

❖ **CSS:**

- CSS stands for **Cascading Style Sheets**.
- **CSS describes** how HTML elements are to be **displayed**.
- **CSS saves a lot of work**. It can control the layout of multiple web pages all at once.
- External stylesheets are stored in **CSS files**

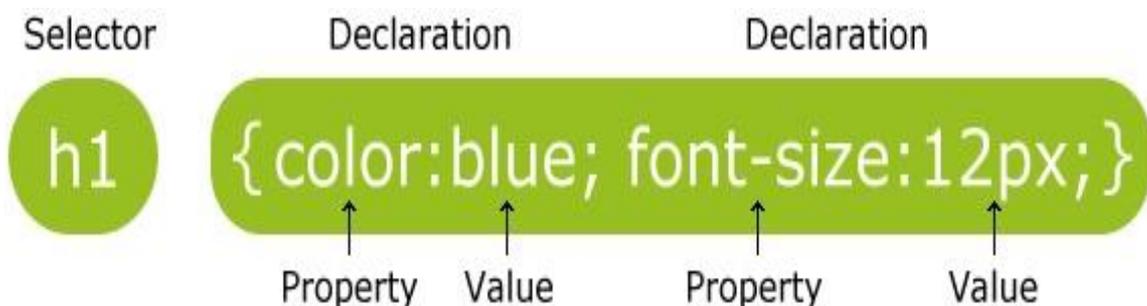
❖ **CSS Solved a Big Problem (how):**

- HTML was created to **describe the content** of a web page, like:

```
<h1>This is a heading</h1>  
<p>This is a paragraph.</p>
```
 - HTML was **NEVER** intended to contain tags for formatting a web page!
 - When tags like ``, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers. Development of large websites, where fonts and color information were added to every single page, became a long and expensive process.
- ✓ To solve this problem, the World Wide Web Consortium (W3C) created CSS. CSS removed the style formatting from the HTML page!

❖ **CSS Syntax and Selectors:**

A CSS rule-set consists of a selector and a declaration block:



- ✓ The selector points to the HTML element you want to style.
- ✓ The declaration block contains one or more declarations separated by semicolons.
- ✓ Each declaration includes a CSS property name and a value, separated by a colon.
- ✓ A CSS declaration always ends with a semicolon, and declaration blocks are surrounded by curly braces.

❖ **CSS Syntax and Selectors:**

CSS selectors are used to "find" (or select) HTML elements based on their element name, id, class, attribute, and more.

- **The element Selector:**

The element selector selects elements based on the element name.

You can select all <p> elements on a page like this (in this case, all <p> elements will be center-aligned, with a red text color):

Ex:

```
p { text-align: center; color: red; }
```

- **The id Selector:**

- The id selector uses the id attribute of an HTML element to select a specific element.

- The id of an element should be unique within a page, so the id selector is used to select one unique element!

- To select an element with a specific id, write a hash (#) character, followed by the id of the element.

- The style rule below will be applied to the HTML element with id="para1":

Ex: #para1 { text-align: center; color: red; }

Note: An id name cannot start with a number!

- **The class Selector:**

- The class selector selects elements with a specific class attribute.

- To select elements with a specific class, write a period (.) character, followed by the name of the class.

- In the example below, all HTML elements with class="center" will be red and center-aligned:

Ex:

```
.center {
  text-align: center;
  color: red;
}
```

You can also specify that only specific HTML elements should be affected by a class. In the example below, only <p> elements with class="center" will be center-aligned:

Ex:

```
p.center
{ text-align:
  center; color:
  red;
}
```

HTML elements can also refer to more than one class. In the example below, the <p> element will be styled according to class="center" and to class="large":

Ex:

```
<p class="center large">This paragraph refers to two classes.</p>
```

Note: An id name cannot start with a number!

- **Grouping Selectors:**

If you have elements with the same style definitions, It will be better to group the selectors, to minimize the code.

➤ To group selectors, separate each selector with a comma.

Ex:

```
h1, h2, p {
  text-align: center;
  color: red; }
```

- **CSS Comments:**

A CSS comment starts with /* and ends with */. Comments can also span multiple lines:

Ex:

```
p { color: red;
  /* This is a single-line comment */
  text-align: center; }
```

```
/* This is  
a multi-line  
comment */
```

❖ Ways to Insert CSS:

There are three ways of inserting a style sheet:

- ✓ External style sheet
- ✓ Internal style sheet
- ✓ Inline style

• External Style Sheet:

- ✓ With an external style sheet, you can change the look of an entire website by changing just one file!
- ✓ Each page must include a reference to the external style sheet file inside the <link> element. The <link> element goes inside the <head> section:

Ex:

```
<head>  
<link rel="stylesheet" type="text/css" href="mystyle.css">  
</head>
```

- ✓ An external style sheet can be written in any text editor. The file should not contain any html tags. The style sheet file must be saved with a .css extension.
- ✓ Here is how the "mystyle.css" looks:

```
body {  
  background-color: lightblue;  
}  
h1 {  
  color: navy;  
  margin-left: 20px;  
}
```

Note: Do not add a space between the property value and the unit (such as margin-left: 20 px;). The correct way is: margin-left: 20px;

• Internal Style Sheet:

- ✓ An internal style sheet may be used if one single page has a unique style.
- ✓ Internal styles are defined within the <style> element, inside the <head> section of an HTML page:

Ex:

```
<head>
<style>
  body {
    background-color: linen;
  }
  h1 {
    color: maroon; margin-left: 40px;
  }
</style>
</head>
```

- **Inline Styles:**

- ✓ An inline style may be used to apply a unique style for a single element.
- ✓ To use inline styles, add the style attribute to the relevant element. The style attribute can contain any CSS property.
- ✓ The example below shows how to change the color and the left margin of a <h1> element:

Ex:

```
<h1 style="color:blue;margin-left:30px;">This is a heading</h1>
```

- **Multiple Style Sheets:**

- ✓ If some properties have been defined for the same selector (element) in different style sheets, the value from the last read style sheet will be used.

Ex:

Assume that an external style sheet has the following style for the <h1> element:

```
h1 {
  color: navy;
}
```

then, assume that an internal style sheet also has the following style for the <h1> element:

```
h1 {
  color: orange;
}
```

If the internal style is defined after the link to the external style sheet, the <h1> elements will be "orange":

Ex:

```
<head>
```

```
<link rel="stylesheet" type="text/css" href="mystyle.css">
<style>
  h1 {
    color: orange;
  }
</style>
</head>
```

However, if the internal style is defined before the link to the external style sheet, the <h1> elements will be "navy":

Ex:

```
<head>
<style>
  h1 {
    color: orange;
  }
</style>
<link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
```

•Cascading Order:

What style will be used when there is more than one style specified for an HTML element?

Generally speaking we can say that all the styles will "cascade" into a new "virtual" style sheet by the following rules, where number one has the highest priority:

1. Inline style (inside an HTML element)
2. External and internal style sheets (in the head section)
3. Browser default

So, an inline style (inside a specific HTML element) has the highest priority, which means that it will override a style defined inside the <head> tag, or in an external style sheet, or a browser default value.

•CSS Backgrounds:

The CSS background properties are used to define the background effects for elements.

CSS background properties:

- background-color
- background-image
- background-repeat
- background-attachment
- background-position

- **Background-Color:**

- ✓ background-color property specifies the background color of an element.

The background color of a page is set like this:

Ex:

```
body {  
    background-color: lightblue;  
}
```

- ✓ With CSS, a color is most often specified by:

- a valid color name - like "red"
- a HEX value - like "#ff0000"
- an RGB value - like "rgb(255,0,0)"

- **Background-Image:**

- ✓ The background-image property specifies an image to use as the background of an element.
- ✓ By default, the image is repeated so it covers the entire element.

The background image for a page can be set like this:

Ex:

```
body {  
    background-image: url("paper.gif");  
}
```

- **background-repeat:**

- ✓ By default, a background-image is repeated both vertically and horizontally.
- ✓ The background-repeat property sets if/how a background image will be repeated.

```
background-repeat: no-repeat; /* No repeat */  
background-repeat: repeat-y; /* Repeat vertically */  
background-repeat: repeat-x; /* Repeat horizontally */
```

Ex:

```
body {  
    background-image: url("paper.gif");  
    background-repeat: no-repeat;  
}
```

- **background-attachment:**

The background-attachment property sets whether a background image is fixed or scrolls with the rest of the page.

Ex:

```
body {
```

```
background-image: url('w3css.gif');
background-repeat: no-repeat;
background-attachment: fixed;
}
```

- **background- position:**

The background-position property sets the starting position of a background image.

Ex:

```
body { background-image: url('smiley.gif');
background-repeat: no-repeat; background-attachment: fixed;
background-position: center;
}
```

- **CSS Box Model:**

- ✓ The CSS box model is essentially a box that wraps around every HTML element. It consists of: margins, borders, padding, and the actual content. The image below illustrates the box model:



Explanation of the different parts:

- **Content** - The content of the box, where text and images appear
- **Padding** - Clears an area around the content. The padding is transparent
- **Border** - A border that goes around the padding and content
- **Margin** - Clears an area outside the border. The margin is transparent

The box model allows us to add a border around elements, and to define space between elements.

Ex:

```
div {
    width: 300px;
    border: 25px solid green;
    padding: 25px;
    margin: 25px;
}
```

❖ CSS Borders:

The CSS border properties allow you to specify the style, width, and color of an element's border.

•Border Style:

The border-style property specifies what kind of border to display.

- ✓ dotted - Defines a dotted border
- ✓ dashed - Defines a dashed border
- ✓ solid - Defines a solid border
- ✓ double - Defines a double border
- ✓ groove - Defines a 3D grooved border. The effect depends on the border-color value
- ✓ ridge - Defines a 3D ridged border. The effect depends on the border-color value
- ✓ inset - Defines a 3D inset border. The effect depends on the border-color value
- ✓ outset - Defines a 3D outset border. The effect depends on the border-color value
- ✓ none - Defines no border
- ✓ hidden - Defines a hidden border

The border-style property can have from one to four values (for the top border, right border, bottom border, and the left border).

Ex:

```
p.dotted {border-style: dotted;}
p.dashed {border-style: dashed;}
p.solid {border-style: solid;}
p.double {border-style: double;}
```

```

p.groove {border-style: groove;}
p.ridge {border-style: ridge;}
p.inset {border-style: inset;}
p.outset {border-style: outset;}
p.none {border-style: none;}
p.hidden {border-style: hidden;}
p.mix {border-style: dotted dashed solid double;}

```

A dotted border.

A dashed border.

A solid border.

A double border.

A groove border. The effect depends on the border-color value.

A ridge border. The effect depends on the border-color value.

An inset border. The effect depends on the border-color value.

An outset border. The effect depends on the border-color value.

No border.

A hidden border.

A mixed border.

- **Border Width:**

- ✓ The border-width property specifies the width of the four borders.
- ✓ The width can be set as a specific size (in px, pt, cm, em, etc) or by using one of the three pre-defined values: thin, medium, or thick.
- ✓ The border-width property can have from one to four values (for the top border, right border, bottom border, and the left border).

