SRI KRISHNADEVARAYA UNIVERSITY

B.Sc. Degree Examination

Third year – Sixth Semester

AUGUST 2021

Computer Applications

Paper VII : **WEB TECHNOLOGIES**

(Regular/Supplementary)

Time : 3 Hours Max : 75 Marks

Section A (5 × 5 = 25 Marks)

Answer any **five** *of the following questions.*

1. What is Hyperlink? Explain.
2. Discuss about multimedia objects.
3. Write about using styles in CSS.
4. Describe about defining your own styles.
5. Write a short notes on DHTML.
6. What are mathematical functions in Java script?
7. Briefly explain about Data validation.
8. What is the purpose of rollover buttons?
9. Write about basic XML.
10. Explain briefly about web services.

Section B (5 × 10 = 50 Marks)

Answer **all** *the questions.*

11. (a) Explain the lists in HTML with example.

(*Or*)

(b) Briefly explain about forms towards interactive in HTML.

12. (a) Discuss in detailed note On Cascading style sheets.

(*Or*)

(b) Write about properties and values in styles?

13. (a) Explain about operators in Java script.

(*Or*)

(b) Describe exception handling in Java script with an example?

14. (a) Explain how messages and confirmations can be done in Java Script.

(*Or*)

(b) Discuss the concept of moving images with examples?

15. (a) Write in detail about document type definition.

(*Or*)

(b) Briefly explain about document object model in XML.

SRI KRISHNADEVARAYA UNIVERSITY

B.Sc. Degree Examination

Third Year

Sixth Semester

SEPTEMBER 2020

Part II : Computer Science

(Regular & Supplementary)

Paper - VII : **Web Technologies**

(Common for B.A. (CA) and B.Sc. (CS))

Time : 3 Hours Max : 75 Marks

Section A (5 × 5 = 25 Marks)

Answer any **five** *questions.*

1. Explain basic structure of HTML with example.
2. Discuss image tag with example.
3. What is style rule ? Give example.
4. Discuss formatting blocks of information.
5. What are data types in JavaScript ?
6. Define Array and object.
7. Write a note on messages and conformations.
8. How to open a new window using a JavaScript.
9. What is XML ? Write its need.
10. Write a note on Document object model in XML.

Section B (5 × 10 = 50 Marks)

Answer **all** *questions.*

11. (a) Discuss different types of links with examples.

(Or)

(b) Discuss form elements with examples.

12. (a) Discuss different formatting styles to implement styles with example.

(Or)

(b) Discuss layers with example.

13. (a) Discuss any Five mathematical functions in Javascript with example.

(Or)

(b) Write a program to handle Exception handling.

14. (a) Discuss about data validation using JavaScript.

(Or)

(b) Write a note on moving images.

15. (a) Write a note on DTD.

(Or)

(b) Discuss about Web services.

SRI KRISHNADEVARAYA UNIVERSITY

B.Sc. Degree Examination

Thid Year – Sixth Semester

MARCH 2019

Computer Applications (Regular)

WEB TECHNOLOGIES

(Common for B.A. (CA) & B.Sc. (CS))

Time : 3 Hours] [Max : 75 Marks

Section A (5 × 5 = 25 Marks)

Answer **five** *questions.*

1. Describe about text formatting tags.
2. Explain the basic structure of an HTML document.
3. What is CSS ? Explain about inline stylesheets.
4. What is style rule ? Give an example.
5. Explain the data types of JavaScript.
6. What is function ? Explain its syntax with example.
7. Write a note on messages and confirmations.
8. Write a JavaScript for opening a new window.
9. What is XML ? Explain its structure with example.
10. Write about XML name spaces.

Section B (5 × 10 = 50 Marks)

Answer **all** *the questions.*

11. (a) How to partitioned a webpage using frameset tag ? Explain.

(*Or*)

(b) Explain the document head in detail.

12. (a) Explain about class and anonymous classes in CSS.

(*Or*)

(b) Write about layers and its attributes with example.

13. (a) How JavaScript included in HTML documents ? Give example.

