

e-PG PATHSHALA- Computer Science
Web Technology
Module 5

Component-I (B) Description of Module

Items	Description of Module
Subject Name	Computer Science
Paper Name	Web Technology
Module Name/Title	HTML
Module Id	CS/WT/5
Pre-requisites	None
Objectives	<ul style="list-style-type: none">• To discuss about Tables and its attributes in HTML.• To know about HTML Forms.
Keywords	HTML, Tables, Forms

e-PG Pathshala
Subject : Computer Science
Paper: Web Technology
Module: HTML
Module No: CS/WT/5
Quadrant 1 – e-text

Learning Objectives

The last module provides an introduction about the different Markup languages and discusses about the basic tags in HTML such as Formatting, Headings, Paragraphs and Lists in HTML. This module explains about Tables and its attributes in HTML. Moreover, this module also explains about HTML Forms .

Introduction

Tables provide a means of organizing the layout of data. A table is divided into rows and columns and these specify the cells of the table. Cells can contain text, images, links, other tables etc. The HTML tables allow web developers to arrange data like text, images, links, other tables, etc. into rows and columns of cells. Tables can also be used for organizing the layout of the web page itself. These HTML tables can be created using the **<table>** tag in which the **<tr>** tag is used to create table rows and **<td>** tag is used to create data cells. In simple, we can understand that the HTML Table Element **<table>** represents data in two dimensions or more.

HTML Tables

The **<TABLE></TABLE>** element has four sub-elements:

1. Table Row **<TR></TR>**.
2. Table Header **<THEAD></THEAD>** and Table Footer **<TFOOT></TFOOT>**.
3. Table Data **<TD></TD>**.
4. Caption **<CAPTION></CAPTION>**.

The Table rows can be grouped into a head, foot, and body sections using the THEAD, TFOOT and TBODY elements, respectively. Head section also called the header cell is defined with a <thead> element that contains the header information such as column names.

The <thead> tag must be used as a child of a <table> element, after any <caption>, and <colgroup> elements, and before any <tbody>, <tfoot>, and <tr> elements.

The Foot section is defined with a <tfoot> element are the footer rows that appears at the bottom of the table. The purpose of this <thead> and <tfoot> elements are that when long tables are printed, the head and foot information may be repeated on each page that contains table data.

The Table body is defined with a <tbody> element. Otherwise they are called as data cells defined with <td> element.

We can use at most one <thead> or <tfoot> but we can use multiple <tbody> elements in a table.

The caption tag is the descriptive title and used to provide a short description about the table's purpose or it can be used to name the table.

Example:

The anatomy of how a table can be created in HTML is shown in the below code.

```
<table border="1">
<tr>
<th> Column 1 header </th>
<th> Column 2 header </th>
</tr>
<tr>
<td> Row1, Col1 </td>
<td> Row1, Col2 </td>
</tr>
<tr>
<td> Row2, Col1 </td>
<td> Row2, Col2 </td>
</tr>
</table>
```

In the above example the table is defined with the <table> tag. Tables are divided into **table rows** with the <tr> tag. Table rows are divided into **table data** with the <td> tag. A table row can also be divided

into **table headings** with the **<th>** tag. By default, all major browsers display table headings as bold and centered:

We can also use **<th></th>** as **row headings** as well as **column headings**. One of the basic differences between **<thead>** and **<th>** is that **<thead>** is a block level element, whereas **<th>** is an inline-block. **<th>** also has a special attribute, **scope** so we can designate what it is a heading of, the column or the row.

Tables Attributes

Some of the table attributes that can be used with the **<table>** element is given below.

- **BGColor:** Some browsers support background colors in a table.
- **Width:** We can specify the table width as an absolute number of pixels or a percentage of the document width. We can set the width for the table cells as well.
- **Border:** We can choose a numerical value for the border width, which specifies the border in pixels.
- **CellSpacing:** Cell Spacing represents the space between cells and is specified in pixels.
- **CellPadding:** Cell Padding is the space between the cell border and the cell contents and is specified in pixels.
- **Align:** Tables can have left, right, or center alignment.
- **Background:** Background Image, will be titled in IE3.0 and above. **BorderColor**, **BorderColorDark**.

