ After all resources are fetched and the web page is fully loaded, the connection between the browser and server may be closed, or it may be kept open for potential further requests. (HTTP keep-Alive)

~~Q.5. Explain meta Tag with~~ For

⇒ Example, the browser translated the URL http://www.test101.com/doc/index.html into the following request message:

GET/docs/index.html HTTP/1.1 Host: www.test101.com
Accept: image/gif, image/jpeg, */* Accept Language: en-US
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)

Q.5 Explain meta Tag with example.

⇒ Meta Tag (<meta>) is a HTML component that gives the metadata about a HTML document. Metadata is data (information about data.

⇒ Meta tag always go inside the <head> element, and are typically used to specify character set, page description, keywords, author of the document, and viewport settings.

⇒ Metadata will not be displayed on the page but is machine parsable.

⇒ It is an empty tag, for example, it just has a initial tag and no end tag.

⇒ Syntax:

```
<head>
    <meta attribute-name = "value" />
</head>
```

⇒ Attributes:

1. name: The attribute is used for indicating the character encoding for the HTML Document.

2. http-equiv: This attribute is used to get the HTTP response message header.

3. Content: This attribute is used to specify properties value.

4. Charset: This is used for indicating the character encoding for the HTML Document.

⇒ Example:

```
<html>
<head>
    <meta charset = "UTF-8">
    <meta name = "description" content = "Free Web tutorials">
    <meta name = "keywords" content = "HTML, CSS, XML, JavaScript">
    <meta name = "author" content = "John Doe">
    <meta name = "viewport" content = "width = device-width, initial-scale = 1.0">
</head>
<body>
<p> All meta information goes in the head section </p>
</body>
</html>
```

Q.6 Write a short note on web page

⇒ Web page is a document accessible world wide web.

⇒ Web pages are stored on web and can be viewed using a web

⇒ Web pages are primarily created HTML, which provides the framework for organizing content

1. Hypertext: Component of web page

⇒ Web pages can contain various content, including text, images, audio, and interactive elements

2. Hyper

⇒ Web

⇒ Cascading style sheets (CSS) Commonly used to control presentation and styling of elements like colors, font

Q6. Write a short note on web pages.

⇒ Web page is a document available on world wide web.

⇒ Web pages are stored on web server and can be viewd using a web browser.

⇒ Web pages are primarily constructed using HTML, which provides the structural framework for organizing content.

1. Hypertext: (component of web page content-wise):

⇒ Web pages can contain various types of content, including text, images, videos, audio, and interactive elements.

2. Hyperlinks:

⇒ Web pages often include hyperlinks that connect to other web pages, enabling users to navigate the web by clicking on these links.

```
Web tutorials">
CSS, XML, JavaScript">
Doe">
device-width, initial-scale=1.0">

section </p>
```

⇒ Cascading style sheets (CSS) are commonly used to control the presentation and styling of web page elements like colors, fonts, layout etc.

⇒ Component of webpage structure curse

1. Page Title: This is a single link text which is displayed on the title bar of the browser displaying web page.

2. Header: This is generally a one or two link text defining the purpose of the web page. It is displayed at the top of the web pag, below the address bar of the browser.

3. Body of the web page: This is the section below the header of the web page and it contains the actual content of web page.

4. Navigational links: These are the hyperlinks placed on the web page using which you can move the linked web pages.

5. Footer: This is the bottom section of the web page. This is the section where usually the copyright notice, website contact information, etc is put.

Q.2 Define term: SEO.

⇒ Search Engine Optimization

⇒ It is a process for increasing the number of viewers of web site by

applying some SEO tools to increase or bring the site on higher position in search engine results.

⇒ Websites can be improved by improving its content, make its content unique and pages to be indexed correctly, etc.

⇒ Features:

- It can improve the friendliness of your website.
- Increase your company's rank for a group of keywords over some months or years.
- A well planned and executed SEO campaign costs a lot.
- Optimize your website's textual content by incorporating a number of key words into the text.

