

- N. B.: (1) **All** questions are **compulsory**.  
 (2) Make **suitable assumptions** wherever necessary and **state the assumptions** made.  
 (3) Answers to the **same question** must be **written together**.  
 (4) Numbers to the **right** indicate **marks**.  
 (5) Draw **neat labeled diagrams** wherever **necessary**.  
 (6) Use of **Non-programmable** calculators is **allowed**.

<b>1. Attempt <u>any three</u> of the following:</b>	<b>15</b>
<p><b>a. What is WWW? [1 marks]</b></p> <p><b>Write difference between WWW and Internet. [4 marks]</b></p> <p><b>Answer:</b>        The World Wide Web (WWW) is an open source information space where documents and other web resources are identified by URLs, interlinked by hypertext links, and can be accessed via the Internet. The World Wide Web was invented by English scientist Tim Berners-Lee in 1989.        Many people use the terms Internet and World Wide Web interchangeably, but in fact the two terms are not synonymous. The Internet and the Web are two separate but related things.</p> <p><b>What is The Internet?</b>        The Internet is a massive network of networks, a networking infrastructure. It connects millions of computers together globally, forming a network in which any computer can communicate with any other computer as long as they are both connected to the Internet. Information that travels over the Internet does so via a variety of languages known as protocols.</p> <p><b>What is The Web (World Wide Web)?</b>        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 top of the Internet. The Web uses the HTTP protocol, only one of the languages spoken over the Internet, to transmit data. Web services, which use HTTP to allow applications to communicate in order to exchange business logic, use the Web to share information. 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.</p> <p><b>The Web is a Portion of The Internet</b>        The Web is just one of the ways that information can be disseminated over the Internet. The Internet, not the Web, is also used for email, which relies on SMTP, Usenet news groups, instant messaging and FTP. So the Web is just a portion of the Internet, although a large portion, but the two terms are not synonymous and should not be confused.</p>	
<p><b>b. List and explain different types of CSS selectors with example. [5 marks]</b></p> <p><b>Answer:(Any Five:1 mark for each)</b>        CSS selectors are used to "find" (or select) HTML elements based on their element name, id, class, attribute, and more.</p> <p><b>1)Element Selector:</b>        The element selector selects elements based on the element name.        We can select all &lt;p&gt; elements on a page like this given below. In this case, all &lt;p&gt; elements will be center-aligned, with a red text color):</p>	

```
p {
  text-align: center;
  color: red;
}
```

**2)Class Selector:**

The class selector selects elements with a specific class attribute.

To select elements with a specific class, write a period (.) character, followed by the name of the class.

In the example below, all HTML elements with class="center" will be red and center-aligned:

```
center {
  text-align: center;
  color: red;
}
```

**3)ID Selector:**

The id selector uses the id attribute of an HTML element to select a specific element.

The id of an element should be unique within a page, so the id selector is used to select one unique element!

To select an element with a specific id, write a hash (#) character, followed by the id of the element.

The style rule below will be applied to the HTML element with id="para1":

```
#para1 {
  text-align: center;
  color: red;
}
```

**4)Grouping Selectors:**

To group selectors, separate each selector with a comma. The rules within the brackets are applied to both selectors:

```
h1,p {
  text-align: center;
  color: red;
}
```

**5)Combining Selectors:**

You can join some selectors to define even more specific rules:

- p.warning{ } matches all paragraphs with the class of warning
- div#example{ } matches the element with the id attribute example, but only when it is a div.

**c. Write short note on Uniform Resource Locator. [5 Marks]**

**Answer:**

A URL (Uniform Resource Locator) is a form of URI and is a standardized naming convention for addressing documents accessible over the Internet or Intranet. An example of a URL is <http://www.computerhope.com>, which is the URL for the Computer Hope website.

**Overview of a URL:**

Below is additional information about each of the sections of the http URL for this page.

<http://www.computerhope.com/jargon/u/url.htm>

Protocol    Subdomain    Domain and domain suffix    Directories    Web page

- The http:// stands for Hyper Text Transfer Protocol and enables the browser to know what protocol it is going to use to access the information specified in the domain. A URL is not explicit to HTTP addresses; HTTPS, FTP, TFTP, Telnet, and other addresses are also considered URLs and may not follow the same syntax as our example.

