WsCube Tech™
System For Satisfaction

HTML CUBE

Building Static Websites with HTML

BY **WsCube Tech**

What is 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 (URL). All publicly accessible websites collectively constitute the World Wide Web.

Or

A **website** is a collection of Web pages, images, videos or other digital assets that is hosted on one or several Web server(s), usually accessible via the Internet, cell phone or a LAN.

The definition of **web page** is a document, typically written in HTML, which is almost always accessible via HTTP, a protocol that transfers information from the Web server to display in the user's Web browser.

What is Website Development :

Website development broadly includes **developing a website** for either World Wide Web (www; internet) or a private network (intranet).

Web development is inclusive of various tasks that are commonly referred as web design, web content development, client association, scripting from client-side/server-side, web server and network security configuration, and e-commerce development. The development ranges from simple static page containing only text to the most complex web-based applications, electronic businesses, and social network services.

It is referred to the main non-design features of building web sites that includes writing markup and coding language by web professionals.

For **website development** at big organizations and businesses a team is required that involves hundreds of people as content writers, graphic designers, web designer and web developers.

The web development can be diverted into two main groups; Client Side Coding and the Server Side Coding. The role of a web developer generally includes **Graphic and Web designing**, information architecture and copy editing with the usability of the web and **optimization of the search engine**. Web developer focus on the technologies sent to the client such as HTML, JavaScript, CSS, and on the server-side frameworks (such as PHP) used to deliver content and scripts to the client. Web developer focus on the interaction between server-side frameworks, the web server, and a database system.

Web development is the back-end of the website that works upon the programming and communications on the pages. A **web developer** focuses on how a site works and how the customers get things done on it. Good web developers know how to program CGI and scripts like PHP. They understand about how web forms work and can keep a site running effectively.

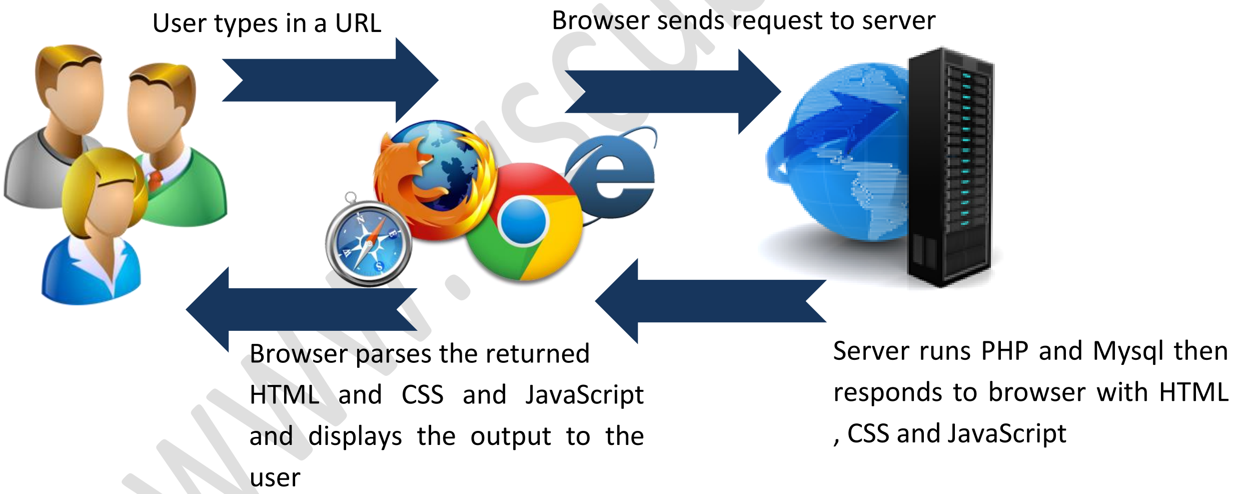
Web development means something different for Micro Businesses than it does for Corporations. A web designer has an absolute knowledge of JavaScript, PHP, HTML and is well aware of the importance of design to a website.

Language	Role	Where it Runs

The web development assesses your business needs, creates plan to meet them and then assembles all the facts that make up the solution to convert your Micro Business into Macro Business.

- Web development means that it looks good as well as works well from every outlook.
- **Website development** takes into consideration the cost control and takes into consideration that every feature is weighed against the benefit for the business.
- It means that every feature of the site is developed such as to give maximum benefit for the available funds.
- It means that the developer has a good understanding of quality issues across graphics, content, code, SEO, and marketing messages.

Identify the different roles that PHP, MySQL, HTML and CSS play in modern Website Development



User types in a URL

Browser sends request to server

Browser parses the returned HTML and CSS and JavaScript and displays the output to the user

Server runs PHP and Mysql then responds to browser with HTML , CSS and JavaScript

HTML	**Content and Structure** (text, images, tables, forms, etc)	Browser
CSS	**Style and Presentation** (color, fonts, background, margins & paddings, etc)	Browser
JavaScript	**Client Side Scripting** (dynamic user interaction, popup, form validation)	Browser
PHP	**Server Side Scripting** (server side logic and data processing)	Server
MySQL	**Data Management** (stores all sorts of data needed by a website)	Server

The Difference Between **STATIC & DYNAMIC Websites**

STATIC WEBSITE

What is a STATIC WEBSITE? ....

Any Site that has fixed / Stagnant content usually written in html code.

1. Every page will have the Code Written separately, just as you see it on the web.

2. Every page has to be saved separately on the server.

3. Changes have to be made manually every time, and you need coding knowledge to make any and all changes.

Advantages & Disadvantages of STATIC Website

Advantages	Disadvantages

▸ Easy to develop	▸ Requires web development expertise to update site
▸ Cheap to develop	▸ Changes and updates are very time consuming
▸ Cheap to host	▸ Site not as useful for the user
	▸ Content can get stagnant
	▸ OUT OF DATE!! YOU ARE A DINOSAUR

DYNAMIC WEBSITE

What is a DYNAMIC Website ?......

A site whose construction is controlled by an application server processed by server side scripts.

1. Pages of the website are not coded and saved separately.
2. The design / template (look and feel) is saved separately.
3. Corresponding content are saved separately.
4. The pages are dynamically populated every time.

Advantages	Disadvantage
▸ **Much more functional website**	▸ **More expensive to develop**
▸ **Much easier to update**	▸ **Slower to develop**
▸ **Much Easier to add new content / pages**	▸ **Hosting costs a little more (This might not be true now ,because of great shared web hosting plans these days**
▸ **New content brings people back to the site and helps in the search engines**	
▸ **Con work as a system to allow staff or users to collaborate**	

What is HTML ?

HTML is a computer language devised to allow website creation. These websites can then be viewed by anyone else connected to the Internet. It is relatively **easy to learn**; and quite powerful in what it allows you to create. It is constantly undergoing revision and evolution to meet the demands and requirements of the growing Internet audience under the direction of the » W3C, the organization charged with designing and maintaining the language.

The definition of HTML is Hypertext Markup Language.

HyperText is the method by which you move around on the web — by clicking on special text called hyperlinks which bring you to the next page. The fact that it is hyper just means it is not linear — i.e. you can go to any place on the Internet whenever you want by clicking on links — there is no set order to do things in.

Markup is what HTML tags do to the text inside them. They mark it as a certain type of text (italicised text, for example).

HTML is a Language, as it has code-words and syntax like any other language.

HTML is a language for describing web pages.

