

Q.1 Attempt the following (any THREE) [15]

Q.1(a) What is CSS? Explain its types. [5]

Ans.: Cascading Style Sheets (CSS) provide easy and effective alternatives to specify various attributes for the HTML tags.

External Style Sheet

If you need to use your style sheet to various pages, then its always recommended to define a common style sheet in a separate file.

A CSS file will have extension as .css and it will be included in HTML files using <link> tag.

Example

```
.red{
    color: red;
}
.thick{
    font-size:20px;
}
.green{
    color:green;
}
```

Save this file as style.css

```
<!DOCTYPE html>
<html>
<head>
<title>HTML External CSS</title>
<link rel="stylesheet" type="text/css" href="/html/style.css">
</head>
<body>
<p class="red">This is red</p>

<p class="thick">This is thick</p>

<p class="green">This is green</p>

<p class="thick green">This is thick and green</p>
</body>
</html>
```

Internal Style Sheet

If you want to apply Style Sheet rules to a single document only then you can include those rules in header section using <style> tag.

```
<html>
<head>
<title>HTML Internal CSS</title>
<style type="text/css">
.red{
    color: red;
```

```

}
.thick{
    font-size:20px;
}
.green{
    color:green;
}
</style>
</head>
<body>
<p class="red">This is red</p>

<p class="thick">This is thick</p>

<p class="green">This is green</p>

<p class="thick green">This is thick and green</p>
</body>
</html>

```

Inline Style Sheet

You can apply style sheet rules directly to any HTML element using style attribute of the relevant tag. This should be done only when you are interested to make a particular change in one HTML element only.

```

<html>
<head>
    <title>HTML Inline CSS</title>
</head>
<body>
    <p style="color:red;">This is red</p>
    <p style="font-size:30px;">This is thick</p>
    <p style="color:green;">This is green</p>
    <p style="color:green;font-size:20px;">This is thick and green</p>
</body>
</html>

```

Q.1(b) What is internet? What are its applications?

[5]

Ans.: Internet is an acronym of interconnected networks. Internet is a means of connecting a computer to any other computer anywhere in the world via dedicated routers and servers. When two computers are connected over the internet they can send and receive all list of information such as text, graphics, audio, video and computer programs. The internet refers to a global network of information that is transmitted to billions of computers and devices.

Application of Internet:

- Email: Electronic mail is a service which allows us to send the message in electronic mode over the internet. It is defined as the transmission of message over communication networks. Earlier email systems required the author and recipient both to be online of the same time in common with instant messaging. Today's email systems are based on a store and forward model. Email consist of header from date, to, subject, CC, BCC, greeting message body and signature.
- E-business: It referes to online transctions including interactions with business partners customers and vendors . Its main aim is to improve business processs and efficiency higher and productivity and cost savings. Key elements of E-business model are value proposition revenue model market strategy etc.

Q.1 (c) Explain any 2 web browsers.**[5]**

- Ans.:**
- A Web Browser is a client-side software application used for retrieving, displaying and navigating through the online resources available on the World Wide Web.
 - Web browser is used to access information from the web servers in network or from files in file systems. Browsers may also be used to save information into the file systems.
 - The browser identifies the URL and fetches the requested information or resource to the user.

Common functions offered by all major browsers include,

- The user enters the URL of the resource to be fetched on the address bar of the browser.
- The Prefix of the URL explains the method to interpret the URL. Most of the URLs enable the resources to be retrieved Hypertext Transfer Protocol while some web browsers also support prefixes like FTP i.e. File Transfer Protocol.
- HTML script is passed 'to the browser for interpretation. Interactive script may be written using a scripting language like Java Script or VB Script.
- The browser interprets text, images, video and interactive scripts .
- The browsers allow users to open multiple tabs or windows for accessing multiple information resources at the same time. Browsers may include popup blockers which do not allow windows to open without user's permission.
- All browsers provide backward and forward button to navigate to the previous and next resource.
- History of the browser keeps a list of previously visited web resources.
- Refresh or reload help to reload the current document in the browser.
- Stop button can be used to cancel the action on loading the resource.
- Home button enables the user to return to the website's home page.
- Search bar is used to allow user to input terms into the search engine.
- Status bar displays the progress of loading the current resource and also the URL of links when the cursor is moved over them.

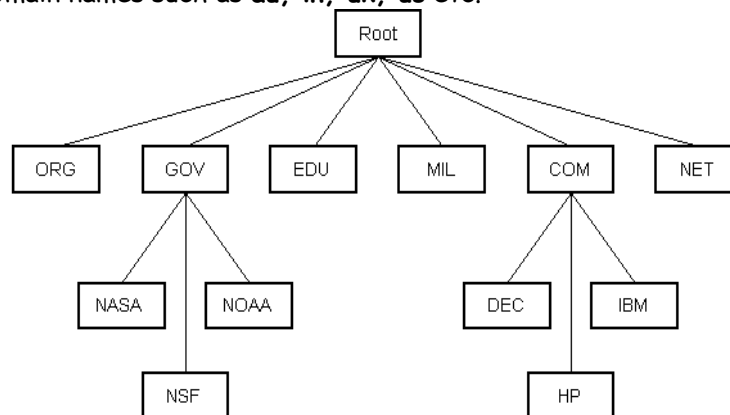
Q.1 (d) Explain Domain Name Server.**[5]**

- Ans.:** It is easier to remember names i.e. URL rather than IP addresses. For eg. to communicate with web server www.google.co.in computers needs IP address of the web server. The DNS are used to translate a URL's into numeric addresses. It converts a computer's host name into IP address on the internet.