	<ul style="list-style-type: none"> <li>• www. stands for World Wide Web and is used to distinguish the content. This portion of the URL is not required and many times can be left out. For example, typing "http://computerhope.com" would still get you to the Computer Hope web page. This portion of the address can also be substituted for an important subpage known as a subdomain. For example, http://support.microsoft.com is the support section of Microsoft's website.</li> <li>• computerhope.com is the domain name for the website. The last portion of the domain is known as the "domain suffix", or TLD, and is used to identify the type or location of the website. For example, .com is short for commercial, .org is short for organization, and .uk is United Kingdom. There are dozens of other domain suffixes available. To get your own domain you would register the name through a domain registrar.</li> <li>• The jargon and u portions of the above URL are the directories of where on the server the web page is located. In this example, the web page is two directories deep, so if you were trying to find the file on the server, it would be in the /public_html/jargon/u directory.</li> <li>• url.htm is the actual web page on the domain you're viewing. The trailing .htm is the file extension of the web page that indicates the file is an HTML file. Other common file extensions on the Internet include .php, .asp, .cgi, .xml, .jpg, and .gif. Each of these file extensions performs a different function, just like all the different types of files on your computer.</li> </ul>	
d.	<p><b>Explain the following HTML tags with the help of example: [5 marks-1 mark each]</b></p> <p><b>Answer:</b></p> <p>(i) &lt; br &gt; inserts a single line break. The &lt;br&gt; tag is an empty tag which means that it has no end tag. <b>Example:</b> A line break is marked up as follows: This text contains&lt;br /&gt;a line break.</p> <p>(ii) &lt;pre&gt; tag defines preformatted text. Text in a &lt;pre&gt; is displayed in a fixed-width font (usually Courier), and it preserves both spaces and line breaks. <b>Example:</b> &lt;pre&gt; This is preformatted text. &lt;/pre&gt;</p> <p>(iii) &lt;h6&gt; tag defines the least important heading. <b>Example:</b> &lt;h6&gt;This is a heading&lt;/h6&gt;</p> <p>(iv) &lt;p&gt; tag defines a paragraph <b>Example:</b> &lt;p&gt;This is a paragraph&lt;/p&gt;</p> <p>(v) &lt;a&gt; tag defines a hyperlink, which is used to link from one page to another. The most important attribute of the &lt;a&gt; element is the href attribute, which indicates the link's destination. <b>Example:</b> &lt;a href="http://www.w3schools.com"&gt;Visit W3Schools.com!&lt;/a&gt;</p>	
e.	<p><b>What is proxy server? [1 Mark]</b></p> <p><b>Discuss its application with reference to internet. [4 Marks]</b></p>	

**Answer:**

In a computer network a proxy server is a server (a computer system or an application program) that acts as an intermediary for requests from clients seeking resources from other servers.

- A client connects to the proxy server, requesting some service, such as a file, connection, web page, or other resource, available from a different server.
- The proxy server evaluates the request according to its filtering rules. For example, it may filter traffic by IP address or protocol.
- If the request is validated by the filter, the proxy provides the resource by connecting to the relevant server and requesting the service on behalf of the client.
- A proxy server may optionally alter the client's request or the server's response, and sometimes it may serve the request without contacting the specified server.
- In this case, it 'caches' responses from the remote server, and returns subsequent requests for the same content directly.

A proxy server has a large variety of potential purposes, including:

- To keep machines behind it anonymous (mainly for security).
- To speed up access to resources (using caching). Web proxies are commonly used to cache web pages from a web server.
- To apply access policy to network services or content, e.g. to block undesired sites.
- To log / audit usage, i.e. to provide company employee Internet usage reporting.
- To bypass security/ parental controls.
- To scan transmitted content for malware before delivery.
- To scan outbound content, e.g., for data leak protection.
- To circumvent regional restrictions.

**f. Explain different types of lists available in HTML with the help of example. [5 Marks]**

**Answer:**

There are three types of lists available in HTML: Ordered Lists, Unordered Lists, Description/Definition Lists

**(1) Ordered Lists**

An ordered list is also a list of items. The list items are marked with numbers.

An ordered list starts with the <ol> tag. Each list item starts with the <li> tag.

```
<ol>
  <li>Coffee</li>
  <li>Milk</li>
</ol>
```

The **type** attribute of <ul> tag can take the following values:

- 1: The list items will be numbered with numbers (default)
- A: The list items will be numbered with numbers (default)
- a: The list items will be numbered with numbers (default)
- I: The list items will be numbered with numbers (default)
- i: The list items will be numbered with numbers (default)

**(2) Unordered Lists**

An unordered list is a list of items. The list items are marked with bullets (typically small black circles).

An unordered list starts with the <ul> tag. Each list item starts with the <li> tag.

```
<ul>
```

```
<li>Coffee</li>
```

```
<li>Milk</li>
```

```
</ul>
```

The **type** attribute of <ul> tag can take the following values:

disc: Sets the list item marker to a bullet (default)

circle: Sets the list item marker to a circle

square: Sets the list item marker to a square.

### (3) Definition Lists

A definition list is not a list of single items. It is a list of items (terms), with a description of each item (term).

A definition list starts with a <dl> tag (definition list).

Each term starts with a <dt> tag (definition term).

Each description starts with a <dd> tag (definition description).

```
<dl>
```

```
<dt>WWW</dt>
```

```
<dd>World Wide Web</dd>
```

```
</dl>
```

## 2. Attempt any three of the following:

15

### a. How to format and position a division on a web page? Explain with example. [5 Marks]

#### Answer:

The <div> tag defines a division or a section in an HTML document. The <div> element is often used as a container for other HTML elements to style them with CSS to layout a web page.

#### Formatting Divisions:

You can use style attribute in <div> tag to specify the font family, font style, font weight, alignment, color etc.

#### Example:

```
<div style="font-family: georgia,garamond,serif; font-weight:bold; font size:large;
background-color:lightblue">
<h3>This is a heading</h3>
<p>This is a paragraph.</p>
</div>
```

#### Positioning a Division on the Page:

If you need a division to be in a specific spot on the page, use the position style rule, which has three possible values:

Attribute	Description
position: absolute	This value specifies a fixed position with respect to the parent element. Unless the element is within some other tag, the parent element is generally the <body> tag; in this case, the element would have a fixed position relative to the upper-left corner of the page.

position: relative	This value specifies an offset from the element's natural position. Other elements on the page are not affected, even if the new position causes elements to overlap.
position: fixed	This value specifies a fixed position within the browser window that doesn't change even when the display is scrolled up or down.