Ex:

```

p.one { border-style: solid; border-width: 5px; }
p.two { border-style: solid; border-width: medium; }
p.three { border-style: solid; border-width: 2px 10px 4px 20px; }

```

- **Border Color:**

- ✓ The border-color property can have from one to four values (for the top border, right border, bottom border, and the left border).
- ✓ If border-color is not set, it inherits the color of the element.

Ex:

```
p.one { border-style: solid;
border-color: red;
}
p.three {
border-style: solid;
border-color: red green blue yellow;
}
```

You can also specify all the individual border properties for just one side:

Ex:

```
p { border-left: 6px solid red;
background-color: lightgrey;
}
```

- ❖ **CSS Margins:**

The CSS margin properties are used to create space around elements, outside of any defined borders.

With CSS, you have full control over the margins. There are properties for setting the margin for each side of an element (top, right, bottom, and left).

- **Margin - Individual Sides:**

CSS has properties for specifying the margin for each side of an element:

- ✓ margin-top
- ✓ margin-right
- ✓ margin-bottom
- ✓ margin-left

Ex:

```
p { margin-top: 100px;
margin-bottom: 100px;
margin-right: 150px;
margin-left: 80px;
}
```

❖ CSS Padding:

- ✓ The CSS padding properties are used to generate space around an element's content, inside of any defined borders.
- ✓ With CSS, you have full control over the padding. There are properties for setting the padding for each side of an element (top, right, bottom, and left).

•Padding - Individual Sides:

CSS has properties for specifying the padding for each side of an element:

- ✓ padding-top
- ✓ padding-right
- ✓ padding-bottom
- ✓ padding-left

Ex:

```
div {  
  padding-top: 50px;  
  padding-right: 30px;  
  padding-bottom: 50px;  
  padding-left: 80px;  
}
```

JavaScript

JavaScript is the world's most popular programming language. JavaScript is the programming language of the Web. JavaScript is easy to learn. JavaScript and [Java](#) are completely different languages, both in concept and design.

Why Study JavaScript?

JavaScript is one of the **3 languages** all web developers **must** learn:

1. [HTML](#) to define the content of web pages
2. [CSS](#) to specify the layout of web pages
3. **JavaScript** to program the behavior of web pages

JavaScript Can Change HTML Content

One of many JavaScript HTML methods is `getElementById()`.

The example below "finds" an HTML element (with id="demo"), and changes the element content (innerHTML) to "Hello JavaScript":

```
document.getElementById("demo").innerHTML = "Hello JavaScript";
```

JavaScript accepts both double and single quotes:

```
document.getElementById('demo').innerHTML = 'Hello JavaScript';
```

```
<!DOCTYPE html>
<html>
<body>

<h2>What Can JavaScript Do?</h2>

<p id="demo">JavaScript can change HTML content.</p>

<button type="button"
onclick='document.getElementById("demo").innerHTML =
"Hello JavaScript!'">Click Me!</button>

</body>
</html>
```

What Can JavaScript Do?

JavaScript can change HTML content.

Click Me!

After Run

What Can JavaScript Do?

Hello JavaScript!

Click Me!

JavaScript Can Change HTML Attribute Values

In this example JavaScript changes the value of the `src` (source) attribute of an `` tag:

```
<!DOCTYPE html>
<html>
<body>

<h2>What Can JavaScript Do?</h2>

<p>JavaScript can change HTML attribute values.</p>

<p>In this case JavaScript changes the value of the src
(source) attribute of an image.</p>

<button
onclick="document.getElementById('myImage').src='pic_bul
bon.gif'">Turn on the light</button>



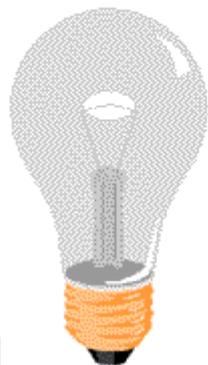
<button
onclick="document.getElementById('myImage').src='pic_bul
boff.gif'">Turn off the light</button>

</body>
</html>
```

What Can JavaScript Do?

JavaScript can change HTML attribute values.

In this case JavaScript changes the value of the src (source) attribute of an image.



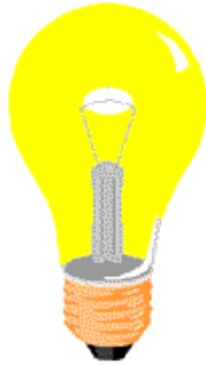
Turn on the light

Turn off the light

What Can JavaScript Do?

JavaScript can change HTML attribute values.

In this case JavaScript changes the value of the src (source) attribute of an image.



Turn on the light

Turn off the light

JavaScript Can Change HTML Styles (CSS)

Changing the style of an HTML element, is a variant of changing an HTML attribute:

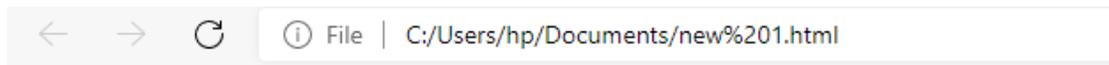
```
<!DOCTYPE html>
<html>
<body>

<h2>What Can JavaScript Do?</h2>

<p id="demo">JavaScript can change the style of an HTML
element.</p>

<button type="button"
onclick="document.getElementById('demo').style.fontSize='3
5px'">Click Me!</button>

</body>
</html>
```



What Can JavaScript Do?

JavaScript can change the style of an HTML element.

Click Me!

After Run



What Can JavaScript Do?

JavaScript can change the style of an HTML element.

Click Me!

JavaScript Can Hide HTML Elements

Hiding HTML elements can be done by changing the **display** style:

```
<!DOCTYPE html>
<html>
<body>

<h2>What Can JavaScript Do?</h2>

<p id="demo">JavaScript can hide HTML elements.</p>

<button type="button"
onclick="document.getElementById('demo').style.display='none'">Click Me!</button>

</body>
</html>
```

What Can JavaScript Do?

JavaScript can hide HTML elements.

Click Me!

After Run



JavaScript Where To

The `<script>` Tag

In HTML, JavaScript code is inserted between `<script>` and `</script>` tags.

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript in Body</h2>

<p id="demo"></p>

<script>
document.getElementById("demo").innerHTML = "My First
JavaScript";
</script>

</body>
</html>
```

JavaScript in Body

My First JavaScript

JavaScript Functions and Events

A JavaScript **function** is a block of JavaScript code that can be executed when "called" for. For example, a function can be called when an **event** occurs, like when the user clicks a button.

JavaScript in <head> or <body>

You can place any number of scripts in an HTML document. Scripts can be placed in the **<body>**, or in the **<head>** section of an HTML page, or in both.

JavaScript in <head>

In this example, a JavaScript **function** is placed in the **<head>** section of an HTML page. The function is invoked (called) when a button is clicked:

```
<!DOCTYPE html>
<html>
<head>
<script>
function myFunction() {
    document.getElementById("demo").innerHTML = "Paragraph
changed.";
}
</script>
</head>
<body>

<h2>JavaScript in Head</h2>

<p id="demo">A Paragraph.</p>

<button type="button" onclick="myFunction()">Try
it</button>

</body>
</html>
```

JavaScript in Head

A Paragraph.

Try it

JavaScript in Head

Paragraph changed.

Try it

JavaScript in <body>

In this example, a JavaScript **function** is placed in the **<body>** section of an HTML page. The function is invoked (called) when a button is clicked:

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript in Body</h2>

<p id="demo">A Paragraph.</p>

<button type="button" onclick="myFunction()">Try
it</button>

<script>
function myFunction() {
  document.getElementById("demo").innerHTML = "Paragraph
changed.";
}
</script>

</body>
</html>
```

JavaScript in Body

A Paragraph.

Try it

External JavaScript

Scripts can also be placed in external files:

```
function myFunction() {  
  document.getElementById("demo").innerHTML = "Paragraph changed.";  
}
```

External scripts are practical when the same code is used in many different web pages. JavaScript files have the file extension **.js**.

```
<script src="myScript.js"></script>
```

You can place an external script reference in **<head>** or **<body>** as you like.

The script will behave as if it was located exactly where the **<script>** tag is located.

External JavaScript Advantages

Placing scripts in external files has some advantages:

- It separates HTML and code
- It makes HTML and JavaScript easier to read and maintain
- Cached JavaScript files can speed up page loads

Lecture 1

Stander HTML Basics

3.1:History of HTML

As we discussed earlier in chapter one, (10.1). HTML is the language used to create web page documents and it is not a programming language; it is a client side markup language.

The first version of HTML didn't have a version number; it was just called "HTML" and was used to put up simple Web pages back in 1989 - 1995. In 1995, the IETF (Internet Engineering Task Force) standardized HTML and numbered it "HTML 2.0".

In 1997, the World Wide Web Consortium (W3C) presented the next version of HTML, HTML 3.2. It was followed by HTML 4.0 in 1998 and 4.01 in 1999.

Then the W3C announced that it would not be creating new versions of HTML, and would begin to focus on extensible HTML or XHTML. They recommend Web designers use HTML 4.01 for their HTML documents.

Around this point, Web designers didn't want to move to the rigid structure of XHTML, so in 2004, the Web Hypertext Application Technology Working Group (WHATWG) began working on a new version of HTML that is not as strict as XHTML called HTML5. The next table summarizes the development of HTML since the early days of the web.

Version	Year
HTML	1991
HTML+	1993
HTML 2.0	1995
HTML 3.2	1997
HTML 4.01	1999
XHTML	2000
HTML5	2012

3.2: Writing your first HTML code

HTML is a format that tells a computer how to display a web page. The documents themselves are plain text files with special "tags" or codes that a web browser uses to interpret and display information on your computer screen. Please consider the following:

- An HTML file is a text file containing small markup tags
- The markup tags tell the Web browser how to display the page
- An HTML file must have an htm or html file extension.

Example 1: First HTML page(Firstpage.html)

Open your text editor and type the following text:

```
1  <!DOCTYPE html>
2  <html>
3
4  <head>
5    <title>My First Web page</title>
6  </head>
7
8  <body>
9    <H1>This is my first homepage</H1>
10   <H6> This text is small text </H6>
11  </body>
12
13 </html>
```

Save the file as **firstpage.html**. Start using your favorite web browser like Internet Explorer, Google Chrome, or Firefox etc. it must show the following output:



Favorite Text Editor

We do recommend using Notepad ++ for writing HTML codes. It's free open source software and you can easy download it from the official web site www.notepad-plus-plus.org

Example Explained

What you just made is a basichtml document. This is the minimum required information for a web document and all web documents should contain these basic components.