No DNS server contains the entire database, they only contain a subset of it.

There are 2 types of domain names:

1. Generic such as **com, edu, gov, net** etc
2. Country level domain names such as **au, in, uk, us** etc.

**Types Of Name Servers****i) Root Server**

It is the top level server which consists of the entire DNS tree.

ii) Primary Servers

It stores a file about its zone. It has authority to create, maintain and update the zone file.

iii) Secondary Server

It transfers complete information about a zone from another server which may be primary or secondary server.

e.g. lets consider domain name as www.mu.ac.in

First it search country name server in i.e. india.

Then it identifies ac i.e. academic and finally mu i.e. Mumbai university.

Q.1(e) What are the types of list?

[5]

Ans.: Types of List

Unordered List

It starts with the tag. Each list item starts with the tag.

Use type property to define the style:

type=disc	Sets the list item marker to a bullet
type=circle	Sets the list item marker to a circle
type=square	Sets the list item marker to a square
type=none	The list items will not be marked

e.g.

```
<ul type=disc>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

Ordered List

It starts with the tag. Each list item starts with the tag.

type attribute defines the type of the list items:

type="1"	numbered with numbers (default)
type="A"	numbered with uppercase letters
type="a"	numbered with lowercase letters
type="I"	numbered with uppercase roman numbers
type="i"	numbered with lowercase roman numbers

start Attribute

specify the starting point of numbering.

e.g.

```
<ol type="1">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

Definition list

It arranges items in the same way as they are arranged in a dictionary. It uses following three tags.

- <dl> - Defines the start of the list
- <dt> - Defines term
- <dd> - Defines description

```
e.g.
<dl>
  <dt>College</dt>
  <dd>A place where students study</dd>
</dl>
```

Q.1(f) Explain text formatting tags.

[5]

Ans.: Text Formatting Tags:

- - It display bold text
- - It display important text. It is similar to
- <i> - It display italic text
- - It display emphasized text. It is similar to <i>
- <mark> - It display marked/highlighted text
- <big>: It display text in large font size
- <small> - It display text in small font size
- - It display deleted text
- <ins> - It display inserted text
- <sub> - It display Subscript text
- <sup> - It display Superscript text
- <u>- It display underline text
- <strike>: It display strike through text
- <tt>: It display teletype text

Example

```
<html>
  <head>
    <title>Text Formatting Tags</title>
  </head>
  <body>
    <b>bold</b><br>
    <strong>strong</strong><br>
    <i>Italics</i><br>
    <em>Emphasize</em><br>
    <tt>Teletype text</tt><br>
    H<sub>2</sub>O<br>
    2<sup>2</sup>=4<br>
    Dozen contains <del>13</del><ins>12</ins>items<br>
    <small>Small</small><br>
    <big>Big</big><br>
    <strike>This text is strike</strike><br>
    <mark>Mark</mark>
  </body>
</html>
```

Q.2 Attempt the following (any THREE)

[15]

Q.2(a) Explain HTML5 semantic tags.

[5]

Ans.: HTML5 offers new semantic tags to define different parts of a web page:

- 1) <header> : The <header> element specifies a header for a document or section. The <header> element should be used as a container for introductory content.

Example:

```
<header> <h1>What Does WWF Do?</h1> <p>WWF's mission:</p> </header>
```

- 2) <nav> : The <nav> element defines a set of navigation links.

Example:

```
<nav> <a href="/html/">HTML</a>
```

```
<a href="/css/">Cascaded Style Sheet</a>
<a href="/js/">JavaScript</a>
<a href="/jquery/">Active Server Pages</a> </nav>
```

- 3) **<section>** : The `<section>` element defines a section in a document. A section is a thematic grouping of content, typically with a heading."

Example:

```
<section> <h1>WWW</h1> <p>The World Wide Web</p> </section>
```

- 4) **<article>** : The `<article>` element specifies independent, self-contained content. `<article>` element can be used for Forum post, Blog post, Newspaper article etc.

Example:

```
<article> <h1>What Does WWF Do?</h1> <p>WWF's mission is to stop the degradation of our planet's natural environment, and build a future in which humans live in harmony with nature.</p> </article>
```

- 5) **<aside>** : The `<aside>` element defines some content aside from the content it is placed in (like a sidebar).

Example:

```
<aside> <h4>Epcot Center</h4> <p>The Epcot Center is a theme park in Disney World, Florida.</p> </aside>
```

- 6) **<footer>** : The `<footer>` element specifies a footer for a document or section.

Example:

```
<footer> <p>Posted by: Hege Refsnes</p> <p>Contact information:
<a href="mailto:profmanojkavedia@gmail.com"> Contact Me</a>.</p> </footer>
```

Q.2(b) What is image map? what are its types?

[5]

Ans.: We can make an image link by `` tag. In cases where we want a particular part of an image to act like a hyperlink then this can be done with image mapping such active area is called as Hotspot.

Server side image map:

- This map attribute specifies that the image is part of a server side image (map can image map is an image with click able areas).
- When clicking on a server-side image-map the click co-ordinates are sent to the server as a URL query string.
- It is best to use server side image map whenever the map has many years or where the areas are not defined by simple shapes or circle, polygons and rectangles.

Eg. `<html>`

```
<body>
```

```
<a href = "serverside map. html">
```

```
<img src = "c:\desert.jpg" alt = "desert" is map = "is map"/>
```

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

On clicking this image, server side map page will open. In this page image, clicking on the specific area based on shape will open different links.