**Example:**

```
<div style="position:absolute; left:80px; top:20px;background color:yellow">
  This div has absolute positioning.
</div>
```

**b. Write HTML code to design given web page using Table tags. [5 Marks]**

Sales Report			
ITEM CODE	UNITS	RATE	SALES
1	2	100	200
2	5	50	250
3	10	250	2500
Total Sales			2950

**Answer:**

```
<html>
<body>
<table border="1">
  <tr>
    <td colspan="4" align="center" >Sales Report</td>
  </tr>
  <tr>
    <td align="center">ITEM CODE</td>
    <td align="center">UNITS</td>
    <td align="center">RATE</td>
    <td align="center">SALES</td>
  </tr>
  <tr>
    <td align="center">1</td>
    <td align="center">2</td>
    <td align="center">100</td>
    <td align="center">200</td>
  </tr>
  <tr>
    <td align="center">2</td>
    <td align="center">5</td>
    <td align="center">50</td>
    <td align="center">250</td>
  </tr>
  <tr>
    <td align="center">3</td>
    <td align="center">10</td>
    <td align="center">250</td>
    <td align="center">2500</td>
  </tr>
  <tr>
    <td colspan="3" align="center">Total Sales</td>
    <td align="center">2950</td>
  </tr>
</table>
```

```

<td align="center">3</td>
<td align="center">10</td>
<td align="center">250</td>
<td align="center">2500</td>
</tr>
<tr>
<td colspan="3" align="right">Total Sales</p></td>
<td align="center">2950</td>
</tr>
</table>
</body>
</html>

```

c. **How will you create graphical navigation bar? Explain with example. [5 Marks]**

**Answer:**

### **Creating a Graphical Navigation Bar**

HTML5 includes a <nav> tag to place the code for a navigation bar. The <nav> tag is designed to help browsers and style sheets identify sets of links as a navigational element, and handle them appropriately.

To create graphical navigation bar use <img> tag within the <a> tag, like this:

```
<a href="product.htm"></a>
```

The **example** below adds a graphic-based navigation bar to the html page

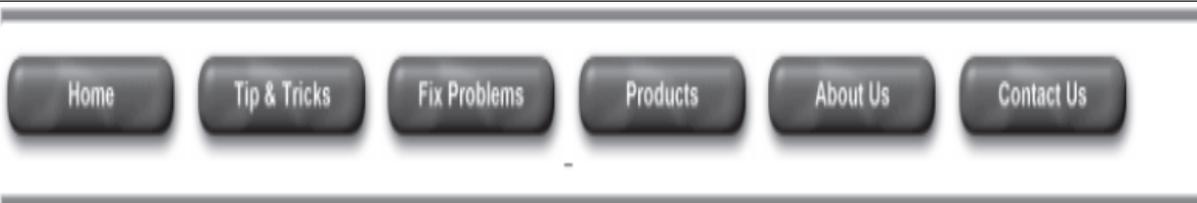
In Notepad, in the upper navigation bar, change the hyperlinks so that they reference the button graphics in the /images folder rather than displaying text.

```

<nav>
<hr>
<p style="margin:0px">
<a href="index.htm"></a>
<a href="tips.htm"></a>
<a href="problems.htm"></a>
<a href="products.htm"></a>
<a href="about.htm"></a>
<a href="contact.htm">
</a></p>
<hr>
</nav>

```

The above code reference the button graphics in the /images folder rather than displaying text.  
Output:



**d. Explain <audio> and <video> tags in HTML 5. [5 Marks]**

**Answer:**

**<audio> tag**

The <audio> tag defines sound, such as music or other audio streams.

Currently, there are 3 supported file formats for the <audio> element: MP3, Wav, and Ogg:

**Attributes:**

Attribute	Value	Description
<u>autoplay</u>	autoplay	Specifies that the audio will start playing as soon as it is ready
<u>controls</u>	controls	Specifies that audio controls should be displayed (such as a play/pause button etc)
<u>loop</u>	loop	Specifies that the audio will start over again, every time it is finished
<u>muted</u>	muted	Specifies that the audio output should be muted
<u>preload</u>	auto metadata none	Specifies if and how the author thinks the audio should be loaded when the page loads
<u>src</u>	URL	Specifies the URL of the audio file

**Example**

<html>

<body>

<audio controls>

<source src="horse.ogg" type="audio/ogg">

<source src="horse.mp3" type="audio/mpeg">

Your browser does not support the audio element.

</audio>

<p><strong>Note:</strong> The audio tag is not supported in Internet Explorer 8 and earlier versions.</p>

</body>

</html>

**<video> tag**

The <video> tag specifies video, such as a movie clip or other video streams.

Currently, there are 3 supported video formats for the <video> element: MP4, WebM, and Ogg:

**Attributes:**

Attribute	Value	Description
<u>autoplay</u>	autoplay	Specifies that the video will start playing as soon as it is ready
<u>controls</u>	controls	Specifies that video controls should be displayed (such as a play/pause button etc).
<u>height</u>	pixels	Sets the height of the video player
<u>loop</u>	loop	Specifies that the video will start over again, every time it is finished
<u>muted</u>	muted	Specifies that the audio output of the video should be muted
<u>poster</u>	URL	Specifies an image to be shown while the video is downloading, or until the user hits the play button
<u>preload</u>	auto metadata none	Specifies if and how the author thinks the video should be loaded when the page loads
<u>src</u>	URL	Specifies the URL of the video file
<u>width</u>	pixels	Sets the width of the video player