~~⇒ SEO services:~~

Q.7 Explain the architecture of web browsers.

⇒ Web browser is an application that we use to access the world wide web.

⇒ ~~Web the~~ There Web browser has certain components which participate in architecture.

The parts playing role in the working of the web browsers are:

1. **Controller/Dispatcher:** It takes in information, interprets it, and gives instructions to the device.
2. **Interpreter:** It receives information from the controller and starts performing step by step.
3. **Client Program:** It specifies protocols such as HTTP(Hypertext Transfer Protocol) and FTP(File Transfer Protocol) to complete a service.

Components of a Web Browser

Let us look at some components of the web browser and their role in the working of the web browser.

1. **User Interface (UI):** It ensures smooth interaction between the browser and the user. It contains features like the search bar, refresh button, bookmarks, etc.
2. **Browser Engine:** It bridges the gap between UI and rendering engines. It is responsible for all actions between UI and rendering engine.
3. **Rendering Engine:** It is the core component of a browser. It transforms simple HTML pages into the webpage that we see. It is responsible for showing the requested content.
4. **Networking:** This is the component of the browser that retrieves URLs (Uniform Resource Locator) using protocols like HTTP. It is responsible for security and communication.
5. **Data Storage (Cookies):** Websites store information about the user in files known as cookies. This is usually done so that the user does not need to enter their information again and again, like username and password. Some cookies, known as third-party cookies, store other detailed information about user interests and web browsing patterns. Third-party cookies are stored so that the site can provide targeted content or ads.
6. **JavaScript Interpreter:** It is responsible for interpreting the JavaScript code. After interpretation, it sends the result to the rendering engine so that it can be displayed.
7. **UI Backend:** It is mainly used for drawing widgets like windows.

```
<html>
  <head>
    <title>
      Tutorial 2
    </title>
  </head>
  <style>
    table{
      border-color: black;
      background-color: rgb(140, 238, 240);
      color:black;
    }
    .row1{
      height: 30px;
    }
  </style>
  <body>
```

8

```
</style>
<body>
  <table border="3px">
    <caption style="caption-side:bottom;"><b>Student Details</b></caption>
    <tr>
      <th>Enrollment No.</th>
      <th>Name</th>
      <th>City</th>
    </tr>
    <div class="row1"></div><tr>
      <td>211240116070</td>
      <td>Peter</td>
      <td>Valsad</td>
    </tr></div>
    <tr>
      <td>211240116055</td>
      <td>Bhargavi</td>
      <td>Sholdhara</td>
    </tr>
    <tr>
      <td>211240116064</td>
      <td>Jagdish</td>
      <td>Bilimora</td>
    </tr>
    <tr>
      <td>211240116016</td>
      <td>Jitendra</td>
      <td>Talavchala</td>
    </tr>
  </table>
  <p><a href="tut4.html">Train Details</a></p>
 </body>
</html>
```

22:34

Use of links in html.Explain various target that can be use in link(_self,_blank,_parent,_top).

9

In HTML, links are used to create hyperlinks that allow users to navigate between web pages and resources on the internet. Links are created using the `<a>` (anchor) element. The most common use of links is to connect one web page to another, but they can also link to other types of resources, such as documents, images, or email addresses.

Here's how you create a basic link in HTML:

```html
<a href="https://www.example.com">Visit Example.com</a>
```

Copy code

In the example above, the `href` attribute specifies the destination URL, and the link text "Visit Example.com" is what the user sees and clicks on.