Q.2(c) Explain types of navigation bar with example.

[5]

Ans.: HTML 5 includes a `<nav>` tag a two sided container tag in which you can optionally 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.

If browser does not support the `<nav>` tag it is ignored. You are free to define style attributes for `<nav>` tag in internal or external style sheets. This can be away to help ensure consistency among the navigation bars through out all the pages in your web site.

Eg. To add a text based navigation bar to the top and bottom of a web page.

- (i) At the first `<hr>` tag add a `<nav>` container and the text for a navigation bar.

```
<nav> <hr> <p> about us </p> </nav>
```

- (ii) Same the file and refresh internet explorer. The text of the intended navigation bar appears, but the items are not clearly separated.

Q.2(d) How to embed audio and video onto web page?

[5]

Ans.: HTML 5 Audio Tag

With the introduction of HTML5, audio tag has spread very quickly and it is very well supported by current browsers. Audio tag is used to add sound and music in the html page. At the moment the audio tag, supports 3 types of audio files:

- mp3 - MIME-type audio/mpeg
- wav - MIME-type audio/wav
- ogg - MIME-type audio/ogg

A video file can be inserted in a html page in two ways. The first way would be using the <embed/> tag. This tag does not need an ending/finishing tag. It works, mostly, the same as a tag for a photo. <embed src="example.mpeg" autostart="false" height="30" width="144" />

Also, you can insert a video file using a link.

film name

Supported extensions for the embed tag

These are:

- .swf - made by Macromedia Flash
- .wmv - Microsoft Windows Media Video
- .mov - Quick Time Movie, belongs to Apple
- .mpeg - created by Moving Pictures Expert Group.

The most used are .mpeg and .swf, because of the compact format.

Attributes of the embed tag :

- autostart - sets whether the file will run automatically after the page is loaded. Can have the true or false value
- hidden - sets whether the buttons are hidden or not. Can have the true or false value.
- volume - can have any value from 0 to 100
- loop- sets whether the file will be replayed after it is finished. Can have the true or false value.
- playcount - this establishes how many times the file will be replayed. For example playcount="2" means it will be replayed two times and after it will stop.

Example of Embedding Audio and Video in Web Page

```
<!DOCTYPE html>
<html>
<head>
<title> Embedding an Audio and Video </title>
</head>
<body>
Video:
<br/>
<video width="400" controls>
<source src="myvideo.mp4" type="video/mp4">
<source src="myvideo.ogg" type="video/ogg">
```

Your browser does not support HTML5 video.

```
</video>
<br/>
```

Audio:

```
<br/>
<audio controls>
```

```
<source src="myaudio.ogg" type="audio/ogg">
<source src="myaudio.mp3" type="audio/mpeg">
```

Your browser does not support the audio element.

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

Q.2(e) Write HTML code to display form for email registration.

[5]

Ans. :

```
<html>
<body>
<form name=form1 method=post action=register.php onsubmit="return validate()">
  <pre>
    Userid:    <input type=text size=20 maxlength=20 name=fname>
    Age:      <input type=text size=2 maxlength=2 name=age>

    alternate email:<input type = text name=email>

    password:  <input type=password name=pass1 >

    Retype Password:<input type=password name=pass2>

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

  </pre>
</form>
</body>
</html>
```

Q.2(f) Explain rowspan, colspan, cellpadding and cellspacing with example.

[5]

Ans. : Rowspan and Colspan

Rowspan and Colspan Attributes : By using ROWSPAN and COLSPAN attributes to the <TH> and <TD> tags, it is possible to create data cells that span a given numbers of rows and columns. The basic idea of the ROWSPAN and COLSPAN attribute for <TD> and <TH> is to extend the size of the cells across two or more rows or columns respectively.

To set a cell to span two rows, use <TD ROWSPAN="2"> and to set a cell to span two columns, use <TD COLSPAN="2">

- You will use colspan attribute if you want to merge two or more columns into a single column.
- you will use rowspan if you want to merge two or more rows.

Example 1 : Table with Rowspan

```
<!DOCTYPE html>
<html>
<head>
<title>HTML table Tag with Rowspan</title>
</head>
<body>
<body>
<table border = "1">
<tr>
<th>Column 1</th>
<th>Column 2</th>
<th>Column 3</th>
```



```

</tr>
<tr>
<td rowspan = "2">Row 1 Cell 1</td>
<td>Row 1 Cell 2</td>
<td>Row 1 Cell 3</td>
</tr>
<tr>
<td>Row 2 Cell 2</td>
<td>Row 2 Cell 3</td>
</tr>
<tr>
<td>Row 3 Cell 1</td>
    <td>Row 3 Cell 2</td>
<td>Row 3 Cell 3</td>
</tr>
</table>
</body>
</html>

```

Output :

Column 1	Column 2	Column 3
Row 1 Cell 1	Row 1 Cell 2	Row 1 Cell 3
	Row 2 Cell 2	Row 2 Cell 3
Row 3 Cell 1	Row 3 Cell 2	Row 3 Cell 3