### Example

```

<html>
<body>
<video width="320" height="240" controls>
<source src="movie.mp4" type="video/mp4">
<source src="movie.ogv" type="video/ogg">
  Your browser does not support the video tag.
</video>
<p>
<strong>Note:</strong>
The video tag is not supported in Internet Explorer 8 and earlier versions.
</p>
</body>
</html>

```

e. Write HTML Code to design a web page with Imagemaps. [5 Marks]

Answer:

```

<html>
<head>
<title> image-map</title>
</head>
<body>

<map name="planetmap">
  <area shape="rect" coords="0,0,82,126" href="sun.htm" alt="Sun">
  <area shape="circle" coords="90,58,3" href="mercur.htm" alt="Mercury">
  <area shape="circle" coords="124,58,8" href="venus.htm" alt="Venus">
</map>

```

```
</body>
</html>
```

**f. List and explain any five HTML Form controls with example. [5 Marks]**

**Answer:**

There are different types of form controls that can be used to collect data using HTML <form> tag –

**(1) Text Input Control:**

This control is used for items that require only one line of user input, such as search boxes or names. They are created using HTML <input> tag.

Example:

```
<input name="firstname" type="text">
```

Output: First name:

**(2) Checkboxes Control:**

Checkboxes are used when more than one option is required to be selected. They are also created using HTML <input> tag but type attribute is set to **checkbox**.

Example:

```
<form>
  <input type = "checkbox" name = "maths" value = "on"> Maths
  <input type = "checkbox" name = "physics" value = "on"> Physics
</form>
```

Output:  Maths  Physics

**(3) Radio Box Control:**

Radio buttons are used when out of many options, just one option is required to be selected. They are also created using HTML <input> tag but type attribute is set to **radio**.

Example:

```
<form>
  <input type = "radio" name = "subject" value = "maths"> Maths
  <input type = "radio" name = "subject" value = "physics"> Physics
</form>
```

Output:  Maths  Physics

**(3) Select Box Control:**

A select box, also called drop down box which provides option to list down various options in the form of drop down list, from where a user can select one or more options

Example:

```
<form>
  <select name = "dropdown">
    <option value = "Maths" selected>Maths</option>
    <option value = "Physics">Physics</option>
  </select>
</form>
```

**Output:**

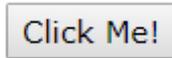
**(5) Button Control:**

The `<button>` Control defines a clickable **button**

Example:

```
<button type="button" onclick="alert('Hello World!')">Click Me!</button>
```

Output:



**3. Attempt *any three* of the following:**

**15**

**a. Write a short note on “for...in” looping statement in JavaScript. [5 Marks]**

**Answer:**

The for...in statement loops through the elements of an array or through the properties of an object.

**Syntax**

```
for (variable in object)  
{  
  code to be executed  
}
```

- The code in the body of the for...in loop is executed once for each element/property.
- The variable argument can be a named variable, an array element, or a property of an object.
- In each iteration, one property from **object** is assigned to **variable** and this loop continues till all the properties of the object are exhausted.

**Example:**

Use of for...in statement to loop through an array:

```
<html>  
<body>  
<script type="text/javascript">  
  var x;  
  var mycars = new Array();  
  mycars[0] = "Saab";  
  mycars[1] = "Volvo";  
  mycars[2] = "BMW";  
  for (x in mycars)  
  {  
    document.write(mycars[x] + "<br />");  
  }  
</script>
```

	<pre> &lt;/body&gt; &lt;/html&gt; <b>Output:</b> Saab Volvo BMW </pre>	
<p><b>b.</b></p>	<p><b>Write a program in JavaScript to accept a sentence from the user and display the number of words in it. (Do not use split () function). [5 Marks]</b></p> <p><b>Answer:</b></p> <pre> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt; &lt;title&gt;Without using split function&lt;/title&gt; &lt;script&gt;     var str=prompt("Enter the sentence=", "");     var count=0;     for(i=0;i&lt;str.length;i++)     {     if(str.charAt(i,1)==" " &amp;&amp; str.charAt(i+1,1)!=" ")     count++;     }     document.write("Number of words are="+count); &lt;/script&gt; &lt;/head&gt; &lt;body&gt; &lt;/body&gt; &lt;/html&gt; </pre>	
<p><b>c.</b></p>	<p><b>Explain following events: [5 Marks:1 Mark each]</b></p> <p><b>Answer:</b></p> <p><b>(i) onclick:</b> This event occurs when user clicks an HTML element like a link or a button.</p> <p><b>(ii) onfocus:</b> This event occurs when an HTML element like document or text fields or textarea or selection receives current focus.</p> <p><b>(iii) onmouseover:</b> This event occurs when user moves the mouse over an HTML element like image.</p>	

**(iv) onload:**

This event occurs when the browser has finished loading the page or document.

**(v) onerror:**

This event occurs when an error occurs.

