C.P.PATEL & F.H.SHAH COMMERCE COLLEGE
(MANAGED BY SARDAR PATEL EDUCATION TRUST)
BCA, BBA(ITM) & PGDCA PROGRAMME
BCA Sem I (WEB APPLICATION DESIGN)
UNIT II: Web Page Designing-II

UNIT III: Web Page Designing-II	
Sr No.	Topics
1.	HTML tables
2.	Frames, framesets, nested framesets
3.	Designing HTML forms
4.	Multimedia tags
5.	Advance Elements of HTML5: !Doctype, meta, Input Controls (number, date, time, calendar, ranges),
6.	Multimedia tags(<audio> , <video>)
Reference Books: 1. Ivan Bayross, "Web enabled Commercial Application Development using HTML, DHTML, Java script, pen CGI" BPB 2004 2. Xavier C: World Wide Web Design with HTML, Tata Mcgraw hill publication 2000	

❖ TABLE HANDLING

TABLE

Tables are commonly used as a layout device for arranging text and images The TABLE element defines a table for multi-dimensional data arranged in rows and columns. The TABLE element contains components like, TR and TD Each entry in the table is called a cell Each cell contains data.

A table is an arrangement of data in rows and columns, or possibly in a more complex structure. Tables are widely used in communication, research, and data analysis.

- Tables are useful for various tasks such as presenting text information and numerical data.
- Tables can be used to compare two cr more items in tabular form layout.
- Tables are used to create databases

Defining Tables in HTML

- An HTML table is defined with the "table" tag. Each table row is defined with the "tr" tag. A table header is defined with the "th" tag. By default, table headings are bold and centred. A table data/cell is defined with the "td" tag.

THE WIDTH OF THE TABLE AND CELLS are measured in any one of the following units:
E g <TABLE BORDER UNITS=en WIDTH= "6O">
OR
<TABLE BORDER UNITS=PIXELS WIDTH=100>
OR
<TABLE BORDER UNITS=Relative WIDTH=50%>

the alignment for each cell.

- Left alignment

- Right alignment
- Center alignment
- Decimal alignment

The attribute for column specification is COLSPEC. The column specification is given as a string.

Syntax:

<TABLE ALIGN=”left” |”center”| “right”
BGCOLOR=”color”
BORDER=”pixels”
CELLPADDING=”pixels”
CELLSPACI NG=”pixels”
BACKGROUND=”URL”
HEIGHT=”height”
WIDTH=”width”>
….
</TABLE>

Attribute	Description
ALIGN=“left”|“center” |“right”	Specifies the Horizontal alignment of the table. The default is left.
BGCOLOR=”color”	Specifies the background color of table cells. The background color of specific cells may differ if specified by attribute BGCOLOR of elements TR, TH and TD.
BORDER=”pixels”	Specifies the width of border around the table. If omitted the table has no border
CELLPADDING=pixels”	Specifies the spacing within cells i.e. between the cell border and cell contents. **Default cellpadding is 1.**
CELLSPACING=”pixels”	Specifies the spacing between cells of the table. **Default cellspacing is 2.**
BACKGROUND=”URL”	Specifies the URL (UNIFORM RESOURCE LOCATOR) of background image for the table.
HEIGHT= “height”	Specifies the table height as: pixels - (e.g. HEIGHT=”60”); -OR percentage of window height - (e.g. HEIGHT=”20%”).
WIDTH= “WIDTH”	You can set a table width using **width** attributes. You can specify table width in terms of pixels or in terms of percentage of available screen area.

TR (Table Row)

The TR element defines a table row. TR must be contained within TABLE. TR contains TD.

