

VER.  
1.1



# CSS

STYLE. STRUCTURE. SIMPLICITY.  
BUILD BEAUTIFUL WEBSITES.

# BASICS OF HTML

DR. SUNIL KUMAR VATS



#### EASY TO LEARN

Simple language  
for everyone.



#### PRACTICAL EXAMPLES

Learn with real  
world examples.



#### BEGINNER FRIENDLY

Step-by-step guide  
for beginners.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Learn HTML</title>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
```

## HTML



BUILD YOUR

WEB  
FUTURE



LEARN TODAY. CODE TOMORROW.

ISBN: 978-93-5258-140-5



*Dedicated to my  
grandparents*

**Late Sh. Radha Krishan**

**&**

**Smt. Bharpi Devi**



*An inspiring source of  
motivation for me...*



**© 2021, IJRTS Publications**

This book is an accurate reproduction of the original. Any marks, names, colophons, imprints, logos, or symbols or identifiers that appear on or in this book, except for those of IJRTS Publications, are used only for historical reference and accuracy and are not intended to designate origin or imply any sponsorship by or license from any third party.

---

### **Limits of Liability and Disclaimer of Warranty**

The authors and publishers of this book have tried their best to ensure that the derivations, procedures & functions contained in the book are correct. However, the author and the publishers make no warranty of any kind, expressed or implied, with regard to these derivations, procedures & functions or the documentation contained in this book. The author and the publishers shall not be liable in any event for any damages, incidental or consequential, in connection with, or arising out of the furnishing, performance or use of these derivations, procedures & functions. Product names mentioned are used for identification purposes only and may be trademarks of their respective persons or companies.

The disclaimer of warranties and limitation of liability provided above shall be interpreted in a manner that, to the extent possible, most closely approximates an absolute disclaimer and waiver of all liability.

**ISBN:** 978-93-5258-140-5

**Price:** ₹ 290

Published by Dr. Vipin Mittal for IJRTS Publications, Crescent ParC, Sector 92, Gurgaon-122001

Printed in India

by Manyu Cyber Cafe

Bound in India

By Twarit Print Solution



**IJRTS Publications**

**Crescent ParC, Sector 92, Gurgaon-122001**

**DISCLAIMER**

While every effort has been made to ensure that the book is free of error, it is inevitable that some errors still remain. Please report any errors, suggestions or questions to the authors at the following email address.

[ijrts13@gmail.com](mailto:ijrts13@gmail.com)

[ijrts14@gmail.com](mailto:ijrts14@gmail.com)

# INDEX

CONTENTS	PAGE NO.
DISCLAIMER .....	ii
Preface .....	iii
Chapter 1: HTML Introduction .....	01
Chapter 2: HTML Tags .....	05
Chapter 3: Marquee .....	13
Chapter 4: HTML Deprecated Attributes .....	35
Chapter 5: Programming Variable .....	54
Chapter 6: Setting of Cookies .....	59
Chapter 7: Images .....	66
Chapter 8: Inserting Table .....	69
Chapter 9: Lists .....	88
Chapter 10: Hyperlink .....	96
Chapter 11: Frames .....	102
Chapter 12: F.A.Q. (HTML) .....	121
Chapter 13: Glossary .....	123
Appendix: Internet Terminology .....	138

## Preface

Hypertext Markup Language (HTML) is the underlying markup language of the World Wide Web. It's the common thread that ties together virtually every Web site, from large scale corporate sites such as Microsoft's to single-page classroom projects at the local grade school.

Don't let the phrase "markup language" intimidate you. A markup language annotates or "marks up" plain text, letting a browser know how to format that text so it looks good on a Web page. It's easy to get started—in fact, you can create a simple Web page in just a few minutes.

While full-featured What You See Is What You Get (WYSIWYG) tools exist that can help speed up the process of writing Web pages, all you really need is an ordinary text-editing program such as Microsoft Notepad. You don't need special software or extensive training.

This book is divided into 11 chapters where you'll learn some basics about HTML. You'll find out how they turn plain text into attractive formatting, how they incorporate graphics and hyperlinks, and how anyone can create Web content in virtually any program that edits text.

This introduction explains what hyperlinking is, and how they make formatting consistent across large Web sites. You'll also discover the differences between HTML4, XHTML, and HTML5, so you can make the important decision about which version of HTML you want your code to conform to. Finally, you'll learn about the conventions used in this book for pointing out special helps like notes, tips, cautions, and references to the data files.

Chapter 12 consists of F.A.Q. and Glossary is included in the form of Chapter 13.

I would like to thank Navodaya Vidyalaya Samiti for giving me the chance and freedom I needed to develop these notes. I would also like to acknowledge the influence of my colleagues. The most beautiful proofs and ideas grew out of material that I learned from him.

There are times in such projects when the clock beats you time and again and you run out of energy and you just want to finish it once and for ever. My parents Sh. Pawan Kumar & Smt. Savitari Devi made me endure such times with their unfailing humor and warm wishes with full of mental support and love.