HTML also provides the ability to specify a target for the link using the `target` attribute. The `target` attribute determines where the linked content will open. Here are some commonly used target values:

1. `_self` **(Default):** When you don't specify a `target` attribute, the link opens in the same window or tab that the user is currently viewing. This is the default behavior.

```html
<a href="https://www.example.com">Visit Example.com</a>
```

1. `_blank`: This target value causes the linked content to open in a new browser window or tab. It's often used for external links to keep the current page open for the user.

```html
<a href="https://www.example.com" target="_blank">Visit Example.com in a new
```

1. `_parent`: If your web page is displayed within a frameset or iframe, using `_parent` as the target will load the linked content in the parent frame. It's useful for navigating within framed web pages.

```html
<a href="https://www.example.com" target="_parent">Load Example.com in the p
```

Copy code

1. `_top`: When used in a nested frame structure, `_top` will load the linked content in the top-level window, effectively breaking out of any frames.

```html
<a href="https://www.example.com" target="_top">Load Example.com in the top-
```

Copy code

10

1. **Frameset:** A frameset is an HTML element used to define the structure of a web page that is divided into multiple frames. The `<frameset>` element is typically placed within the `<html>` element but outside the `<body>` element. It specifies the layout of frames within a web page and how much space each frame should occupy. Framesets are usually defined in the main HTML file of a webpage.

   Example of a simple frameset definition:

```html
<!DOCTYPE html>
<html>
<frameset cols="25%,75%">
  <frame src="menu.html">
  <frame src="content.html">
</frameset>
</html>
```

   In this example, a frameset divides the page into two columns, with the left column containing a menu (loaded from `menu.html`) and the right column containing content (loaded from `content.html`).

2. **Frame:** A frame is an individual section within a frameset that displays a separate HTML document or content. The `<frame>` element is used to define each frame and its source content. Frames are contained within the `<frameset>` element and specify where the content should be loaded.

   Example of defining frames within a frameset:

   html — Copy code

```
<frameset cols="25%,75%">
  <frame src="menu.html">
  <frame src="content.html">
</frameset>
```

   In this frameset, there are two frames, one for the menu and one for the content. Each frame is specified with a `src` attribute that points to the HTML file it should load.

While framesets and frames were once used to create multi-part web layouts, they have several drawbacks, including:

- **Usability Issues:** Frames can make it difficult for users to bookmark or share specific pages or navigate back and forth within the website.
- **Search Engine Optimization (SEO) Problems:** Search engines may struggle to index content within frames, leading to poor search engine rankings.
- **Accessibility Challenges:** Frames can pose accessibility issues for users with disabilities and screen readers.
- **Compatibility and Responsiveness:** Framesets and frames are not well-suited for responsive web design, and they may not work correctly on mobile devices.

Due to these limitations and the advent of more modern web design techniques, framesets and frames are largely discouraged in modern web development. Instead, web designers and developers use HTML, CSS, and JavaScript to create responsive, user-friendly, and accessible layouts for websites.

Here's a basic example of how a frameset might have been used in a simple website design:

html

Copy code

```html
<!DOCTYPE html>
<html>
<head>
    <title>Sample Frameset Example</title>
</head>
<frameset rows="20%, 80%">
    <!-- First Frame (Header) -->
    <frame src="header.html" name="header_frame" scrolling="no">
    <!-- Second Frame (Content) -->
    <frame src="main.html" name="content_frame">
</frameset>
</html>
```

Regenerat

In this example:

- The `<frameset>` element is used to create a vertical frameset with two rows: one for the header and one for the content.
- Two `<frame>` elements are defined within the frameset. Each frame has a `src` attribute pointing to an HTML document ("header.html" and "main.html" in this case) to be loaded into the frame.
- The `name` attribute is used to give each frame a unique name. This name can be used as a target for links and navigation within the frames.
- The "scrolling" attribute is set to "no" for the header frame to prevent scrollbars from appearing in that frame.

Once this frameset is loaded in a web browser, it would display the "header.html" content in the top frame and the "main.html" content in the bottom frame. Any links or navigation within these frames can be targeted to open within the same frame or a different frame by specifying the frame's name in the `target` attribute of an anchor (`<a>`) element.

Regenerate

11

HTML Lists

← Prev Next →

HTML Lists are used to specify lists of information. All lists may contain one or more list elements. There are three different types of HTML lists:

1. Ordered List or Numbered List (ol)
2. Unordered List or Bulleted List (ul)
3. Description List or Definition List (dl)

Note: We can create a list inside another list, which will be termed as nested List.