Syntax:

<TR ALIGN=”left” “center” “right” VALIGN = “top” | “middle” |“bottom”
BGCOLOR=”color” HEIGHT= “SIZE” BACKGROUND= “URL/
IMAGENAME.EXTENSION”>
</TR>

Attributes:

Attribute	Description
ALIGN= “left” “center” “right”	Specifies the horizontal alignment for each cell in the row The default is left

"right"	
"top" VALIGN="middle" "bottom"	Specifies the vertical position of the cell's contents top - positions data at the top of the cell; middle - centers the cell data vertically (the default value), Bottom - positions data at the bottom of the cell.
BGCOLOR="color"	Specifies the background color for the row of the table. The background' color of specific cells may differ if specified by attribute BGCOLOR of elements TH and TD.
HEIGHT	Specifies the table height as: pixels - (e.g. HEIGHT="60"); -OR percentage of window height - (e.g. HEIGHT="20%").
BACKGROUND	Specifies the URL (UNIFORM RESOURCE LOCATOR) of background image for the table.

TD (Table Data)

The TD element defines a data cell in a table TD is contained within TR Browsers display text contained.

Syntax:

```
<TD
ALIGN="left" | "center" |"right"
VALIGN="top" | "middle"| "bottom"
BGCOLOR="color"
BACKGROUND="URL"
COLSPAN="number"
ROWSPAN="number"
WIDTH= "WIDTH">
.....</TD>
```

Attributes of TH and TD are:

Attribute	Description
"left" ALIGN= "center" "right"	Specifies the horizontal alignment for the cell's content. The default is left.
"top" VALIGN= "middle" "bottom"	Specifies the vertical position of the cell's content. Top - positions data at the top of the cell; middle - centers the cell data vertically (the default value); bottom - positions data at the bottom of the cell.
BGCOLOR="color"	Specifies the background color for the cell.
BACKGROUND="URL"	Specifies the URL (UNIFORM RESOURCE LOCATOR) of background
COLSPAN= "number"	Specifies the number of columns that are spanned by the cell.
ROWSPAN= "number"	Specifies the number of rows that are spanned by the cell.
WIDTH= "WIDTH"	You can set a table width using **width** attributes. You can specify table width in terms of pixels or in terms of percentage of available screen area.

❖ **FRAMES**

what is a frame?

Frames are a way of presenting two or more web pages to the visitor at once. The Technical definition "A frame is a rectangular region within the browser window that displays a web page, alongside other pages in other frames".
The Browser shows the web page through a window. We scroll the web page and see the entire document through the window of the browser. The windows are called the container. HTML frames divide the container / window to present documents in sub windows Sub windows Provide a way to keep certain information visible, while other views are scrolled or replaced.

Characteristics of Frames
- Each frame is given a name.
- Each frame will be targeted by an HTML document.
- Each frame resizes itself dynamically in response to the changes in the size of the visible. client area

FRAMESET
The FRAMESET element is a frame container for dividing a window into rectangular subspaces called frames. In a Frameset document, the outermost FRAMESET element takes the place of BODY and immediately follows the HEAD. The FRAMESET element contains one or more FRAMESET or FRAME elements, along with an optional NOFRAMES element. A frameset document has no content...it just tells the browser which pages to load, and how to arrange them in the window. When <FRAMESET> is being coded within an HTML document, the <BODY> </BODY> tags are not used.
THE SIZES OF THE FRAMES are measured in any one of the following units:

- Pixels e.g. <frameset cols="150,70,70">
- Percentage e.g. <frameset rows="70%,30%>
- Fraction e.g. <frameset rows="*4*, *4*, *, *" > Here the frame sizes are 4/10, 4/10, *1/10,* 1/10 of the overall container size.

Syntax:
```
<FRAMESET
COLS="list"
ROWS="list"
BORDER="pixels"
BORDERCOLOR="color"
FRAMEBORDER="yes" "no"
FRAMESPACING="pixels">
</FRAMESET>
```

Attributes:

Attribute	Description
COLS"list"	Defines a comma-separated list of widths for division of window on columns. (For example, to divide the window on two columns 20% and 80% width, write COLS=(20%,80%) or COLS=(20%,*). The asterisk * replaces the remaining percentage to 100%.)
ROWS="list"	• Specifies the comma-separated list of heights for division of window on rows. (For example, to divide the window on two rows 20% and 80% height, write ROWS=(20%,80%) or ROWS=(20%,*). The asterisk* replaces the remaining percentage to 100%.)