To my wife Smt. Sharma Sharma, I owe more than what I can mention. Unintentional love & support of my brother Arvind, sister Neelam and my twin chaps Abhishek & Abhinav helped me profoundly in making this coding script even livelier.

This acknowledgement would not be completed without extending my thanks to my role model Sh. Dharamvir Arya (B.E.E.O.), who helped me clear any doubts that arose during my writing.

Any and all remaining errors or inconsistencies are mine. I will gladly take reader and user feedback to correct them, along with other suggestions to improve the text.

Dr. Sunil Kumar Vats (*MCA, PhD*)

PGT Computer Science

Jawahar Navodaya Vidyalaya

Mohindergarh, Haryana

e: sunilvats1981@gmail.com

t: +91-941 6266 966

# CHAPTER-1

## HTML INTRODUCTION

### History of World Wide Web

Internet is such a wonder box that whenever we need some information on any topic, we refer to internet. Information is available on almost all the topics - from Ayurveda to Advanced Medical Surgeries, from basic colors to advanced designs, from fundamentals of computers to latest developments in the field of supercomputers - you name it and internet has it. This information is actually stored on some computers on the net. These computers are called the servers. The information is stored in the form of some documents called hypertext documents. All the related documents on a server are linked together using hyperlinks. Therefore using hyperlinks we can move from one document to another. This is formally called navigation. There are a number of ways to create hypertext documents. There are many specialized software packages like Dreamweaver, CoffeCup etc. which are used to create web documents. The simplest way to create a web document is to use a text editor like notepad, notepad2 etc. and write code in HTML.

A hypertext document on the web is also called a web page. The information over the web is shared using a protocol called Hyper Text Transfer Protocol (HTTP). The concept of WWW was designed in 1989 by Tim Berners-Lee and scientists at CERN (Geneva), the European centre for High Energy Physics. Their purpose was to make sharing and retrieval of research material simpler. A year later they had developed a 'browser/editor' program and had named the program World Wide Web. The World Wide Web grew rapidly and attained its present form. Its further development is guided by the WWW Consortium (W3C) based at the Massachusetts Institute of Technology in Cambridge, Massachusetts

### Uniform Resource Locator

The uniform resource locator (URL) is the unique identifier of a web page. The address or URL of the current page you are on appears in the "Address Bar" of the web browser. You can go directly to a web page if you know its URL by simply typing the URL in the address bar. You can click in the address bar at any time and overwrite the current address with another URL to jump to a different web page. The most general form of a URL syntax is as follows:

Protocol://domain name/<directory path>/<object name>

HTML stands for **H**yper **T**ext **M**arkup **L**anguage, which is the most widely used language on Web to develop web pages. HTML was created Berners-Lee in late 1991 but "HTML 2.0" was the first standard HTML specification which was published in 1995. HTML 3.2 and 4.01 was a major version of HTML and it was published in 1997 and 1999 respectively. Though HTML 4.01 version

is widely used but currently we are having HTML-5 version which is an extension to HTML 4.01, and this version was published in 2014.

## HTML Versions

Since the early days of the web, there have been many versions of HTML:

Version	Year
HTML	1991
HTML 2.0	1995
HTML 3.2	1997
HTML 4.01	1999
XHTML	2000
HTML5	2014

HTML stands for **H**ypertext **M**arkup **L**anguage, and it is the most widely used language to create Web Pages in the current time.

- **Hypertext** refers to the way in which a web page link to another web pages. Thus the link available on a webpage are called Hypertext.
- As its name suggests, HTML is a **Markup Language** which means you use HTML to simply "mark up" a text document with tags that tell a Web browser how to the content of the document will display.

Originally, HTML was developed with the intent of defining the structure of documents like headings, paragraphs, lists, and so forth to facilitate the sharing of scientific information between researchers.

## HTML Code

HTML can be edited by using professional HTML editors like Microsoft WebMatrix, Sublime Text. However, for learning HTML we recommend a text editor like Notepad (PC) or TextEdit (Mac).

To simply create the web page, a user must follow the four steps:

---

### Step 1: Open Notepad

To Open Notepad in the earlier version of Windows before Windows 8

- ✓ Click **Start** (bottom left on your screen). Click **All Programs**. Click **Accessories**. Click **Notepad**.

To open Notepad in Windows 8 or later:

- ✓ Open the **Start Screen** (the window symbol at the bottom left on your screen). Type **Notepad**.

---

## Step 2: Write Some HTML

Write codes of HTML into Notepad using required tags like:

```
<!DOCTYPE html>
```

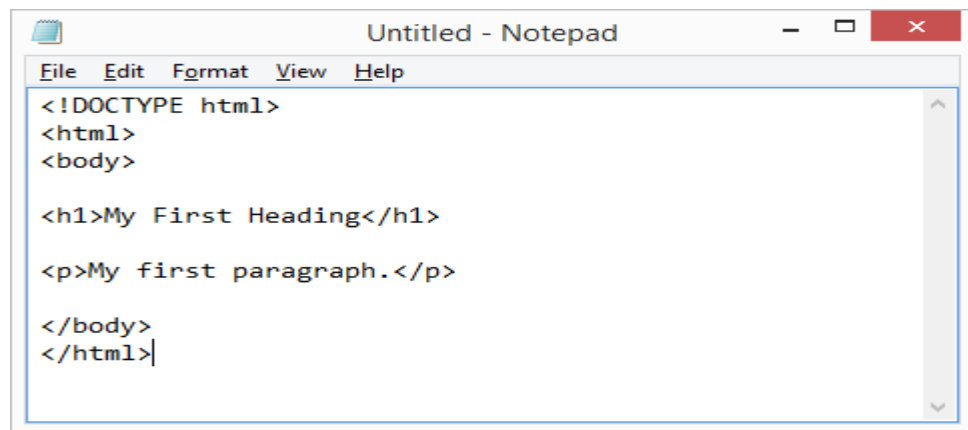
```
<html>
```

```
<body>
```

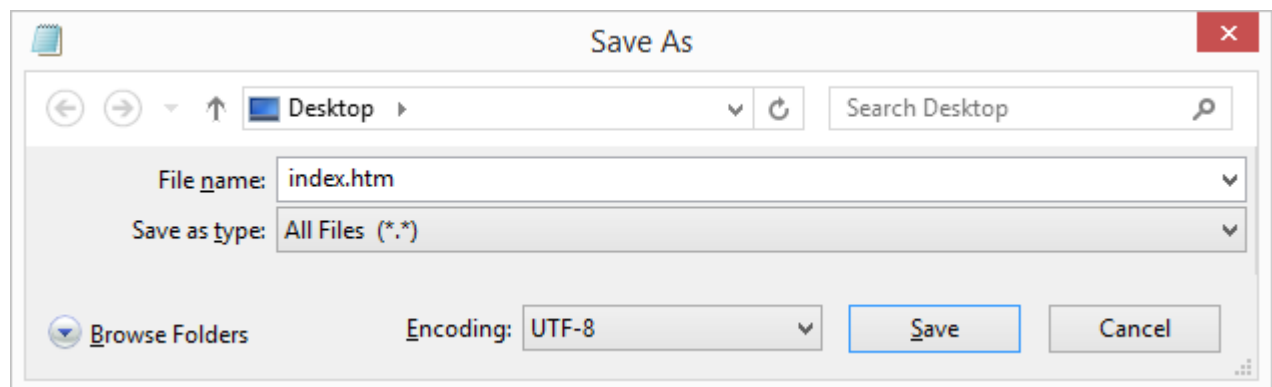
```
<h1>My First Heading</h1>
```

```
<p>My first paragraph.</p>
```

```
</body>
```



</html>



---

## Step 3: Save the HTML Page

Save the file on your computer.

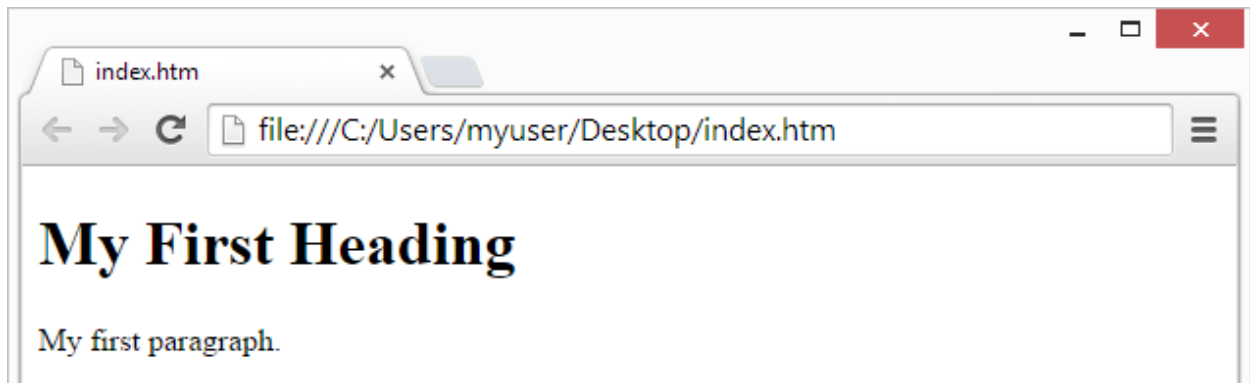
Select **File > Save as** in the Notepad menu.

Name the file followed by .html or .htm extension. UTF-8 is the preferred encoding for HTML files. ANSI encoding covers US and Western European characters only.

---

#### Step 4: View HTML Page in Your Browser

Open the saved HTML file in your favourite browser. The result will look much like this:



To open a file in a browser, double click on the file, or right-click, and choose open with.

## CHAPTER-2

### HTML TAGS

As told earlier, HTML is a markup language and makes use of various tags to format the content. These tags are enclosed within angle braces **<Tag Name>**. Except few tags, most of the tags have their corresponding closing tags. For example **<html>** has its closing tag **</html>** and **<body>** tag has its closing tag **</body>** tag etc. Above example of HTML document uses following tags:

Tag	Description
<!DOCTYPE...>	This tag defines the document type and HTML version.
<html>	This tag encloses the complete HTML document and mainly comprises of document header which is represented by <b>&lt;head&gt;...&lt;/head&gt;</b> and document body which is represented by <b>&lt;body&gt;...&lt;/body&gt;</b> tags.
<head>	This tag represents the document's header which can keep other HTML tags like <b>&lt;title&gt;</b> , <b>&lt;link&gt;</b> etc.
<title>	The <b>&lt;title&gt;</b> tag is used inside the <b>&lt;head&gt;</b> tag to mention the document title.
<body>	This tag represents the document's body which keeps other HTML tags like <b>&lt;h1&gt;</b> , <b>&lt;div&gt;</b> , <b>&lt;p&gt;</b> etc.
<h1>	This tag represents the heading.
<p>	This tag represents a paragraph.

To learn HTML, you will need to study various tags with their purpose and understand how they behave while their use in document. Learning HTML is simple as users have to learn the usage of different tags in order to format the text or images to make a beautiful webpage. World Wide Web Consortium (W3C) recommends to use lowercase tags starting from HTML 4.

#### HTML Document Structure

A typical HTML document will have following structure:

Document declaration tag

```
<html>
```

```
  <head>
```

```
    Document header related tags
```

```
  </head>
```

```
  <body>
```

```
    Document body related tags
```

```
  </body>
```

</html>

We will study all the header and body tags in subsequent chapters, for now let's see what is document declaration tag.

### **The <!DOCTYPE> Declaration**

The <!DOCTYPE> declaration tag is used by the web browser to understand the version of the HTML used in the document. Current version of HTML is 5 and it makes use of the following declaration:

```
<!DOCTYPE html>
```

There are many other declaration types which can be used in HTML document depending on what version of HTML is being used. We will see more details on this while discussing <!DOCTYPE...> tag along with other HTML tags. HTML is not case sensitive, i.e., tags and attributes can be written in small or capital alphabets. They can also be written using a mix of capital and small alphabets.

### **HTML Element**

It is a container element started by <HTML> tag and ended by </HTML> tag. It identifies the document as an HTML document. It does not have any effect on the appearance of the document, but tells the browser that the current document is an HTML document.

Syntax: <HTML> ...</HTML>

HTML element further contains HEAD and the BODY elements, which can further contain a number of other elements.

### **HEAD Element**

It is a container element started by <HEAD> tag and ended by </HEAD> tag. It defines the HTML document header and does not affect the appearance of the document in the browser window. The header contains information about the document.

Syntax: <HEAD> ...</HEAD>

### **TITLE Element**

It is a container element started by <TITLE> tag and ended by </TITLE> tag. Every HTML document should contain the title to be displayed in the title bar of the browser window. If an HTML document does not contain a title, then the file name of the HTML document is displayed in the title bar. System Security is written in the TITLE element and thus it is displayed in the browser's title window. The TITLE element is placed in HEAD element.

Syntax: <TITLE>...</TITLE>

### **BODY Element**

It is a container element started by <BODY> tag and ended by </BODY> tag. It contains the main contents of the document as parameter.

Syntax: <BODY> . . .</BODY>

BODY tag contains many attributes. Some of the important attributes are discussed below:-

### 1. Background

It is used to specify the path and filename of an image that has to be used as the Background of the document. If the referenced image is smaller than the browser window, it will be tiled to fit and will scroll with the text on the page. The filename extension has to be specified along with the filename.

**BACKGROUND= "path/filename.jpg"**

It sets the background colour of the web page. Most of the browsers recognize the popular colour names like RED, GREEN, YELLOW, GREY, AQUA ETC.

### 2. BGCOLOR

If we want to specify a colour which does not have a specific name but we know its RRGGBB composition, then we can specify this RRGGBB composition in the BGCOLOR attribute to get the background color. instead of writing BGCOLOR = "GREY", we could have written BGCOLOR = "#999999" and would have got the same effect. If a background image is also present, the BGCOLOR specified shines through regions where the background image is transparent.

**BGCOLOR= "colorname" OR BGCOLOR= "#rrggb"**

### 3. Text

It sets the colour of the normal text in the document. Colour values can be given in the same way as that of the BGCOLOR attribute. The default is black (hexadecimal code

#000000).TEXT = "colorname" OR TEXT = "#rrggbb"

## Elements and Attributes

### Container Elements

A container element is specified by a pair of tags - Start tag and End tag. These tags also called ON tags and OFF tags. Start tag consists of the tag name enclosed in left and right angular brackets. The end tag is identical to the start tag, except for a slash (/) that precedes the text within angular brackets of the end tag. e.g.,

**<BODY> ... </Body>**

Container elements contain parameters and the parameters of an element are given between the start tag and end tags.

**<BODY> ... </BODY>**



### Parameters

Elements in HTML may also contain attributes that can be given along with the tag name in the angular brackets of the start tag. E.g.,

**<BODY BGCOLOR="BLUE" TEXT="RED">**



### Attributes

### Empty Elements

Empty elements have only a start tag and no end tag. Hence, an empty element has no parameters, but can take attributes, which are given within angular brackets of the start tag.

Example <BR> tag

### Heading Tags

Any document starts with a heading. You can use different sizes for your headings. HTML also has six levels of headings, which use the elements <h1>, <h2>, <h3>, <h4>, <h5>, and <h6>. While displaying any heading, browser adds one line before and one line after that heading.

<!DOCTYPE html> <html>

<head>

<title>Heading Example</title>

</head>

<body>

<h1>This is heading 1</h1>

<h2>This is heading 2</h2>

<h3>This is heading 3</h3>

**This is heading 1**

**This is heading 2**

**This is heading 3**

**This is heading 4**

**This is heading 5**

**This is heading 6**

```
<h4>This is heading 4</h4>
```

```
<h5>This is heading 5</h5>
```

```
<h6>This is heading 6</h6>
```

```
</body>
```

```
</html>
```

## Paragraph Tag

The **<p>** tag offers a way to structure your text into different paragraphs. Each paragraph of text should go in between an opening **<p>** and a closing **</p>** tag as shown below in the example:

Example

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Paragraph Example</title>
```

```
</head>
```

```
<body>
```

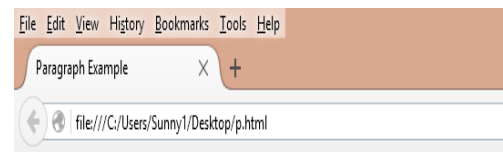
```
<p>Here is a first paragraph of text.</p>
```

```
<p>Here is a second paragraph of text.</p>
```

```
<p>Here is a third paragraph of text.</p>
```

```
</body>
```

```
</html>
```



Here is a first paragraph of text.

Here is a second paragraph of text.

Here is a third paragraph of text.

## Line Break

Whenever you use the **<br />** element, anything following it starts from the next line. This tag is an example of an **empty** element, where you do not need opening and closing tags, as there is nothing to go in between them.