Example 2 : Table with Colspan

```

<!DOCTYPE html>
<html>
<head>
<title>HTML table Tag with colspan</title>
</head>
<body>
<body>
<body>
<table border = "1">
<tr>
<th>Column 1</th>
<th>Column 2</th>
<th>Column 3</th>
</tr>
<tr>
<td colspan = "2">Row 1 Cell 1</td>
<td>Row 1 Cell 2</td>
</tr>
<tr>
<td>Row 2 Cell 1</td>
<td>Row 2 Cell 2</td>
    <td>Row 2 Cell 3</td>
</tr>
<tr>
<td>Row 3 Cell 1</td>

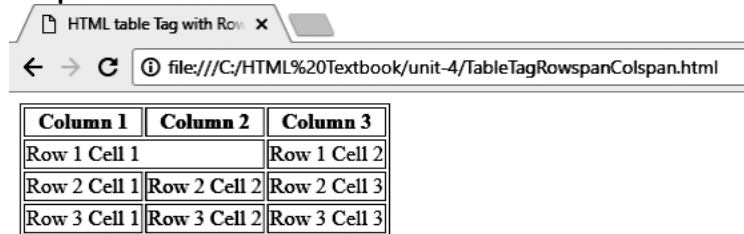
```

```

        <td>Row 3 Cell 2</td>
<td>Row 3 Cell 3</td>
</tr>
</table>
</body>
</html>

```

Output :



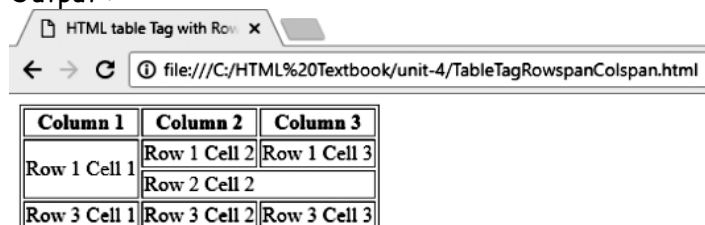
Example 3 : Table with Rowspan and Colspan

```

<!DOCTYPE html>
<html>
<head>
<title>HTML table Tag with Rowspan and colspan</title>
</head>
<body>
<body>
<body>
<table border = "1">
<tr>
<th>Column 1</th>
<th>Column 2</th>
<th>Column 3</th>
</tr>
<tr>
<td rowspan = "2">Row 1 Cell 1</td>
<td>Row 1 Cell 2</td>
        <td>Row 1 Cell 3</td>
</tr>
<tr>
<td colspan = "2">Row 2 Cell 2</td>
</tr>
<tr>
<td>Row 3 Cell 1</td>
        <td>Row 3 Cell 2</td>
<td>Row 3 Cell 3</td>
</tr>
</table>
</body>
</html>

```

Output :



Cellspacing and Cellpadding :

Cellspacing Attribute : This attribute sets the amount of the space between individual cells and between cells and outside borders of the table. The value for this attribute is expressed in pixels, as a whole number. The default value may vary according to the browsers. A value of 0 turns off the cell spacing.

Cellpadding Attribute : This attribute sets the amount of the space between the edge of the cell and its contents. The value for this attribute is expressed in pixels, as a whole number. The default value may vary according to the browsers. A value of 0 turns off the cell padding.

Note : Following code gives the most compact table possible.

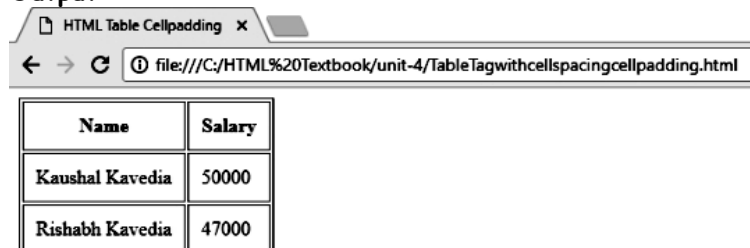
```
<TABLE BORDER="0" CELSPACING="0" CELLPADDING="0">
```

Example 1 : Table with cell padding cellpadding attribute represents the distance between cell borders and the content within a cell

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Table Cellpadding </title>
</head>

<body>
<table border = "1" cellpadding = "10">
<tr>
<th>Name</th>
<th>Salary</th>
</tr>
<tr>
<td>Kaushal Kavedia</td>
<td>50000</td>
</tr>
<tr>
<td>Rishabh Kavedia</td>
<td>47000</td>
</tr>
</table>
</body>
</html>
</body>
```

Output :



Name	Salary
Kaushal Kavedia	50000
Rishabh Kavedia	47000

Example 2 : Table with cell spacing

The cellspacing attribute defines space between table cells

```
<!DOCTYPE html>
<html>
<head>
```

```

<title>HTML Table Cellspacing </title>
</head>
<body>
<table border = "1" cellspacing = "10">
<tr>
<th>Name</th>
<th>Salary</th>
</tr>
<tr>
<td>Kaushal Kavedia</td>
<td>50000</td>
</tr>
<tr>
<td>Rishabh Kavedia</td>
<td>47000</td>
</tr>
</table>
</body>
</html>
</body>

```

Output :



Example 3 : Table with cellspacing , cellpadding and border

```

<!DOCTYPE html>
<html>
<head>
<title>HTML Table Cellspacing </title>
</head>
<body>
<table border = "5" cellpadding="10" cellspacing = "10">
<tr>
<th>Name</th>
<th>Salary</th>
</tr>
<tr>
<td>Kaushal Kavedia</td>
<td>50000</td>
</tr>
<tr>
<td>Rishabh Kavedia</td>
<td>47000</td>
</tr>
</table>
</body>
</html>
</body>

```

Output :

Name	Salary
Kaushal Kavedia	50000
Rishabh Kavedia	47000

Q.3 Attempt the following (any THREE)

[15]

Q.3(a) Explain void, new, this, comma and delete operators in Javascript.

[5]

Ans.: (i) **new** : The new operator is used to Create a new instance of the object.

Example:

```
var x= new Fn( )
```

In the above statement x becomes new instance of function Fn and gets all its properties and methods.

(ii) **delete** : The delete operator is deletes the properties from objects and array elements from arrays making them undefined.

Example:

```
delete myobj;
```

```
delete myarray[3];
```

(iii) **this** : this operator is used to refer to the current object.

Example:

```
this.window
```

```
this.value
```

(iv) **void** : The void operator evaluates expression and returns undefined.

Example :

```
void expr
```

(v) **,(Comma)-Comma operator** executes all expressions from left to right. It evaluates both the operands from left to right and returns the result.

Example x=4, y= 5;

Q.3(b) Differentiate between client side and server side Javascript.

[5]

Ans.: **Server side scripting** is when the users browser makes a request back to the server and the script creates the dynamic page based a number of conditions. Languages usually used are PHP or ASP.

Example: On a website and the user clicks on the show the products link. The server then runs the PHP/ASP code to create a page by finding all the products in the data base. The page is created and sent back to the users browser for viewing.

Client side scripting is when all the code is already on the users browser and the page alters based on the users input. Language is usually JavaScript.

Example: A JavaScript drop down menu. The JavaScript alters how the menu is displayed but all the code is run locally while it is doing it.

Client side JavaScript:

1. Client side JavaScript runs on your computer after you load a web page.
2. The script should be included in or referenced by an HTML document for the code to be interpreted by the browser.

3. It means a web page need not be a static HTML, but can include code that interact with the user, controls the browser and dynamically create HTML contents.
4. The JavaScript Code is executed when the user submits the form, and only if all the entries are valid, they would be submitted to the web server.
5. It enables web pages on browsers to run active online content.
6. It uses <SCRIPT>tag in HTML document.
7. This script is interpreted by the browser at run time.

Server side JavaScript:

1. Server side JavaScript refers to JavaScript that runs on the server side and is therefore not downloaded to the browser.
2. There are many languages that support Server side scripts like ASP, JSP etc.
3. The advantage of the Server side JavaScript is that the code can validate data on both the client for immediate user validations and on the server for security.
4. It enables back-end access to databases, file systems and servers.
5. It uses <SERVER>tag in HTML document.
6. The Server side JavaScript is deployed only after compilation

Client-Side Scripting	Server side Scripting
Scripts that execute in client side. In context of websites, it is scripts that execute in the browser of the user. Eg: Javascript, VB etc.	Scripts that execute in the Server. In context of website, it is scripts that execute on application servers. Eg: PHP, Python, Ruby etc
Interact with temporary storage	Interaction with servers/storages and with databases
Sends requests to the server and Retrieval of data from Server	It processes the user input and Displays the requested pages
Provides remote access for client server program	Encoding of data into HTML
Not database require	Querying the database and Operations over databases like delete, update.
here are many client-side scripting languages too. 1. JavaScript 2. VBScript 3. HTML (Structure) 4. CSS (Designing) 5. AJAXjQuery etc.	There are several languages that can be used for server-side programming: 1. PHPASP.NET (C# OR Visual Basic) 2. C++ 3. Java 4. JSP 5. PythonRuby on Rails and so on.

Q.3(c) Write a program to display prime numbers between 1 to 100.

[5]

Ans. : <!DOCTYPE html>

```

<html>
<head>
<title>Generation of Prime number between 1 and 100 </title>
</head>
<body >
<H1> Prime Number Between 1 to 100 </h1>
<script>
for(var i=2;i<=100;i++)
{
var flag=0;
for(var j=2;j<=i/2;j++)
{
if(i%j==0)

```

```

{
flag=1;
break;
}
}
if(flag==0)
{
document.write(i+"<br>");
}
}
</script>
</body>
</html>

```

Q.3(d) What is event and event handler? Explain Mouse and keyboard event handlers.

[5]

Ans. : An HTML event can be something the browser does, or something a user does.

Here are some examples of HTML events:

- An HTML web page has finished loading
- An HTML input field was changed
- An HTML button was clicked

Often, when events happen, you may want to do something.

JavaScript lets you execute code when events are detected.

Events are actions that can be detected by JavaScript. Every element on a web page has certain events which can trigger a JavaScript. For example, we can use the onClick event of a button element to indicate that a function will run when a user clicks on the button.

Examples of events:

- A mouse click
- A web page or an image loading
- Mousing over a hot spot on the web page
- Selecting an input field in an HTML form
- Submitting an HTML form
- A keystroke

Events are normally used in combination with functions, and the function will not be executed before the event occurs.

Event handlers are JavaScript code that are not added inside the <script> tags, but rather, inside the html tags, that execute JavaScript when something happens, such as pressing a button, moving your mouse over a link, submitting a form etc.

The basic syntax of these event handlers is:

```
name_of_handler="JavaScript code here"
```

Example:

```
<a href="http://google.com" onClick="alert('hello!')">Google</a>
```

When events are associated with functions, the functions are written in the head section within the <script> tag and are called from the event handlers.

List various types of The event occurs when...

mouse event: Events

onmousedown	A mouse button is pressed
onmousemove	The mouse is moved
onmouseout	The mouse is moved off an element
onmouseover	The mouse is moved over an element
onmouseup	A mouse button is released

Examples:

```
< h1 onmouseover= "style.color= 'RED'" onmouseout= "style.color= 'RED'">Mouse over this text</h1>
<button onclick="displayDate()">The time is?</button>
```

Example with Complete code

```
<html>
  <head>
    <script type="text/javascript">
      <!--
        function sayHelloWorld() {
          alert("Hello World")
        }
      //-->
    </script>
  </head>
  <body>
    <p>Click the following button and Observe the Changes Happening</p>
    <form>
      <input type="button" onclick="sayHelloworld()" value="Say Hello" />
    </form>
  </body></html>
```

What can JavaScript Do?

Event handlers can be used to handle, and verify, user input, user actions, and browser actions:

- Things that should be done every time a page loads
- Things that should be done when the page is closed
- Action that should be performed when a user clicks a button
- Content that should be verified when a user inputs data
- And more ...

Many different methods can be used to let JavaScript work with events:

- HTML event attributes can execute JavaScript code directly
- HTML event attributes can call JavaScript functions
- You can assign your own event handler functions to HTML elements
- You can prevent events from being sent or being handled
- And more ...

Q.3(e) Explain document object properties and methods.

[5]

Ans. : Document object

Properties -

- **cookie**
Returns all name/value pairs of cookies in the document.
- **documentMode**
Returns the mode used by the browser to render the document.
- **domain**
Returns the domain name of the server that loaded the document.

- **lastModified**
Returns the date and time the document was last modified.
- **readyState**
Returns the (loading) status of the document.
- **referrer**
Returns the URL of the document that loaded the current document.
- **title**
Sets or returns the title of the document.
- **URL**
Returns the full URL of the document.

Methods

- **close()**
Closes the output stream previously opened with document.open()
- **getElementById()**
Accesses the first element with the specified id.
- **getElementsByName()**
Accesses all elements with a specified name.
- **getElementsByTagName()**
Accesses all elements with a specified tagname
- **open()**
Opens an output stream to collect the output from document.write() or document.writeln()
- **write()**
Writes HTML expressions or JavaScript code to a document.
- **writeln()**
Same as write(), but adds a newline character after each statement.

Q.3(f) Explain date object with its methods.

[5]

Ans.: **Date Object**

It is used to work with dates and times.

Date objects are created with new Date().

There are four ways to create date object:

```
var d = new Date();
```

```
var d = new Date(milliseconds);
```

```
var d = new Date(dateString);
```

```
var d = new Date(year, month, day, hours, minutes, seconds, milliseconds);
```

Methods

Method	Description
getDate()	Returns the day of the month (from 1-31)
getDay()	Returns the day of the week (from 0-6)
getFullYear()	Returns the year
getHours()	Returns the hour (from 0-23)
getMilliseconds()	Returns the milliseconds (from 0-999)
getMinutes()	Returns the minutes (from 0-59)
getMonth()	Returns the month (from 0-11)
getSeconds()	Returns the seconds (from 0-59)
getTime()	Returns the number of milliseconds since midnight Jan 1 1970, and a specified date
setDate()	Sets the day of the month of a date object
setFullYear()	Sets the year of a date object
setHours()	Sets the hour of a date object
setMilliseconds()	Sets the milliseconds of a date object
setMinutes()	Set the minutes of a date object

setMonth()	Sets the month of a date object
setSeconds()	Sets the seconds of a date object
setTime()	Sets a date to a specified number of milliseconds after/before January 1, 1970
toDateString()	Converts the date into a string

Q.4 Attempt the following (any THREE)

[15]

Q.4(a) Write a program in PHP to accept a number from user and display its factorial.

[5]

Ans.: <!DOCTYPE html>

```
<html>
  <head>
    <title>Factorial</title>
  </head>
  <body>
    <script>

      var i,f=1;
      var no=prompt("Enter no","");
      for(i=2;i<=no;i++)
        f=f*i;
      alert("Factorial is:"+f);

    </script>

  </body>
</html>
```

Q.4(b) Differentiate between get and post method.

[5]

Ans.: There are two ways the browser client can send information to the web server.

- The GET Method
- The POST Method

Before the browser sends the information, it encodes it using a scheme called URL encoding. In this scheme, name/value pairs are joined with equal signs and different pairs are separated by the ampersand.

```
name1=value1&name2=value2&name3=value3
```

The GET Method :

The GET method sends the encoded user information appended to the page request. The page and the encoded information are separated by the ? character.

```
http://www.test.com/index.htm?name1=value1&name2=value2
```

- The GET method produces a long string that appears in your server logs, in the browser's Location: box.
- The GET method is restricted to send upto 1024 characters only.
- Never use GET method if you have password or other sensitive information to be sent to the server.
- GET can't be used to send binary data, like images or word documents, to the server.
- The data sent by GET method can be accessed using QUERY_STRING environment variable.
- The PHP provides \$_GET associative array to access all the sent information using GET method.

Example:

```
<html>
<body> <form action="welcome_get.php" method="get">
<b>Name:</b> <input type="text" name="name"><br>
<b>E-mail:</b> <input type="text" name="email"><br>
<input type="submit" value="submit">
</form>
</body>
</html>
"welcome_get.php":
<html> <body> Welcome <?php echo $_GET["name"]; ?><br>
```

