BSc.(Information Technology)

(Semester II)

2018-19

Web Programming

(USIT203 Core)

University Paper Solution

By

Mrs. Pallavi Tawde

Question 1

Q1a. What is WWW? Write difference between WWW and Internet.

Ans: WWW:

The World Wide Web (WWW) is a network of online content that is formatted in HTML and accessed via HTTP. The term refers to all the interlinked HTML pages that can be accessed over the Internet. The World Wide Web was originally designed in 1991 by Tim Berners-Lee while he was a contractor at CERN. The World Wide Web is most often referred to simply as "the Web."

	Internet	**World Wide Web**
Estimated year of Origin	1969, though opening of the network to commercial interests began only in 1988	1993
Name of the first version	ARPANET	NSFnet
Comprises	Network of Computers, copper wires, fibre-optic cables & wireless networks	Files, folders & documents stored in various computers
Governed by	Internet Protocol	Hyper Text Transfer Protocol
Dependency	This is the base, independent of the World Wide Web	It depends on Internet to work
Nature	Hardware	Software

Q1b. List and Explain different types of CSS selectors with example.

Ans:

The class Selector

With the class selector you can define different styles for the same type of HTML element. Say that you would like to have two types of paragraphs in your document: one right–aligned paragraph, and one center– aligned paragraph. Here is how you can do it with styles:

```
p.right          {text – align: right}

p.center   {text – align: center}
```

You have to use the class attribute in your HTML document:

```
<p class="right">

This paragraph will be right–aligned.</p>

<p class="center">

This paragraph will be center–aligned.</p>
```

The id Selector

The id selector is different from the class selector! While a class selector may apply to SEVERAL elements on a page, an id selector always applies to only ONE element. An ID attribute must be unique within the document.

The style rule below will match a p element that has the id value "para1":

```
#paral

{text–align: center;

color: red}
```

The rule above will match this h1 element:

```
<h1 id= "wer345">Some text</h1>
```

Q 1c. write a short note on Uniform Resource Locator.

***Ans:* URL**

URL is a abbreviation of **U***niform* **R***esource* **L***ocator,* the global address of documents and other resources on the World Wide Web.

Eg: http://www.vsit.edu.in

The first part of the address is called a *protocol identifier* and it indicates what protocol to use, and the second part is called a *resource name* and it specifies the IP address or the domain name where the resource is located. The protocol identifier and the resource name are separated by a colon and two forward slashes.

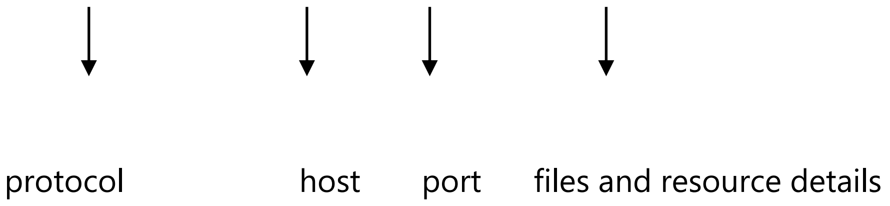
For example, the two URLs below point to two different files at the domain *pcwebopedia.com*. The first specifies an executable file that should be fetched using the FTP protocol, the second specifies a Web page that should be fetched using the HTTP protocol.

- **ftp://www.pcwebopedia.com/stuff.exe**
- **http://www.pcwebopedia.com/index.html**

Structure of an URL:

The following is an outline of the most common form of a URL:

http://www.address.edu:1234/path/subdir/file.txt



Q1d. Explain the following tags with the help of example:

i. <br> ii. <pre> iii. <h6> iv. <p> v. <a>

Ans:

HTML <br> tag

The <br> tag inserts a single line break.

The <br> tag is an empty tag which means that it has no end tag.

Example : A line break is marked up as follows

This text contains<br />a line break.

HTML <pre> tag

The <pre> tag defines preformatted text.

Text in a pre element is displayed in a fixed-width font (usually Courier), and it preserves both spaces and line breaks.