<u>Line</u>	<u>Tag</u>	<u>Explanation</u>
1	<!DOCTYPE html>	<p>Used by the web browser to understand the version of the HTML used in the document. If you are using another HTML version like HTML 4.01 the declaration type will be like this:</p> <pre><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd"></pre> <p>If you are using HTML 5. Declaration will be like this:</p> <pre><!DOCTYPE html></pre>
2	<html>	<p>This tag tells your browser that this is the start of an html document. The last tag in your document is </html> in line 13. This tag tells your browser that this is the end of the html document.</p>
4	<head>	<p>Contain information header, which is not displayed in the browser window. The text between the <head> tag and the </head> tag is The text between the <title> tags</p>
5	<title>	<p>The <title> tag is used to uniquely identify each document and is also displayed in the title bar of the browser window.</p>
8	<body>	<p>The text between the <body> tags is the text that will be displayed in your browser.</p>

9	<h1>	The text between the <h1> and </h1> tags will be displayed in a bold font. <h6> is the smallest font you can have.
---	-------------------	--

HTM or HTML Extension?

When you save an HTML file, you can use either the .htm or the .html extension. The .htm extension comes from the past when some of the commonly used software only allowed three letter extensions. It is perfectly safe to use either .html or .htm, but be consistent. Firstpage.htm and mypage.html are treated as different files by the browser.

3.3:HTML elements

HTML elements are written with a **start** tag, with an **end** tag, with the content in between; <tagname>content</tagname>.

Keep in your mind that some HTML elements don't have an end tag like
.Back to our example, see the next table:

Start tag	Element contents	End tag
<title>		</title>
<h1 >	This is my first homepage	</h1 >
<h6 >	This text is small text	</h6 >

HTML elements can be nested (elements can contain elements).All HTML documents consist of nested HTML elements. Think of nesting tags Nesting means that when you put several pairs of tags into a document, you must balance each pair of tags, making sure they do not overlap.

It is easiest to understand with an example. So here is an example using the emphasis or italics tags, ``, and the strong or bold tags, ``:

Incorrect:

` ` ← this end tag is misplaced

Correct:

` `

HTML Tips: Use Lowercase Tags in
*HTML tags are not case sensitive: `<P>` means the same as `<p>`.
The HTML5 standard does not require lowercase tags, but W3C **recommends** lowercase in HTML4, and **demands** lowercase for stricter document types like XHTML.*

3.4: HTML Attributes

Attributes provide additional information about HTML elements. Attributes are always specified in the start tag comes in name/value pairs like: `name="value"`. See the next example:

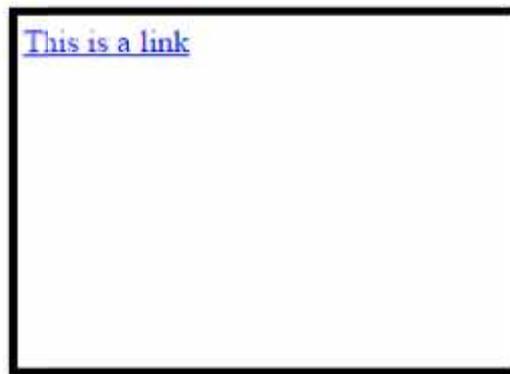
Example 2: Hyperlink Attributes (attributes.html)

```

1 <!DOCTYPE html>
2 <html>
3 <head >
4
5   <title> This is HTML attribute </title>
6
7 </head>
8 <body>
9
10  <a href="http://www.w3schools.com">This is a link</a>
11
12 </body>
13 </html>
14

```

The output of this code will be like this:



The Lang Attribute

The document language can be declared in the **<html>** tag. Declaring a language is important for accessibility applications (screen readers) and search engines:

Example 3: Languages attributes (lang.html)

```
1  <!DOCTYPE html >
2  <html lang="en-US">
3  <body>
4
5  <h1>My First Heading</h1>
6
7  <p>My first paragraph.</p>
8
9  </body>
10 </html>
11
```

The first two letters specify the language (en). If there is a dialect, use two more letters (US).

Size Attributes

HTML images are defined with the **** tag. The filename of the source (**src**), and the size of the image (**width** and **height**) are all provided as **attributes**:

```

```

The image size is specified in pixels: width="104" means 104 screen pixels wide. You will learn more about images and the tag later in this tutorial.

The altAttribute

The **alt** attribute specifies an alternative text to be used, when an HTML element cannot be displayed. The value of the attribute can be read by "screen readers". This way, someone "listening" to the webpage, i.e. a blind person, can "hear" the element.

Example

```

```

By using alt attribute. The text "W3Schools" will show up in your browser instead of `w3schools` image.

Always Use Lowercase Attributes

*The HTML5 standard does not require lower case attribute names. The title attribute can be written with upper or lower case like **Title** and/or **TITLE**. W3C **recommends** lowercase in HTML4, and **demands** lowercase for stricter document types like XHTML.*

Below is an alphabetical list of some attributes often used in HTML:

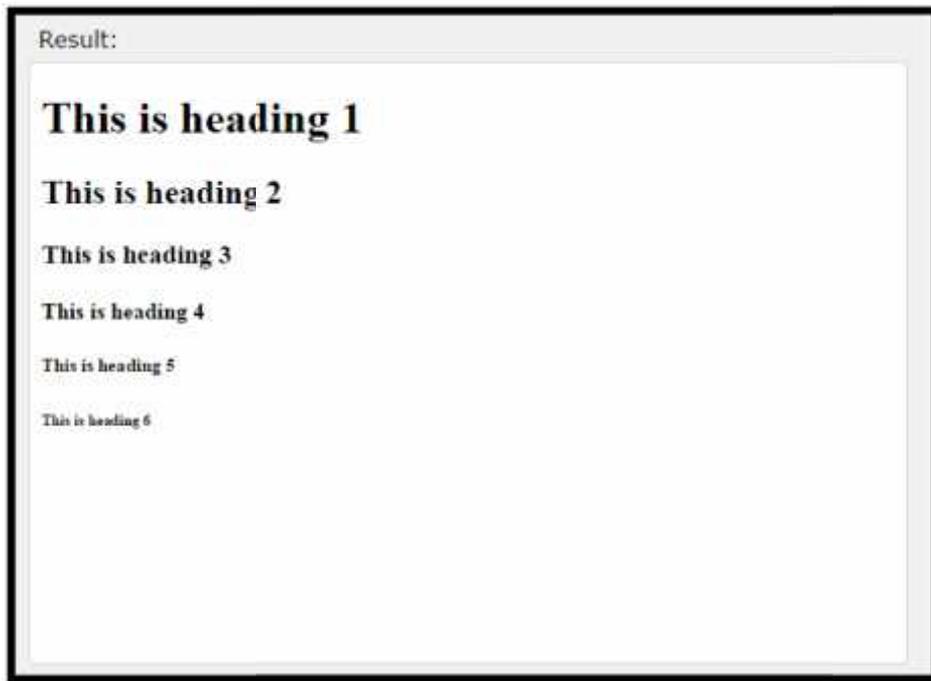
Attribute	Description
alt	Specifies an alternative text for an image
disabled	Specifies that an input element should be disabled
href	Specifies the URL (web address) for a link
id	Specifies a unique id for an element
src	Specifies the URL (web address) for an image
style	Specifies an inline CSS style for an element
title	Specifies extra information about an element (displayed as a tool tip)
value	Specifies the value (text content) for an input element.

5: HTML Heading and HTML paragraph

Headings are defined with the <h1> to <h6> tags.<h1> defines the most important heading. <h6> defines the least important heading.

Example 3: Languages attributes (lang.html)

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4
5 <h1>This is heading 1</h1>
6 <h2>This is heading 2</h2>
7 <h3>This is heading 3</h3>
8 <h4>This is heading 4</h4>
9 <h5>This is heading 5</h5>
10 <h6>This is heading 6</h6>
11
12 </body>
13 </html>
14
```



Use HTML headings for headings only. Don't use headings to make text **BIG** or **bold**. Search engines use your headings to index the structure and content of your web pages.

Users browse your pages by its headings. h1 headings should be main headings, followed by h2 headings, then the less important h3, and so on.

HTML Paragraphs

HTML documents are divided into paragraphs. As we discussed earlier, Don't use headings (h1 to h6) to make text **BIG** or **bold**. The HTML `<p>` element defines a **paragraph**.

```
<p>This is a paragraph</p>
```

```
<p>This is another paragraph</p>
```

3.6: HTML Display

You cannot be sure how HTML will be displayed. Large or small screens and resized windows will create different results. With HTML, you cannot change the output by adding extra spaces or extra lines in your HTML code.

The browser will remove extra spaces and extra lines when the page is displayed. Any number of spaces, and any number of new lines, counts as **only one space**.

Example 4: HTML Display (display.html)

```
1  <!DOCTYPE html>
2  <html>
3  <body>
4  <p>
5  This paragraph
6  contains a lot of lines
7  in the source code,
8  but the browser
9  ignores it.
10 </p>
11
12 <p>
13 This paragraph
14 contains      a lot of spaces
15 in the source  code,
16 but the      browser
17 ignores it.
18 </p>
19 </body>
20 </html>
```

The output of this code will be like this:

This paragraph contains a lot of lines in the source code, but the browser ignores it.
This paragraph contains a lot of spaces in the source code, but the browser ignores it.

The HTML `
` element defines a line break. It has no end tag. Use `
` if you want a line break (a new line) without starting a new paragraph:

Try this code:

```
<p>This is<br>a para<br>graph with line breaks</p>
```

The output will be like this:

This is
a para
graph with line breaks

Don't Forget the End Tag

Most browsers will display HTML correctly even if you forget the end tag, but don't rely on it. Forgetting the end tag can produce unexpected results or errors. Thus, stricter versions of HTML, like XHTML, do not allow you to skip the end tag.

The HTML `<pre>` tag defines a block of **pre-formatted** text, with structured spaces and lines. To display anything, with

right spacing and line-breaks, you must wrap the text in a **<pre>** element:

Example 5: HTML Pre-Format (preformat.html)

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4
5 <p>The pre tag is needed for displaying poems:</p>
6
7 <pre>
8     My Bonnie lies over the ocean.
9
10    My Bonnie lies over the sea.
11
12    My Bonnie lies over the ocean.
13
14    Oh, bring back my Bonnie to me.
15 </pre>
16
17 </body>
18 </html>
19
```

Try it yourself to see the output of this code:

4.7: HTML Format

If you use a word processor, you must be familiar with the ability to make text bold, italicized, or underlined; these are just three of the ten options available to indicate how text can appear in HTML and XHTML.

Bold Text

Anything that appears within `...` element, is displayed in bold as shown below:

Example 6: HTML Format (boldtext.html)

```
<!DOCTYPE html>
<html>
<head>
<title>Bold Text Example</title>
</head>
<body>
<p>The following word uses a bold typeface.</p>
</body>
</html>
```

Italic Text

Anything that appears within `<i>...</i>` element is displayed in italicized as shown below:

Example 7: HTML Format (italictext.html)

```
<!DOCTYPE html>
<html>
<head>
<title>Italic Text Example</title>
</head>
<body>
<p>The following word uses a italicized typeface.</p>
</body>
</html>
```

Underlined Text

Anything that appears within `<u>...</u>` element, is displayed with underline as shown below:

Example 8: HTML Format (underlinetext.html)

```
<!DOCTYPE html>
<html>
<head>
<title>Underlined Text Example</title>
</head>
<body>
<p>The following word uses a underlined typeface.</p>
</body>
</html>
```

Strike Text

Anything that appears within `<strike>...</strike>` element is displayed with strikethrough, which is a thin line through the text as shown below:

Example 9: HTML Format (striketext.html)

```
<!DOCTYPE html>
<html>
<head>
<title>Strike Text Example</title>
</head>
<body>
<p>The following word uses a strikethrough typeface.</p>
</body>
</html>
```

This will produce following result:

The following word uses a ~~strikethrough~~ typeface.