(*Or*)

(b) Explain about basic array functions in JavaScript.

14. (a) Explain the data validation process with examples.

(*Or*)

(b) What are rollover buttons ? Explain with example.

15. (a) Explain about document type definition in XML.

(*Or*)

(b) Explain about XML schema with example.

SRI KRISHNADEVARAYA UNIVERSITY

B.Sc. Degree Examination

Thid Year – Sixth Semester

APRIL 2018

Computer Applications (Regular)

WEB TECHNOLOGIES

(Common for B.A. & B.Sc.)

Time : 3 Hours] [Max : 75 Marks

Section A (5 × 5 = 25 Marks)

Answer **five** *questions.*

Q. 1 ***Explain HTML Document structure with example.***

HTML, which stands for Hyper Text Markup Language, is the predominant markup language for web pages. It is written in the form of HTML elements consisting of "tags" surrounded by angle brackets within the web page content.

It allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts in languages such as JavaScript which affect the behavior of HTML webpages.

HTML can also be used to include Cascading Style Sheets (CSS) to define the appearance and layout of text and other material. The W3C, maintainer of both HTML and CSS standards, encourages the use of CSS over explicit presentational markup.

Structure of HTML document :

An HTML document consists of text, which comprises the content of the document, and tags, which define the structure and appearance of the document. The structure of an HTML document is simple, consisting of an outer <html> tag enclosing the document header and body:

```
<html>
<head>
<title>Barebones HTML Document</title>
</head>
<body>
This illustrates in a very <i>simple</i> way, the basic structure of an HTML document.
</body>
</html>
```

Each document has a *head* and a *body*, delimited by the <head> and <body> tags. The head is where you give your HTML document a title and where you indicate other parameters the browser may use when displaying the document. The body is where you put the actual contents of the HTML document. This includes the text for display and document control markers (tags) that advise the browser how to display the text. Tags also reference special-effects files like graphics and sound, and indicate the hot spots (*hyperlinks* or *anchors*) that link your document to other documents.

Q. 2 ***Write about image tag and its attributes. Give example.***

In HTML, images are defined with <img> tag to show images on a web page.

<img> tag attributes are:

(a) src attribute is used to get the url location of the image from where we get the image.

(b) Alt attribute is used to show the alternative text when the server not able to find the image in the given location.

(c) Width attribute is used to define the width of the image.

(d) Height attribute is used to define the height of the image.

The syntax to define an image is:

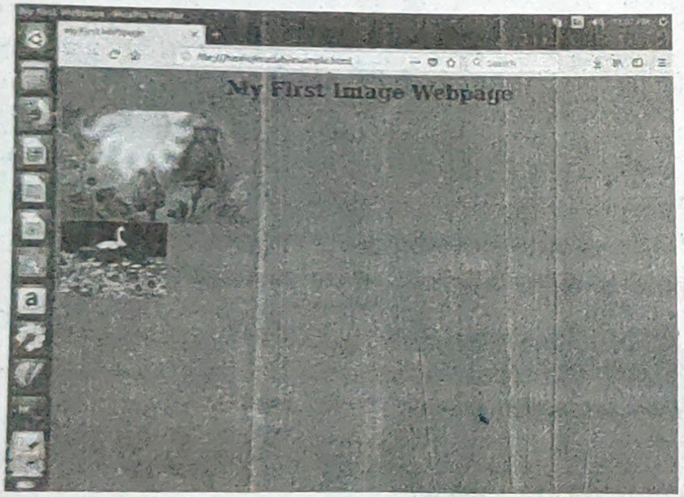
<img src="url">

for example:

```
<html>
<head>
```

```
<title>My First Webpage</title>
</head>
<body bgcolor="#FF00FF">
<h1 align="center">My First Image Webpage</h1>
<img src="/home/matlab/Pictures/rose.jpeg">
<br>
<img src="/home/matlab/Pictures/hamsa.jpeg" height=120
width:120>
</body>
</html>
```

output:



My First Image Webpage

Q. 3 ***List and describe any five properties and its values in CSS.***

Refer to Q. 6, Question Bank, Page 84

Q. 4 ***Explain external stylesheet with example.***

Stylesheet definitions created with CSS (cascading style sheet) can be inserted into an HTML document in 3 ways.