Example : Preformatted text

<pre>

Text in a pre element is displayed in a fixed-width font, and it preserves both spaces and line breaks

</pre>

Heading Tags

- The <h1> to <h6> tags are used to define HTML headings.
- <h1> defines the most important heading. <h6> defines the least important heading.

HTML <p> tag

The <p> tag defines a paragraph.

The p element automatically creates some space before and after itself. The space is automatically applied by the browser, or you can specify it in a style sheet.

Example : A paragraph is marked up as follows

<p>This is some text in a paragraph.</p>

Creating Hyperlinks and Anchors

HTML uses the <a> (anchor) tag to create a link to another document.

An anchor can point to any resource on the Web: an HTML page, an image, a sound file, a movie, etc.

The syntax of creating an anchor :

<a href="url">Text to be displayed</a>

Q1e. What is proxy server? Discuss its application with reference to internet.

Ans:

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.

Q1f. Explain different types of lists available in HTML with the help of example.

Ans:

HTML List (Ordered, Unordered Definition and Nested List)

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>
```

Here is how it looks in a browser :

- Coffee
- Milk

Inside a list item you can put paragraphs, line breaks, images, links, other lists, etc.

Attribute	Value	Description
Type	disc square circle	Specifies the style of the bullet points of the list items

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>
```

Here is how it looks in a browser:

1. Coffee

2. Milk

Inside a list item you can put paragraphs, line breaks, images, links, other lists, etc.

Optional Attributes

Attribute	**Value**	**Description**
Start	*Number*	Specifies the start point in a list
Type	decimal (1, 2, 3, 4 (the default)) decimal-leading-zero (01, 02, 03, 04) lower-roman (i, ii, iii, iv) upper-roman (I, II, III, IV) lower-alpha (a, b, c, d) upper-alpha (A, B, C, D) none(nothing)	Specifies which kind of bullet points will be used.

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>Coffee</dt>
<dd>Black hot drink</dd>
```

```
<dt>Milk</dt>
<dd>White cold drink</dd>
</dl>
```

Here is how it looks in a browser:

```
Coffee

    Black hot drink

Milk

        White cold drink
```

Inside the <dd> tag you can put paragraphs, line breaks, images, links, other lists, etc.

Question 2

Q2a. How to format and position a division on a web page? Explain with example.

Ans:

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 or to perform certain tasks with JavaScript.

format

```
<html>

  <head>
    <title>HTML div Tag</title>
    <link rel = "stylesheet" href = "style2.css">
  </head>

  <body>
    <div id = "contentinfo">
      <p>Welcome to our website. We provide tutorials on various subjects.</p>
    </div>
  </body>

</html>
```

Here is the csss file *style2.css*

```
#contentinfo p {
  line-height: 20px;
  margin: 30px;
  padding-bottom: 20px;
  text-align: justify;
  width: 140px;
  color: red;
}
```

The position Property

The position property specifies the type of positioning method used for an element.

There are five different position values:

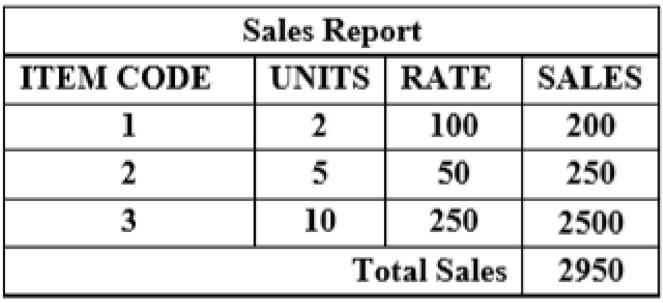
- static
- relative
- fixed
- absolute
- sticky

Elements are then positioned using the top, bottom, left, and right properties. However, these properties will not work unless the position property is set first. They also work differently depending on the position value.