- HTML stands for **H**yper **T**ext **M**arkup **L**anguage
- HTML is not a programming language, it is a **markup language**
- A markup language is a set of **markup tags**
- HTML uses **markup tags** to describe web pages

So The HTML Teaches you to create your own Website easily. So lets move further.

What Is Markup Language

A markup language combines text and extra information about the text.

The extra information, for example about the text's structure or presentation, is expressed using markup, which is intermingled with the primary text. The best-known markup language in modern use is HTML (HyperText Markup Language), one of the foundations of the World Wide Web. Historically, markup was (& is) used in the publishing industry in the communication of printed work between authors, editors, and printers.

What Is An HTML Element

An HTML element indicates structure in an HTML document and a way of arranging content hierarchically. An HTML element is an SGML element that meets the requirements of one or more of the HTML DTDs (Document Type Definitions). These elements have properties: both attributes and content, as specified (both allowable and required) according to the appropriate HTML DTD . Elements may represent paragraphs, headings, hypertext links, lists, embedded media, and a variety of other structures.

Syntactically HTML elements are constructed with:

1) a start tag marking the beginning of an element;

2) any number of attributes (and their associated values);
3) some amount of content (characters and other elements)'; and
4) an end tag.

Many HTML elements include attributes in their start tags. Attributes are defining desired behavior or indicating additional element properties. The end tag is optional for many elements. There are a few elements that are not part of any official DTDs, yet are supported by some browsers and used by some web pages. Such elements may be ignored or displayed improperly on browsers not supporting them.

What are HTML Tags ?

The tags are what separate normal text from HTML code. You might know them as the words between the <angle-brackets>. They allow all the cool stuff like images and tables and stuff, just by telling your browser what to render on the page. Different tags will perform different functions. The tags themselves don’t appear when you view your page through a browser, but their effects do. The simplest tags do nothing more than apply formatting to some text, like this:

<b>These words will be bold</b>, and these will not.

In the example above, the <b> tags were wrapped around some text, and their effect will be that the contained text will be bolded when viewed through an ordinary web browser.

HTML markup tags are usually called HTML tags

- HTML tags are keywords surrounded by **angle brackets** like <html>
- HTML tags normally **come in pairs** like <b> and </b>
- The first tag in a pair is the **start tag,** the second tag is the **end tag**
- Start and end tags are also called **opening tags** and **closing tags**

Why do We Use Lowercase Tags?

Start using lowercase tags, if you want to prepare yourself for the next generations of HTML. The World Wide Web Consortium (W3C) recommends lowercase tags in their HTML 4 recommendation, and XHTML - the next generation HTML - demands lowercase tags.

Introduction To Tags

The tag are represented by angle bracket < TagName >

HTML have set of tags. There are mainly two types of tag as given below:

1. Paired Tag

2. Unpaired Tag

Paired Tag

There are many paired tag which have two end tag: starting tag and closing tag

The starting tag are represented by angle bracket < TagName >

The closing tag are represented by closing bracket < / TagName >

Example:

<div> content.. </div>

<h1> heading 1 </h1>

Unpaired Tag

A tag without closing tag is called Unpaired tag. It is also called as stand-alone tag or singular Tag.

Example :
e.g. <br />, <hr />

HTML Basic Tags

The basic tags include <HTML>, <title>, <meta>, and <body>.

<HTML>

This tag is used to indicate that this is an HTML document.

Most HTML documents should start and end with this tag.

<head>

This tag is used to indicate the header section of the HTML document, which typically includes the <title> and <meta> tags, and is not displayed in the main window of the browser.

<title>
This tag indicates the title of this HTML page. The title is what is displayed on the upper left corner of the browser when you view a web page. For example, right now you can see there "Basic Tags: HTML, head, title, meta, body". That is the title of this page.

The title tag is important when it comes to search engine ranking. Many of the search engines pay special attention to the text in the <title> tag. This is because (logically) that words in the <title> tag indicate what the page content is.

<meta>
The <meta> tag information is not directly displayed when the page is rendered on the browser. Rather, this is used for the author of the HTML page to record information related to this page. Two common attributes are name and content. The <meta> tag used to hold great importance in search engine optimization, with authors carefully drafting what's inside the tag to gain better search engine ranking, but recently its importance has been decreasing steadily.
<body>

The <body> tag includes the HTML body of the document. Everything inside the <body> tag (other than those within the <script> tag) is displayed on the browser inside the main browser window.
The <body> tag may contain several attributes. The most commonly used ones are listed below:

- **bgcolor:** This is the background color of the entire HTML document, and may be specified either by the color name directly or by the six-digit hex code.
- **alink:** The color of the links.
- **vlink:** The color of the visited links.
- **topmargin:** The margin from the top of the browser window.
- **leftmargin:** The margin from the left of the browser window.

Tags Based on their Utility

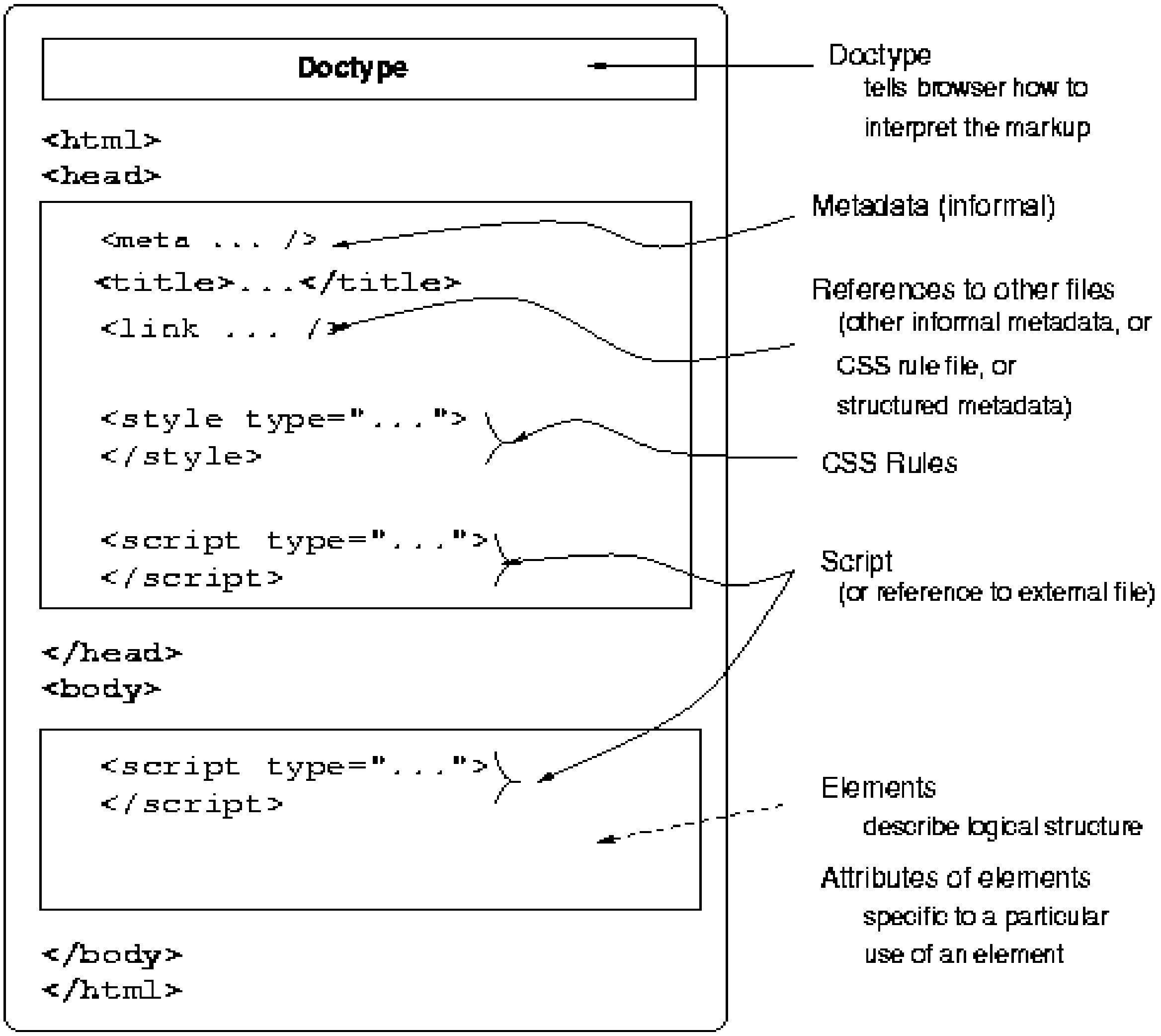
We can differentiate tags based on the purpose they used. Basically we have three types here