```
Email address is: <?php echo $_GET["email"]; ?>
</body>
</html>
```

The POST Method :

The POST method transfers information via HTTP headers. The information is encoded as described in case of GET method and put into a header called QUERY_STRING.

- The POST method does not have any restriction on data size to be sent.
- The POST method can be used to send ASCII as well as binary data.
- The data sent by POST method goes through HTTP header so security depends on HTTP protocol. By using Secure HTTP you can make sure that your information is secure.
- The PHP provides \$_POST associative array to access all the sent information using POST method.

Example:

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

Q.4(c) Explain PHP error handling. [5]

Ans. : The default error handling in PHP is very simple. An error message with filename, line number and a message describing the error is sent to the browser. When creating scripts and web applications, error handling is an important part. If your code lacks error checking code, the program may look very unprofessional and may be open to security risks.

Different error handling methods:

- Simple "die()" statements
- Custom errors and error triggers
- Error reporting

Basic Error Handling: Using the die() function

The first example shows a simple script that opens a text file:

```
<?php
$file=fopen("welcome.txt","r");
?>
```

If the file does not exist and get an error message

```
<?php
if(!file_exists("welcome.txt"))
{
die("File not found");
}
else
{
$file=fopen("welcome.txt","r");
}
?>
```

Now if the file does not exist you get an error like this :
File not found

Q.4(d) Explain superglobal arrays in PHP.

[5]

Ans. : Superglobal Arrays in PHP

1. \$GLOBALS

It is used to access global variables from anywhere in the script.

It can be accessed by using \$GLOBALS[name] where name be the name of variable.

e.g.

```
<?php
$x = 75;
$y = 25;
function addition() {
$GLOBALS['z'] = $GLOBALS['x'] + $GLOBALS['y'];
}
addition();
echo $z;
?>
```

2. \$_SERVER

\$_SERVER holds information about server properties like server name, software, protocol, request method, etc.

```
<?php
echo $_SERVER['SERVER_NAME'];
echo $_SERVER['REQUEST_METHOD'];

echo $_SERVER['SERVER_SOFTWARE'];
echo $_SERVER['SERVER_PROTOCOL'];
?>
```

3. \$_REQUEST

It is used to collect data after submitting HTML form.

```
<?php
$name = $_REQUEST['fname'];
if (empty($name))
    echo "Pls enter name";
```

```

else
    echo $name;
?>

```

4. \$_POST

It is used to collect form data after submitting HTML form with method="post".

```

<?php
    $name = $_POST['fname'];
    if (empty($name))
        echo "Pls enter name";
    else
        echo $name;
?>

```

5. \$_GET

It is used to collect form data after submitting HTML form with method="get".

```

<?php
echo "I like Subject:" . $_GET['subject'];
?>

```

6. \$_COOKIE

A cookie is often used to identify a user. A cookie is a small file that the server embeds on the user's computer.

```

<?php
echo "Cookie named " . $_COOKIE['name'];
echo "Value is: " . $_COOKIE[$_COOKIE['name']];
?>

```

7. \$_SESSION

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

```

<?php

$_SESSION["favcolor"] = "green";
$_SESSION["favanimal"] = "cat";
?>

```

Q.4(e) Write a PHP program from user to accept 2 numbers from user and display greater number. [5]

Ans.:

```

greater.html
<html>
    <head>
        <title>Greater number</title>

    </head>
    <body>
        <form name=f method=get action=greater.php>
            Enter no1:<input type=number name=n1><br>
            Enter no2:<input type=number name=n2><br>
            <input type=submit>
        </form>
    </body>
</html>
greater.php
<?php

```

```

$no1=$_GET["n1"];
$no2=$_GET["n2"];

if($no1 == $no2)
    echo "Equal";
elseif($no1 < $no2)
    echo "No2 is greater";
else
    echo "No1 is greater";
?>

```

Q.4(f) Explain String functions in PHP.

[5]

Ans. : A string can be any text inside double or single quotes:

```
var carname = "BMW";
```

Methods

Method	Description
charAt()	Returns the character at the specified index
charCodeAt()	Returns the Unicode of the character at the specified index
concat()	Joins two or more strings
endsWith()	Checks whether a string ends with specified string/characters
includes()	Checks whether a string contains the specified string/characters
indexOf()	Returns the position of the first occurrence of a specified value in a string
lastIndexOf()	Returns the position of the last occurrence of a specified value in a string
match()	Searches a string for a match.
repeat()	Returns a new string with a specified number of copies
replace()	Replace old string by new one.
search()	Searches a string for a specified value.
slice()	Extracts a part of a string
split()	Splits a string into substrings
startsWith()	Checks whether a string begins with specified characters
substr()	Extracts the characters from a string
substring()	Extracts the characters from a string, between two specified indices
toLowerCase()	Converts a string to lowercase letters
toString()	Returns the value of a String object
toUpperCase()	Converts a string to uppercase letters
trim()	Removes whitespace from both ends of a string
valueOf()	Returns the primitive value of a String object

Q.5 Attempt the following (any THREE)

[15]

Q.5(a) Write a PHP program to display data from table.

[5]

```

Ans. : <?php
$con = mysql_connect("localhost","root","");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}
mysql_select_db("test");
$result = mysql_query("SELECT * FROM emp");
echo "Id    Name    Address"<br>";
echo "-----<br>";
while($row = mysql_fetch_array($result))
{

```

```

    echo $row['id'] . " " . $row['name'] . " " . $row['address'];
    echo "<br>";
    }
mysql_close($con);
?>

```

Q.5(b) Explain PHP session.**[5]**

Ans.: A PHP session variable is used to store information about, or change settings for a user session.

- Session variables hold information about one single user, and are available to all pages in one application.
- Web server does not maintain the client state as HTTP is a stateless protocol.
- PHP session allow to store user information on the server for later use (i.e. username, shopping items, etc).
- Session information is temporary and will be deleted once the user closed the web browser
- Sessions work by creating a unique id (UID) for each visitor and store variables based on this UID. The UID is either stored in a cookie or is propagated in the URL.

Q.5(c) Explain PHP mail.**[5]**

Ans.: PHP Mail

mail() function is used.

Syntax:

mail(to, subject, message, headers, parameters)

Required fields:

To	The recipient's email address.
subject	Subject of the email to be sent.
message	Defines the message to be sent.
Optional	
headers	This is typically used to add extra headers such as "From", "Cc", "Bcc".
parameters	Used to pass additional parameters.