Example1: HTML Table: table1.html

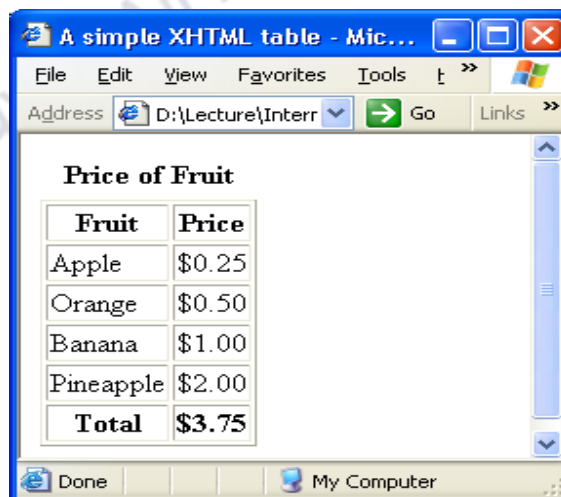
```
<table border = "1" width = "40%"  
  summary = "This table provides information about the price of fruit">  
  <caption><strong>Price of Fruit</strong></caption>  
  <thead>  
    <tr>  
      <th>Fruit</th>  
      <th>Price</th>  
    </tr>  
  </thead>  
  <tfoot>  
    <tr>  
      <th>Total</th>  
      <th>$3.75</th>  
    </tr>  
  </tfoot>
```

```

<tbody>
  <tr>
    <td>Apple</td>
    <td>$0.25</td>
  </tr>
  <tr>
    <td>Orange</td>
    <td>$0.50</td>
  </tr>
  <tr>
    <td>Banana</td>
    <td>$1.00</td>
  </tr>
  <tr>
    <td>Pineapple</td>
    <td>$2.00</td>
  </tr>
</tbody>
</table>

```

The above code displays the table with a caption as "Price of Fruit". The table has a <thead> element and a <tfoot> element. The table body starts with a <tbody> element which holds the content or data of the table. The table body has four rows. Table attributes such as <border>, <width> and <summary> has been used in the example. The attribute <summary> is new in HTML5. The output of the above code is shown in the below Figure 5.1.



Fruit	Price
Apple	\$0.25
Orange	\$0.50
Banana	\$1.00
Pineapple	\$2.00
Total	\$3.75

Figure 5.1

Example 2: HTML Table: table2.html

```
<html>
```

```

<head><title>Table Cells</title></head>
<body>

<table cellspacing="15" cellpadding="0">
  <tr><td>First</td>
    <td>Second</td></tr>
</table>

<br/>

<table cellspacing="0" cellpadding="10">
  <tr><td>First</td><td>Second</td></tr>
</table>
</body>
</html>

```

Example 2 explains about the table attributes cellspacing and cellpadding. The example specifies that there are two tables to differentiate and understand about cell spacing and cell padding. The first table has a cell spacing of 15 pixels that represents the space between cells. The second table has a cellpadding of 10 pixels which is the space between the cell border and the cell contents. The output of this HTML table is shown in the below Figure 5.2.