Superscript Text

The content of a `^{...}` element is written in superscript; the font size used is the same size as the characters surrounding it but is displayed half a character's height above the other characters.

Example 10: HTML Format (`superscripttext.html`)

```
<!DOCTYPE html>
<html>
<head>
<title>Superscript Text Example</title>
</head>
<body>
<p>The following word uses a <sup>superscript</sup> typeface.</p>
</body>
</html>
```

This will produce following result:

The following word uses a ^{superscript} typeface.

Inserted Text

Anything that appears within `<ins>...</ins>` element is displayed as inserted text.

Example 11: HTML Format (`inserttext.html`)

```
<!DOCTYPE html>
<html>
<head>
<title>Inserted Text Example</title>
</head>
<body>
<p>I want to drink <del>cola</del><ins>wine</ins></p>
</body>
```

```
</html>
```

This will produce following result:

I want to drink ~~cola~~ wine

Larger Text

The content of the `<big>...</big>` element is displayed one font size larger than the rest of the text surrounding it as shown below:

Example 12: HTML Format (**large**text.html)

```
<!DOCTYPE html>
<html>
<head>
<title>Larger Text Example</title>
</head>
<body>
<p>The following word uses a <big>big</big> typeface.</p>
</body>
</html>
```

This will produce following result:

The following word uses a **big** typeface.

For smaller Text, The content of the `<small>...</small>` element is displayed one font size smaller than the rest of the text surrounding it.

Grouping Content

The `<div>` and `` elements allow you to group together several elements to create sections or subsections of a page.

For example, you might want to put all of the footnotes on a page within a <div> element to indicate that all of the elements within that <div> element relate to the footnotes. You might then attach a style to this <div> element so that they appear using a special set of style rules.

Example 12: HTML Format (**grouping.html**)

```
<!DOCTYPE html>
<html>
<head>
<title>Div Tag Example</title>
</head>
<body>
<div id="menu" align="middle">
<a href="/index.htm">HOME</a> |
<a href="/about/contact_us.htm">CONTACT</a> |
<a href="/about/index.htm">ABOUT</a>
</div>

<div id="content" align="left" bgcolor="white">
<h5>Content Articles</h5>
<p>Actual content goes here.....</p>
</div>
</body>
</html>
```

This will produce following result:

HOME | CONTACT | ABOUT

CONTENT ARTICLES

Actual content goes here.....

3.8: HTML Special Characters

Some characters have a special meaning in HTML, like the less than sign (<) that defines the start of an HTML tag. If we want the browser to actually display these characters we must insert character entities in place of the actual characters themselves. The Most Common Character Entities:

Entity	Entity Name	Description	Resu
 	 	non-breaking	
<	<	less than	<
>	>	greater than	>
&	&	ampersand	&
"	"	quotation mark	"
'	'	apostrophe (does not work in)	'

A character entity has three parts: an ampersand (&), an entity name or an entity number, and finally a semicolon (;). The & means we are beginning a special character, the; means ending a special character and the letters in between are sort of an abbreviation for what it's for. To display a less than sign in an HTML document we must write: < or < the advantage of using a name instead of a number is that a name is easier to remember. The disadvantage is that not all browsers support the newest entity names, while the support for entity numbers is very good in almost all browsers.

Non-breaking Space is the most common character entity in HTML is the non-breaking space ** **. Normally HTML will truncate spaces in your text. If you add 10 spaces in

If you want to use a background image, you should keep in mind:

- Will the background image increase the loading time too much?
- Will the background image look good with other images on the page?
- Will the background image look good with the text colors on the page?
- Will the background image look good when it is repeated on the page?
- Will the background image take away the focus from the text?

Try this out:

Example 13: HTML Background (**background.html**)

Example: HTML Background

```
<html>
<head>
<title>My First Webpage</title>
</head>
<body background="http://profdevtrain.austincc.edu/html/graphics/clouds.gif" bgcolor="#E9D9E9">
<h1 align="center">My First Webpage</h1>
<p>Welcome to my <strong>first</strong> webpage. I am writing this page using a text editor and plain old
html.</p>
<p>
By learning html, I'll be able to create webpages like a <del>beginner</del> pro....<br>
which I am of course.
</p>
</body>
</html>
```

The output will be like this:



3.10: Color Values

Colors are defined using a hexadecimal notation for the combination of red, green, and blue color values (RGB). The lowest value that can be given to one light source is 0 (hex #00). The highest value is 255 (hex #FF). This table shows the result of combining red, green, and blue:

Color	Color HEX
	#000000
	#FF0000
	#00FF00
	#0000FF
	#FFFF00
	#00FFFF
	#FF00FF
	#C0C0C0
	#FFFFFF

Here is the example to set background of an HTML tag by color code in hexadecimal

Example 14: HTML Colors (**color.html**)

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Colors by Hex</title>
</head>
<bodytext="#0000FF"bgcolor="#00FF00">
<p>Use different color hexa for for body and table and see the result.</p>
<tablebgcolor="#000000">
<tr>
<td>
<fontcolor="#FFFFFF">This text will appear white on black background.</font>
</td>
</tr>
</table>
</body>
</html>
```

3.11: HTML Lists

HTML provides a simple way to show unordered lists (bullet lists) or ordered lists (numbered lists).

Unordered Lists

An unordered list is a list of items marked with bullets (typically small black circles). An unordered list starts with the `` tag. Each list item starts with the `` tag.

Would Display	This Code
<ul style="list-style-type: none">• f Coffee• f Milk	<pre> Coffee Milk </pre>

Ordered Lists

An ordered list is also a list of items. The list items are marked with numbers. An ordered list starts with the `` tag. Each list item starts with the `` tag.

Would Display	This Code
<ol style="list-style-type: none">1. Coffee2. Milk	<pre> Coffee Milk </pre>

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

Try It Out

Example 15: HTML lists (`lists.html`)

```
1 <html>
2 <head>
3   <title>My Webpage</title>
4 </head>
5 <body bgcolor="#EDDD9E">
6   <h1 align="center">My Webpage</h1>
7   <p>By learning html, I'll be able to create web pages like a pro....<br>
8   which I am of course.</p> Here's what I've learned:
9   <ul>
10    <li>How to use HTML tags</li>
11    <li>How to use HTML colors</li>
12    <li>How to create Lists</li>
13  </ul>
14  <ol>
15    <li>How to use HTML tags</li>
16    <li>How to use HTML colors</li>
17    <li>How to create Lists</li>
18  </ol>
19 </body>
20 </html>
```

3.11: HTML Links

A webpage can contain various links that take you directly to other pages and even specific parts of a given page. These links are known as hyperlinks.

Hyperlinks allow visitors to navigate between Web sites by clicking on words, phrases, and images. Thus you can create hyperlinks using text or images available on a webpage.

Linking Documents

A link is specified using HTML tag `<a>`. This tag is called anchor tag and anything between the opening `<a>` tag and the closing `` tag becomes part of the link and a user can click that part to reach to the linked document. Following is the simple syntax to use `<a>` tag.

```
<ahref="Document URL" ... attributes-list>Link Text</a>
```

Example 16: HTML Links (**links.html**)

```
<!DOCTYPE html>
<html>
<head>
<title>Hyperlink Example</title>
</head>
<body>
<p>Click following link</p>
<a href="http://www.facebook.com" target="_self">Facebook</a>
</body>
</html>
```

This will produce following result, where you can click on the link generated Facebook reach to the home page of google.

Click following link

[Facebook](http://www.facebook.com)

The target Attribute

We have used target attribute in our previous example. This attribute is used to specify the location where linked document is opened. Following are possible options:

Option	Description
_blank	Opens the linked document in a new window or tab.
_self	Opens the linked document in the same frame.
_parent	Opens the linked document in the parent frame.
_top	Opens the linked document in the full body of the window.

targetframe Opens the linked document in a named *targetframe*.

After trying the following example, you will understand the basic difference in few options given for target attribute.

Example 17: HTML link2 (**link2.html**)

```
<!DOCTYPE html>
<html>
<head>
<title>Hyperlink Example</title>
<basehref="http://www.facebook.com/">
</head>
<body>
<p>Click any of the following links</p>
<a href="/html/index.htm" target="_blank">Opens in New</a> |
<a href="/html/index.htm" target="_self">Opens in Self</a> |
<a href="/html/index.htm" target="_parent">Opens in Parent</a> |
<a href="/html/index.htm" target="_top">Opens in Body</a>
</body>
</html>
```

This will produce following result, where you can click on different links to understand the difference between various options given for target attribute.

Click any of the following links

Use of Base Path

When you link HTML documents related to the same website, it is not required to give a complete URL for every link. You can get rid of it if you use <base> tag in your HTML document header. This tag is used to give a base path for all the links. So your browser will concatenate given relative path to this base path and will make a complete URL.

Linking to a Page Section

You can create a link to a particular section of a given webpage by using name attribute. This is a two-step process.

First create a link to the place where you want to reach within a webpage and name it using <a...> tag as follows:

```
<h1>HTML Text Links <aname="top"></a></h1>
```

Second step is to create a hyperlink to link the document and place where you want to reach:

```
<a href="/html/html_text_links.htm#top">Go to the Top</a>
```

This will produce following link, where you can click on the link generated Go to the Top to reach to the top of the HTML Text Link **web page**.

[Go to the Top](#)

Setting Link Colors

You can set colors of your links, active links and visited links using **link**, **alink** and **vlink** attributes of <body> tag.

Try the following example to see how **link**, **alink** and **vlink** attribute work.

Example 18: HTML link3 (**link3**.html)

```
<!DOCTYPE html>
<html>
<head>
<title>Hyperlink Example</title>
<basehref="http://www.tutorialspoint.com/">
</head>
<bodyalink="#54A250"link="#040404"vlink="#F40633">
<p>Click following link</p>
<a href="/html/index.htm"target="_blank">HTML Tutorial</a>
</body>
</html>
```

Just check color of the link before clicking on it, next check its color when you activate it and when the link has been visited.

Download Links

You can create text link to make your PDF, or DOC or ZIP files downloadable. This is very simple, you just need to give complete URL of the downloadable file as follows:

Example 19: HTML link4 (**link4.html**)

```
<!DOCTYPE html>
<html>
<head>
<title>Hyperlink Example</title>
</head>
<a href="http://www.tutorialspoint.com/page.pdf">Download PDF File</a>
</body>
</html>
```

This will produce following link and will be used to download a file.

Download PDF File

3.13: HTML Images

Images are very important to beautify as well as to depict many complex concepts in simple way on your web page. This tutorial will take you through simple steps to use images in your web pages.

Insert Image

You can insert any image in your web page by using `` tag. Following is the simple syntax to use this tag.