**Example:**

```
<input type="button" onclick="sayHello()" value="Say Hello" />
```

**d. Write a JavaScript program using various methods of Date Object. [5 Marks]**

**Answer:**

**JavaScript program:**

```
<html>
<body>
<script language="javascript">
    var d = new Date();
    document.write("Today's date and time:"+d+"<br>");
    document.write("Day:-"+d.getDate()+"<br>");
    document.write("Day of week:-"+d.getDay()+"<br>");
    document.write("Year:-"+d.getFullYear()+"<br>");
    document.write("Hours:-"+d.getHours()+"<br>");
    document.write("Minutes:-"+d.getMinutes()+"<br>");
    document.write("Month:-"+d.getMonth()+"<br>");
    document.write("Seconds:-"+d.getSeconds()+"<br>");
    document.write("Time:-"+d.getTime()+"<br>");
    d.setDate(20)
    document.write("Set Date:-"+d+"<br>");
    d.setFullYear(2018);
    document.write("Set FullYear:-"+d+"<br>");
    d.setHours(12);
    d.setMinutes(46);
    d.setMonth(4);
    d.setSeconds(12);
    document.write("Modified Date:-"+d);
</script>
</body>
</html>
```

**Output:**

```
Today's date and time: Thu Feb 4 08:07:33 UTC+0530 2016
Day:-4
Day of week:-4
Year:-2016
```

Hours:-8  
 Minutes:-7  
 Month:-1  
 Seconds:-33  
 Time:-1454553453419  
 Set Date:-Sat Feb 20 08:07:33 UTC+0530 2016  
 Set FullYear:-Tue Feb 20 08:07:33 UTC+0530 2018  
 Modified Date:-Sun May 20 12:46:12 UTC+0530 2018

e. **Write a short note on comparison and logical operators in JavaScript. [5 Marks]**  
**Answer:**

Comparison and Logical operators are used to test for true or false.

**Comparison Operators**

Comparison operators are used in logical statements to determine equality or difference between variables or values.

Given that **x=5**, the table below explains the comparison operators:

Operator	Description	Example
==	is equal to	x==8 is false
===	is exactly equal to (value and type)	x===5 is true x==="5" is false
!=	is not equal	x!=8 is true
>	is greater than	x>8 is false
<	is less than	x<8 is true
>=	is greater than or equal to	x>=8 is false
<=	is less than or equal to	x<=8 is true

Comparison operators can be used in conditional statements to compare values and take action depending on the result:

```
if (age<18) document.write("Too young");
```

**Logical Operators**

Logical operators are used to determine the logic between variables or values.

Given that **x=6** and **y=3**, the table below explains the logical operators:

	Operator	Description	Example	
	&&	and	(x < 10 && y > 1) is true	
		or	(x==5    y==5) is false	
	!	not	!(x==y) is true	
<b>f.</b>	<b>List various features of JavaScript. [5 Marks]</b> <b>Answer:</b> <ul style="list-style-type: none"> <li>• Javascript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.</li> <li>• Javascript that runs at the client side (ie at the client's browser) is client side java script (CCJS) and javascript that runs at the server is serverside java script (SSJS)</li> <li>• The <u>ECMA-262 Specification</u> defined a standard version of the core JavaScript language. <ul style="list-style-type: none"> <li>- JavaScript is a lightweight, interpreted programming language.</li> <li>- Designed for creating network-centric applications.</li> <li>- Complementary to and integrated with Java.</li> <li>- Complementary to and integrated with HTML.</li> <li>- Open and cross-platform</li> </ul> </li> </ul> <b>Advantages of JavaScript</b> <ul style="list-style-type: none"> <li>• <b>Less server interaction</b> – You can validate user input before sending the page off to the server. This saves server traffic, which means less load on your server.</li> <li>• <b>Immediate feedback to the visitors</b> – They don't have to wait for a page reload to see if they have forgotten to enter something.</li> <li>• <b>Increased interactivity</b> – You can create interfaces that react when the user hovers over them with a mouse or activates them via the keyboard.</li> <li>• <b>Richer interfaces</b> – You can use JavaScript to include such items as drag-and-drop components and sliders to give a Rich Interface to your site visitors.</li> </ul>			
<b>4.</b>	<b>Attempt <u>any three</u> of the following:</b>			<b>15</b>
<b>a.</b>	<b>What is PHP ?[1 Mark]</b> <b>Write the advantages of using PHP for server-side web scripting. [4 Marks]</b> <b>Answer:</b>			

The PHP Hypertext Preprocessor (PHP) is a programming language that allows web developers to create dynamic content that interacts with databases. PHP is basically used for developing web based software applications. This tutorial helps you to build your base with PHP.

### Advantages of PHP

- Open source: It is developed and maintained by a large group of PHP developers, this will help in creating a support community, abundant extension library.
- Speed: It is relatively fast since it uses much system resource.
- Easy to use: It uses C like syntax, so for those who are familiar with C, it's very easy for them to pick up and it is very easy to create website scripts.
- Stable: Since it is maintained by many developers, so when bugs are found, it can be quickly fixed.
- Powerful library support: You can easily find functional modules you need such as PDF, Graph etc.
- Built-in database connection modules: You can connect to database easily using PHP, since many websites are data/content driven, so we will use database frequently, this will largely reduce the development time of web apps.
- Can be run on many platforms, including Windows, Linux and Mac, it's easy for users to find hosting service providers.