BORDER="pixels"	Specifies the border width around frames.
BORDERCOLOR="color"	Specifies the border color around frames. If omitted, the default color is gray.
FRAMEBORDER="yes" "no"	Specifies whether or not the frames have a visible border. If FRAMEBORDER= "yes" the space between frames is filled with BORDERCOLOR, if FRAMEBORDER="no" the space between frames is left blank (white color).
FRAMESPACING="pixels"	Specifies the space between frames.

Note: The attribute FRAMESPACING overrides the attribute BORDER, i.e. if FRAMESPACING="20" and BORDER="O", the space between frames is 20 pixels.

FRAME

The FRAME element defines a frame a rectangular subspace within a Frameset document. Each FRAME must be contained within a FRAMESET that defines the dimensions of the frame.

Syntax:

```
<FRAME
NAME="name"
MARGINHEIGHT="pixels"
MARGINWIDTH="pixels"
BORDERCOLOR="color"
FRAMEBORDER="yes" "no"
NORESIZE
SCROLLING="yes" "no" "auto"
SRC="URL">
```

Attributes:

Attribute	Description
NAME="name"	Specifies the name of the frame for use with the TARGET attribute of a element. The name of the frame begins with an alphabetic letter.
MARGINHEIGHT="pixels"	Specifies the number of pixels to use as the top/bottom margins, within the frame.
MARGINWIDTH="pixels"	Specifies the number of pixels to use as the left/right margins, within the frame.
BORDERCOLOR="color"	Specifies the border color around the frame. If omitted, the default color is gray. This attribute overrides the same attribute of FRAMESET element.
FRAMEBORDER="yes"|"no"	Specifies whether or not the frame has a visible border. 'If FRAMEBORDER= "yes" the space between the frame and all adjoining frames is filled with BORDERCOLOR, if FRAMEBORDER="no" the space is left blank (white color).
NORESIZE	Prevents the user from resizing the frame.
"yes" SCROLLING="no" "auto"	Specifies whether scrollbars are provided for the frame. yes - gives scrollbars at all times; no - suppresses scrollbars--even when they are needed to see all the content;

	value). If the content is more than the frame size, a scroll bar appears.
SRC"URL"	Specifies the L for the document to render in the frame.

Designing HTML forms

FORMS

What is a form?

Forms are the tools to improve **user** interface in the web. Using forms, we design a web page on which a user can communicate his wish, opinion, suggestion, etc. When an advertisement is made for a seminar applicant can apply in a form attached to the advertisement. The form can contain interlace elements such as text fields, buttons, checkboxes, radio buttons, and selection lists that let users enter text and make choices, and then submit the form by clicking a "Submit" button.

The form can be submitted to a program on server or to an E-mail address.

FORM

The FORM element defines an interactive form. The form tag has three attributes.

action: Forms are used to get inputs from users. The user input s submitted to the server. The action attribute informs the browser the location of the server to which the form input has to be submitted.

For e.g. <form action= "index.php">

The name of the executable program and the location of the directory depend upon the web server.

Method: The method attribute has only two choices of values.

They are GET and POST.

These denote the protocol the server uses in implementing the, form's features. Usually the value used for the method attribute is POST.

This is the recommended protocol. With the post method, the information from the user is put into the data stream of the HTTP, and the back-end program can read the data as input through the "standard input" data stream.

In the case of GET, the data received in the form are placed at the end of the URL. If the form is very big and has a number of inputs, the GET method causes the URL to be very long. So this option is often discouraged.

Syntax:

<FORM

ACTION="action"

METHOD="GET"|"POST">

.....

</FORM>

Attributes:

Attribute	Description
ACTION="action"	Specifies the action the browser has to take on submit. action can be URL of a program on server that handles the form or mailto: E_mail_address.
METHOD="GET" "POST"	Specifies how the form input is sent to the server.

METHOD="GET" "POST"	Specifies how the form input is sent to the server.

INPUT

The INPUT element defines a form control for the user to INPUT must be contained within FORM.