External stylesheet :

Use an external stylesheet when we want to apply one style to many pages. If we make one change in an external stylesheet, the change is universal on all the pages where the stylesheet is used.

An external stylesheet is declared in an external file with a .css extension. It is called by pages whose interface it will affect. External stylesheets are called using the <link> tag which should be placed in the head section of an HTML document. This tag takes three attributes.

Attributes of the <link> tag :

1. rel - When using an external stylesheet on a webpage, this attribute takes the value "stylesheet"
2. type - When using an external stylesheet on a webpage, this attribute takes the value "text/css"
3. href - Denotes the name and location of the external stylesheet to be used.

Example :

```
<html>
<head>
<link rel="stylesheet" type="text/css" href="style1.css" />
</head>
<body>
<p> The text in this paragraph will be blue. </p>
</body>
</html>
```

Q. 5 ***Write about mathematical functions in JavaScript.***

Refer to Q. 10, Question Bank, Page 105

Q. 6 ***Write about features of JavaScript.***

Javascript has a number of benefits:

(a) It is widely supported in web browsers.

(b) It gives easy access to the document objects and can manipulate most of them.

(c) It gives interesting animations with low downloading times.

(d) Web surfer don't need a special plug-in to use scripts.

(e) It is relatively secure.

Q. 7 ***Write a JavaScript for opening a new window.***

Refer to Q. 17, Question Bank, Page 117

Q. 8 ***How to move images in JavaScript. Explain.***

Refer to Q. 11, Question Bank, Page 159

Q. 9 ***Write differences between XML and DHTML.***

XML :

(a) XML stands for eXtensible Markup Language.

(b) XML is a cross-platform, software and hardware independent tool for transmitting information.

(c) XML tags are not predefined, users have to define their own tags for their specific application before using tags.

(d) XML uses Document Type Definition (DTD) or an XML Schema to define tags.

(e) XML was designed to describe data, XML aims to deal with the logic meaning of data, or semantics.

DHTML :

(a) DHTML stands for Dynamic Hyper Text Markup Language.

(b) The technique of making Web pages dynamic by client-side scripting to manipulate the document content and presentation.

(c) Standard-based DHTML involves three aspects:

(d) Javascript for cross-browser scripting.

(e) Cascading Style Sheets(CSS) for style and presentation control.

(f) Document Object Model (DOM) for a uniformprogramming interface to access and manipulate the Web page as a document.

Q. 10 ***Define XML and its features.***

XML means extensible Markup Language. XML is a self-describing data interchange language that not only contains data but also contains information about the structure and nature of the data.

The vast majority of desktop applications are now able to view and manipulate XML data with the XML standard being consistent across languages and operating systems.

The development of XML and its adoption as a formal recommendation of the World Wide Web Consortium provides significant opportunities to improve data exchange and interoperability for modelling and simulation software projects.

XML is a language that is able to represent not only data itself, but the nature and structure of that data. This combination of data and metadata in the one file means XML has considerable advantages over traditional data formats. As XML files are plain text and comply with a globally accepted standard, the files are very useful in almost any application where interoperability is required. XML is a cross-platform format that is not exclusive to any particular operating system or development platform.

Features :

1. XML seperates data from HTML.
2. XML simplifies data sharing.
3. XML Simplifies data transport.
4. XML simplifies Platform Changes.
5. XML makes data more available.
6. XML is used to create new internet languages.

Section B (5 × 10 = 50 Marks)

Answer **all** *the questions.*

Q. 11 (a) ***Explain form tag and its input types in detail.***

Forms are a mechanism that allow to type information into fields on a browser screen and submit the information to a webserver. They allows to create interactive webpages. We can fill out a form, submit it and then the information is uploaded to the designated server to handle the form. A program on the server processes the information and then returns a new HTML document.