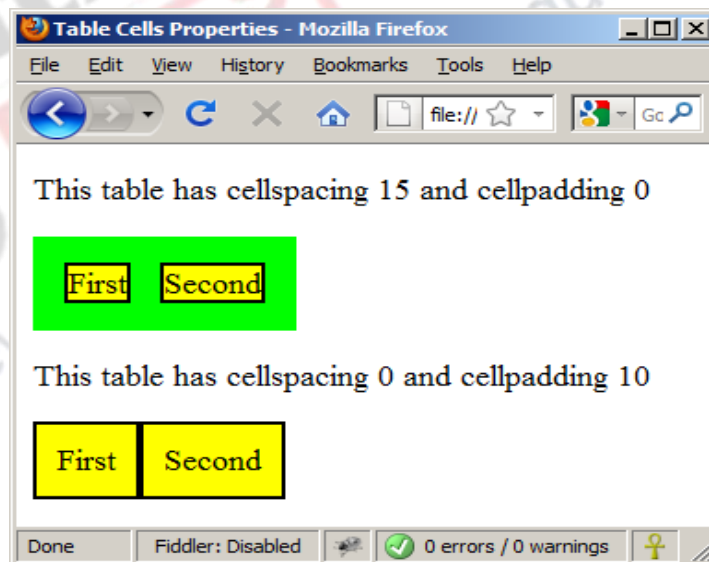


Figure 5.2

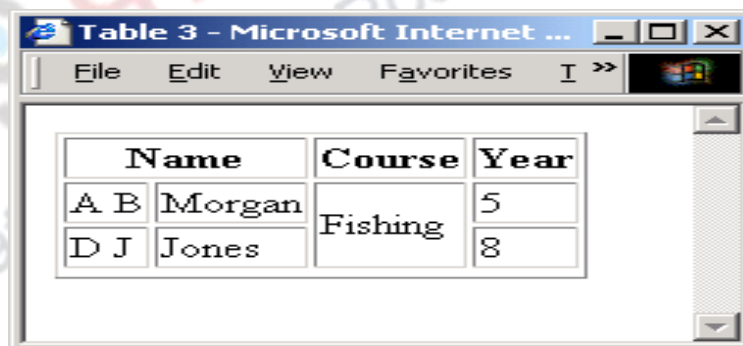
Rows and Columns

Cells can span multiple columns and multiple rows with the **colspan** and **rowspan** attributes. In order to make a cell span more than one column we can use the **colspan** attribute and to make a cell span more than one row we can use the **rowspan** attribute.

An example to show that how we use the colspan and rowspan attributes. The table is defined with three rows. The first row specifies the heading where the heading cell 'name' spans over two columns. In the second row, the data cell 'Fishing' spans over two rows. The output of the below code is shown in Figure 5.3.

Example 3:

```
<table border="1">
<tr>
  <th colspan="2">Name</th>
  <th>Course</th>
  <th>Year</th>
</tr>
<tr>
  <td>A B</td>
  <td>Morgan</td>
  <td rowspan="2">Fishing</td>
  <td>5</td>
</tr>
<tr>
  <td>D J</td>
  <td>Jones</td>
  <td>8</td>
</tr>
</table>
```



Name		Course	Year
A B	Morgan	Fishing	5
D J	Jones		8

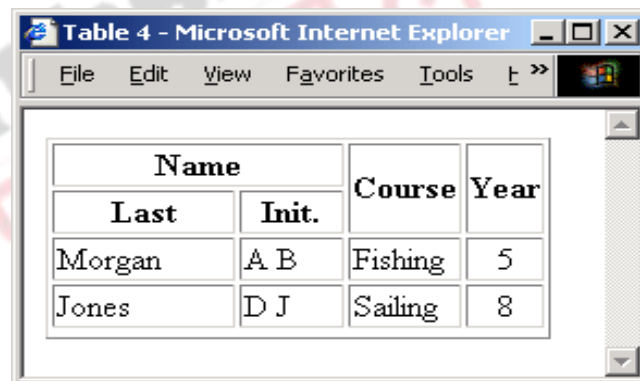
Figure 5.3

Example 4:

The table is defined with four rows. In the first row, the heading cell 'name' spans over two columns and the heading cell 'course' and 'year' spans over two rows. The second row has the data cell 'last' and 'init'.