```
<html>
<head>
<style>
div.relative {
  position: relative;
  width: 400px;
  height: 200px;
  border: 3px solid #73AD21;
}

div.absolute {
  position: absolute;
  top: 80px;
  right: 0;
  width: 200px;
  height: 100px;
  border: 3px solid #73AD21;
}
</style>
</head>
<body>
```

```
<h2>position: absolute;</h2>


<div class="relative">This div element has position: relative;
  <div class="absolute">This div element has position: absolute;</div>
</div>

</body>
</html>
```

Q2b. Write HTML code to design given web page using table tags.



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

Ans:

<html>

```
<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="right">Total Sales</p></td>     <td
align="center">2950</td>    </tr>  </table>
```

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

Q2c. How will you create graphical navigation bar? Explain with example.
Ans:

Creating a Graphical Navigation Bar

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

```
<a href="product.htm"><img src="product_button.gif"></a>
```

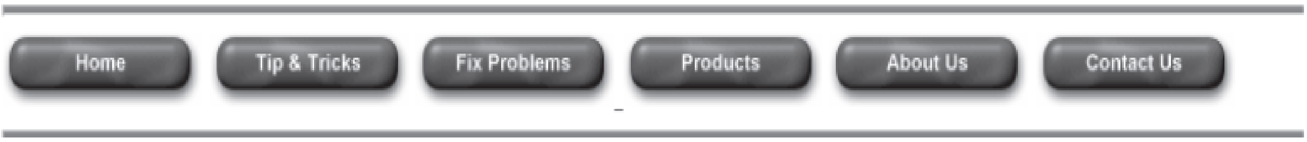
The example below adds a graphic-based navigation bar to the html page

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"><img src="images/btn_home.gif" style="border:none"></a>
        <a href="tips.htm"><img src="images/btn_tips.gif" style="border:none"></a>
<a href="problems.htm"><img src="images/btn_problem.gif" style="border:none"></a>
<a href="products.htm"><img src="images/btn_products.gif" style="border:none"></a>
        <a href="about.htm"><img src="images/btn_about.gif" style="border:none"></a>
        <a href="contact.htm"><img src="images/btn_contact.gif" style="border:none">
        </a></p>
        <hr>
        </nav>
```

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

__**Output:**__

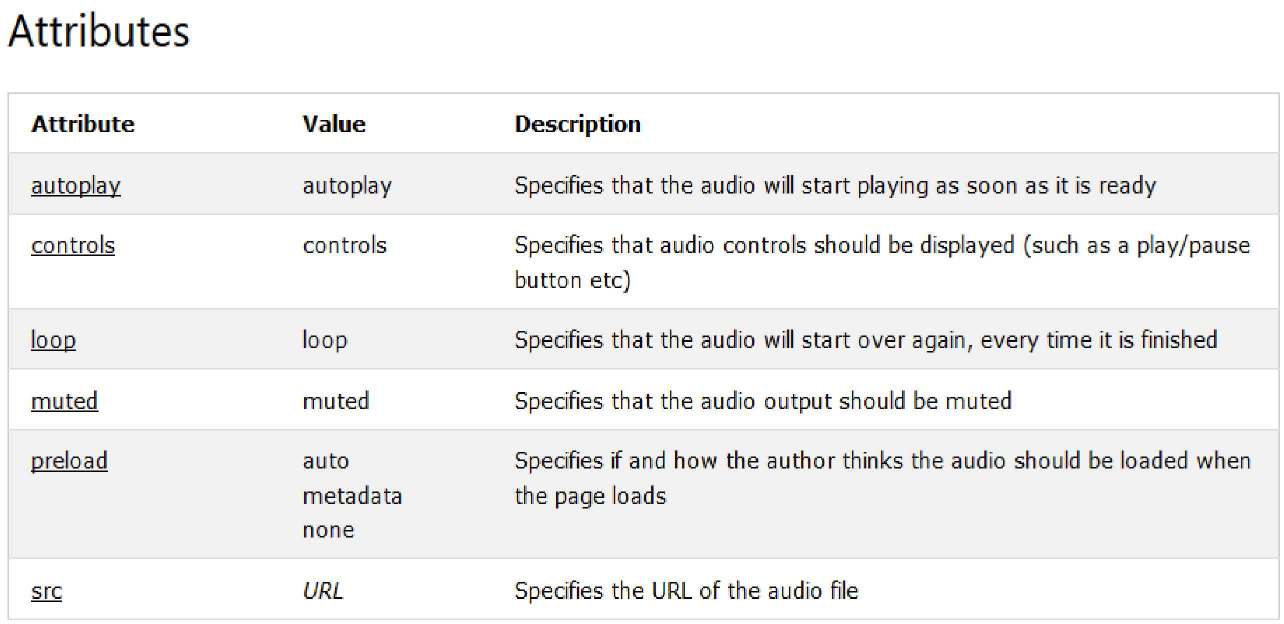


Q2d. Explain <audio> and <video> tags in HTML5

Ans:

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

Example

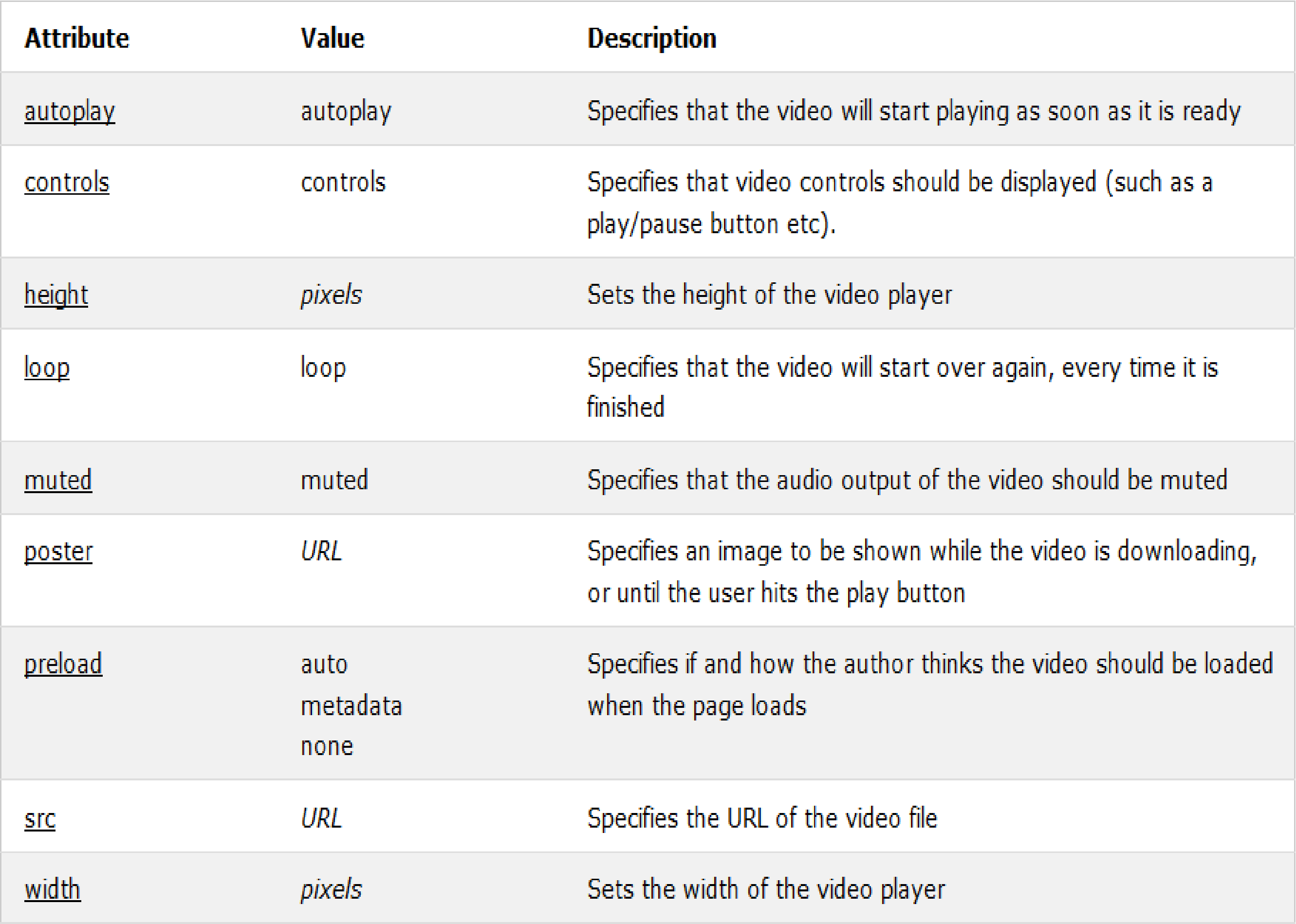
```
<html>
<body>

<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:

- MP4 = MPEG 4 files with H264 video codec and AAC audio codec
- WebM = WebM files with VP8 video codec and Vorbis audio codec
- Ogg = Ogg files with Theora video codec and Vorbis audio codec



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

Q2e. Write HTML code to design a web page with Image maps.

Ans:

HTML <area> tag

Definition and Usage

The <area> tag defines an area inside an image-map (an image-map is an image with clickable areas).

The area element is always nested inside a <map> tag.

Attribute	Value	Description
shape	rect, circle, poly	Specifies the shape of area
coords	number	Specifies the co-ordinates of particular shape
href	url	Specifies the page that will be displayed on the click of that area