Syntax:

<INPUT
NAME ="name"
SIZE="number"
TYPE="text | checkbox | radio | submit | reset| password |
VALUE="value" DISABLED
CHECKED
MAXLENGTH=number">

Attributes:

Attribute	Description
NAME="name"	Specifies the name of the control. Attribute is required, because data is sent as: name="value". The value is entered text for text control; value of VALUE attribute for radio and checkbox controls. Same variable name for all the buttons is given because the user can select only one of the radio button.
SIZE="number"	Specifies the size of text control
"text" "checkbox" "radio" TYPE= "submit" "reset" "password" "hidden"	Specifies the type of the control. **Text** - provides a single-line text input field; Used by default. **checkbox** - provides switch (square button) that can be turned on and off; **radio** - provides switch (radio button) that can be turned on and off; **submit** - defines a button for submitting the form. This button is used at the end of the user input. The data given by the user is sent to the server. **reset** - defines a button for resetting the form to its initial values; **Password** - Password field gets and sends text input from the user. But when the user types the input, only asterisks (*) are displayed on the screen; **Hidden** — hides the text typed by the user. This is only for maintaining privacy from other persons watching the monitor of the computer.
VALUE="value"	The value depends on the form control. The VALUE attribute: specifies the initial value for the text field; provides the text/label of the button for reset and submit controls; gives the value of the control when **it** is checked for radio and checkbox controls
CHECKED	Specifies that the control is initially checked for radio and checkbox controls.
MAXLENGTH	It specifies the maximum length that can be entered in the textbox.
DISABLED	It allows to disable the option

SELECT

The SELECT element defines a form control for the selection of options. SELECT must be contained within FORM. A Dropdown List is done using SELECT tag.

SIZE="number" MULTIPLE>
</SELECT>

Attributes:

Attribute	Description
NAME="name"	Specifies the name of the control. Attribute is required, because data is sent as: name="value". The value is determined by VALUE attribute of the selected OPTION
SIZE—"number"	Specifies the number of visible options. Using this option drop down list can be converted to list box.
MULTIPLE	Allows multiple selections.

OPTION:

The OPTION element defines a menu choice within a SELECT menu.

Syntax:

<OPTION
SELECTED
VALUE='value">
</OPTION>

Attributes:

Attribute	Description
SELECTED	Defines the OPTION to be initially selected.
VALUE="value"	Specifies the value of the option, sent with the submitted form. In the absence of a VALUE attribute, the value is the content of the OPTION element.

TEXTAREA

The TEXTAREA element defines a form control for the user to enter multi-line text input.
TEXTAREA must be contained within FORM.

Syntax:

<TEXTAREA
NAM E="name"
COLS="number"
ROWS="number">
</TEXTAREA>

Attributes:

Attributes:	Description
NAME="name"	Specifies the name of the control. Attribute is required, because data is sent as: name="value" The value is entered text.
COLS="number"	Specifies the number of visible columns.
ROWS="number"	Specifies the number of visible rows

ADVANCED ELEMENT OF HTML

<!Doctype> (Document type of Declaration)

- <!DOCTYPE> declaration is written at the top of the page before the <html> tag.
- It must only appear once in html document.

<!Doctype> (Document type of Declaration)

The <!DOCTYPE> declaration is written at the top of the page before the <html> tag.

- It must only appear once in html document.
- It is an instruction to the web browser about what version of HTML the page is written in.

(Such as HTML 1.0, HTML 2.0, HTML 3.0, HTML 3.2, HTML 4.01 , XHTML 1.0 HTML5).

- The <!Doctype> declaration is not case sensitive.
- The DOCTYPE declaration for HTML5 is very simple. The doctype declaration tells the Browser that this page is written in HTML5 as follows:

<!DOCTYPE html> OR <!doctype HTML >

<META>

- Metadata is data (information) about data.
- The <Meta> tag provides metadata about the HTML document.
- Metadata will not be displayed on the page.
- Meta elements are typically used to specify page description, keywords, author of the document, last modified and other metadata.
- <Meta> tag always written inside the <head> section.

Syntax:

```
<head>
<meta property/attribute =“value” property/attribute =“ value”>
</head>
```

Example:

```
<head>
<meta name=”description”  content=”Free Web tutorials”>
<meta name=”keywords” content=”HTML,CSS,XML,JavaScript”> </head>
```

INPUT TYPE ELEMENTS IN HTML5

(NUMBER, DATE, TIME, CALENDAR, RANGES):

1. Number:

- This element is used for input fields that should contain a numeric value.
- Numeric restrictions will apply in the input field.