```

<?php
$to = 'santosh.tamboli@vit.edu.in';
$subject = 'Regarding WP Exam';
$message = 'Hi Sir, When WP exam start?';
$from = 'santosh_tamboli2000@yahoo.co.in';

// Sending email
if(mail($to, $subject, $message)){
    echo 'Your mail has been sent successfully.';
} else{
    echo 'Unable to send email. Please try again.';
}
?>

```

Drawbacks of mail():

- No support for SMTP authentication.
- Difficult to send HTML-formatted emails.
- Difficult to add attachments.

To overcome above limitations, PHPMailer class can be used.

Q.5(d) Explain PHP-MySQL functions.

[5]

- (a) `mysql_query()`
- (b) `mysql_connect()`
- (c) `mysql_affected_rows`
- (d) `mysql_data_seek()`
- (e) `mysql_fetch_array()`

Ans.: `mysql_query()`

This function executes a query on a MySQL database.

This function returns the query for SELECT queries, TRUE/FALSE for other queries, or FALSE on failure.

Syntax

`mysql_query(query, connection)`

Query	Specifies the SQL query
Connection	Specifies the MySQL connection.

mysql_connect()

This function opens a non-persistent MySQL connection.

This function returns the connection on success, or FALSE and an error on failure.

Syntax

`mysql_connect(server, user, pwd, newlink, clientflag)`

Server	Specifies the server name
User	Specifies the username.
Pwd	Specifies the password.
Newlink	No new connection will be established
Clientflag	It specifies whether to use SSL or Compression

mysql_affected_rows()

It returns the number of affected rows in the previous MySQL operation or -1 if the last operation failed.

Syntax

`mysql_affected_rows(connection)`

where:

connection	Specifies the MySQL connection.
------------	---------------------------------

mysql_data_seek()

This function moves the internal row pointer.

This function returns TRUE on success, or FALSE on failure.

Syntax

`mysql_data_seek(data, row)`

data	Specifies which data pointer to use.
row	Specifies which record to move to.

Q.5(e) Explain PHP cookies.

[5]

- Ans.:
- 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.
 - PHP, can be used to both create and retrieve cookie values.

1. Create a Cookie

- setcookie() function can be used to set a cookie.
- setcookie() function must appear BEFORE the <html> tag.

Syntax

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

Example

PHP code to create a cookie named "user" and assign the value "abc" to it. Specify that the cookie should expire after one hour.

```
<?php
setcookie("user", "abc", time()+3600);
?>
<html>
```

2. Retrieve a Cookie Value

The PHP \$_COOKIE variable can be used to retrieve a cookie value

Example

Retrieve the value of the cookie named "user" and display it on a page

```
<?php
//print a cookie
echo $_COOKIE["user"];

//A way to view all cookies
print_r($_COOKIE);
?>
```

- isset() function can be used to check if a cookie has been already set or not.

```
<html>
<body>
<?php
if (isset($_COOKIE["user"]))
echo "Welcome" . $_COOKIE["user"] . "!<br / >";
else
echo "Welcome guest!<br />";
?>
</body>
</html>
```

3. Delet a Cookie

To delet a cookie set the expiration date in the past.

Example : PHP code to delete a cookie named 'user'

```
<?php
//set the expiration date to one hour ago
setcookie ("user", "", time()-3600);
?>
```

Q.5(f) Explain regular expression in PHP.**[5]**

Ans.: A regular expression pattern is a string consisting of plain text and pattern meta-characters. The regexmeta characters define the type and number of characters that can match part of a pattern. Character Glasses allow a pattern to match multiple characters simultaneously.

Character classes are

- d - Digits 0-9
- D - Anything not a digit
- w - Any alphanumeric character or an underscore (_)
- W - Anything not an alphanumeric character or an underscore
- s - Any whitespace
- S - Any nonwhitespace character
- .

- Character class meta characters match a single character.
- enumeration operators can specify how many times they must match.

The enumeration operators are

- ? - 0 or 1 time
- * - 0 or more times
- + - -1 or more times
- {,n} - at most n times
- {m,} - m or more times
- {m, n} - between m and n times

Example

```
if (preg_match("/d{5}-d{4}/", $subject)) {  
  // matches a ZIP+4  
}
```

preg_match() takes two arguments, the first argument is the pattern, and the second argument is the subject string. The pattern is enclosed in forward slashes.

Example

```
$string = 'My zipcode is 21797 -2046';  
if (preg_match("/(d{5})-(d{4})/", $string, $matches)) {  
  print_r( $matches);  
}
```

This will print:

```
Array  
(  
  [0] => 21797-2046  
  [1] => 21797  
  [2] => 2046  
)'
```