**b. Write a PHP code to find the greater of two numbers. Accept the numbers from the user.**

**[5 Marks]**

**Answer:**

**Input.html**

```
<html>
<head>
<title>Greater of two no.s</title>
</head>
<body>
<form method="post" action="check.php">
  1st Number: <input type="text" name="n1"><br>
  2nd Number: <input type="text" name="n2"><br>
<input type="submit" value="Check">
</form>
</body>
</html>
```

**check.php**

```
<?php
$n1=(int)$_POST['n1'];
$n2=(int)$_POST['n2'];
if($n1>$n2)
echo $n1." is greater than ".$n2;
else if($n2>$n1)
```

	<pre> echo \$n2." is greater than ".\$n1; else echo "Both the numbers are equal"; ?&gt; </pre>	
<p><b>c.</b></p>	<p><b>Explain any five string functions available in PHP with example. [5 Marks]</b></p> <p><b>Answer: Any Five string functions with one example each [1 mark for each]</b></p> <p><b>(1) strlen()</b> function returns the length of a string.  Example:  <pre> &lt;?php echo strlen("Hello"); ?&gt; </pre> Output:5</p> <p><b>(2)substr()</b> function returns a part of a string.  Example:  <pre> &lt;?php echo substr("Hello world",6); ?&gt; </pre> Output: world</p> <p><b>(3)strcmp( )</b> function compare two strings(case-sensitive).  Example:  <pre> &lt;?php echo strcmp("Hello world!","Hello world!"); ?&gt; </pre> Output:0</p> <p><b>(4) str_replace( )</b> function replaces a part of a string with another string.  Example:  <pre> &lt;?php echo str_replace("world","Peter","Hello world!"); ?&gt; </pre> Output: Hello Peter!</p> <p><b>(5) strtolower( )</b> function converts a string to lowercase letters.  Example:  <pre> &lt;?php echo strtolower("HELLO WORLD"); ?&gt; </pre> Output:  hello world</p>	
<p><b>d.</b></p>	<p><b>What are the different methods available in PHP for passing the information from one page to another? Explain. [5 Marks]</b></p> <p><b>Answer:</b>  There are two different methods available in PHP for passing the information from one page to other:</p> <ul style="list-style-type: none"> <li>• The GET Method</li> </ul>	

- The POST Method

### **The GET Method:**

- In GET method the data is sent as URL parameters that are usually strings of name and value pairs separated by ampersands (&). In general, a URL with GET data will look like this:
- <http://www.example.com/action.php?name=john&age=24>
- Since the data sent by the GET method are displayed in the URL, it is possible to bookmark the page with specific query string values.
- The GET method is not suitable for passing sensitive information such as the username and password, because these are fully visible in the URL query string as well as potentially stored in the client browser's memory as a visited page.
- Because the GET method assigns data to a server environment variable, the length of the URL is limited. So, there is a limitation for the total data to be sent.

- **Example:**

#### **“Get.html”:**

```
<html>
<body>
<form action="welcome_get.php" method="get">
  Name: <input type="text" name="name"><br>
  E-mail: <input type="text" name="email"><br>
<input type="submit">
</form>
</body>
</html>
```

#### **"welcome\_get.php":**

```
<html>
<body>
  Welcome
  <?php
  echo $_GET["name"]; ?><br>
  Your email address is: <?php echo $_GET["email"];
  ?>
</body>
</html>
```

### **The POST Method:**

- In POST method the data is sent to the server as a package in a separate communication with the processing script. Data sent through POST method will not visible in the URL.
- Advantages and Disadvantages of Using the POST Method.
- It is more secure than GET because user-entered information is never visible in the URL query string or in the server logs.
- There is a much larger limit on the amount of data that can be passed and one can send text data as well as binary data (uploading a file) using POST.
- Since the data sent by the POST method is not visible in the URL, so it is not possible to bookmark the page with specific query.

- **Example:**

#### **“Post.html”**

```
<html>
<body>
<form action="welcome.php" method="post">
```

```

Name: <input type="text" name="name"><br>
E-mail: <input type="text" name="email"><br>
<input type="submit">
</form>
</body>
</html>
"welcome.php":
<html>
<body>
Welcome
<?php
    echo $_POST["name"]; ?><br>
    Your email address is: <?php echo $_POST["email"];
?>
</body>
</html>

```

e. **Write a short note on PHP data types. [5 Marks]**

**Answer:**

Variables can store data of different types, and different data types can do different things. PHP supports the following data types:

**PHP String**

A string is a sequence of characters, like "Hello world!".

Example: \$x = "Hello world!";

**PHP Integer**

An integer data type is a non-decimal number between -2,147,483,648 and 2,147,483,647.

Example: \$x = 5985;

**PHP Float**

A float (floating point number) is a number with a decimal point or a number in exponential form.

Example: \$x = 10.365;

**PHP Boolean**

A Boolean represents two possible states: TRUE or FALSE.

Example:

\$x = true;

\$y = false;

**PHP Array**

An array stores multiple values in one single variable.

Example: \$cars = array("Volvo", "BMW", "Toyota");

**PHP Object**

An object is a data type which stores data and information on how to process that data.

In PHP, an object must be explicitly declared.