HTML Ordered List or Numbered List

In the ordered HTML lists, all the list items are marked with numbers by default. It is known as numbered list also. The ordered list starts with <ol> tag and the list items start with <li> tag.

```
<ol>
 <li>Aries</li>
 <li>Bingo</li>
 <li>Leo</li>
 <li>Oracle</li>
</ol>
```

Test it Now

Output:

1. Aries
2. Bingo
3. Leo
4. Oracle

Click here for full details of HTML ordered list. HTML Ordered List

HTML Unordered List or Bulleted List

In HTML Unordered list, all the list items are marked with bullets. It is also known as bulleted list also. The Unordered list starts with <ul> tag and list items start with the <li> tag.

```
<ul>
 <li>Aries</li>
 <li>Bingo</li>
 <li>Leo</li>
 <li>Oracle</li>
</ul>
```

Test it Now

Output:

- Aries
- Bingo
- Leo
- Oracle

Click here for full details of HTML unordered list. HTML Unordered List

HTML Description List or Definition List

HTML Description list is also a list style which is supported by HTML and XHTML. It is also known as definition list where entries are listed like a dictionary or encyclopedia.

The definition list is very appropriate when you want to present glossary, list of terms or other name-value list.

The HTML definition list contains following three tags:

1. **<dl> tag** defines the start of the list.
2. **<dt> tag** defines a term.
3. **<dd> tag** defines the term definition (description).

```
<dl>
 <dt>Aries</dt>
 <dd>-One of the 12 horoscope sign.</dd>
 <dt>Bingo</dt>
 <dd>-One of my evening snacks</dd>
<dt>Leo</dt>
<dd>-It is also an one of the 12 horoscope sign.</dd>
 <dt>Oracle</dt>
 <dd>-It is a multinational technology corporation.</dd>
</dl>
```

Test it Now

```
</dl>
```

Test it Now

Output:

Aries

-One of the 12 horoscope sign.

Bingo

-One of my evening snacks

Leo

-It is also an one of the 12 horoscope sign.

Oracle

-It is a multinational technology corporation.

Q-13 Use of form tag. Write a program to create form with use of different forms tags.

- The '<form>' tag in HTML is used to create interactive forms that allow users to input data.

- Forms are a crucial part of web application and websites for tasks like user registration, login, search, and more.

x:

<html>
<body>

```
<html>
  <head>
    <title>Registration Form</title>
  </head>
  <style>
    body{
      background-color: aquamarine;
    }
  </style>
  <body>
    <fieldset>
      <legend>Registration Form</legend>
    <form action="#">
      <label>First Name:</label><br>
      <input type="text" name="First_Name"><br>

      <label>Last Name:</label><br>
      <input type="text" name="Last_Name"><br>

      <label>Email:</label><br>
      <input type="email" name="Email"><br>

      <label>Password:</label><br>
      <input type="password" name="Password"><br>

      <label>Gender:</label><br>

      <input type="radio" id="Male" name="html" value="Male">
      <label for="Male">Male </label>

      <input type="radio" id="Female" name="html" value="Female">
      <label for="Female">Female </label><br>

      <label>Country:</label><br>
```

Type a message

Search

```
<label>Country:</label><br>
<input type="checkbox" id="Country1" name="Country1" value="India">
<label for="Country1">India</label><br>
<input type="checkbox" id="Country2" name="Country2" value="China">
<label for="Country2">China</label><br>
<input type="checkbox" id="Country3" name="Country3" value="Nepal">
<label for="Country3">Nepal</label><br>

<label>Division:</label>
<select name="Division" id="Division">
    <option value="A">A</option>
    <option value="B">B</option>
    <option value="C">C</option>
    <option value="D">D</option>
    <option value="None" selected>None</option>
</select><br>

<input type="submit" value="Submit">
<input type="reset" value="Reset">
</form>
</fieldset>
</body>
</html>
```