Forms have many functions. They can be used for gathering information about an user, conducting a survey, selecting something of interest, placing an order online, submitting a query to a search engine and so on. In each case, when the user submits a form, values are uploaded to the receiving server the server that processes the form does not have to be the server from which the form was retrieved, although it usually is. If the user filled out the form correctly, the HTML document may contain a message asking the user to redo the form and resubmit it.

When we retrieve a webpage containing a form, the connection is released immediately after the form is delivered. Submitting it requires a new connection to the server be established.

Form Tags : We can include multiple forms in a single HTML document. Also, forms can have a number of different types of fields such as tent input fields, radio buttons, check bones, select feilds, submit and reget button.

The beginning and ending tags for form creation are <FORM> and <\FORM>. The most important attributes of FORM tag are ACTION and METHOD. For example the following HTML code.

```
<FORM ACTION = "http: / / WWW.Stockers.com/
cgibin/stock-process" METHOD = "get">
< /FORM >
```

Specifies that the program that process the form resides on the server WWW. stockers.com;

In the cgibin directory and is called stock-process. Most form processing programs are kept in cgibin directory are something similar. The CGI stands for common gateway interface and bin stands for binary.

Form Methods : There are two different ways of sending values to a web server. They are 'GET' and 'POST'. The GET method appends the values of input parameters to the URL specified in ACTION attribute the ACTION URL is separated from the parameter names and values by question mark symbol. For example, when the query

http: / / WWW.XYZMail.com / login?user=srinivas & password = 123ca2z is submitted, the ACTION URL is

http: / / WWW.XYZMail.com / login

The input parameter names and values are included in the string user = srinivas & password = 123cazz. the parameter name / value pair is superated by & sign.

In the post METHOD the server specified in ACTION attribute in contacted. After the communication is made input values are sent to the server.

Whether a form uses GET or POST method the program that handles the form must be able to decode the parameter and use them. Form processing programs are usually written in C, C++, Java, or Perl.

Comparision GET and POST :

(1) For a novice programmer GET method may be preferable, since the POST method requires more programming expertise.

(2) If you have a small list of input parameter, GET is preferable, but for a long list of input parameters POST method is preferable.

(3) Neither method is totally secure, but the POST METHOD is preferable to the GET METHOD if security is an issue.

From input tags : The basic structure of a form is <FORM ACTION = "http: / / WWW.Server,com / cgibin / program" METHOD = "get">..... Here goes input tags.

</FORM>

We include nearly any type of HTML formatting command inside of a form tag, and forms may be nested. To create most of the form elements, use the input tag, <INPUT>. The program on the server-side must do type checking and must handle errors.

Text boxs : A rectangular- shaped field in which a user can enter text is called a text box. A text box is produced using INPUT tag and specifying the appropriate attributes. The following HTML code is used to produce a text box and the question posed above it.

```
<DL>
        <DT> What is your nick name ?
        <DD> <INPUT TYPE = "text" NAME = "nn" SIZE
= "40" MAXLENGTH = "60">
</DL>
```

The value assigned to the TYPE attribute is text which is used for producing a text box. The VALUE attribute of the input tag is used to specify a default value for a parameter.

Checkboxes : A checkbox is represented by an icon that the user can select or deselect by clicking on it. A selected checkbox is usually shown in dark gray, and an unselected checkbox is usually shown in light gray. Checkboxes are used in series, so that a user can easily specify all of their preferences. The following code produces three checkboxes and allows the user to select their favorite running distance : 5 km, 10 km or 262 kmiles.

```
    <DL>
        <DT> What kind is your favorite distance ?
        <DD> <INPUT TYPE + "checkbox" NAME '
"DISTANCE" VALUE = "5" CHECKED> 5 KM
            <INPUT TYPE = "Checkbox" NAME =
        "DISTANCE" VALUE = "10" CHECKED > 10 KM
            <INPUT TYPE = "Checkbox" NAME =
"DISTANCE" VALUE = "26.2" CHECKED> 26.2 miles.
```

The value assigned to the TYPE attribute is checkbox, which creates a checkbox.

Another interesting point is that we specified the CHECKED attribute of the input tag for the entry corresponding to the 5K. This allows us to set a default value.