Attribute	Description
max	Specifies the maximum number value for an input field
min	Specifies the minimum number value for an input field
step	Specifies the legal number intervals for an input field

Syntax:

```
<input type=”type of element name”
        max=”maximum value for input field”
        min=”minimum value for an input field”>
```

Example:

```
<form>
<input type=”number” min=”1” max=”5”> Quantity (between 1 and 5):
</form>
```


2. Date:

- This element is used for input fields that should contain a date.
- The input field can be display a date picker.

Syntax:

```
<input type="element name"
max="maximum value date for input field"
min="minimum value for date an input field">
```

Example:

```
<form>
<input type="date" name="bday" min="1979-12-31"><br>
<input type="date" name="bday" max="20 16-09-31 ">cbr>
</form>
```

3. Time:

- This element is use for allows the user to select a time (no time zone).
- The input field can be displayed a time picker.

Syntax:

```
<input type="element name" name="any name">
```

Example:

```
<form>
<input type= "time" name="usr time"> </form>
```

4. Range:

- This element is used for input fields that should contain a value within a range.
- The input field can be displayed as a slider control.

Attribute Description

max Specifies the maximum range of value for an input field.

min Specifies the minimum range of value for an input field

step Specifies the legal number intervals for an input field

Syntax:

```
<input type= "element name"
max= "maximum value of range for input field"
min= "minimum value of range for an input field">
```

Example:

```
<form>
<input type="range" name="points" min="0" max="10">
</form>
```

MULTIMEDIA TAGS:

- Web pages often contain multimedia elements of different types and formats.

Examples: Pictures, music, sound, videos, records, films, animations, and more.

- Before HTML5, there was no standard for showing videos/playing audio files on a web page.
- Before HTML5, videos could only be played with a plug-in (like flash).
- The HTML5 <video>&<audio> element specifies a standard way to embed a video& audio in a web page.

- To play an audio file in HTML, use the <audio> tag.
- You can also stream sound such as music or other audio streams.
- Any text inside the between <audio> and </audio> will be displayed in browsers that do not

. <audio>

- To play an audio file in HTML, use the <audio> tag.
- The <audio> tag defines sound, such as music or other audio streams.
- Any text inside the between <audio> and </audio> will be displayed in browsers that do not support the <audio> tag.
- Use <source> tag inside the <audio> tag.

Syntax:

```
<body>
<audio attributes/properties>
<source src="URL"  type="media_type">
<source src="URL"  type="media....type ">
any text type in here that do not support the <audio> tag
</audio>
</body>
```

Example:-

```
<body>
<audio controls autoplay>
<source src=" URL"  type="media_type">
<source src=" URL" type="media_type ">
Your browser does not support the audio element.
</audio >
</body>
```

Attribute property of <audio> tag are:

1. Controls:

o It Specifies that audio controls should be displayed (such as a play/pause/volume button etc)

Syntax:

<audio controls>

Example:

<audio controls> … </audio>

2. Autoplay:

o It Specifies that the audio will start playing as soon as it is ready

Syntax:

<audio autoplay>

Example:

<audio autoplay>
</audio>

3. loop:

o It specifies that the audio will start over again, every time it is finished.

Syntax:

<audio loop>

Example:

<audio controls loop>
</audio>

4. Muted:

It specifies that the audio output should be muted.

- it Specifies the URL of the audio file

Syntax:

<audio src="URL">

Example:

<audio src='horse.ogg" controls>
your browser does not support the audio element.
</audio>

<Source>:

- The <source> tag is used to specify multiple media resources for media elements, such as <video> and <audio>.
- The <source> tag allows you to specify alternative video/audio files.
- <source> tag defines inside between the <audio>&<video>tags.

  Syntax:

  <body>
  <audio attributes/properties>
  <source src="URL" type="media type">
  <source src=" URL" type=" media type">
  </audio>

- <video attributes/properties>
  <source src="URL" type="media type">
  <source src=" URL" type=" media type">
  </video>
  </body>