Formatting Tags
We manage the size of the font, underline part of the text, make the text bold etc by using tags like <font>, <u>,<b> etc.

Page Structure Tags
Description , title , head , body etc are part of the page structure tags. They are part of the basic html page and does not directly affect the formatting of text or image.

Control Tags
Form tags, Script tags, Radio buttons etc are part of the control tags

HTML Structure



Getting Started with HTML

What you need?

HTML is written simply in a plain text editor like Notepad. Advanced developers also use software's like Frontpage, Expression Web & Dreamweaver. But we will be using advanced text editor Notepad++ (provided to you on wscubetech.com Website).

How to Save HTML Web Page. HTM or HTML Extension ?

When you save an HTML file, you can use either the .htm or the .html extension. We use .htm in our examples. It is a habit from the past, when the software only allowed three letters in file extensions.

With new software it is perfectly safe to use .html.

Understand the HTML Web Page Structure.

```
<html>

<body>

        Hello, How are you. Welcome to WsCube Tech
</body>

</html>
```

Example Explained

- The text between <html> and </html> describes the web page
- The text between <body> and </body> is the visible page content

This is the common structure of all the HTML Web Pages. Always you have to Start the HTML Page with opening HTML and Body Tags <html> and <body> and after putting all the content you need in the Web Page, You need to close the opened tags by </body> and </html>, in the order they were opened.

HTML Elements.

An HTML element is everything from the start tag to the end tag.

Illustration

```
<p>This is Some Content. Welcome to WsCube Tech</p>
```

Illustration Explained

Here everything for start paragraph tag to end tag is the HTML Element. The content between the tags “This is Some Content.” is known as the element content.

Comments in HTML

The comment tag is used to insert a comment in the HTML source code. A comment will be ignored by the browser. You can use comments to explain your code, which can help you when you edit the source code at a later date.

<!-- This is a comment -->

Note that you need an exclamation point after the opening bracket, but not before the closing bracket.

HTML Element Syntax

- An HTML element starts with a **start tag / opening tag.**
- An HTML element ends with an **end tag / closing tag.**
- The **element content** is everything between the start and the end tag.
- Some HTML elements have **empty content.**
- Empty elements are **closed in the start tag.**
- Most HTML elements can have **attributes.**

Nested HTML Elements

The HTML Elements can be nested, that is one element can be inside the other element.

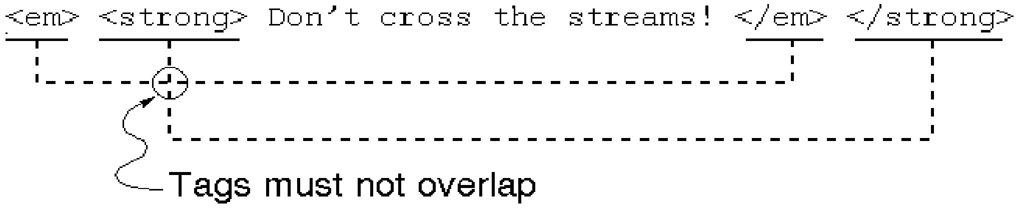
Illustration

```
<html>

<body>

<p>Welcome to WsCube Tech</p>

</body>

</html>
```

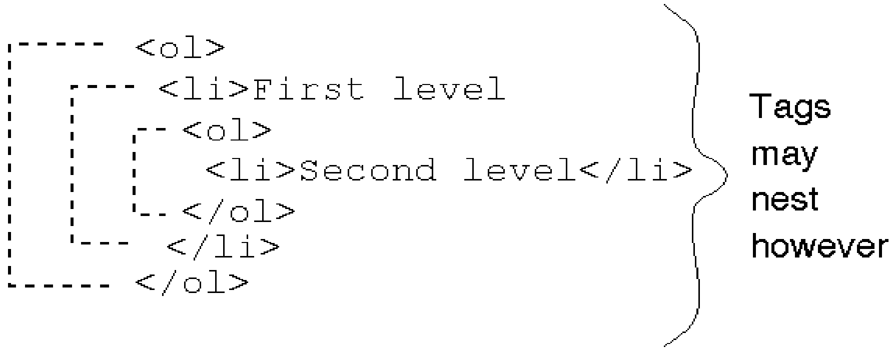
Here there are three elements <html>, <body> and <p>.

<p> tag is nested inside <body> which in turn is nested inside <html>.

Always Remember to put the End Tags after opening the tags.



```
<em> <strong> Don't cross the streams! </em> </strong>
```



```
<ol>
  <li>First level
    <ol>
      <li>Second level</li>
    </ol>
  </li>
</ol>
```

HTML Basic Tags Explained.

Heading Tags

There are six levels of **headings** in **HTML** specified by <**H1**>, <H2>, <H3>, <H4>, <H5> and <H6> **tags**. <h1> defines the largest heading and <h6> defines the smallest heading.

Illustration

```
<html>
<body>

<h1>My First Heading.</h1>

 <h2>My Second Heading.</h2>

</body>
</html>
```

Illustration Output:

My First Heading.

My Second Heading.

Heading Tag Illustration Explained

HTML headings are defined with the <h1> to <h6> tags.

<h1> is the biggest font size and <h6> is the smallest.

Paragraph Tags

HTML paragraph tags are used to define the **HTML paragraph** element. The **paragraph** element begins with the **<p> tag** and ends with the **</p> tag**

Illustration

```
<html>

<body>

<p>My First Paragraph.</p>

</body>
</html>
```

Illustration Output

My First Paragraph.

Paragraph Tag Illustration Explained

HTML Paragraph element are defined with the <p> tags. Always remember to close the paragraph tag when a particular paragraph is written completely.

These are the tags that really made web pages unique, by letting a person click on text or an image and go to other webpage. Hyperlink tags are container tags, which means there must be an initial tag and an ending tag. All the WebPages in a website are connected to one another using the Hyperlinks.

Illustration

```
<html>

<body>

<a href="http://www.wscubetech.com">Wscube Tech Home Page.</a>

</body>
</html>
```

Illustration Output

WsCube Tech Home Page.