```
<img src="planets.gif" width="145" height="126" alt="Planets" usemap="#planetmap"  />

     <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>
```

Q2f. List and explain any five HTML form controls with example.

Ans:

A. Text Fields

Text fields are used when you want user to type letters, numbers etc. in a form.

```
<form>

First name :

<input type= “text” name= “firstname”><br>
```

```
    Last name :

    <input type= “text” name= “lastname”>

  Password:

  <input type= “password” name= “pass”>

    < /form>
```

The output on previous page shows how it looks in a browser. Note that the form itself is not visible. Also note that in most browsers, the width of the text field is 20 characters by default.

B. Radio Buttons

Radio Buttons are used when you want the user to select one of a limited number of choices. Note that only one option can be chosen.

```
    <form>

    <input type= “radio” name= “gender” value= “male”> Male<br>

    <input type= “radio” name= “gender” value= “female”> Female

    < /form>
```

C. Checkboxes

Checkboxes are used when you want the user to select one or more options of a limited number of choices.

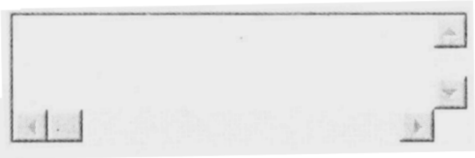
```
    <form>

    <input type= “checkbox” name= “bike”>I have a bike<br>

    <input type= “checkbox” name= “car” >I have a car

    < /form>
```

D. Text Area Box

The difference between the Text Box (above) and the Text Area is that the Text Box only allows for one line. The Text Area, however, is much larger and will allow for as many words as you want.

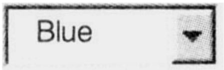
Here’s a Text Area Box:

```
        < textarea name= “comment” rows= 6 cols= 40 >< /textarea>
```



E. Pop–Up Box(Combo Box)

The Pop–Up box, unless clicked on, only shows one item. Click on it to see all the choices. The below Pop–Up box is for people to choose their favorite color :



Here are the commands that placed the Pop–Up box on the page:

```
<select name= “Favorite_Color” size= “1”>

<option selected>Blue

<option>Red

<option>Yellow

<option>Green

<option>Black

<option>Orange

<option>Purple

< /select>
```

Question 3

Q3a. Write a short note on “for in” looping statement in javascript.

Ans: JavaScript For...In Statement

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
}
```

Note: The code in the body of the for...in loop is executed once for each element/property.

Note: The variable argument can be a named variable, an array element, or a property of an object.

Example

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

Example

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

Q3b. 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).

Ans:

```
<html>
        <head><title>Counting number of words</title></head>
        <body>
                <script>
                        var i,c=0;
                        var s =prompt("enter any sentence");
                        //var c=s.split(" ");
                        //document.write("Count of words=",s[0]);

                        for(i=0;i<s.length;i++)
                        {
                                if(s[i]==' ')
                                {
                                        c++;
                                }
                        }

                        document.write("Count of words=",(c+1))
                </script>
        </body>
</html>
```