The **<br />** tag has a space between the characters **br** and the forward slash. If you omit this space, older browsers will have trouble rendering the line break, while if you miss the forward slash character and just use **<br>** it is not valid in XHTML

```
<!DOCTYPE html>
```

```
<html>
```

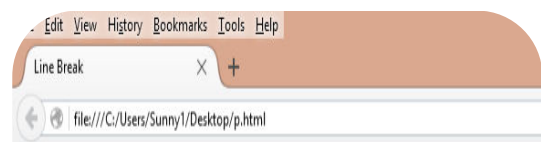
```
<head>
```

```
<title>Line Break</title>
```

```
</head>
```

```
<body>
```

```
<p>Hello</br>
```



Hello

You delivered your assignment.

Thanks

Mahnaz

You delivered your assignment.</br>

Thanks</br>

Mahnaz</p>

</body>

</html>

## Centering Content

You can use **<center>** tag to put any content in the center of the page or any table cell.

<!DOCTYPE html>

<html>

<head>

<title>Centring Content Example</title>

</head>

<body>

<p>This text is not in the center.</p>

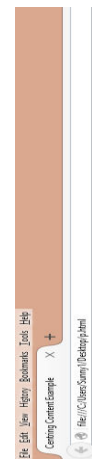
<center>

<p>This text is in the center.</p>

</center>

</body>

</html>



This text is not in the center.

This text is in the center.

## Horizontal Lines

Horizontal lines are used to visually break up sections of a document. The **<hr>** tag creates a line from the current position in the document to the right margin and breaks the line accordingly. For example you may want to give a line between two paragraphs as in the given example below:

<!DOCTYPE html>

<html>

<head>

<title>Horizontal Line Example</title>

</head>

<body>

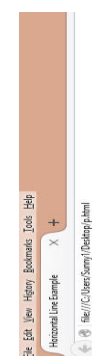
<p>This is paragraph one and should be on top</p>

<hr>

<p>This is paragraph two and should be at bottom</p>

</body>

</html>



This is paragraph one and should be on top

This is paragraph two and should be at bottom

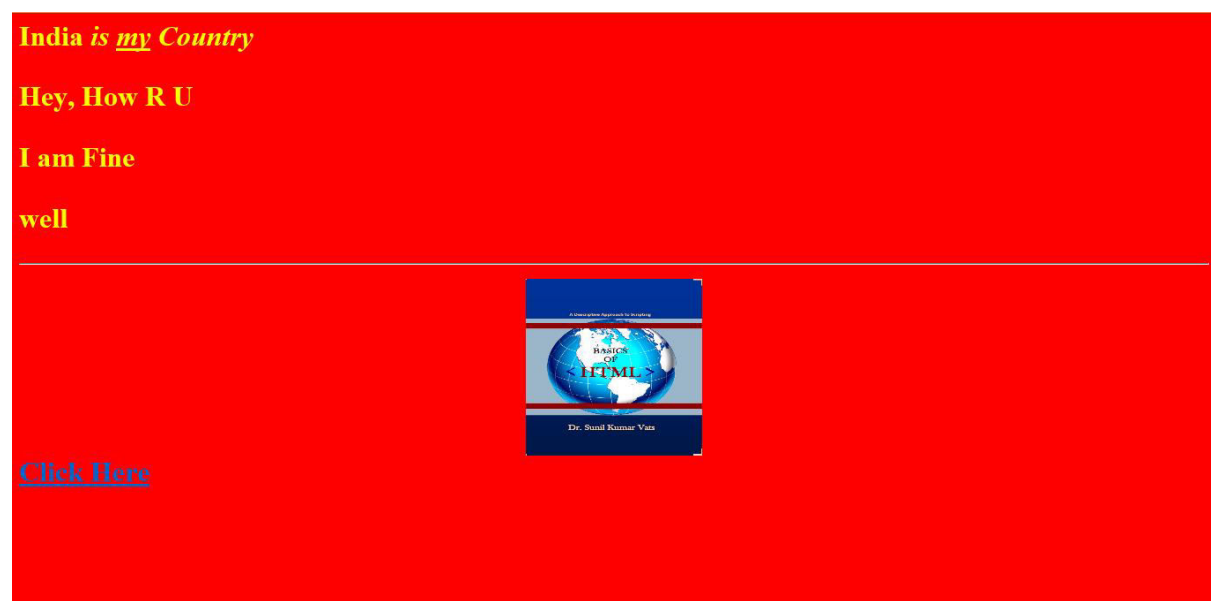
Again **<hr>** tag is an example of the **empty** element, where you do not need opening and closing tags, as there is nothing to go in between them. **<hr>** tag insert the horizontal line in the paragraph.

**Note:-** If you are using XHTML then the syntax for **<hr>** is changed and then **<hr >** element has a space between the characters **hr** and the forward slash. If you omit this space, older browsers will have trouble rendering the horizontal line, while if you miss the forward slash character and just use **<hr>** it is not valid in XHTML

## REVISION

```
<head>
<title>Html Tags</title>
</head>
<body bgcolor="red">
<h1> <font color="Yellow">India <i>is <u>my</u><b> Country</b></i>
<p>Hey, How R U</p>
<p>I am Fine</p>
<p>well</p>
<hr><center>
</img><br></center>
<a href="C:\Users\SUNNY\Desktop\s1.jpg">Click Here<a>
</body>
</html>
```