Hyperlink Tag Illustration Explained

WsCube Tech Home Page.

Clicking on the above sentence on the web page takes you to the www.wscubetech.com Website Home Page.

Image Tags

This Tag is used for putting pictures or images on your web page. The images can be of any format that is JPG, GIF, PNG etc. The <img> tag is empty, which means that it contains attributes only and it has no closing tag.

Illustration

```
<html>

<body>

<img src="wscubetech.jpg" width="250" height="100" />

</body>
</html>
```

Illustration Output:

Image Tag Illustration Explained

The Image Tags contains number of attributes like src that is source of the file, width, height of the image, al attribute etc. We will discuss more about attributes in subsequent chapters.

Some other Useful Tags.

We will now discuss some other useful tags like HTML Rules, HTML Comments, Line Breaks etc.

Drawing HTML Rules

<HR> (Horizontal Rule) is a standalone tag that generates a horizontal **line**. There is no corresponding **HTML** command for a verticle **line**.

Illustration

```
<html>

<body>

<h1>My First Heading.</h1>

<hr>

<h2>My Second Heading.</h2>

</body>
</html>
```

Illustration Output.

My First Heading.

My Second Heading.

Rules Tag Illustration Explained

The <hr> Tag generates a horizontal line between the 2 Headings.

HTML Comments

All combinations of text placed within the **comment tags** will be ignored by the web browser, this includes any **HTML** tags, scripting language(s), etc.

Illustration

```
<html>

<body>

 <!-- <h1>My First Heading.</h1> -->

<h2>My Second Heading.</h2>

</body>
</html>
```

Illustration Output

My Second Heading.

Heading Tag Illustration Explained

First Heading would not be displayed as it is commented. So the browsers just ignore everything inside the comment tag.

Line Breaks

To break the line and move to the next line. Use the <br> tag.

Illustration

```
<html>

<body>

<p>I am writing a Very Long Paragraph. <br> You Would need like it.</p>

</body>
</html>
```

Illustration Output

I am writing a Very Long Paragraph.

You Would need like it.

Heading Tag Illustration Explained

We Inserted the line break tag <br> between the 2 lines so the second line breaks and gets after the first line.

HTML Text Formatting Tags

Now we will discuss how do we format to Bold, Italicize, Underline, Subscript or Superscript any text in the HTML Page.

Bold

<b> (Bold) is a tag that makes any text between its text to be formatted as Bold. You always had to remember closing the bold tag after you have formatted the required text.

Illustration

```
<html>

<body>

<b>This will appear Bold. </b> While This will not.

</body>
</html>
```

Illustration Output

This will appear Bold. While This will not.

Bold Tag Illustration Explained

The <b> Tag bolds the text which appears in between its tags.

Italic

<i> (Italic) is a tag that makes any text between its text to be formatted as Italic. You always had to remember closing the bold tag after you have formatted the required text.

Illustration

```
<html>

<body>

<b>This will appear Bold. </b>

<i>While This will be italic. </i> <br>

<b><i>And This would be Bold and Italic.</i></b>
```

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

Illustration Output

This will appear Bold. *While This will be italic.*

And This would be Bold and Italic.

Italic Tag Illustration Explained

The <i> Tag Italicize the text which appears in between its tags. We also used the <br> tags in between so a line break had also took place.

Underline

<u> (Underline) is a tag that makes any text between <u> tags to be formatted as Underlined. You always had to remember closing the bold tag after you have formatted the required text.

Illustration

```
<html>

<body>

<u>This will appear Underline. </u>

</body>
</html>
```

Illustration Output

This will appear Underline.

Underline Tag Illustration Explained

The <u> Tag Underline the text which appears in between its tags.

Subscript

<sub> (Subscript) is a tag that makes any text between <sub> tags to be formatted as subscripted. You always had to remember closing the subscript tag after you have formatted the required text.

Illustration

```
<html>
<body>
The Element is CO<sub>2</sub>.
```

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

Illustration Output

The Element is CO2.

Subscript Tag Illustration Explained

The 2 between <sub> Tags is subscripted.

Superscript

<sup> (Superscript) is a tag that makes any text between <sup> tags to be formatted as subscripted. You always had to remember closing the superscript tag after you have formatted the required text.

Illustration

```
<html>

<body>

The cube of 2 that is 2<sup>3</sup> is 8. This will appear Underline. </sup>

</body>
</html>
```

Illustration Output

The cube of 2 that is 2^3 is 8.

Superscript Tag Illustration Explained

The <sup> Tag Superscripts the text which appears in between its tags.

Marquee

<sup> (Superscript) is a tag that makes any text between <sup> tags to be formatted as subscripted. You always had to remember closing the superscript tag after you have formatted the required text.

Illustration

```
<html>
<body>
The cube of 2 that is 2<sup>3</sup> is 8. This will appear Underline. </sup>
```

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

Illustration Output

The cube of 2 that is 2^3 is 8.

Superscript Tag Illustration Explained

The <sup> Tag Superscripts the text which appears in between its tags.

HTML Styling Tags

You may want to change the way the generated HTML output looks. The best way to do that is with a Cascading Style Sheet (CSS), which modern browsers support. Font family, type size, colors, and other styles can be controlled with CSS for each kind of element.

Connect CSS to a Webpage

Insert the link of CSS Files into the HTML file. The link is to be put in the <HEAD> element.

Illustration

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

Illustration Explained

The above tag, links the CSS file named “stylesheet.css” to the current Web page.

HTML Tables

The HTML table model allows authors to arrange data -- text, preformatted text, images, links, forms, form fields, other tables, etc. -- into rows and columns of cells.

Tables

- Tables are defined with the <table> tag.
- A table is divided into rows (with the <tr> tag), and
- each row is divided into data cells (with the <td> tag).
- The letters td stands for "table data," which is the content of a data cell.
- A data cell can contain text, images, lists, paragraphs, forms, horizontal rules, tables, etc.

```
<table border="1">
   <tr>
       <td>row 1, cell 1</td>
       <td>row 1, cell 2</td>
   </tr>
   <tr>
      <td>row 2, cell 1</td>
      <td>row 2, cell 2p</td>
   </tr>
</table>
```

Headings in a Table

Headings in a table are defined with the <th> tag.

```
<table border="1">
   <tr>
      <th>Heading</th>
      <th>Another Heading</th>
   </tr>
   <tr>
      <td>row 1, cell 1</td>
      <td>row 1, cell 2</td>
   </tr>
   <tr>
      <td>row 2, cell 1</td>
      <td>row 2, cell 2p</td>
   </tr>
</table>
```

Tables and the Border Attribute

The table will be displayed without any borders if you do not specify a border attribute . Sometimes this can be useful, but most of the time, you want to show the borders.
To display a table with borders, you will have to use the border attribute:

```
<table border="1">
   <tr>
      <td>row 1, cell 1</td>
      <td>row 1, cell 2</td>
   </tr>
</table>
```

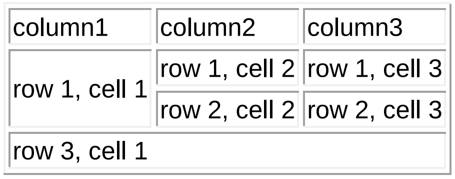
Spanning Multiple Rows and Cells

Use rowspan to span multiple rows & colspan to span multiple columns.
To display a table with borders, you will have to use the border attribute:

```
<table border="1">
   <tr>
      <td>column1</td>
      <td>column2</td>
      <td>column3</td>
   </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>
```

O/P :



column1	column2	column3
row 1, cell 1	row 1, cell 2	row 1, cell 3
	row 2, cell 2	row 2, cell 3
row 3, cell 1		

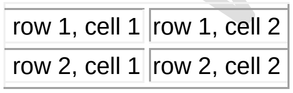
Cell Padding and Spacing

You will be able to adjust the white space on your tables with the cellpadding and cellspacing attributes .