```

```

The `` tag is an empty tag, which means that it can contain only list of attributes and it has no closing tag. To try following example, let's keep our HTML file `image.htm` and image file `test.png` in the same directory:

Example 20: HTML Images (**image1.html**)

```
<html>
<head>
<title>Using Image in Webpage</title>
</head>
<body>
<p>Simple Image Insert</p>
<imgsrc="test.png"alt="Test Image"/>
</body>
</html>
```

You can use PNG, JPEG or GIF image file based on your comfort but **make sure you specify correct image file name in src attribute**. Image name is always case sensitive.

The **alt** attribute is an optional attribute which specifies an alternate text for an image, if the image cannot be displayed.

Set Image Location

Usually we keep our all the images in a separate directory. So let's keep HTML file `image.htm` in our home directory and create a subdirectory **images** inside the home directory where we will keep our image `test.png`.

Assuming our image location is "image/test.png", try the following example:

Example 21: HTML Images (**image2.html**)

```
<html>
<head>
<title>Using Image in Webpage</title>
</head>
<body>
<p>Simple Image Insert</p>
<imgsrc="images/test.png"alt="Test Image"/>
</body>
</html>
```

Set Image Width/Height

You can set image width and height based on your requirement using **width** and **height** attributes. You can specify width and height of the image in terms of either pixels or percentage of its actual size.

Example 22: HTML Images (**image3.html**)

```
<html>
<head>
<title>Set Image Width and Height</title>
</head>
<body>
<p>Setting image width and height</p>
<imgsrc="test.png"alt="Test Image"width="150"height="100"/>
</body>
</html>
```

Set Image Border

By default image will have a border around it, you can specify border thickness in terms of pixels using border attribute. A thickness of zero (0) means, no border around the picture.

Example 23: HTML Images (**image4.html**)

```
<!DOCTYPE html>
<html>
<head>
<title>Set Image Border</title>
</head>
<body>
<p>Setting image Border</p>
<imgsrc="test.png"alt="Test Image"border="3"/>
</body>
</html>
```

Set Image Alignment

By default image will align at the left side of the page, but you can use align attribute to set it in the center or right.

Example 24: HTML Images (**image5.html**)

```
<!DOCTYPE html>
<html>
<head>
<title>Set Image Alignment</title>
</head>
<body>
<p>Setting image Alignment</p>
```

```
<imgsrc="test.png"alt="Test Image"border="3"align="right"/>
</body>
</html>
```

3.14: HTML Tables

The HTML tables allow web authors to arrange data like text, images, links, other tables, etc. into rows and columns of cells.

The HTML tables are 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.

Example 25: HTML Tables (**table1.html**)

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Tables</title>
</head>
<body>
<tableborder="1">
<tr>
<td>Row 1, Column 1</td>
<td>Row 1, Column 2</td>
</tr>
<tr>
<td>Row 2, Column 1</td>
<td>Row 2, Column 2</td>
</tr>
</table>
```

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

This will produce following result:

Row 1, Column 1	Row 1, Column 2
Row 2, Column 1	Row 2, Column 2

Here border is an attribute of <table> tag and it is used to put a border across all the cells. If you do not need a border then you can use border="0".

Table Heading

Table heading can be defined using <th> tag. This tag will be put to replace <td> tag, which is used to represent actual data cell. Normally you will put your top row as table heading as shown below, otherwise you can use <th> element in any row.

Example 26: HTML Tables (**table2.html**)

```
<!DOCTYPE html>  
<html>  
<head>  
<title>HTML Table Header</title>  
</head>  
<body>  
<tableborder="1">  
<tr>  
<th>Name</th>
```

```
<th>Salary</th>
</tr>
<tr>
<td>Ramesh Raman</td>
<td>5000</td>
</tr>
<tr>
<td>Shabbir Hussein</td>
<td>7000</td>
</tr>
</table>
</body>
</html>
```

This will produce following result:

Name	Salary
Ramesh Raman	5000
Shabbir Hussein	7000

Cellpadding and Cellspacing Attributes

There are two attributes called **cellpadding** and **cellspacing** which you will use to adjust the white space in your table cells. The **cellspacing** attribute defines the width of the border, while **cellpadding** represents the distance between cell borders and the content within a cell.

Example 27: HTML Tables (**table3.html**)

```
<!DOCTYPE html>
```

```
<html>
<head>
<title>HTML Table Cellpadding</title>
</head>
<body>
<tableborder="1"cellpadding="5"cellspacing="5">
<tr>
<th>Name</th>
<th>Salary</th>
</tr>
<tr>
<td>Ramesh Raman</td>
<td>5000</td>
</tr>
<tr>
<td>Shabbir Hussein</td>
<td>7000</td>
</tr>
</table>
</body>
</html>
```

This will produce following result:

Name	Salary
Ramesh Raman	5000
Shabbir Hussein	7000

Colspan and Rowspan Attributes

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

Example 28: HTML Tables (**table4.html**)

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Table Colspan/Rowspan</title>
</head>
<body>
<tableborder="1">
<tr>
<th>Column 1</th>
<th>Column 2</th>
<th>Column 3</th>
</tr>
<tr><tdrowspan="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><tdcolspan="3">Row 3 Cell 1</td></tr>
</table>
</body>
</html>
```

This will produce following result:

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		

Tables Backgrounds

You can set table background using one of the following two ways:

- **bgcolor** attribute - You can set background color for whole table or just for one cell.
- **background** attribute - You can set background image for whole table or just for one cell.

You can also set border color also using **bordercolor** attribute.

Example 29: HTML Tables (**table5.html**)

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Table Background</title>
</head>
<body>
<tableborder="1"bordercolor="green"bgcolor="yellow">
<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 colspan="3">Row 3 Cell 1</td></tr>
</table>
</body>
</html>
```

This will produce following result:

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

Here is an example of using background attribute. Here we will use an image available in /images directory.

Example 30: HTML Tables (**table6.html**)

```

<!DOCTYPE html>
<html>
<head>
<title>HTML Table Background</title>
</head>
<body>
<tableborder="1"bordercolor="green"background="/images/test.png">
<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 colspan="3">Row 3 Cell 1</td></tr>
</table>
</body>
</html>

```

This will produce following result. Here background image did not apply to table's header.

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		

Table Height and Width

You can set a table width and height using `width` and `height` attributes. You can specify table width or height in terms of pixels or in terms of percentage of available screen area.

Example 31: HTML Tables (`table7.html`)

```

<!DOCTYPE html>
<html>
<head>
<title>HTML Table Width/Height</title>
</head>
<body>
<table border="1" width="400" height="150">
<tr>
<td>Row 1, Column 1</td>
<td>Row 1, Column 2</td>
</tr>
<tr>
<td>Row 2, Column 1</td>
<td>Row 2, Column 2</td>
</tr>
</table>
</body>
</html>

```

This will produce following result:

Row 1, Column 1	Row 1, Column 2
Row 2, Column 1	Row 2, Column 2

Table Caption

The caption tag will serve as a title or explanation for the table and it shows up at the top of the table. This tag is **depracated** in newer version of HTML/XHTML.

Example 32: HTML Tables (**table8.html**)

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Table Caption</title>
</head>
<body>
<tableborder="1"width="100%">
<caption>This is the caption</caption>
<tr>
<td>row 1, column 1</td><td>row 1, columnn 2</td>
</tr>
<tr>
<td>row 2, column 1</td><td>row 2, columnn 2</td>
</tr>
</table>
</body>
</html>
```

This will produce following result:

This is the caption	
row 1, column 1	row 1, columnn 2
row 2, column 1	row 2, columnn 2

Table Header, Body, and Footer

Tables can be divided into three portions: a header, a body, and a foot. The head and foot are rather similar to headers and footers in a word-processed document that remain the same for every page, while the body is the main content holder of the table.

The three elements for separating the head, body, and foot of a table are:

- **<thead>** - to create a separate table header.
- **<tbody>** - to indicate the main body of the table.
- **<tfoot>** - to create a separate table footer.

A table may contain several `<tbody>` elements to indicate different pages or groups of data. But it is notable that `<thead>` and `<tfoot>` tags should appear before `<tbody>`

Example 33: HTML Tables (**table9.html**)

```

<!DOCTYPE html>
<html>
<head>
<title>HTML Table</title>
</head>
<body>
<tableborder="1"width="100%">
<thead>

```

```
<tr>
<td colspan="4">This is the head of the table</td>
</tr>
</thead>
<tfoot>
<tr>
<td colspan="4">This is the foot of the table</td>
</tr>
</tfoot>
<tbody>
<tr>
<td>Cell 1</td>
<td>Cell 2</td>
<td>Cell 3</td>
<td>Cell 4</td>
</tr>
</tbody>
</table>
</body>
</html>
```

This will produce following result:

This is the head of the table			
This is the foot of the table			
Cell 1	Cell 2	Cell 3	Cell 4

HTML Cheat Sheet

Work Sheet

Q1: ?

Work Sheet

Lecture 2

Advance HTML

4.1:HTMLforms

HTML Forms are required when you want to collect some data from the site visitor. For example during user registration you would like to collect information such as name, email address, credit card, etc.

A form will take input from the site visitor and then will post it to a back-end application such as CGI, ASP Script or PHP script etc. The back-end application will perform required processing on the passed data based on defined business logic inside the application.

There are various form elements available like text fields, textarea fields, drop-down menus, radio buttons, checkboxes, etc.

The HTML <form> tag is used to create an HTML form and it has following syntax:

```
<form action="Script URL" method="GET|POST">  
form elements like input, textarea etc.  
</f
```

Form Attributes

Apart from common attributes, following is a list of the most frequently used form attributes:

Attribute	Description
action	Backend script ready to process your passed data.
method	Method to be used to upload data. The most frequently used are GET and POST methods.

target	Specify the target window or frame where the result of the script will be displayed. It takes values like <code>_blank</code> , <code>_self</code> , <code>_parent</code> etc.
enctype	<p>You can use the <code>enctype</code> attribute to specify how the browser encodes the data before it sends it to the server. Possible values are:</p> <ul style="list-style-type: none"> • Application/x-www-form-urlencoded - This is the standard method most forms use in simple scenarios. • multipart/form-data - This is used when you want to upload binary data in the form of files like image, word file etc.

HTML Form Controls

There are different types of form controls that you can use to collect data using HTML form:

- Text Input Controls
- Checkboxes Controls
- Radio Box Controls
- Select Box Controls
- File Select boxes
- Hidden Controls
- Clickable Buttons
- Submit and Reset Button

4.1.1 Text Input Controls

There are three types of text input used on forms:

Single-line text input controls

This control is used for items that require only one line of user input, such as search boxes or names. They are created using HTML `<input>` tag.

Example

Here is a basic example of a single-line text input used to take first name and last name:

```
<!DOCTYPE html>
<html>
<head>
<title>Text Input Control</title>
</head>
<body>
<form>
First name: <input type="text" name="first_name" />
<br>
Last name: <input type="text" name="last_name" />
</form>
</body>
</html>
```

This will produce following result:

First name:

Last name:

Attributes

Following is the list of attributes for <input> tag for creating text field.

Attribute	Description
type	Indicates the type of input control and for text input control it will be set to text .
name	Used to give a name to the control which is sent to the server to be recognized and get the value.
value	This can be used to provide an initial value inside the control.
size	Allows to specify the width of the text-input control in terms of characters.
maxlength	Allows to specify the maximum number of characters a user can enter into the text box.

4.1.2 Password input controls

This is also a single-line text input but it masks the character as soon as a user enters it. They are also created using HTML<input> tag.

Example

Here is a basic example of a single-line password input used to take user password:

```
<!DOCTYPE html>
<html>
<head>
<title>Password Input Control</title>
</head>
<body>
<form>
User ID : <input type="text" name="user_id" />
<br>
Password: <input type="password" name="password" />
</form>
</body>
</html>
```

This will produce following result:

This will produce following result:

UserID:

Password:

Attributes

Following is the list of attributes for <input> tag for creating password field.

Attribute	Description
type	Indicates the type of input control and for password input control it will be set to password .
name	Used to give a name to the control which is sent to the server to be recognized and get the value.
value	This can be used to provide an initial value inside the control.
size	Allows to specify the width of the text-input control in terms of characters.
maxlength	Allows to specify the maximum number of characters a user can enter

into the text box.

Multi-line text input controls - This is used when the user is required to give details that may be longer than a single sentence. Multi-line input controls are created using HTML **<textarea>** tag.

Example

Here is a basic example of a multi-line text input used to take item description:

```
<!DOCTYPE html>
<html>
<head>
<title>Multiple-Line Input Control</title>
</head>
<body>
<form>
Description :<br/>
<textarea rows="5" cols="50" name="description">
Enter description here...
</textarea>
</form>
</body>
</html>
```

This will produce following result:

Description :

Attributes

Following is the list of attributes for **<textarea>** tag.

Attribute	Description
name	Used to give a name to the control which is sent to the server to be recognized and get the value.
rows	Indicates the number of rows of text area box.

cols	Indicates the number of columns of text area box
-------------	--

4.1.3 Checkbox Control

Checkboxes are used when more than one option is required to be selected. They are also created using HTML `<input>` tag but type attribute is set to checkbox.

Example

Here is an example HTML code for a form with two checkboxes:

```
<!DOCTYPE html>
<html>
<head>
<title>Checkbox Control</title>
</head>
<body>
<form>
<input type="checkbox" name="maths" value="on"> Maths
<input type="checkbox" name="physics" value="on"> Physics
</form>
</body>
</html>
```

This will produce following result:

Maths Physics

Attributes

Following is the list of attributes for `<checkbox>` tag.

Attribute	Description
type	Indicates the type of input control and for checkbox input control it will be set to checkbox .
name	Used to give a name to the control which is sent to the server to be recognized and get the value.
value	The value that will be used if the checkbox is selected.
checked	Set to <i>checked</i> if you want to select it by default.

4.2.4 Radio Button Control

Radio buttons are used when out of many options, just one option is required to be selected. They are also created using HTML `<input>` tag but type attribute is set to `radio`.

Example

Here is example HTML code for a form with two radio buttons:

```
<!DOCTYPE html>
<html>
<head>
<title>Radio Box Control</title>
</head>
<body>
<form>
<input type="radio" name="subject" value="maths"> Maths
<input type="radio" name="subject" value="physics"> Physics
</form>
</body>
</html>
```

This will produce following result:

Maths Physics

Attributes

Following is the list of attributes for radio button.

Attribute	Description
type	Indicates the type of input control and for checkbox input control it will be set to radio .
name	Used to give a name to the control which is sent to the server to be recognized and get the value.
value	The value that will be used if the radio box is selected.
checked	Set to <i>checked</i> if you want to select it by default.

4.1.5 Select Box Control

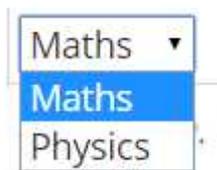
A select box, also called drop down box which provides option to list down various options in the form of drop down list, from where a user can select one or more options.

Example

Here is example HTML code for a form with one drop down box

```
<!DOCTYPE html>
<html>
<head>
<title>Select Box Control</title>
</head>
<body>
<form>
<selectname="dropdown">
<optionvalue="Maths"selected>Maths</option>
<optionvalue="Physics">Physics</option>
</select>
</form>
</body>
</html>
```

This will produce following result:



Attributes

Following is the list of important attributes of <select> tag:

Attribute	Description
name	Used to give a name to the control which is sent to the server to be recognized and get the value.
size	This can be used to present a scrolling list box.

multiple	If set to "multiple" then allows a user to select multiple items from the menu.
-----------------	---

Following is the list of important attributes of <option> tag:

Attribute	Description
value	The value that will be used if an option in the select box box is selected.
selected	Specifies that this option should be the initially selected value when the page loads.
label	An alternative way of labeling options

4.1.5 File Upload Box

If you want to allow a user to upload a file to your web site, you will need to use a file upload box, also known as a file select box. This is also created using the <input> element but type attribute is set to file.

Example

Here is example HTML code for a form with one file upload box:

```
<!DOCTYPE html>
<html>
<head>
<title>File Upload Box</title>
</head>
<body>
<form>
<input type="file" name="fileupload" accept="image/*" />
</form>
</body>
</html>
```

Attributes

Following is the list of important attributes of file upload box:

Attribute	Description
-----------	-------------

name	Used to give a name to the control which is sent to the server to be recognized and get the value.
accept	Specifies the types of files that the server accepts.

4.1.7 Button Controls

There are various ways in HTML to create clickable buttons. You can also create a clickable button using `<input>` tag by setting its type attribute to button. The type attribute can take the following values:

Type	Description
submit	This creates a button that automatically submits a form.
reset	This creates a button that automatically resets form controls to their initial values.
button	This creates a button that is used to trigger a client-side script when the user clicks that button.
image	This creates a clickable button but we can use an image as background of the button.

Example

Here is example HTML code for a form with three types of buttons:

```
<!DOCTYPE html>
<html>
<head>
<title>File Upload Box</title>
</head>
<body>
<form>
<input type="submit" name="submit" value="Submit" />
<input type="reset" name="reset" value="Reset" />
<input type="button" name="ok" value="OK" />
<input type="image" name="imagebutton" src="/html/images/logo.png" />
</form>
</body>
</html>
```

This will produce following result:

4.2. HTML Embed Multimedia

Sometimes you need to add music or video into your web page. The easiest way to add video or sound to your web site is to include the special HTML tag called `<embed>`. This tag causes the browser itself to include controls for the multimedia automatically provided browser supports `<embed>` tag and given media type.

You can also include a `<noembed>` tag for the browsers which don't recognize the `<embed>` tag. You could, for example, use `<embed>` to display a movie of your choice, and `<noembed>` to display a single JPG image if browser does not support `<embed>` tag.

Example

Here is a simple example to play an embedded midi file:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML embed Tag</title>
</head>
<body>
<embedsrc="/html/yourfile.mid"width="100%"height="60">
<noembed><imgsrc="yourimage.gif"alt="Alternative Media"></noembed>
</embed>
</body>
</html>
```

Try this code to see the production, you can put any media file in src attribute. You can try it yourself by giving various types of files.

The <embed> Tag Attributes

Following is the list of important attributes which can be used with <embed> tag.

Attribute	Description
align	Determines how to align the object. It can be set to either <i>center</i> , <i>left</i> or <i>right</i> .
autostart	This boolean attribute indicates if the media should start automatically. You can set it either true or false.
loop	Specifies if the sound should be played continuously (set loop to true), a certain number of times (a positive value) or not at all (false)
playcount	Specifies the number of times to play the sound. This is alternate option for <i>loop</i> if you are using IE.
hidden	Specifies if the multimedia object should be shown on the page. A false value means no and true values means yes.
width	Width of the object in pixels
height	Height of the object in pixels
name	A name used to reference the object.
src	URL of the object to be embedded.
volume	Controls volume of the sound. Can be from 0 (off) to 100 (full volume).

4.2.3: Supported Video Types

You can use various media types like Flash movies (.swf), AVI's (.avi), and MOV's (.mov) file types inside embed tag.

- .swf files - are the file types created by Macromedia's Flash program.
- .wmv files - are Microsoft's Window's Media Video file types.
- .mov files - are Apple's Quick Time Movie format.
- .mpeg files - are movie files created by the Moving Pictures Expert Group.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML embed Tag</title>
</head>
<body>
<embedsrc="/html/yourfile.swf"width="200"height="200">
<noembed><imgsrc="yourimage.gif"alt="Alternative Media"></noembed>
</embed>
</body>
</html>
```

2.4.4 HTML Object tag

HTML 4 introduces the <object> element, which offers an all-purpose solution to generic object inclusion. The <object> element allows HTML authors to specify everything required by an object for its presentation by a user agent, here are two examples:

Example

You can embed an HTML document in an HTML document itself as follows:

```
<objectdata="data/test.htm"type="text/html"width="300"height="200">
```

```
alt: <ahref="data/test.htm">test.htm</a>  
</object>
```

Here alt attribute will come into picture if browser does not support objecttag.

Example - 2

You can embed a PDF document in an HTML document as follows:

```
<objectdata="test.pdf" type="application/pdf" width="300" height="200">  
alt :<ahref="test.pdf">test.htm</a>  
</object>
```

2.5 HTML Layouts

A webpage layout is very important to give better look to your website. It takes considerable time to design a website's layout with great look and feel.

Now a days, all modern websites are using CSS and Javascript based framework to come up with responsive and dynamic websites but you can create a good layout using simple HTML tables or division tags in combination with other formatting tags. This chapter will give you few examples on how to create a

simple but working layout for your webpage using pure HTML and its attributes.

2.5.1 HTML Layout - Using Tables

The simplest and most popular way of creating layouts is using HTML `<table>` tag. These tables are arranged in columns and rows, so you can utilize these rows and columns in whatever way you like.

Example

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Layout using Tables</title>
</head>
<body>
<tablewidth="100%"border="0">
<tr>
<tdcolspan="2"bgcolor="#b5dcb3">
<h1>This is Web Page Main title</h1>
</td>
</tr>
<trvalign="top">
<tdbgcolor="#aaa"width="50">
<b>Main Menu</b><br/>
    HTML<br/>
    PHP<br/>
    PERL...
</td>
<tdbgcolor="#eee"width="100"height="200">
    Technical and Managerial Tutorials
</td>
```

```
</tr>
<tr>
<td colspan="2" bgcolor="#b5dcb3">
<center>
    Copyright © 2007 Tutorialspoint.com
</center>
</td>
</tr>
</table>
</body>
</html>
```

This will produce following result:

This is Web Page Main title

Main Menu

Technical and Managerial Tutorials

HTML

PHP

PERL...