Example:

```

<?php
class Car {
    function Car() {

```

	<pre> \$this-&gt;model = "VW"; } }  // create an object \$herbie = new Car();  // show object properties echo \$herbie-&gt;model; ?&gt; </pre> <p><b>PHP NULL Value</b>  Null is a special data type which can have only one value: NULL.  Example: \$x = null;</p>	
f.	<p><b>Explain associative array in PHP with the help of example. [5 Marks]</b>  <b>Answer:</b></p> <ul style="list-style-type: none"> <li>• Associative arrays are arrays that use named keys that you assign to them.</li> <li>• An associative array, each ID key is associated with a value.</li> <li>• When storing data about specific named values, a numerical array is not always the best way to do it.</li> <li>• With associative arrays we can use the values as keys and assign values to them.</li> <li>• There are two ways to create an associative array:</li> </ul> <p>Example 1  In this example we use an array to assign ages to the different persons:  \$ages = array("Peter"=&gt;32, "Quagmire"=&gt;30, "Joe"=&gt;34);</p> <p>Example 2  This example is the same as example 1, but shows a different way of creating the array:  \$ages['Peter'] = "32";  \$ages['Quagmire'] = "30";  \$ages['Joe'] = "34";</p> <p>The ID keys can be used in a script:  &lt;?php  \$ages['Peter'] = "32";  \$ages['Quagmire'] = "30";  \$ages['Joe'] = "34";  echo "Peter is " . \$ages['Peter'] . " years old.";  ?&gt;</p> <p>The code above will output:  Peter is 32 years old.</p>	
5.	<b>Attempt <u>any three</u> of the following:</b>	<b>15</b>
a.	<b>Explain following PHP/MYSQL functions: [5 Marks:1 Mark each]</b> <b>Answer:</b>	

(i) **mysql\_connect**- PHP provides **mysql\_connect** function to open a database connection.

**Syntax:**

```
mysql_connect(server,user,password);
```

(ii) **mysql\_close**- PHP provides function **mysql\_close** to close a database connection.

**Syntax:**

```
mysql_close ( resource $link_identifier );
```

(iii) **mysql\_query**- PHP uses **mysql\_query** function to create a MySQL database and table.

**Syntax:**

```
mysql_query( sql, connection );
```

(iv) **mysql\_select\_db**- PHP provides function **mysql\_select\_db** to select a database. It returns TRUE on success or FALSE on failure.

**Syntax:**

```
mysql_select_db( db_name, connection );
```

(v) **mysql\_error**-Returns the text of the error message from previous MySQL Operation.

**Syntax:**

```
mysql_error();
```

**Example:**

```
<?php
    $dbhost = 'localhost:3036';
    $dbuser = 'root';
    $dbpass = 'rootpassword';
    $conn = mysql_connect($dbhost, $dbuser, $dbpass);
    if(! $conn ) {
        die('Could not connect: ' . mysql_error());
    }
    $sql = 'INSERT INTO employee '
    '(emp_name,emp_address, emp_salary, join_date) '
    'VALUES ( "guest", "XYZ", 2000, NOW() )';
    mysql_select_db('test_db');
    $retval = mysql_query( $sql, $conn );
    if(! $retval ) {
        die('Could not enter data: ' . mysql_error());
    }
    echo "Entered data successfully\n";
    mysql_close($conn);
?>
```

**b. Write a PHP program to demonstrate the use of cookies in PHP. [5 Marks]**

**Answer:**

The following example creates a cookie named "user" with the value "John Doe". The cookie will expire after 30 days (86400 \* 30). The "/" means that the cookie is available in entire website (otherwise, select the directory you prefer).

We then retrieve the value of the cookie "user" (using the global variable \$\_COOKIE). We also use the isset() function to find out if the cookie is set:

**Code: Creating Cookie**

```
<?php
```

```

$cookie_name = "user";
$cookie_value = "John Doe";
setcookie($cookie_name, $cookie_value, time() + (86400 * 30), "/"); // 86400 = 1 day
?>
Code : Retrieving the Cookie value
<html>
<body>
<?php
    if(isset($_COOKIE[$cookie_name]))
    {
        echo "Cookie named " . $cookie_name . " is not set!";
    }
    else
    {
        echo "Cookie " . $cookie_name . " is set!<br>";
        echo "Value is: " . $_COOKIE[$cookie_name];
    }
?>
</body>
</html>

```

**c. Compare POSIX and PERL style regular expressions of PHP. [5 Marks]**

**Answer:**

Regular expressions are a sequence or pattern of characters itself. They provide the foundation for pattern-matching functionality.

PHP offers functions specific to two sets of regular expression functions, each corresponding to a certain type of regular expression.

**1) POSIX Regular Expressions**

The structure of a POSIX regular expression is not dissimilar to that of a typical arithmetic expression: various elements (operators) are combined to form more complex expressions.

Brackets ([]) have a special meaning when used in the context of regular expressions. They are used to find a range of characters like:

- [0-9]:matches any decimal digit from 0 to 9.
- [a-z]:matches any character from lowercase a through lowercase z.
- [A-Z]:matches any character from uppercase A through uppercase Z.

**PHP's Regexp POSIX Functions**

PHP uses **ereg()** and **eregi()** functions for searching strings using POSIX-style regular expressions.

**Example:**

```

<?php
$password = "abc";
if (! eregi ("[:alnum:]{8,10}", $password))
{
    print "Invalid password! Passwords must be from 8 - 10 chars";
}
else
{
    print "Valid password";
}
?>

```

This will produce the following result –  
Invalid password! Passwords must be from 8 - 10 chars

## 2) PERL Style Regular Expressions

Perl-style regular expressions are similar to their POSIX counterparts.

### Meta characters

A meta character is simply an alphabetical character preceded by a backslash that acts to give the combination a special meaning.

Following is the list of meta characters which can be used in PERL Style Regular Expressions.

- . -a single character
- \s -a whitespace character (space, tab, newline)
- \S -non-whitespace character
- \d -a digit (0-9)
- \D -a non-digit
- \w -a word character (a-z, A-Z, 0-9, \_)
- \W -a non-word character
- [aeiou] -matches a single character in the given set
- [^aeiou] -matches a single character outside the given set
- (foo|bar|baz)- matches any of the alternatives specified

### PHP's Regexp PERL Compatible Functions

PHP uses **preg\_match()** and **preg\_match\_all()** functions for searching strings using Perl-compatible regular expressions.