Output:-



## CHAPTER-3

### MARQUEE

The HTML `<marquee>` tag is used for scrolling piece of text or image displayed either horizontally across or vertically down your web site page depending on the settings. It is a non-standard **HTML element** which causes text to scroll up, down, left or right automatically.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML marquee Tag</title>
</head>
<body>
<marquee>This is basic example of marquee</marquee>
<marquee direction="up">The direction of text will be from bottom to top.</marquee>
</body>
</html>
```

#### Specific Attributes

The HTML `<marquee>` tag also supports following additional attributes:

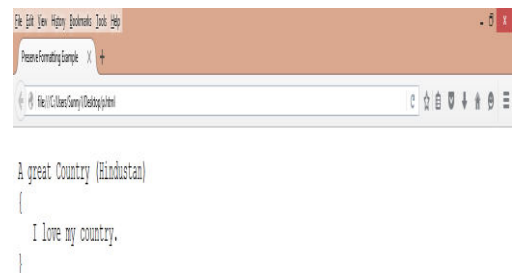
Attribute	Value	Description
behavior	scroll slide alternate	Defines the type of scrolling.
bgcolor	rgb(x,x,x) #xxxxxx colorname	<i>Deprecated</i> -Defines the direction of scrolling the content.
direction	up down left right	Defines the direction of scrolling the content.
height	pixels or %	Defines the height of marquee.
hspace	Pixels	Specifies horizontal space around the marquee.
loop	number	Specifies how many times to loop. The default value is INFINITE, which means that the marquee loops endlessly.
scrolldelay	seconds	Defines how long to delay between each jump.

scrollamount	number	Defines how how far to jump.
width	pixels or %	Defines the width of marquee.
vspace	Pixels	Specifies vertical space around the marquee.

## Preserve Formatting

Sometimes you want your text to follow the exact format of how it is written in the HTML document. In those cases, you can use the preformatted tag `<pre>`. Any text between the opening `<pre>` tag and the closing `</pre>` tag will preserve the formatting of the source document.

```
<!DOCTYPE html>
<html>
<head>
<title>Preserve Formatting Example</title>
</head>
<body>
<pre>
A great Country (Hindustan)
{
  I love my country.
}
</pre>
</body>
</html>
```



You can Try same code without keeping it inside `<pre>...</pre>` tags

## Nonbreaking Spaces

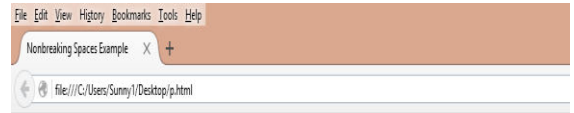
Suppose you want to use the phrase "The Brown Fox Jumps over the Little Lazy dog." Here you would not want a browser to split the "12, Angry" and "Men" across two lines:

In cases where you do not want the client browser to break text, you should use a nonbreaking space entity **&nbsp;** instead of a normal space. For example, when coding the "12 Angry Men" in a paragraph, you should use something similar to the following code:

```

<!DOCTYPE html>
<html>
<head>
<title>Nonbreaking Spaces Example</title>
</head>
<body>
<p>An example of this technique appears in the movie "12&nbsp;Angry&nbsp;Men."</p>
</body>
</html>

```



An example of this technique appears in the movie "12 Angry Men."

An **HTML element** is defined by a starting tag. If the element contains other content, it ends with a closing tag, where the element name is preceded by a forward slash as shown below with few tags:

Start Tag	Content	End Tag
<p>	This is paragraph content.	</p>
<h1>	This is heading content.	</h1>
<div>	This is division content.	</div>

So here <p>....</p> is an HTML element, <h1>...</h1> is another HTML element. There are some HTML elements which don't need to be closed, such as <img.../>, <hr /> and <br /> elements. These are known as **void elements**.

HTML documents consist of a tree of these elements and they specify how HTML documents should be built, and what kind of content should be placed in what part of an HTML document. An HTML element is defined by a *starting tag*. If the element contains other content, it ends with a *closing tag*.

For example <p> is starting tag of a paragraph and </p> is closing tag of the same paragraph but <p>**This is paragraph**</p> is a paragraph element.

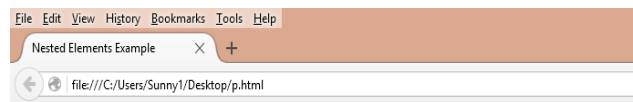
### Nested HTML Elements

It is very much allowed to keep one HTML element inside another HTML element:

```

<!DOCTYPE html>
<html>
<head>
<title>Nested Elements Example</title>

```



**This is *italic* heading**

This is underlined paragraph

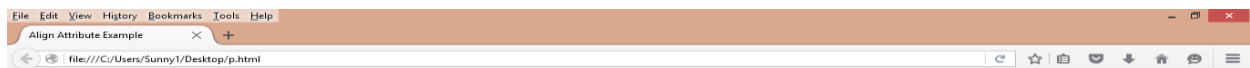
```
</head>
<body>
<h1>This is <i>italic</i> heading</h1>
<p>This is <u>underlined</u> paragraph</p>
</body>
</html>
```

We have seen few HTML tags and their usage like heading tags <h1>, <h2>, paragraph tag <p> and other tags. We used them so far in their simplest form, but most of the HTML tags can also have attributes, which are extra bits of information. An attribute is used to define the characteristics of an HTML element and is placed inside the element's opening tag. All attributes are made up of two parts: a **name** and a **value**:

- ✓ The **name** is the property you want to set. For example, the paragraph <p> element in the example carries an attribute whose name is **align**, which you can use to indicate the alignment of paragraph on the page.
- ✓ The **value** is what you want the value of the property to be set and always put within quotations. The below example shows three possible values of align attribute: **left**, **center** and **right**.

Attribute names and attribute values are case-insensitive. However, the World Wide Web Consortium (W3C) recommends lowercase attributes/attribute values in their HTML 4 recommendation.