Padding represents the distance between cell borders and the content within, while spacing defines the width of the border.

```
<table border="1">
    <tr>
        <td >row 1, cell 1</td>
        <td>row 1, cell 2</td>
    </tr>
    <tr>
        <td>row 2, cell 1</td>
        <td>row 2, cell 2</td>
    </tr>
</table>
```

O/P: Table without Cell Padding and Spacing

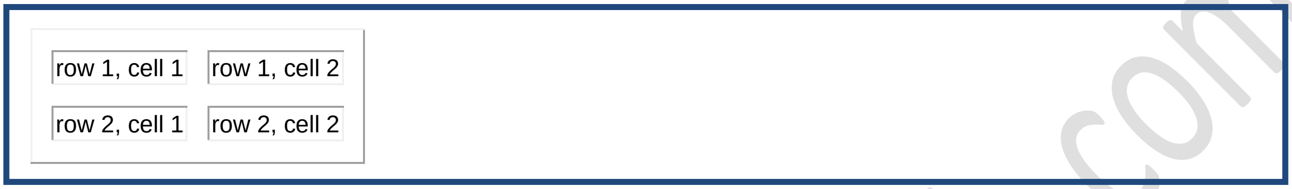


row 1, cell 1	row 1, cell 2
row 2, cell 1	row 2, cell 2

Example 2

```
<table border="1" cellspacing="10">
    <tr>
        <td >row 1, cell 1</td>
        <td>row 1, cell 2</td>
    </tr>
    <tr>
        <td>row 2, cell 1</td>
        <td>row 2, cell 2</td>
    </tr>
</table>
```

O/P: Table with Cellspacing

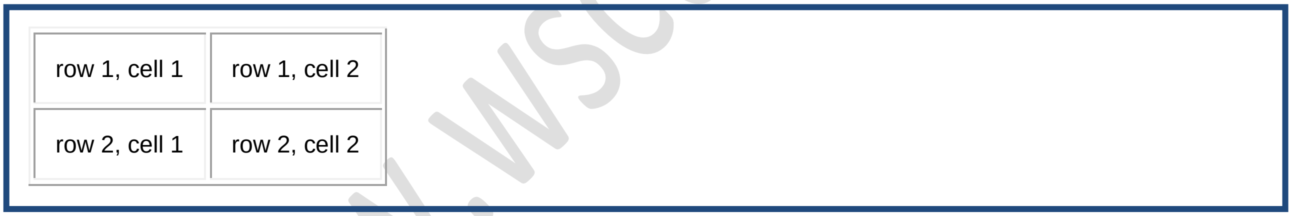


row 1, cell 1	row 1, cell 2
row 2, cell 1	row 2, cell 2

Table with Cellpadding

```
<table border="1" cellpadding="10">
    <tr>
        <td >row 1, cell 1</td>
        <td>row 1, cell 2</td>
    </tr>
    <tr>
        <td>row 2, cell 1</td>
        <td>row 2, cell 2</td>
    </tr>
</table>
```

O/P:



row 1, cell 1	row 1, cell 2
row 2, cell 1	row 2, cell 2

<font>

The <font> tag is used to change the format of the text on the web page.
The most important attributes are as follows:
face: The type of font. Common ones include "Time New Roman", "Verdana", and "Helvetica."
size: This indicates the size of the text. This can be absolute (0 .. 6), or relative ("+1", ... or "-1",...)
color: This attribute indicates the color of the text. Either the color name or the six-character color code may be used to specify color.
<font size=2 face="Helvetica" color=red>This illustrates the attributes of the font tag.</font>

HTML List Tags

Words or phrases which need to be set apart from the rest of the body of text can be emphasized with a “bullet” (a heavy dot used for calling attention to a particular section of text). An empty tag called a “list” tag is used to do this.

List Tag

<LI>: creates a bullet in front of text which is to be set apart for emphasis and causes all text after it to be indented, either until another list tag is detected or until the end of the list is reached. It is used to itemize elements of “unordered” and “ordered” lists.

Unordered List

An unordered list is a list of items. The list items are marked with bullets (typically small black circles). An unordered list starts with the <ul> tag. Each list item starts with the <li> tag.

Illustration

```
<html>
<body>

<ul>
        <li>January</li>
        <li>February</li>
</ul>

</body>
</html>
```

Illustration Output

```
    □    January
    □    February
```

Unordered List Illustration Explained

January & February are put in Bullet as they are inside an <ul> unordered list tags.

Ordered List

An ordered list is also a list of items. The list items are marked with numbers. An ordered list starts with the <ol> tag. Each list item starts with the <li> tag.

Illustration

```
<html>
<body>
<ol>
        <li>January</li>
        <li>February</li>
```

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

Illustration Output

1. January
2. February

Ordered List Illustration Explained

January & February are put in a numbered list as they are inside an <ol> Ordered list tags.

HTML Links

- A link is a connection from one Web resource to another
- A link has two ends(anchors) and a direction.
- The link starts at the "source" anchor and points to the "destination" anchor,
- which may be any Web resource (e.g., an image, a video clip, a sound bite, a program, an HTML document, an element within an HTML document, etc.).

The Anchor Tag and the Href Attribute

HTML uses the anchor (<a>) tag to create a link to another document.
An anchor can point to any resource on the Web: an HTML page, an image, a sound file, a movie, etc.

The syntax of creating an anchor:

<a href="url">Text to be displayed</a>
The <a> tag is used to create an anchor to link from, the href attribute is used to address the document to link to, and the words between the open and close of the anchor tag will be displayed as a hyperlink.

The Anchor Tag and the Name Attribute

The name attribute is used to create a named anchor. We can create links that can jump directly into a specific section on a page, using named anchors instead of letting the user scroll around to find what he/she is looking for.

Below is the syntax of a named anchor:

<a name="label">Text to be displayed</a>
To create a named anchor the name attribute is used. The name of the anchor can be any text you care to use.
The line below defines a named anchor:

<a name="main">Main Section</a>
You should notice that a named anchor is not displayed in a special way.

To link directly to the a section, add a # sign and the name of the anchor to the end of a URL, like this:
<a href="http://www.testsworld.com/HTML_links.asp#main"> Jump to the Main Section</a>

A hyperlink to the Main Section from within the file "HTML_links.asp" will look like this:

<a href="#main">Jump to the Main Section</a>

The Target Attribute

With the target attribute, you can define where the linked document will be opened.
The line below will open the document in a new browser window:

Below is the syntax of a named anchor:

<a href="http://www.testsworld.com/" target="_blank">Visit testsworld!</a>

Designing HTML Forms

A form is an area that can contain form elements.

Form elements are elements that allow the user to enter information (like text fields, textarea fields, drop-down menus, radio buttons, checkboxes, etc.) in a form.