Radio Buttons : Radio buttons are a group of buttons from which only one can be selected at a time. The following code includes two radio buttons to determine whether or not the person filling out the form had ever run before.

```
Ever Run Before ?
<INPUT TYPE = "radio"
    NAME = "Run"
    VALUE = "YES"
    CHECKED > YES
<INPUT TYPE = "radio"
```

```
    NAME = "Run"
    VALUE = "NO" > NO
```

Buttons that have the same name are the elements of the same set of the radio buttons, and only one of them may be selected at any given time. The default button is the first button, unless the CHECKED attribute is included. A user can deselect the default by clicking on the other button.

Action Button : There are two types of action buttons. They are submit and reset. When user clicks on the submit button, the values that have been entered into the form are sent to the program that processes the form. The code for submit button is

```
<INPUT TYPE = "Submit"
    VALUE = "SUBMIT">
```

The function of the reset button is to allow the user to clear all of the input they have thus far entered into the form. The code for the reset button is

```
<INPUT TYPE = "reset"
    VALUE = "RESET">
```

Select : The select tag lets you choose any subset of items from a group and it does not take a lot of screen space. The select tag <SELECT> with its ending tag </SELECT>. The items in a given select tag are usually rendered in the style of a pop-up menu. For example

```
Favorite beverage ?
<SELECT NAME = "drink">
    <OPTION> COKE
    <OPTION> Thums up
    <OPTION> Pepsi
    <OPTION> Mazza
    <OPTION> Sprite
    <OPTION SELECTED> Water.
</SELECT>
```

The SELECTED attribute is used to preselect an item. The VALUE attribute allows you to specify the content to be returned if a given option is selected.

Passwords : We can create a password text field using the following code :

```
<INPUT TYPE = "Password" NAME = "nn"
SIZE = "40" MAXLENGTH = "20">
```

This creates a masked field in which a user can enter their password; the password is not shown on the screen. However, no security, other than a masking of the password, is provided in the way the password is transmitted to the server.

(*Or*)

Q. 11 (*b*) *Explain table tags and its attributes with example.*

Tables in HTML pages allow you to organize information in a row and column format. HTML tables are used to present any type of information for which we want a lot of control over the positioning of the material.

There are many ways to use tables to format information on the web. There is a good chance it was produced using tables. The following are some situations for which you might use tables :

Present tabular information : If you have information or data that is naturally divided into rows and columns, it can probably be easy and effectively formatted using HTML.

Control layout : If you want to control the layout of text, position a group of images, or present an extensive menu, you may decide to use tables to achieve the desired appearance.

Express relationships : If you need to display relationships between a group of items, tables are usually a good machanism to use.