#### Example:

```
<?php
    $line = "Vi is the greatest word processor ever created!";
    // perform a case-Insensitive search for the word "Vi"
    if (preg_match("/\bVi\b/i", $line, $match)) :
        print "Match found!";
    endif;
?>
```

This will produce the following result –  
Match found!

- d. List various HTTP functions available in PHP. [3 Marks]**  
**Explain header () function in detail. [2 Marks]**

#### Answer:

The HTTP functions let s manipulate information sent to the browser by the Web server, before any other output has been sent.

#### HTTP Functions:

Function	Description
header()	Sends a raw HTTP header to a client
headers_list()	Returns a list of response headers sent (or ready to send)
headers_sent()	Checks if / where the HTTP headers have been sent
setcookie()	Defines a cookie to be sent along with the rest of the HTTP headers

setrawcookie()	Defines a cookie (without URL encoding) to be sent along with the rest of the HTTP headers
----------------	--

**PHP header() Function:**

The header() function sends a raw HTTP header to a client. header() is called before any actual output is sent.

**Syntax:**

```
header(string,replace,http_response_code)
```

- string- Required. Specifies the header string to send
- replace- Optional. Indicates whether the header should replace previous or add a second header.
- http\_response\_code- Optional. Forces the HTTP response code to the specified value. it indicates whether response is success or not.

**Example:**

The given example redirect to the browser to google.com website

```
<?php
header ("Location: https://www.google.com/");
exit;
?>
```

e. **Write a PHP program to create a database named “employee”. Create a table named “salary” with following fields (eid, ename, esalary). Insert 3 records of your choice. Display the names of the employees whose salary is between 15000 to 20000 in a tabular format. [5 Marks]**

**Answer:**

**Code:**

```
<html>
<head>
<title>Database insert and select</title>
</head>
<body>
<?php
$con=mysql_connect("localhost","root","");
if(!$con)
die('could not connect:'.mysql_error());
if(mysql_query("create database employee",$con))
echo"Database Created successfully";
else
echo "Error creating database:".mysql_error();
mysql_select_db("employee",$con);
$query="create table salary(eid smallint, ename varchar(50),esalary decimal(7,2))";
if(mysql_query($query,$con))
echo"Table Created successfully";
else
```

```

echo "Error creating table:".mysql_error();
$query1="insert into salary values(1,'ABC',15000)";
if(mysql_query($query1,$con))
echo"Record 1 inserted successfully";
else
echo "Error inserting record 1:".mysql_error();
$query2="insert into salary values(2,'XYZ',45000)";
if(mysql_query($query2,$con))
echo"Record 2 inserted successfully";
else
echo "Error inserting record 2:".mysql_error();
$query3="insert into salary values(3,'PQR',23000)";
if(mysql_query($query3,$con))
echo"Record 3 inserted successfully";
else
echo "Error inserting record 3:".mysql_error();
$sql="select * from salary where esalary>=35 and esalary<=75";
$result=mysql_query($sql,$con);
if(mysql_num_rows($result)>0)
{
echo "<table border='1'><tr><th>Employee ID</th>
<th>Employee Name</th><th>Salary</th></tr>";
while($row=mysql_fetch_assoc($result))
{
echo "<tr>";
echo "<td>".$row['eid']."</td>";
echo "<td>".$row['ename']."</td>";
echo "<td>".$row['esalary']."</td>";
echo "</tr>";
}
echo "</table>";
}
else
{
echo "Table is empty";
}
mysql_close($con);
?>
</body>
</html>

```

**f. Write a short note on PHP Session. [5 Marks]**

**Answer:**

A session is a way to store information (in variables) to be used across multiple pages. PHP Session is used to make data accessible across the various pages of an entire website. A session creates a file in a temporary directory on the server where registered session variables and their values are stored. This data will be available to all pages on the site during that visit.

The location of the temporary file is determined by a setting in the **php.ini** file called **session.save\_path**. Before using any session variable make sure you have setup this path.

When a session is started following things happen –

- PHP first creates a unique identifier for that particular session which is a random string of 32 hexadecimal numbers such as 3c7foj34c3jj973hjkop2fc937e3443.
- A cookie called **PHPSESSID** is automatically sent to the user's computer to store unique session identification string.
- A file is automatically created on the server in the designated temporary directory and bears the name of the unique identifier prefixed by sess\_ ie sess\_3c7foj34c3jj973hjkop2fc937e3443.

When a PHP script wants to retrieve the value from a session variable, PHP automatically gets the unique session identifier string from the PHPSESSID cookie and then looks in its temporary directory for the file bearing that name and a validation can be done by comparing both values.

A session ends when the user loses the browser or after leaving the site, the server will terminate the session after a predetermined period of time, commonly 30 minutes duration.

#### **Starting a PHP Session:**

A session is started with the `session_start()` function.

Session variables are set with the PHP global variable: `$_SESSION`.

#### **Destroying a PHP Session:**

A PHP session can be destroyed by `session_destroy()` function.

If you want to destroy a single session variable then you can use `unset()` function to unset a session variable.

#### **Example:**

```
<?php
    //start the session
    session_start();
?>
<!DOCTYPE html>
<html>
<body>
<?php
    // remove all session variables
    session_unset();
    // destroy the session
    session_destroy();
?>
</body>
</html>
```