Copyright © 2007 Tutorialspoint.com

2.5.2 Multiple Columns Layout - Using Tables

You can design your webpage to put your web content in multiple pages. You can keep your content in middle column and you can use left column to use menu and right column can be used to put advertisement or some other stuff. This layout will

be very similar to what we have at our website tutorialspoint.com.

Example

Here is an example to create three column layout:

```
<!DOCTYPE html>
<html>
<head>
<title>Three Column HTML Layout</title>
</head>
<body>
<tablewidth="100%"border="0">
<tr valign="top">
<td bgcolor="#aaa"width="20%">
<b>Main Menu</b><br/>
    HTML<br/>
    PHP<br/>
    PERL...
</td>
<td bgcolor="#b5dcb3"height="200"width="60%">
    Technical and Managerial Tutorials
</td>
<td bgcolor="#aaa"width="20%">
<b>Right Menu</b><br/>
    HTML<br/>
    PHP<br/>
    PERL...
</td>
</tr>
</table>
</body>
</html>
```

This will produce following result:

Main Menu	Technical and Managerial Tutorials	Right Menu
HTML		HTML
PHP		PHP
PERL...		PERL...

2.5.3 HTML Layouts - Using DIV, SPAN

The `<div>` element is a block level element used for grouping HTML elements. While the `<div>` tag is a block-level element, the HTML `` element is used for grouping elements at an inline level.

Although we can achieve pretty nice layouts with HTML tables, but tables weren't really designed as a layout tool. Tables are more suited to presenting tabular data.

Note: This example makes use of Cascading Style Sheet (CSS), so before understanding this example you need to have a better understanding on how CSS works.

Example

Here we will try to achieve same result using `<div>` tag along with CSS, whatever you have achieved using `<table>` tag in previous example.

```

<!DOCTYPE html>
<html>
<head>
<title>HTML Layouts using DIV, SPAN</title>
</head>
<body>
<div style="width:100%">
<div style="background-color:#b5dcb3; width:100%">
<h1>This is Web Page Main title</h1>
</div>
<div style="background-color:#aaa; height:200px;width:100px;float:left;">
<div><b>Main Menu</b></div>
    HTML<br/>
    PHP<br/>
    PERL...
</div>
<div style="background-color:#eee; height:200px;width:350px;float:left;">
<p>Technical and Managerial Tutorials</p>
</div>
<div style="background-color:#aaa; height:200px;width:100px;float:right;">
<div><b>Right Menu</b></div>
    HTML<br/>
    PHP<br/>
    PERL...
</div>
<div style="background-color:#b5dcb3;clear:both">
<center>
    Copyright © 2007 Tutorialspoint.com
</center>
</div>
</div>
</body>
</html>

```

This will produce following result:

This is Web Page Main title

Main Menu

HTML

PHP

PERL...

Technical and Managerial Tutorials

Right Menu

HTML

PHP

PERL...

Copyright © 2007 Tutorialspoint.com

You can create better layout using DIV, SPAN along with CSS.

For more information on CSS, please refer to CSS Tutorial.

Diyala University
College of education for pure Science
Department of computer Science
Lecturer of Web Design

Lecturer (1)

What is the website?

A website, also written as web site, or simply site, is a set of related web pages typically served from a single web domain. A website is hosted on at least one web server, accessible via a network such as the Internet or a private local area network through an Internet address known as a Uniform resource locator. All publicly accessible websites collectively constitute the World Wide Web.

What is the Web page?

A webpage is a document, typically written in plain text interspersed with formatting instructions of Hypertext Markup Language (HTML, XHTML). A webpage may incorporate elements from other websites with suitable markup anchors.

Webpages are accessed and transported with the Hypertext Transfer Protocol (HTTP), which may optionally employ encryption (HTTP Secure, HTTPS) to provide security and privacy for the user of the webpage content. The user's application, often a web browser, renders the page content according to its HTML markup instructions onto a display terminal.

The pages of a website can usually be accessed from a simple Uniform Resource Locator (URL) called the web address. The URLs of the pages organize them into a hierarchy, although hyperlinking between them conveys the reader's perceived site structure and guides the reader's navigation of the site which generally includes a home page with most of the links to the site's web content, and a supplementary about, contact and link page.

Some websites require a subscription to access some or all of their content. Examples of subscription websites include many business sites, parts of news websites, academic journal websites, gaming websites, file-sharing websites, message boards, web-based email, social networking websites, websites providing real-time stock market data, and websites providing various other services (e.g., websites offering storing and/or sharing of images, files and so forth).

Web site functions:-

Websites have many functions and can be used in various fashions; a website can be a personal website, a commercial website, a government website or a non-profit organization website. Websites can be the work of an individual, a business or other organization, and are typically dedicated to a particular topic or purpose. Any website can contain a hyperlink to any other website, so the distinction between individual sites, as perceived by the user, can be blurred.

What is the WWW?

The term WWW refers to the World Wide Web or simply the Web. The World Wide Web consists of all the public Web sites connected to the Internet worldwide, including the client devices (such as computers and cell phones) that access Web content. The WWW is just one of many applications of the Internet and computer networks.

Web design process:-

When building a website there is a process that most designers use. This process covers all the steps from deciding on a website to building it and putting them live.

While all of the steps are important, the amount of time you spend on them is up to you. Some designers prefer to plan a lot before building while others spend little or no time on marketing. But if you know what the steps are you can decide which ones you don't need.

Definition - What does Web-Based Application mean?

A Web-based application refers to any program that is accessed over a network connection using HTTP, rather than existing within a device's memory. Web-based applications often run inside a Web browser. However, Web-based applications also may be client-based, where a small part of the program is downloaded to a user's desktop, but processing is done over the Internet on an external server. Web-based applications are also known as Web apps.

Lecturer (2)

Types of Web-Based Application

However, from the aspect of you the user, you really only care about what the web application does, and how it will help you. So, it is easier to talk about types of web applications based on use:

- social networking websites (Facebook, MySpace, hi5.com, Xanga, Classmates.com).
- Image websites (Snapfish, Flickr, Photobucket, Shutterfly)
- Online calendars (Google Calendar, O2 Calendar, Yahoo! Calendar)
- Chat sites (chat-avenue, omgpop, czateria, Meebo)
- Online shopping websites (just about ANY store -- the grandfather of online shopping is Amazon)
- Online telephone directories (Yellow.com, White Pages, AnyWho)
- Blog pages (Blogger, Word Press, Tumbir)
- Online news publishers (Press Smart, The Drudge Report, Open Publish)
- Online dictionaries (ExamplesThe American Heritage Dictionary, Dictionary.com)
- Webmail services (Gmail, Yahoo Mail, HotMail)
- Map servers (Google Maps, MapQuest)
- Internet search engines (Google, Bing, Yahoo)
- Alternatives to desktop software applications (Google Docs, Slide Rocket, Avairy, Mint)

What is the HTTP?

is short for *Hyper Text Transfer Protocol*. HTTP is the underlying protocol used by the World Wide Web and this protocol defines how messages are formatted and transmitted, and what actions Web servers and browsers should take in response to various commands.

For example, when you enter a URL in your browser, this actually sends an HTTP command to the Web server directing it to fetch and transmit the requested Web page. The

other main standard that controls how the World Wide Web works is HTML, which covers how Web pages are formatted and displayed.

What is the URL?

URL is the abbreviation of Uniform Resource Locator. URL is the global address of documents and other resources on the World Wide Web.

Parts of a URL

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

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

What is the website?

A Web site is a related collection of World Wide Web (WWW) files that includes a beginning file called a home page. A company or an individual tells you how to get to their Web site by giving you the address of their home page. From the home page, you can get to all the other pages on their site. For example, the Web site for IBM has the home page address of <http://www.ibm.com>.

What is a Client?

A client is a piece of computer hardware or software that accesses a service made available by a server. The server is often (but not always) on another computer system, in which case the client accesses the service by way of a network.

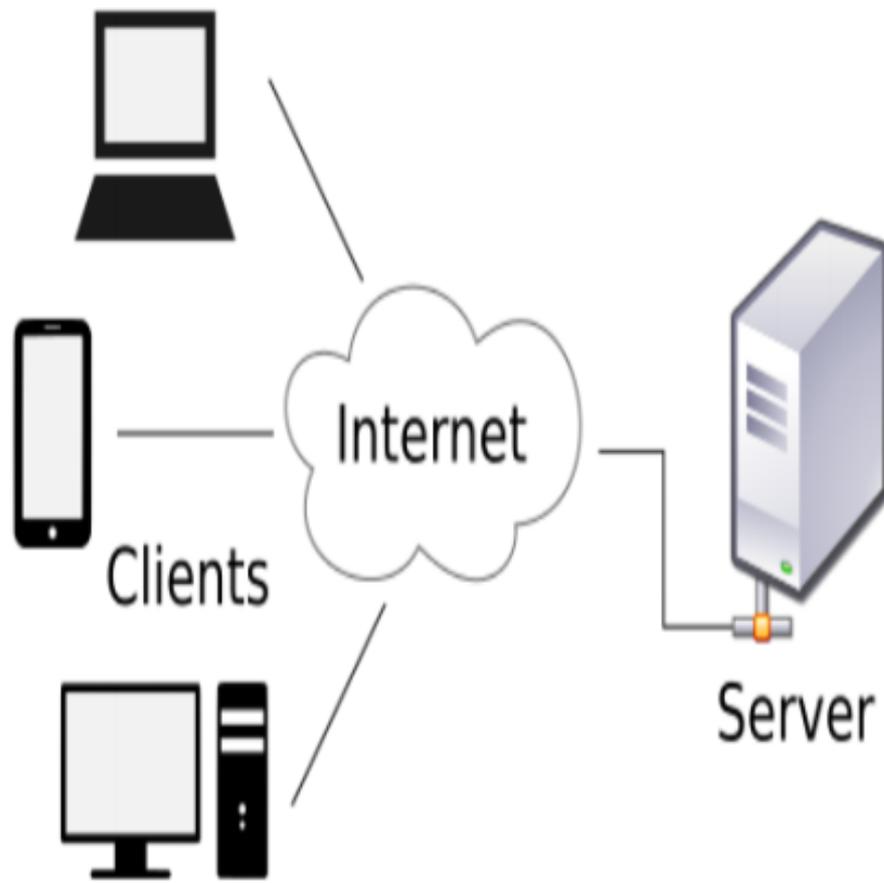


Figure Client

Lecturer (3)

What are the benefits of website?

- **#1 Online Presence 24/7**

Having a website means customers are always able to find you – everyday, 24/7. Meaning even at weekends and evenings your business is still accessible. It offers the customer convenience as they are able to access the information they need in the comfort of their own home, with no added pressure to buy. In general, it is much easier to click a few buttons online than to make the trip to a physical store to hunt someone down for information. It also prevents little annoyances such as waiting in queues.