```
<!DOCTYPE html>
<html>
<head>
<title>Align Attribute Example</title>
</head>
<body>
<p align="left">This is left aligned</p>
<p align="center">This is center aligned</p>
<p align="right">This is right aligned</p>
</body>
</html>
```



This is left aligned

This is center aligned

This is right aligned

## Core Attributes

The four core attributes that can be used on the majority of HTML elements (although not all) are:

- ✓ id
- ✓ title
- ✓ class
- ✓ style

### The Id Attribute

The **id** attribute of an HTML tag can be used to uniquely identify any element within an HTML page. There are two primary reasons that you might want to use an id attribute on an element:

- If an element carries an id attribute as a unique identifier it is possible to identify just that element and its content.
- If you have two elements of the same name within a Web page (or style sheet), you can use the id attribute to distinguish between elements that have the same name.

We will discuss style sheet in separate. For now, let's use the id attribute to distinguish between two paragraph elements as shown below.

```
<p id="html">This para explains what is HTML</p>
```

```
<p id="css">This para explains what is Cascading Style Sheet</p>
```

### Title Attribute

The **title** attribute gives a suggested title for the element. They syntax for the **title** attribute is similar as explained for **id** attribute. The behavior of this attribute will depend upon the element that carries it, although it is often displayed as a tooltip when cursor comes over the element or while the element is loading.

Now try to bring your cursor over "Titled Heading Tag Example" and you will see that whatever title you used in your code is coming out as a tooltip of the cursor.

### The Class Attribute

The **class** attribute is used to associate an element with a style sheet, and specifies the class of element. You will learn more about the use of the class attribute when you will learn cascading Style Sheet (CSS). So for now you can avoid it.

The value of the attribute may also be a space-separated list of class names. For example:

```
class="className1 className2 className3"
```

## Style Attribute

The style attribute allows you to specify Cascading Style Sheet (CSS) rules within the element.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>The style Attribute</title>
```

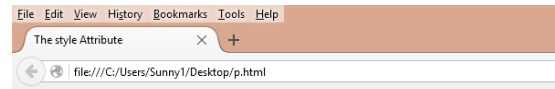
```
</head>
```

```
<body>
```

```
<p style="font-family:arial; color:#FF0000;">Some text...</p>
```

```
</body>
```

```
</html>
```



Some text...

At this point of time, we are not learning CSS, so just let's proceed without bothering much about CSS. Here you need to understand what HTML attributes are and how they can be used while formatting content.

## Internationalization Attributes

There are three internationalization attributes, which are available for most (although not all) XHTML elements.

- o dir
- o lang
- o xml:lang

## The Dir Attribute

The **dir** attribute allows you to indicate to the browser the direction in which the text should flow.

The dir attribute can take one of two values, as you can see in the table that follows:

Value	Meaning
ltr	Left to right (the default value)
rtl	Right to left (for languages such as Hebrew or Arabic that are read right to left)

```
<!DOCTYPE html>
```

```
<html dir="rtl">
```

```
<head>
```

```
<title>Display Directions</title>
```

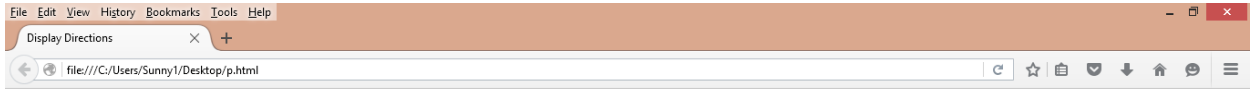
</head>

<body>

This is how IE 5 renders right-to-left directed text.

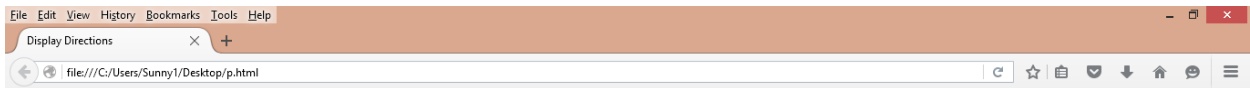
</body>

</html>



.This is how IE 5 renders right-to-left directed text

Output for <ltr> tag is follow:-



This is how IE 5 renders right-to-left directed text.

When *dir* attribute is used within the <html> tag, it determines how text will be presented within the entire document. When used within another tag, it controls the text's direction for just the content of that tag.

### The Lang Attribute

The **lang** attribute allows you to indicate the main language used in a document, but this attribute was kept in HTML only for backwards compatibility with earlier versions of HTML. This attribute has been replaced by the **xml:lang** attribute in new XHTML documents. The values of the *lang* attribute are ISO-639 standard two-character language codes. List are as follow:-

Language	ISO Code	Windows Name	Win Code
Abkhazian	Ab		
Afar	Aa		
Afrikaans	Af	LANG_AFRIKAANS	0x36

Albanian	Sq	LANG_ALBANIAN	0x1c
Amharic	Am	(no constant defined)	0x5e
Arabic	Ar	LANG_ARABIC	0x01
Armenian	Hy	LANG_ARMENIAN	0x2b
Assamese	As	LANG_ASSAMESE	0x4d
Aymara	Ay		