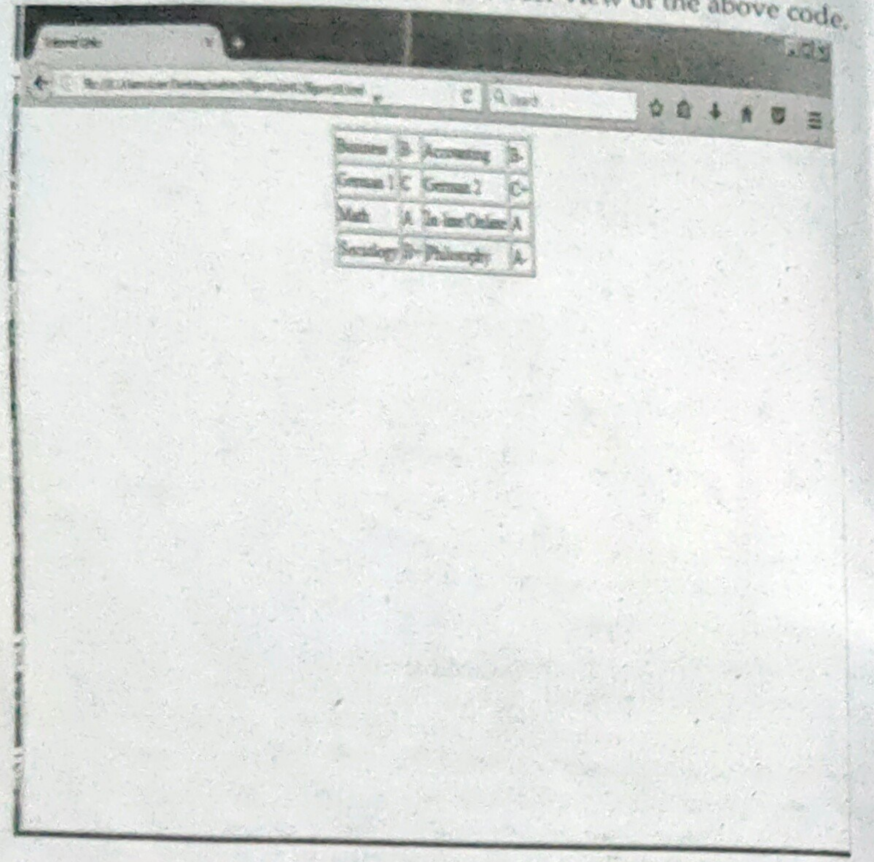
Tables are created using the table tag <TABLE>, with the ending tag </TABLE>. The most important tag that goes inside the table tag is the table row tag <TR> its corresponding end tag </TR> is omitted. Browsers can determine where the next row starts, when encounters another beginning table row tag. When the browser sees the </TABLE> tag. The table is indeed. The items a row are specified using the table data tag <TD>. Its corresponding end tag </TD> may be omited for the same reason that </TR> can.

There is no table column or <TC> tag. The number of columns is determined by the row containing the most items, expressed by table data and table header tag. If any row has fewer than the maximum number of items. Its elements will be positioned starting from the left most column and continuing to the right, as necessary.

Eg :

```
<CENTER>
<TABLE>
    <TR>
        <TD> Business
        <TD> B-
        <TD> Accounting
        <TD> B-
    <TR>
        <TD> German 1
        <TD> C
        <TD> German 2
        <TD> C+
    <TR>
        <TD> Math
        <TD> A
        <TD> In-line / on - line
        <TD> A
    <TR>
        <TD> Sociology
        <TD> B+
        <TD> Philosophy
        <TD> A-
    <TR>
        <TD> Tennis
        <TD> A
    </TABLE>
    </CENTER>
```

The following figure shows the browser view of the above code.



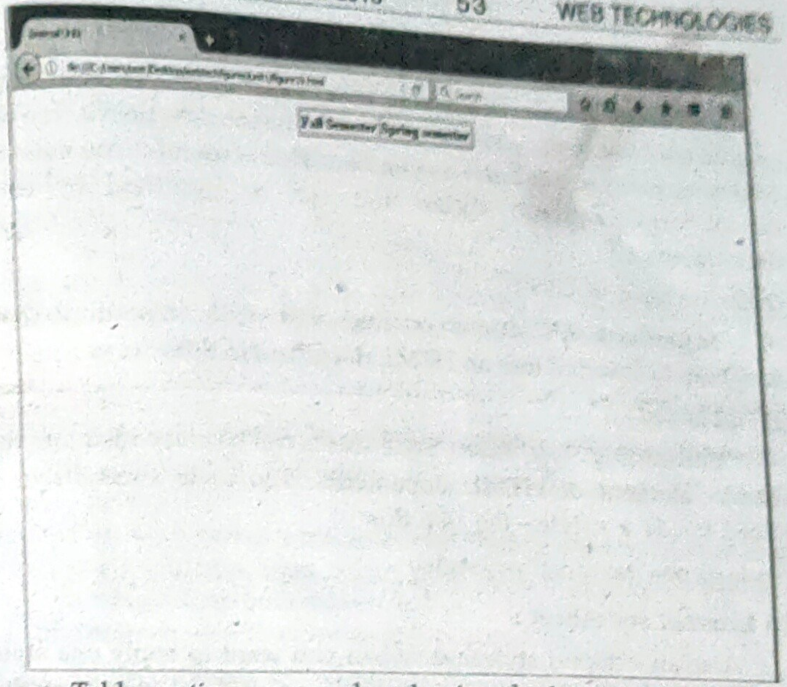
Business	B	Accounting	B
German 1	C	German 2	C-
Math	A	In-line Online	A
Sociology	B-	Philosophy	A-

A simple version of Table

The table header tag is used to include headers and format them. The table header tag <TH> and its corresponding end tag </TH> is omitted. Browsers can determine where the next header starts, when encounters another table header tag.