- <form> is just another kind of HTML tag
- HTML forms are used to create (rather primitive) GUIs on Web pages
    - Usually the purpose is to ask the user for information
    - The information is then sent back to the server
- A form is an area that can contain form elements
    - The syntax is: <form *parameters*> *...form elements...* </form>
    - Form elements include: buttons, checkboxes, text fields, radio buttons, drop-down menus, etc
        - Other kinds of HTML tags can be mixed in with the form elements
    - A form usually contains a Submit button to send the information in he form elements to the server
    - The form's *parameters* tell JavaScript how to send the information to the server (there are two different ways it could be sent)
    - Forms can be used for other things, such as a GUI for simple programs

The <form> Tag

- The <form *arguments*> ... </form> tag encloses form elements (and probably other HTML as well)
- The arguments to form tell what to do with the user input
    - action="*url*" (required)
        - Specifies where to send the data when the Submit button is clicked
    - method="get" (default)
        - Form data is sent as a URL with ?form_data info appended to the end
        - Can be used *only* if data is all ASCII and not more than 100 characters
    - method="post"
        - Form data is sent in the body of the URL request
        - Cannot be bookmarked by most browsers
    - target="*target*"
        - Tells where to open the page sent as a result of the request
        - *target*= _blank means open in a new window
        - *target*= _top means use the same window

The <input> tag

- Most, but not all, form elements use the input tag, with a type="..." argument to tell which kind of element it is
    - type can be text, checkbox, radio, password, hidden, submit, reset, button, file, or image
- Other common input tag arguments include:
    - name: the name of the element
    - value: the "value" of the element; used in different ways for different values of type
    - readonly: the value cannot be changed
    - disabled: the user can't do anything with this element
    - Other arguments are defined for the input tag but have meaning only for certain values of type

A form is defined with the <form> tag.

Illustration

```
<form>
.
input elements
.
</form>
```

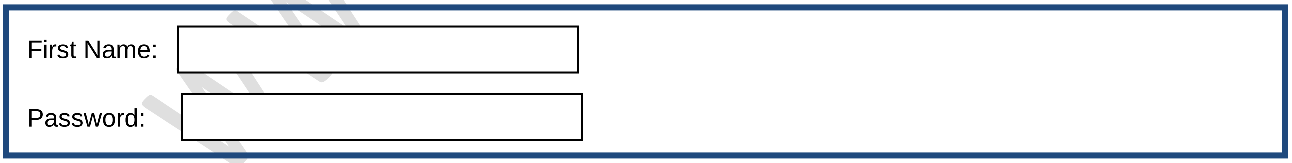
Textfield

Text fields are used when you want the user to type letters, numbers, etc. in a form.

Illustration

```
<form>
First name: <input type="text" name="firstname" /> <br />
Password: <input type="password" name="upass" />
</form>
```

Illustration Output



First Name:

Password:

Radio Button

Radio Buttons are used when you want the user to select one of a limited number of choices.

Illustration

```
<form>
```

```
<input type="radio" name="gender" value="male" /> Male <br />
<input type="radio" name=" gender " value="female" /> Female
</form>
```

Illustration Output

● Male

○ Female

Radio Button Illustration Explained

We used input type=”radio” for getting the Radio Buttons.

Checkbox

Checkboxes are used when you want the user to select one or more options of a limited number of choices.

Illustration

```
<form>
I have a bike: <input type="checkbox" name="vehicle" value="Bike" /> <br />
I have a car  :   <input type="checkbox" name="vehicle" value="Car" /> <br />
I have an airplane: <input type="checkbox" name="vehicle" value="Airplane" />
</form>
```

Illustration Output

I have a bike: □

I have a car: □

I have an airplane: □

Checkbox Illustration Explained

We used input type=”checkbox” for getting the Checkboxes.

Textarea

A text-area is a multi-line text input control. A user can write text in the text-area. In a text-area you can write an unlimited number of characters.

Rows and **columns** need to be specified as attributes to the <textarea> tag. Rows are roughly 12pixels high, the same as in word programs and the value of the columns reflects how many characters wide the text area will be.

Another attribute to be aware of is the **wrap**. Wrap has 3 values.

wrap= "off"
"virtual"
"physical"

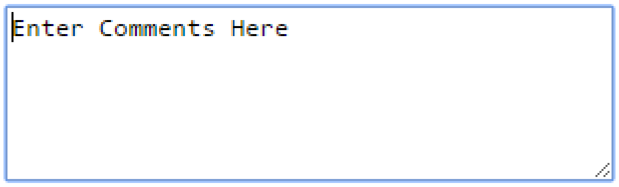
Virtual means that the viewer will see the words wrapping as they type their comments, but when the page is submitted to you, the web host, the document sent will not have wrapping words.

Physical means that the text will appear both to you, the web host, and the viewer including any page breaks and additional spaces that may be inputed. The words come as they are.

Off turns off word wrapping within the text area. One ongoing line.

```
<form>
      <textarea rows="5" cols="20" wrap="physical" name="comments">
      Enter Comments Here
      </textarea>
</form>
```

Output :



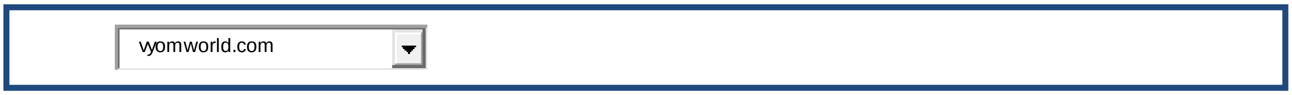
Enter Comments Here

Drop Down Lists

With the **<select>** and **<option>** tags drop down menues are created.
<select> is the list itself and each <option> is an available choice for the user.

```
<form>
      <select name="ourSites">
           <option>vyomworld.com</option>
           <option>testsworld.com</option>
           <option>jobsassist.com</option>
           <option>vyomlinks.com</option>
           <option>sourcecodesworld.com</option>
           <option>coolinterview.com</option>
           <option>tasty-food.com</option>
      </select>
</form>
```

Output :


vyomworld.com

HTML Images

You can display images in a document.

Placing An Image On Your Page

Here's the format for placing an image:

```
<IMG SRC="image.gif" ALT="some text"  width="500" height="200" />
```

Here's What's Happening

IMG

IMG stands for "image."

It announces to the browser that an image will go here on the page. Yes, the image will pop up right where you write in the image tag.

SRC

SRC stands for "source." This again is an attribute, a command inside a command. The value of the src attribute is the URL of the image you want to display on your page. It's telling the browser where to go to find the image. Again, it's best for you to place the images you want to use in the same directory as the page. This way you can call for the image by name alone. If you start to place your images all over the place, you'll have to start adding directories and sub-directories to the SRC attribute. And at this point, that is way too confusing. Just place the image in the same place as the HTML document that will call for it and then call for the image by name alone. You can get fancy later. Right now, let's just get it to work.

image.gif

image.gif is the name of the image. Notice it's following the same type of format as your HTML documents. There is a name (image) then a dot and then there is a suffix (gif).

ALT

ALT stands for "alternate text". This tells the browser that if it can't find the image, then just display this text. It also tells anyone who can't view your image what the image is about. For example a disabled user using a screen reader, or dare I mention it, Search Engines.

"some text"

"some text" is where you put the text describing your image.

HTML Backgrounds

Make textures, colors, and pictures as your page background!

Backgrounds

The <body> tag has two attributes where you can specify backgrounds. The background can be a color or an image.