The next two rows doesn't span over columns or rows. The output of the below code is shown in Figure 5.4.

```
<table border="1" align="center">
<tr>
  <th colspan="2" width="60%">Name</th>
  <th rowspan="2">Course</th>
  <th rowspan="2">Year</th>
</tr>
<tr>
  <th>Last</th>
  <th>Init.</th>
</tr>
<tr>
  <td>Morgan</td>
  <td>A B</td>
  <td>Fishing</td>
  <td align="center">5</td>
</tr>
<!-- etc -->
```



Name		Course	Year
Last	Init.		
Morgan	A B	Fishing	5
Jones	D J	Sailing	8

Figure 5.4

Introduction to Forms

An HTML form is an area of a document that contains normal content, markup, special elements called form elements or controls such as checkboxes, radio buttons, menus, text fields etc., and labels on those controls. Forms are used to create (rather primitive) GUIs on Web pages. Users generally use a form in a web page to send information to a web server or a mail server by entering text, selecting menu items, etc., for processing.

<form> is just another kind of HTML tag. The syntax to create a form is

<form *parameters*> ...*form elements*... </form>

Form elements include: buttons, checkboxes, text fields, radio buttons, drop-down menus, etc. Other kinds of tags can be mixed in with the form elements.

A form usually contains a **Submit button** to send the information in the form elements to the server

The form's parameters tell JavaScript how to send the information to the server. There are two different ways it could be sent to the server.

Forms can also be used for other things, such as a GUI for simple programs

The <form> tag

The <form **arguments**> ... </form> tag encloses form elements and probably other elements as well.

The arguments to form tell what to do with the user input. Below is the list of arguments that can be used with the <form> tag.

- action="**url**" (required)
 - Specifies where to send the data when the Submit button is clicked
- method="get" (default)
 - Form data is sent as a URL with ?form_data information appended to the end of the URL. This method can be used only if data is all ASCII and not more than 100 characters
- method="post"
 - Form data is sent in the body of the URL request and cannot be bookmarked by most browsers
- target="**target**"
 - Specifies where to open the page sent as a result of the request. If '**target=_blank**' means the page has to be opened in a new window. If '**target=_top**' means use the same window when the page is opened.

The <input> tag

Most, but not all, form elements use the input tag, with a type="..." argument to tell which kind of element it is. The type can be text, checkbox, radio, password, hidden, submit, reset, button, file, or image.

Other common input tag arguments include:

- name: the name of the element.

- id: a unique identifier for the element.
- value: the “value” of the element; used in different ways for different values of type.
- readonly: the value cannot be changed.
- disabled: the user can’t do anything with this element.

Other arguments are defined for the input tag but have meaning only for certain values of type

Text input

A text field can be created in a form using the following syntax.

```
<input type="text" name="textfield" value="with an initial value" />
```

The output of the above code is displayed in Figure 5.5.

A text field:

Figure 5.5

A multi-line text field can be created in a form using the below syntax.

```
<textarea name="textarea" cols="24" rows="2">Hello</textarea>
```

The output of the above code is displayed in Figure 5.6.

A multi-line text field

Figure 5.6

Note that two of these use the input tag, but one uses textarea.

Password field

```
<input type="password" name="textfield3" value="secret" />
```

The output of the above code is displayed in Figure 5.7.

A password field:

Figure 5.7

Buttons

A submit button can be created using the below syntax

```
<input type="submit" name="Submit" value="Submit" />
```

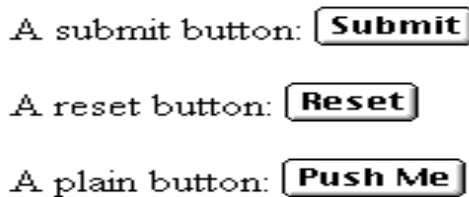
A reset button can be created using the below syntax

```
<input type="reset" name="Submit2" value="Reset" />
```

A plain button can be created using the below syntax

```
<input type="button" name="Submit3" value="Push Me" />
```

The output of the above code is displayed in Figure 5.8.