Q3c. Explain following events:
i. onclick() ii. onfocus() iii. onmouseover() iv. onload() v. onerror()
Ans:

onclick(): when button is clicked.
onfocus():When the input field gets focus
onmouseover(): when the mouse passes over an element
onload(): When the page has been loaded
onerror(): When an error occurs when loading an image

```
<html>
<head>
<script>
function myFunction1() {
  alert("Page is loaded");
}
function myFunction2(x) {
  x.style.background = "yellow";
}

function myFunction() {
  document.getElementById("demo").innerHTML = "Hello World";
}
function imgError() {
  alert('The image could not be loaded.');
}
</script>
</head>
<body onload="myFunction1()">
```

```
Enter your name: <input type="text" onfocus="myFunction2(this)">
<p>Click the button to trigger a function.</p>
<button onclick="myFunction()">Click me</button>
<h1 onmouseover="style.color='red'" onmouseout="style.color='black'">Mouse over this text</h1>
<p id="demo"></p>
<img src="image.gif" onerror="imgError()">
</body>
</html>
```

Q3d. Write a javascript program using various methods of date object.

Ans:

```
<html>
<body>
<script language="javascript">

   var d = new Date();

document.write("Todays 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(2020);
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>
```

Q3e. Write a short note on comparison and logical operators in javascript.

Ans:

JavaScript Comparison and Logical Operators

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

How Can it be Used

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");
```

You will learn more about the use of conditional statements in the next chapter of this tutorial.

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**

<table>
<tr><td>&&</td><td>and</td><td>(x < 10 && y > 1) is true</td></tr>
<tr><td>||</td><td>or</td><td>(x==5 || y==5) is false</td></tr>
<tr><td>!</td><td>not</td><td>!(x==y) is true</td></tr>
</table>

Q3f.List various features of JavaScript.

Ans:

- ***JavaScript gives HTML designers a programming tool :*** HTML authors are normally not programmers, but JavaScript is a scripting language with a very simple syntax! Almost anyone can put small of code into their HTML pages
- ***JavaScript can put dynamic text into an HTML page :*** A JavaScript statement like this: document.write("<h1>" + name + "</h1>") can write a variable text into an HTML page
- ***JavaScript can react to events :*** A JavaScript can be set to execute when something happens, like when a page has finished loading or when a user clicks on an HTML element
- ***JavaScript can read and write HTML elements :*** A JavaScript can read and change the content of an HTML element
- ***JavaScript can be used to validate data :*** A JavaScript can be used to validate form data before it is submitted to a server. This saves the server from extra processing
- ***JavaScript can be used to detect the visitor's browser :*** A JavaScript can be used to detect the visitor's browser, and - depending on the browser - load another page specifically designed for that browser
- ***JavaScript can be used to create cookies :*** A JavaScript can be used to store and retrieve information on the visitor's computer

Question 4

Q4a. What is PHP? Write the advantages of using PHP for server side web scripting.

Ans: PHP:

- PHP is an acronym for "PHP: Hypertext Preprocessor"
- PHP is a widely-used, open source scripting language
- PHP scripts are executed on the server
- PHP is free to download and use

Advantages of using PHP for server side web scripting:

- 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
- 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

Q4b. Write a PHP code to find the greater of two numbers. Accept the numbers from the user.

Ans:

```
<html>

<head><title>Greater Number</title></head>

<body>

<form method="post" action="greater.php">

Value 1: <input type="number"

name="v1">

<br>

Value 2: <input type="number" name="v2">

<br><input type="submit" value="OK">

</form>

<?php
```

```
if(isset($_POST["v1"]) && isset($_POST["v2"]))

{

$n1 = $_POST["v1"];

$n2 = $_POST["v2"];

if($n1 > $n2)

echo "Greater no. = " . $n1;

else if($n2 > $n1)

echo "Greater no. = " . $n2;

else

echo "Both numbers are equal";