Bgcolor

This attribute specifies a background-color for an HTML page. The value of this attribute can be a hexadecimal number, an RGB value, or a color name:

```
<body bgcolor="#000000">
<body bgcolor="rgb(0,0,0)">
<body bgcolor="black">
```

The lines above all set the background-color to black.

Background

The background attribute specifies a background-image for an HTML page. The value of this attribute is the URL of the image you want to use. If the image is smaller than the browser window, the image will repeat itself until it fills the entire browser window.

```
<body background="clouds.gif">
<body background="http://www.academictutorials.com/clouds.gif">
```

The URL can be relative (as in the first line above) or absolute (as in the second line above).

HTML Colors

Colors are displayed combining RED, GREEN, and BLUE light sources.

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).

RGB Values

RGB stands for Red, Green, Blue. Each can have a value from 0 (none of that color) to 255 (fully that color). The format for RGB is - rgb(RED, GREEN, BLUE), just like the name implies.

We do not recommend that you use RGB for safe web design because non-IE browsers do not support HTML RGB. However, if you plan on learning CSS then you should glance over this topic.

```
bgcolor="rgb(255,255,255)" White

bgcolor="rgb(255,0,0)" Red

bgcolor="rgb(0,255,0)" Green

bgcolor="rgb(0,0,255)" Blue

bgcolor="rgb(0,0,0)" Black
```

Hexadecimal

Hexadecimals are far more reliable and widely compatible among web browsers and are the standard for colors on the internet.

A hexadecimal is a 6 digit representation of a color. The first two digits(RR) represent a red value, the next two are a green value(GG), and the last are the blue value(BB).

Hex numbers use 16 digits:

0 1 2 3 4 5 6 7 8 9 A B C D E F

Zero, "0", is the smallest representations of a color. It's almost the total absence of color. F is 15 times the intensity of the color of 0. Combinations of these digits create different shades of a particular color. Double Zero, "00," is equal to zero hue. FF is equal to a pure color.

This color representation is done three times, once for red, once for green, and once for blue, in that order. Put the three, two-digit, codes together and you get a 6-digit hex code. The hex code is just a representation of the red, green, and blue intensity, in that order. The computer creates the three intensities, mashes them together, and you get a single shade of color.

<div> Tags

The <div> tag in XHTML is a tag that defines logical divisions within the content of a page. What this means is that a <div> tag defines sections of a Web page to make it easier to manage, style, and manipulate. You can use the <div> tag when you want to center a block of content or position a content block on the page. The <div> tag is a very powerful tool for Web developers.

It's a good idea to label your <div> tags as you place them in your document. For example, if you're defining the main content area of your site, you should name that DIV tag: "maincontent". <div id="maincontent">

The <div> element is typically used in XHTML+CSS documents to design and position portions of the page.

We will cover the usage of <div> tags in more detail, while using CSS.

Illustration

```
<html>
<body>

<div class="header" id="1" name="top" >
        // You can give the div id, name or class attribute to specify the particular div and then can style //and
        position the various div elements using the CSS.
</div>

</body>
</html>
```

<div> Tag Illustration Explained

The <div> tag creates a division in the Web page. We can then give any design or position to the easily to the <div> using the CSS.

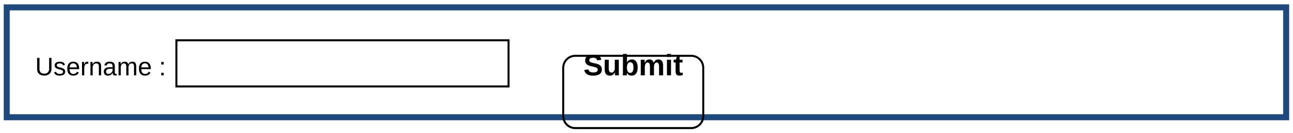
Form Action Attribute & Submit Button

When the user clicks on the "Submit" button, the content of the form is sent to the server. The form's action attribute defines the name of the file to send the content to. The file defined in the action attribute usually does something with the received input.

Illustration

```
<form name="input" action="html_form_submit.asp" method="get">
Username: <input type="text" name="user" />
<input type="submit" value="Submit" />
</form>
```

Illustration Output



Form Action Illustration Explained

If you type some characters in the text field above, and click the "Submit" button, the browser will send your input to a page called "html_form_submit.asp". The page will show you the received input.

HTML 4.01 Quick List

HTML Basic Document

```
<HTML>
        <head>
                <title>Document name goes        here</title>
        </head>

        <body>
                Visible text goes here
        </body>
</HTML>
```

Heading Elements

```
<h1>Largest Heading</h1>
<h2> . . . </h2>
<h3> . . . </h3>
<h4> . . . </h4>
<h5> . . . </h5>

<h6>Smallest Heading</h6>
```

Text Elements

```
<p>This is a paragraph</p>
<br> (line break)
<hr> (horizontal rule)
<pre>This text is preformatted</pre>
```

Logical Styles

```
<em>This text is emphasized</em>
<strong>This text is strong</strong>
<code>This is some computer code</code>
```

Physical Styles

```
<b>This text is bold</b>
<i>This text is italic</i>
```

Links, Anchors, and Image Elements

```
<a href="http://www.example.com/">This is a Link</a>
<a href="http://www.example.com/"><img src="URL" alt="Alternate Text"></a>
<a href="mailto:webmaster@example.com">Send e-mail</a>
```

A named anchor:
<a name="tips">Useful Tips Section</a>
<a href="#tips">Jump to the Useful Tips Section</a>

Unordered list

```
<ul>
        <li>First item</li>
        <li>Next item</li>
</ul>
```

Ordered list

```
<ol>
        <li>First item</li>
        <li>Next item</li>
</ol>
```

Definition list

```
<dl>
        <dt>First term</dt>
                <dd>Definition</dd>
        <dt>Next term</dt>
                <dd>Definition</dd>
</dl>
```

Tables

```
<table border="1">
        <tr>
                <th>someheader</th>
                <th>someheader</th>
        </tr>
        <tr>
                <td>sometext</td>
                <td>sometext</td>
        </tr>
</table>
```

Frames

```
<frameset cols="25%,75%">
<frame src="page1.htm">
<frame src="page2.htm">
</frameset>
```

Forms

```
<form action="http://www.example.com/test.asp"    method="post/get">
<input type="text" name="lastname" value="Nixon" size="30" maxlength="50">
<input type="password">
<input type="checkbox" checked="checked">
<input type="radio" checked="checked">
<input type="submit">
<input type="reset">
```

```
<input type="hidden">
<select>
            <option>Apples </option>
            <option selected>Bananas</option>
            <option>Cherries </option>
</select>

<textarea name="Comment" rows="60" cols="20"></textarea>

</form>
```

Entities

```
&lt; is the same as <
&gt; is the same as          >
&#169; is the same as ©
```

Other Elements

```
<!-- This is a comment -->

<blockquote>
            Text quoted from some source.
</blockquote>

<address>
            Address 1<br>
            Address 2<br>
            City<br>
</address>
```

CSS CUBE

Building Attractive Websites with CSS

BY WsCube Tech

CSS (Cascading Style Sheet)

CSS to control the style and layout of multiple Web pages all at once.
CSS Saves a Lot of Work!, CSS defines HOW HTML elements are to be displayed.

What is CSS?