A submit button:

A reset button:

A plain button:

Figure 5.8

Radio buttons

Radio buttons are used when it is required to select one option from a set of alternatives. Radio buttons can be created using the below syntax,

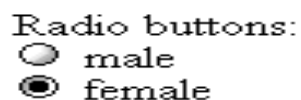
Radio buttons can be created using the below syntax,

Radio buttons:


```
<input type="radio" name="radiobutton" value="myValue1" />male<br>
```

```
<input type="radio" name="radiobutton" value="myValue2"  
checked="checked" />female
```

The output of the above code is displayed in Figure 5.9.



Radio buttons:
☐ male
☒ female

Figure 5.9

When a user clicks on a radio-button, it becomes checked, and all other radio-buttons with equal name become unchecked.

Labels

In many cases, the labels for controls are not part of the control. For example,

```
<input type="radio" name="gender" value="m" />male
```

In this case, clicking on the word “male” has no effect.

A label tag will bind the text to the control

```
<label><input type="radio" name="gender" value="m" />male</label>
```

With the above syntax, clicking on the word “male” now clicks the radio button.

Checkboxes

Checkboxes are used when more options are to be allowed at the same time. A checkbox can be created using the below syntax,

```
<input type="checkbox" name="checkbox" value="checkbox" checked="checked">
```

The output of the above code is displayed in Figure 5.10.

A checkbox: ☒

Figure 5.10

The attributes of this control are,

- type: "checkbox"
- name: used to reference this form element from JavaScript
- value: value to be returned when element is checked

Note that there is no text associated with the checkbox Unless we use a label tag, only clicking on the box itself has any effect.

Drop-down menu or list

A menu or list can be created using the following syntax,

```
<select name="select">
```

```

<option value="red">red</option>
<option value="green">green</option>
<option value="BLUE">blue</option>
</select>

```

The <select> element is used to create a drop-down list and the <option> tags inside the <select> element define the available options in the list.

The output of the above code is displayed in Figure 5.11.

A menu or list: 

Figure 5.11

The additional attributes of this control are,

- *size*: the number of items visible in the list (default is "1")
- *multiple*
 - If set to "true" (or just about anything else), any number of items may be selected.
 - If omitted, only one item may be selected.
 - If set to "false", behavior depends on the particular browser.

Hidden fields

Hidden fields are similar to text fields that does not show on the page.

A hidden field:

```

<input type="hidden" name="hiddenField" value="nyah">
  &lt;!-- right there, don't you see it?

```

The output of the above code is displayed in Figure 5.12.

A hidden field: <!-- right there, don't you see it?

Figure 5.12

The purpose of this hidden field is that all input fields are sent back to the server, including hidden fields. This is a way to include information that the user doesn't need to see or that we don't want others to see. The value of a hidden field can be set programmatically (by JavaScript) before the form is submitted

Now let us look at an example that includes the controls what we have seen earlier.

Example

In this example, a form is created with a text field and radio buttons.

```
<html>
<head>
<title>Get Identity</title>
</head>
<body>
<p><b>Who are you?</b></p>
<form method="post" action="">
  <p>Name:
    <input type="text" name="textfield">
  </p>
  <p>Gender:
    <label><input type="radio" name="gender" value="m" />Male</label>
    <label><input type="radio" name="gender" value="f" />Female</label>
  </p>
</form>
</body>
</html>
```

The output of the above code is displayed in Figure 5.13.

Who are you?

Name:

Gender: ☐ Male ☐ Female

Figure 5.13

Let us now look at a complete example that includes all the controls. The form that is displayed in the web page is shown in Figure 5.14.

The list of all controls has been summarized as below,

- text
- checkbox
- radio (buttons)

- select (options)
- textarea
- password
- button
- submit
- reset
- hidden
- file
- image

Tell us what you think

Name

Address

How did you hear about this web site?

A friend told me ☐

Via a search engine ☐

Followed a link (URL) ☐

How do you rate this site?

Good
 Good
 Bad
 Ugly

Please write your comments:

Do you want to receive any further information:

Thank you

Figure 5.14

Summary

In this module we looked at HTML in detail and explored about the Tables in HTML. This module also explains about the HTML Forms with syntax and examples in detail.