Eg :

```
<TABLE>
        <TR>
                <TH> Fall Semester
                <TH> Spring semester
        </TABLE>
```



Fall Semester Spring semester

Table captions are produced using the beginning and ending caption tags. <CAPTION> and </CAPTION>. The caption tag is best placed either immediatly after the table tag or immediately inside the ending table tag. The following code places the caption directly

```
<TABLE>
        <CAPTION>
                <STRONG>
                        A sample tabular format consisting of a
                        student's courses and grades.
                </STRONG>
        </CAPTION>
</TABLE>
```

The table tag has a WIDTH attribute that allows you to control the width of the table measured from the outer perimeter, including borders.

```
<TABLE WIDTH = "600">
```

Q. 12 (*a*) ***List and explain various types of style sheets.***

Cascading style sheets : Styles are cascaded. This means that you do not have to use a single set of styles inside a document, you can import as many style sheets as you like. This is useful if you define a set of organizational styles that can be modified by each department.

Different types of CSS :

Stylesheet definitions created with CSS (cascading style sheet) can be inserted into an HTML document in three ways.

(I) Embedded :

Embedding Style Sheets Style sheets can be embedded into the <head> element of HTML documents. The style sheet itself is placed inside a <style> tag like this:

<style type="text/css"></style>

(II) External stylesheet :

Use an external stylesheet when you want to apply one style to many pages. If we make one change in an external stylesheet, the change is universal on all the pages where the stylesheet is used.

An external stylesheet is declared in an external file with a .css extension. It is called by pages whose interface it will affect. External stylesheets are called using the <link> tag which should be placed in the head section of an HTML document.

(III) Inline :

Style rules can be specified right in the HTML tag, with style="style rules". Here is an example:

<p style="font-family: trebuchet, times, serif; font-size: 20px; font-weight: 600; color:#7C3030; background-color: #B5D6A9"> The style rules above look like this! </p>.

Inline style sheet declares an individual element's format using the XHTML attribute style. Inline style override any other styles applied using the techniques.

(*Or*)

Q. 12 (*b*) ***Explain block formatting context in CSS.***

In the CSS Rule, we can set various formatting options. In block category we can do Text alignment and indentation, along with other text block properties.

Block-level Elements :

Block-level elements are used to format whole blocks of text — they stand out on their own, spanning the available screen-width and usually adding line breaks before and after themselves.

The block-level elements are:

1. <address>

   <address> should be wrapped around contact information, including email addresses.

2. <blockquote>

   <blockquote> is a block-level tag that's used to enclose multi-line quotations from other sources. It is usually displayed as indented from both sides.

3. <div>

   <div> defines a division in the HTML document. It is used as a container for other HTML elements to style them with CSS.

4. <fieldset>

   <fieldset> tag is used to group related elements in a form.

5. <h1> — <h6>

   <h1>–<h6> create headings. They should flow sequentially. The title of the page should always appear as a level 1 heading, with sub-headings cascading down from it. Text is usually displayed in a large, bold font.

6. <hr>

   <hr> represents a horizontal rule in a visual browser.

7. <legend>

   The <legend> element is used to add a caption to a group of related form <input> elements that have been grouped together into a <fieldset>.

8. <p>

   <p> tag defines a paragraph. Browsers automatically add some space before and after each <p> element.

9. <pre>

<pre> is a block-level element that displays text in a fixed-width font exactly how it was typed in the source code.

10. <ul>, <ol>, <li>, <dl> and <dd>

These are the unordered information, ordered information and definitions.