}

?>

</body>

</html>
```

Q4c. Explain any five string functions available in PHP with example.

Ans:

i. Lcfirst() - Make a string's first character to lowercase.

   Example:

   ```
   $str = 'HelloWorld';
   $str = lcfirst($str);            // helloWorld
   ```

ii. str_suffle() - Randomly shuffles a string.
   Example:

   ```
   $str = 'abcdef';
   $shuffled = str_shuffle($str);
   ```

iii. strrev() – reverses the string.
   Example:

   ```
   echo strrev("Hello world!"); // !dlrow olleH
   ```

iv. str_replace() - Replace all occurrences of the search string with the replacement string

Example:

str_replace("world","Pallavi","Hello world!"); // Hello Pallavi!

v. substr() - Find the first occurrence of a string
Example:

echo substr("Hello world",6); // world

Q4d. What are the different methods available in PHP for passing the information from one page to another? Explain.

Ans:

$_GET is an array of variables passed to the current script via the URL parameters.

$_POST is an array of variables passed to the current script via the HTTP POST method.

$_REQUEST is used to collect data after submitting an HTML form.

Example of POST method:

File 1 : 1.htm

```
<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>
```

To display the submitted data you could simply echo all the variables. The "welcome.php" looks like this:

File 2: welcome.php

```
<html>
<body>

Welcome <?php echo $_POST["name"]; ?><br>
Your email address is: <?php echo $_POST["email"]; ?>
```

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

Q4e. Write a short note on PHP data types

Ans:

PHP supports the following data types:

PHP String

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

A string can be any text inside quotes. You can use single or double quotes.

PHP Integer

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

The PHP var_dump() function returns the data type and value.

PHP Float

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

PHP Boolean

A Boolean represents two possible states: TRUE or FALSE.

```
$x = true;
$y = false;
```

Booleans are often used in conditional testing.

PHP Array

An array stores multiple values in one single variable.

Example

```
<?php
$x = "Hello world!";
$y = 'Hello world!';
echo $x;
echo "<br>";
echo $y;

$x1 = 5985;
var_dump($x1);

$x2 = 10.365;
var_dump($x2);

$cars = array("Volvo","BMW","Toyota");
var_dump($cars);
?>
```

Q4f. Explain associative array in PHP with the help of example.

Ans: **Associative arrays** - Arrays with named keys. Associative arrays are arrays that use named keys that you assign to them.

There are two ways to create an associative array:

```
$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");
```

or:

```
$age['Peter'] = "35";
$age['Ben'] = "37";
$age['Joe'] = "43";
```

```
<?php
$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");

foreach($age as $x => $x_value) {
   echo "Key=" . $x . ", Value=" . $x_value;
   echo "<br>";
}
?>
```

Output:

```
Key=Peter, Value=35
Key=Ben, Value=37
Key=Joe, Value=43
```

Question 5

Q5a. Explain following PHP, MYSQL functions:

i. mysql_connect ii. mysql_close iii. mysql_query iv. mysql_select_db v. mysql_error()

Ans:

i. mysql_connect:

PHP provides **mysql_connect()** function to open a database connection. This function takes five parameters and returns a MySQL link identifier on success or FALSE on failure.

```
mysql_connect('localhost', 'mysql_user', 'mysql_password')
```

ii. mysql_close:

You can disconnect from the MySQL database anytime using another PHP function **mysql_close()**. This function takes a single parameter, which is a connection returned by the **mysql_connect()** function.

mysql_close(connection) -   closes connection

iii. mysql_query:

To create new table in any existing database you would need to use PHP function **mysql_query()**. You will pass its second argument with a proper SQL command to create a table.

mysql_query(query, connection) - Query handle for successful CREATE,
SELECT, SHOW, DESCRIBE, EXPLAIN and other statements or
FALSE on error.

iv. mysql_select_db:

PHP provides function **mysql_select_db** to select a database. It returns TRUE on success or FALSE on failure.

mysql_select_db(database,connection) – selects the database

v. mysql_error():Returns the text of the error message from previous MySQL operation

Q5b. Write a PHP program to demonstrate the use of cookies in PHP.

Ans:

A cookie is often used to identify a user. A cookie is a small file that the server embeds on the user's computer. Each time the same computer requests a page with a browser, it will send the cookie too. With PHP, you can both create and retrieve cookie values.

Create Cookies With PHP

A cookie is created with the setcookie() function.

Syntax

setcookie(name, value, expire, path, domain, secure, httponly);

Only the name parameter is required. All other parameters are optional.