- **#2 Credibility**

In this day and age it is much more credible for a business to have an online presence. People would likely be distrusting to any business that didn't have a telephone number or a physical address, and that soon could be said for having a website and e-mail. They are useful tools to help share further information about your business with customers answering all the 'What's?' and 'Whys?' that people may have. And too, if your website is of a high standard then it is likely to receive positive feedback with customers spreading positive word-of-mouth about your business.

- **#3 It Cuts Costs**

Your business will save money in a number of ways from spending less on printing costs (letters, catalogues etc.), to wages, as a website may mean you no longer need that extra pair of hands, due to automated processes. A website is a much more long term form of advertising and more than likely will produce better results. You can also use your website as ways of collecting cheap market research by setting up visitor polls and online surveys, helping you to find out what customers like best and how they feel.

- **#4 Market Expansion**

As the internet is accessible to anyone all over the world the ability to break through geographical barriers has become a lot easier. Anyone from any country will be able to access you and therefore be a potential customer.

- **#5 Potential Customers Already Out There**

As the internet itself is widely popular, millions of users are already online, meaning that out there already are millions of potential customers. Without a website, you are losing out on this opportunity. Some people often will just use the internet to research items but seeing as they have an exact product in mind they are more likely to go through with a purchase. Online shopping is certainly on the rise. Selling products and services online gives your business a major competitive advantage and is a very cost efficient way of doing business.

- **#6 Information Exchange**

A website provides easy ways of communication between buyers and sellers and allows customers to give quick and easy feedback on any aspect of the business.

- **#7 Ability to Change Information Whenever**

Having a website means a business can change information instantly, from pricing changes to location changes. By having most of your information on printed out documents there is a high chance they'll become out of date, fast. It is much more cost efficient to change information on a website than to reprint updated information which will likely be old news before they even arrive.

- **#8 Customer Service Online**

Websites provide an easier way to handle customer service, providing answers to regularly asked questions in a FAQ (Frequently Asked Questions) section you can reduce customer service costs and save yourself time and money, as well as providing much more information.

- **#9 Competitors Online**

It is more than likely that your competitors will already be set up online and will be attracting the exact same customers as you. So to get ahead it is definitely the way forward, and may even help you as a business become more competitive.

- **#10 Growth Opportunity**

Websites in general are great ways to in providing a place that potential investors can be referred too. It shows what your company is about, what it has achieved and what it can achieve in future.

Web – Basic Concepts

- **What is the Internet?**

The Internet is a global network connecting millions of computers. More than 190 countries are linked into exchanges of data, news and opinions.

Is Web and Internet the Same?

The Internet is not synonymous with World Wide Web. The Internet is a massive network of networks, a networking infrastructure. It connects millions of computers together globally, forming a network in which any computer can communicate with any other computer as long as they are both connected to the Internet. The World Wide Web, or simply Web, is a way of accessing information over the medium of the Internet. It is an information-sharing model that is built on top of the Internet.

Internet based services

1- World Wide Web (WWW)

The World Wide Web is commonly known as *web*. It is a network of web servers that stores web pages. The web pages are connected to each other using hyperlinks. The user can jump from one page to another by clicking the hyperlinks. The web web pages are accessed using web browsers. The HTTP (Hyper Text Transfer Protocol) protocol is used for communication between browser and web servers.

2- E-Mail

E-mail stands for electronic email. It is the most popular service provided by the Internet. It provides the fast and efficient way to send and receive messages through Internet. One

message can be sent to many persons with a single e-mail. Different types of files can also be sent through e-mail.

3- Telnet

Telnet is an abbreviation for Terminal Network. It is a software. It is used to connect to a remote or host computer for accessing information. Through this service, the user can also access information on the Internet.

4- File Transfer Protocol (FTP)

FTP stands for File Transfer Protocol. It is a way for transferring files from one computer to another. The process of transferring a file from a server (or remote computer) to local computer is called downloading. Similarly, the process for transferring a file from local computer to the server on the Internet is called *uploading*.

5- Gopher

A gopher is a menu-based information retrieval system. It is used for retrieving files and programs on the Internet. Gopher allows access to files found on FTP servers and web servers.

6- Chat Groups

Internet provides the facility to Internet users to chat (talk) with people online all over the world. Different programs like MSN Messenger, Yahoo Messenger, AOL online etc. are available for chatting on the Internet. Chat rooms allow the users to participate in a chat on the Internet. Chat rooms are locations on the web that provide facilities to the users to chat with each other online over the Internet.

Lecturer (4)

What is the Web Server?

A Web server is a program that uses HTTP (Hypertext Transfer Protocol) to serve the files that form Web pages to users, in response to their requests, which are forwarded by their computers' HTTP clients. Dedicated computers and appliances may be referred to as Web servers as well.

On the other words, Web servers are computers that deliver (serves up) Web pages. Every Web server has an IP address and possibly a domain name. For example, if you enter the URL.

What is the web browser?

Short for Web browser, a browser is a software application used to locate, retrieve and display content on the World Wide Web, including Web pages, images, video and other files.

On the other words, Web Browsers are software installed on your PC. To access the Web, you need a web browser, such as Netscape Navigator, Microsoft Internet Explorer or Mozilla Firefox.



Figure shows types of web browser

What is the hyperlink?

A hyperlink is an element in an electronic document that links to another place in the same document or to an entirely different document. Typically, you click on the hyperlink to follow the link. Hyperlinks are the most essential ingredient of all hypertext systems, including the World Wide Web.

What is the W3C?

Short for *World Wide Web Consortium*, an international consortium of companies involved with the Internet and the Web. The W3C was founded in 1994 by Tim Berners-Lee, the original architect of the World Wide Web. The organization's purpose is to develop open standards so that the Web evolves in a single direction rather than being splintered among competing factions.

What is the DNS?

DNS is a short for *Domain Name System (or Service or Server)*, an Internet service that translates *domain names* into IP addresses. Because domain names are alphabetic, they're easier to remember. The Internet however, is really based on IP addresses. Every time you use a domain name, therefore, a DNS service must translate the name into the corresponding IP address. For example, the domain name *www.example.com* might translate to *198.105.232.4*.

What is the HTML?

HTML (Hypertext Markup Language) is the set of markup symbols or codes inserted in a file intended for display on a World Wide Web browser page. The markup tells the Web browser how to display a Web page's words and images for the user. Each individual markup code is referred to as an element (but many people also refer to it as a tag). Some elements come in pairs that indicate when some display effect is to begin and when it is to end.

The purpose of building the website

Whether your website is for business or personal interest, there are a number of things you have to consider before you start actually building your site. The first step should be deciding on the purpose of the website.

Purpose of Website - Why are you building this website?

Ask yourself, "**Why am I building this website?**"

- Are you building this site to make money from your website?
- To share information?
- or for your friends and family to keep in touch?

The purpose of the website will influence the style, the website technologies required, hosting costs and budget required for your website.

Lecturer (5)

Types of websites

1. Make Money with Website

A site built to make money from your website will require things like a way for the visitor to pay for the items, a shopping cart if you have a lot of things for them to purchase and security features to protect the purchaser's information. To give a good impression of your business your site needs to give a professional appearance.

- **Electronic commerce**, commonly written as **e-commerce** or **eCommerce**, is the trading or facilitation of trading in products or services using computer networks, such as the Internet or online social networks.^[1] Electronic commerce draws on technologies such as mobile commerce, electronic funds transfer, supply chain management, Internet marketing, online transaction processing, electronic data interchange (EDI).

The screenshot displays the Faiza Fashion website interface. At the top right, there are links for "My Cart (0) | Log In | Register" and a "SEARCH PRODUCT" input field. The main banner features a man on the left and a woman on the right, with the text "faiza FASHION" in the center. Below the banner, the website is divided into three columns: "Categories" on the left, "Featured Product" in the middle, and "New Products" on the right. The "Categories" column lists: Home, Shoes Men and Women, Handbags and Accessories, Accessories, and Test. The "Featured Product" column shows two items: a "shoulderbag" with a retail price of \$760.00 and a Louis Vuitton price of \$688.00, and "newbags" with a retail price of \$760.00 and a Louis Vuitton price of \$688.00. The "New Products" column is currently empty. At the bottom, there is a footer with copyright information: "Copyright © 2008 faizafashion.com. All Rights Reserved.", navigation links: "Contact us | About us | Privacy Policy", and "Web Designer:- Pavan.org".

2. Websites Built to Share Information

A website to share information may not require as many features as a site that sells something. One thing you have to consider is if you are going to allow visitors to add comments or additional resources to your site. This would require additional features from your website hosting.

The screenshot shows the homepage of the 'ROOM for moms' website. The header features the site's logo, 'ROOM for moms', with the tagline 'a place to connect with moms like you'. Navigation links include 'Home', 'FAMILY HEALTH', 'PARENTING & FAMILY LIFE', 'IN YOUR NEIGHBORHOOD', and 'CONNECT WITH MOMS'. A search bar is located in the top right corner. The main content area is titled 'To get started, just click on the mom that's most like you.' and includes a registration prompt. Below this, four profiles are displayed: Kiyana (Starting a family), Sarah (Having my first baby), Jackie (Having another one), and Christina (Not pregnant, with kids). Each profile has a corresponding image of the woman. A 'Vote' button is provided for each profile. The footer contains a 'Printer-friendly version' link and a navigation menu with links to 'home', 'about', 'contact us', 'privacy', 'terms', 'site map', 'search', and 'alegent.com'. The website is presented by Alegant Health.

ROOM for moms
a place to connect with moms like you

username password [log in](#)
[register](#)

[Home](#) [FAMILY HEALTH](#) [PARENTING & FAMILY LIFE](#) [IN YOUR NEIGHBORHOOD](#) [CONNECT WITH MOMS](#) search [go](#)

WISDOM CORNER
How is it possible that I'm 55 and still feel guilty for things that I did when I was 15, or 6 for that matter?
[more wisdom >](#)

Prefereed Trait for your Child?
A genie has given you one wish for the child you are expecting: an abundance of one trait. Which one would you choose?

good looks
 intelligence
 kindness
 positive outlook
 determination

[Vote](#)

To get started, just click on the mom that's most like you.
Then, just follow the prompts to set up your online experience. Or, you can [register here](#) if none of these fit you well. Once you are registered and you have entered your profile information, you will enter an online community like you've never seen before -- made just for moms just like you!

Kiyana
Starting a family

Sarah
Having my first baby

Jackie
Having another one

Christina
Not pregnant, with kids

If none of these fit, just skip to [registration](#) to get started.

[Printer-friendly version](#)

presented by **Alegant Health**
This is your healthcare

[home](#) • [about](#) • [contact us](#) • [privacy](#) • [terms](#) • [site map](#) • [search](#) • [alegent.com](#)

Dr. Traver on Stress During Pregnancy
[watch video](#)

Recipes and grocery lists
[go >>](#)

Baby Expo at the Quest
[go >>](#)

Week 21
Visit the [forum >>](#)

Breastfeeding vs. bottle-fed
Check out the [blog >>](#)

Websites Built for Family and Friends

Family and friends websites would have a less formal feel. For this type of site you would also have to consider if your family and friends will be permitted to automatically add to the content of your website. Again, this would require additional features from your website hosting.