We can follow the below rules between block-level elements and Inline elements:

11. Block-level elements can contain other block-level elements and inline elements

12. Inline elements cannot contain block-level elements.

Q. 13 (a) *Explain Date object methods in JavaScript.*

Refer to Q. 20, Question Bank, Page 120

(Or)

Q. 13 (b) *Explain operations in JavaScript.*

Refer to Q. 4, Question Bank, Page 94

Q. 14 (a) *Write the differences between DHTML and JavaScript.*

DHTML :

(a) DHTML refer to the technique of making Web pages dynamic by client-side scripting to manipulate the document content and presentation. Web pages can be made more lively, dynamic, or interactive by DHTML techniques.

(b) DHTML is not a markup language or a software tool. It is a technique to make dynamic Web pages via client-side programming.

(c) Animation in DHTML use a number of DOM elements (<img />, <div> or otherwise) are moved around the page according to some sort of pattern determined by a logical equation or function

(d) The idea behind Javascript-based animation is fairly simple; a number of DOM elements (<img />, <div> or otherwise) are moved around the page according to some sort of pattern determined by a logical equation or function. To achieve the effect of animation, elements must be moved at a given interval or frame rate; from a programming perspective, the simplest way to do this is to set up an animation loop with a delay.

Javascript :

(a) Scripts are programs written in a simple and easy-to-use language to specify control of other programs. Javascript language to control browsers actions in response to events.

(b) Javascript is a widely used scripting language originally developed by Netscape for both client-side and server-side scripting.

(c) Javascript promises to be the language of choice, in combination with XHTML, CSS, and DOM to deliver dynamism and interactivity for the Web.

(d) Javascript-based animation is fairly simple; With JavaScript, it is possible to execute some code at specified time-intervals called timing events. It's very easy to time events in JavaScript. The two key methods that are used are:

(i) setInterval() - executes a function, over and over again, at specified time intervals.

(ii) setTimeout() - executes a function, once, after waiting a specified number of milliseconds.

(Or)

Q. 14 (b) *Explain rollover buttons in JavaScript with example.*

Refer to Q. 10, Question Bank, Page 157

Q. 15 (a) *Explain various presenting methods of XML.*

Refer to Q. 4, Question Bank, Page 174

(Or)

Q. 15 (b) *Explain DOM events.*

With cooperation from major browser vendors, the W3C is establishing the Document Object Model (DOM) as a standard application programming interface (API) for scripts to access and manipulate HTML and XML documents.

Document Object Model Events Level 3, a platform- and language-neutral interface that allows programs and scripts to dynamically access and update the content, structure and style of documents.

The DOM Level 3 Event Model is designed with two main goals.

(a) The first goal is the design of a generic event system which allows registration of event handlers, describes event flow through a tree structure, and provides basic contextual information for each event.

(b) The second goal of the event model is to provide a common subset of the current event systems used in DOM Level 0 browsers.

DOM Level 3 events can be classified into six types. They are:

(a) UI Events :

User Interface Events (UI Events) are generated by user interaction through an external device (mouse, keyboard, etc.)

(b) The different types of UI Events are listed below:

(i) DOMFocusIn

The DOMFocusIn event occurs when an EventTarget receives focus, for instance via a pointing device being moved onto an element or by tabbing navigation to the element.

(ii) DOMFocusOut

The DOMFocusOut event occurs when a EventTarget loses focus, for instance via a pointing device being moved out of an element or by tabbing navigation out of the element.

(iii) DOMActivate

The activate event occurs when an element is activated, for instance, thru a mouse click or a keypress.

(c) UI Logical Events :

These events are device independent user interface events such as focus change message or element triggering notifications.

(d) Mutation Events :

Events caused by any action which modifies the structure of the document.

(e) Capturing :

Capturing is the process by which an event can be handled by one of the event's target ancestors before being handled by the event's target.

(f) Bubbling :

Bubbling is the process by which an event propagates upward through its ancestors after being handled by the event's target.

(g) Cancelable :

It is a designation for events which indicates that upon handling the event the client may choose to prevent the DOM implementation from processing any default action associated with the event.