```
<?php

$cookie_name = "user";

$cookie_value = "Pallavi Tawde";

setcookie($cookie_name, $cookie_value, time() + (86400 * 30), "/"); // 86400 = 1 day

?>

<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>
```

Q5c. Compare POSIX and PERL style Regular expressions of PHP.
Ans:

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

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

PHP offers functions specific to two sets of regular expression functions, each corresponding to a certain type of regular expression. You can use any of them based on your comfort.

- POSIX Regular Expressions
- PERL Style Regular Expressions

PHP's Regexp POSIX Functions

PHP currently offers seven functions for searching strings using POSIX-style regular expressions −

Sr.No	Function & Description
1	ereg() The ereg() function searches a string specified by string for a string specified by pattern, returning true if the pattern is found, and false otherwise.
2	ereg_replace()

	The ereg_replace() function searches for string specified by pattern and replaces pattern with replacement if found.

PHP's Regexp PERL Compatible Functions

PHP offers following functions for searching strings using Perl-compatible regular expressions −

Sr.No	Function & Description
1	preg_match() The preg_match() function searches string for pattern, returning true if pattern exists, and false otherwise.
2	preg_match_all() The preg_match_all() function matches all occurrences of pattern in string.

Q5d. List various HTTP functions available in PHP. Explain header(function in details.

Ans:

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

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

The header() function sends a raw HTTP header to a client.

It is important to notice that header() must be called before any actual output is sent (In PHP 4 and later, you can use output buffering to solve this problem):

```
<html>
<?php
// This results in an error.
// The output above is before the header() call
header('Location: http://www.example.com/');
?>
```

Q5e. Write a PHP program to create a database named “Employee”. Create a table named “Salary” with fields (eid, ename, esalary). Insert 3 records of your choice. Display the names of the Employees whose salary is between 15000 and 20000 in a tabular format.

Ans:

```
<?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("College",$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(101,'pallavi',11111)";

 if(mysql_query($query1,$con))

  echo"Record 1 inserted successfully";

 else

  echo "Error inserting record 1:".mysql_error();
 $query2="insert into Salary values(102,'Dev',22222)";
 if(mysql_query($query2,$con))
  echo"Record 2 inserted successfully";
 else   echo "Error inserting record 2:".mysql_error();
 $query3="insert into Salary values(103,'Anita',33333)";
 if(mysql_query($query3,$con))
  echo"Record 3 inserted successfully";
 else
  echo "Error inserting record 3:".mysql_error();
 $sql="select * from Salary where esalary >=15000 and esalary <= and 20000
";
 $result=mysql_query($sql,$con);
 if(mysql_num_rows($result)>0)
 {
   echo "<table border='1'><tr><th>ID</th><th>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);

?
```

Q5f. Write a short note on PHP Session.

Ans:

Sessions

- A session is a way to store information (in variables) to be used across multiple pages.
- Unlike a cookie, the information is not stored on the user's computer.
- Session variables hold information about one single user, and are available to all pages in one application.
- A session creates a file in a temporary directory on the server.
- PHP first creates a unique ID for each session which is a random string of 32 hexadecimal numbers.
- A cookie called PHPSESSID is automatically sent to the user's computer to store unique session ID.
- Information stored in session variables will be deleted once the browser is closed.

Starting a PHP Session

A PHP session is easily started by making a call to the **session_start()** function.This function first checks if a session is already started and if none is started then it starts one. It is recommended to put the call to **session_start()** at the beginning of the page.

Session variables are stored in associative array called **$_SESSION[]**. These variables can be accessed during lifetime of a session.
Make use of **isset()** function to check if session variable is already set or not.

Destroying a PHP Session

A PHP session can be destroyed by **session_destroy()** function. This function does not need any argument and a single call can destroy all the session variables. If you want to destroy a single session variable then you can use **unset()** function to unset a session variable.

Example:

```
<?php
   session_start();

   if( isset( $_SESSION['counter'] ) ) {
      $_SESSION['counter'] += 1;
   }else {
      $_SESSION['counter'] = 1;
   }

   $msg = "You have visited this page ".  $_SESSION['counter'];
   $msg .= "in this session.";
?>

<html>   <head>        <title>Setting up a PHP session</title>     </head>
      <body>
      <?php  echo ( $msg ); ?>
   </body>

</html>
```