- CSS stands for Cascading Style Sheets
- Styles define how to display HTML elements
- Styles were added to HTML 4.0 to solve a problem
- External Style Sheets can save a lot of work
- External Style Sheets are stored in CSS files

Benefits of CSS

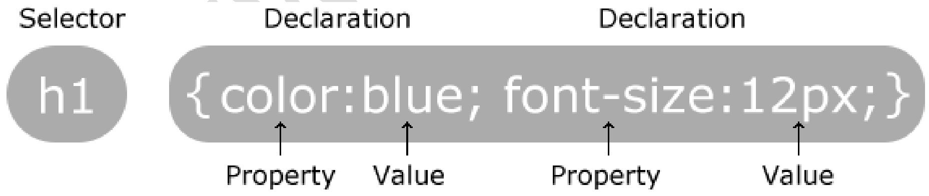
- **Pages download faster**, sometimes by as much as 50%.
- Separates structure from presentation
- Provides advanced control of presentation
- Easy maintenance of multiple pages
- Better accessibility for disabled users
- Easy to learn
- You have to type less code, and your pages are shorter and neater.

CSS Syntax

The CSS syntax is made up of three parts: a selector, a property and a value:

```
selector { property: value; property: value; property: value; }
```

The selector is normally the HTML element/tag you wish to define, the property is the attribute you wish to change, and each property can take a value. The property and value are separated by a colon, and surrounded by curly braces:



Rules :

- The **selector** is usually the name of a tag, without its surrounding angle-brackets.
- The **braces are {curly}**, not [square] or (round).

- After the property name there is a colon, and between each individual part there is a**semicolon**. Each of these pairs of properties and values is a **declaration**.

Note: If the value is multiple words, put quotes around the value:

```
p {font-family: "sans serif"}
```

Note: If you wish to specify more than one property, you must separate each property with a semicolon. The example below shows how to define a center aligned paragraph, with a red text color:

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

To make the style definitions more readable, you can describe one property on each line, like this:

```
p{
 text-align: center;
 color: black;
 font-family: arial
}
```

Three Ways to Insert CSS

There are three ways of inserting a style sheet:

- External style sheet (Use one CSS file for all your pages. This is the best way to do it.)
- Internal style sheet (Integrate CSS commands (<style></style>) into the head of each of your documents.)
- Inline style (Use the style attribute to put CSS code directly into a HTML element.)

1.) External Style Sheet

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

2.) Internal Style Sheet

```
<head>
<style>
        hr { color : sienna; }
        p  { margin-left : 20px; }
        body { background-image : url("images/background.gif"); }
</style>
</head>
```

3. Inline Styles

```
 <p style="color:red; margin-left:20px;"> Welcome to WsCube Tech </p>
```

Grouping

You can group selectors. Separate each selector with a comma. In the example below we have grouped all the header elements. All header elements will be displayed in green text color:

```
h1, h2, h3, h4, h5, h6
{
color: green
}
```

Cascading Order

a) Browser default
b) External style sheet
c) Internal style sheet (inside the <head> tag)
d) Inline style (inside an HTML element)

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

Comment in CSS

- Explain the purpose of the coding
- Help others read and understand the code
- Serve as a reminder to you for what it all means

```
p  { color: #ff0000; }   /* Company */
```

IDs and Classes & HTML Tags

- IDs (#) are unique and can only be used once on the page
- Classes (.) can be used as many times as needed
- HTML TAGS can be used html tag in css

CSS Properties

Text and Fonts

1. **font :** italic bold 1.5em/2 arial, Helvetica, sans-serif;
2. **font-family :** "Times New Roman", Times, serif;
3. **font-size :** 20px;   (px Represents Pixels)
4. **font-weight :** bold;
5. **font-style :** italic;
6. **text-align :** justify;
7. **text-decoration :** none;
8. **text-transform :** uppercase;

9. **vertical-align :** top;

Colors and Backgrounds and Borders

1. **color :** red; (For Text Color)
   color : #ff6600;
2. **background-color :** red;
3. **background :** url(images/wscubetech.jpg) no-repeat top left;
4. **background-image :** url(images/wscubetech.jpg);
5. **background-repeat :** no-repeat;
   repeat (default) - tiled, repeating the image both horizontally and vertically.
   repeat-x - repeating the image horizontally only.
   repeat-y - repeating the image vertically only.
   no-repeat - not repeating the image at all, showing just one instance.
   background-repeat : repeat-x;
6. **background-position**: bottom right;
7. **background-attachment :** fixed;
8. **border :** solid 1px red; (Types – solid, dotted, dashed, double etc.)
9. **border-top:** solid 1px red;
10. **border-right:** solid 1px red;
11. **border-bottom:** solid 1px red;
12. **border-left:** solid 1px red;
13. **border-radius :** 20px;

The Box Model - dimensions, padding, margin

1. **Padding :** 20px; (Padding is used to Move Content)
2. **Padding :** 20px 30px; (1st is for top-bottom Direction and 2nd is left-right Direction)
3. **Padding :** 10px 20px 30px 40px; (top right bottom left direction)
4. **padding-top :** 10px;
5. **padding-right :** 20px;
6. **padding-bottom :** 30px;
7. **padding-left :** 40px;

8. **margin:** 20px; (Margin is used to Move Block / Div)
9. **margin:** 20px 30px; (1st is for top-bottom Direction and 2nd is left-right Direction)
10. **margin :** 10px 20px 30px 40px; (top right bottom left direction)
11. **margin-top :** 10px;
12. **margin-right :** 20px;
13. **margin-bottom :** 30px;
14. **margin-left :** 40px;

15. **width :** 212px;
16. **height :** 3px;

Other Properties

1. **overflow :** hidden; **overflow :** auto;
2. **z-index** (The CSS z-index property is used to layer elements in front or behind each other. This is referred to as the 'stack level'. This only applies to positioned elements (i.e. elements positioned using the position property).)
   z-index : 2
   z-index : 1

3. **float :** left; **float :** right;
4. **cursor :** pointer ;
5. **position :** relative / absolute;
6. **top :** 20px;
7. **right :** 30px;
8. **bottom :** 40px;
9. **right :** 25px;

10. **list-style-type:** square ;
11. **list-style-image :** url(images/wscubetech.jpg) ;

12. **line-height :** 22px;

13. **letter-spacing :** 3px;

14. **word-spacing** : 2px;

15. **clear**
    clear : both; **clear :** left; **clear :** right;

16. **display**
    display : block;
    display : none;

Note: To reset or clear the value of float, we use the clear property in css at the end of last column of the row.

CSS Positioning

- Decide which element to display in front!
- Elements can overlap!

The CSS positioning properties allow you to position an element. It can also place an element behind another, and specify what should happen when an element's content is too big.

Elements can be positioned using the top, bottom, left, and right properties.

a.) Static Positioning
HTML elements are positioned static by default. A static positioned element is always positioned according to the normal flow of the page.

Static positioned elements are not affected by the top, bottom, left, and right properties.
An element with fixed position is positioned relative to the browser window.

position : fixed;

b.) Relative Positioning

A relative positioned element is positioned relative to its normal position.

position : relative;

c.) Absolute Positioning

An absolute position element is positioned relative to the first parent element that has a position other than static. If no such element is found, the containing block is <html>:

position : absolute;

d.) Fixed Positioning

An element with fixed position is positioned relative to the browser window.
It will not move even if the window is scrolled:

position : fixed;