Azerbaijani	Az	LANG_AZERI	0x2c
Bashkir	Ba		
Basque	Eu	LANG_BASQUE	0x2d
Bengali (Bangla)	Bn	LANG_BENGALI	0x45
Bhutani	Dz		
Bihari	Bh		
Bislama	Bi		
Breton	Br		
Bulgarian	Bg	LANG_BULGARIAN	0x02
Burmese	my	(no constant defined)	0x55
Byelorussian (Belarussian)	be	LANG_BELARUSIAN	0x23
Cambodian	km	(no constant defined)	0x53
Catalan	ca	LANG_CATALAN	0x03
Cherokee		(no constant defined)	0x5c
Chewa			
Chinese (Simplified)	zh	LANG_CHINESE (SUBLANG_CHINESE_SIMPLIFIED)	0x04 (0x0804)
Chinese (Traditional)	zh	LANG_CHINESE (SUBLANG_CHINESE_TRADITIONAL)	0x04 (0x0404)

Corsican	co		
Croatian	hr	LANG_CROATIAN	0x1a
Czech	cs	LANG_CZECH	0x05
Danish	da	LANG_DANISH	0x06
Divehi		LANG_DIVEHI	0x65
Dutch	nl	LANG_DUTCH	0x13
Edo		(no constant defined)	0x66
English	En	LANG_ENGLISH	0x09
Esperanto	Eo		
Estonian	Et	LANG_ESTONIAN	0x25
Faeroese	Fo	LANG_FAEROESE	0x38
Farsi	Fa	LANG_FARSI	0x29
Fiji	Fj		
Finnish	Fi	LANG_FINNISH	0x0b
Flemish		LANG_DUTCH (SUBLANG_DUTCH_BELGIAN)	0x13 (0x0813)
French	Fr	LANG_FRENCH	0x0c
Frisian	Fy	(no constant defined)	0x62
Fulfulde		(no constant defined)	0x67
Galician	Gl	LANG_GALICIAN	0x56
Gaelic (Scottish)	Gd	(no constant defined)	0x3c (0x043c)
Gaelic (Manx)	Gv		

Georgian	Ka	LANG_GEORGIAN	0x37
German	De	LANG_GERMAN	0x07
Greek	El	LANG_GREEK	0x08
Greenlandic	Kl		
Guarani	Gn	(no constant defined)	0x74
Gujarati	Gu	LANG_GUJARATI	0x47
Hausa	ha	(no constant defined)	0x68
Hawaiian		(no constant defined)	0x75
Hebrew	he, iw*	LANG_HEBREW	0x0d
Hindi	hi	LANG_HINDI	0x39
Hungarian	hu	LANG_HUNGARIAN	0x0e
Ibibio		(no constant defined)	0x69
Icelandic	is	LANG_ICELANDIC	0x0f
Igbo		(no constant defined)	0x70
Indonesian	id, in*	LANG_INDONESIAN	0x21
Interlingua	ia		
Interlingue	ie		

Inuktitut	iu	(no constant defined)	0x5d
Inupiak	ik		
Irish	ga	(no constant defined)	0x3c (0x083c)
Italian	it	LANG_ITALIAN	0x10
Japanese	ja	LANG_JAPANESE	0x11
Javanese	jav		
Kannada	Kn	LANG_KANNADA	0x4b
Kanuri		(no constant defined)	0x71
Kashmiri	Ks	LANG_KASHMIRI	0x60
Kazakh	Kk	LANG_KAZAKH	0x3f
Kinyarwanda (Ruanda)	Rw		
Kirghiz	Ky	LANG_KYRGYZ	0x40
Kirundi (Rundi)	Rn		
Konkani		LANG_KONKANI	0x57
Korean	Ko	LANG_KOREAN	0x12
Kurdish	Ku		
Laothian	Lo	(no constant defined)	0x54
Latin	La	(no constant defined)	0x76
Latvian (Lettish)	Lv	LANG_LATVIAN	0x26

Limburgish (Limburger)	Li		
Lingala	Ln		
Lithuanian	Lt	LANG_LITHUANIAN	0x27
Macedonian	mk	LANG_MACEDONIAN	0x2f
Malagasy	mg		
Malay	ms	LANG_MALAY	0x3e
Malayalam	ml	LANG_MALAYALAM	0x4c
		LANG_MANIPURI	0x58
Maltese	mt	(no constant defined)	0x3a
Maori	mi		
Marathi	mr	LANG_MARATHI	0x4e
Moldavian	mo		
Mongolian	mn	LANG_MONGOLIAN	0x50
Nauru	na		
Nepali	ne	LANG_NEPALI	0x61
Norwegian	no	LANG_NORWEGIAN	0x14
Occitan	oc		
Oriya	or	LANG_ORIYA	0x48

Oromo (Afan, Galla)	om	(no constant defined)	0x72
Papiamentu		(no constant defined)	0x79
Pashto (Pushto)	Ps	(no constant defined)	0x63
Polish	Pl	LANG_POLISH	0x15
Portuguese	Pt	LANG_PORTUGUESE	0x16
Punjabi	Pa	LANG_PUNJABI	0x46
Quechua	Qu		
Rhaeto-Romance	Rm	(no constant defined)	0x17
Romanian	Ro	LANG_ROMANIAN	0x18
Russian	Ru	LANG_RUSSIAN	0x19
Sami (Lappish)		(no constant defined)	0x3b
Samoan	Sm		
Sangro	Sg		
Sanskrit	Sa	LANG_SANSKRIT	0x4f
Serbian	Sr	LANG_SERBIAN (SUBLANG_SERBIAN_LATIN or SUBLANG_SERBIAN_CYRILLIC)	0x1a (0x081a or 0x0c1a)

Serbo-Croatian	Sh		
Sesotho	St		
Setswana	tn		
Shona	Sn		
Sindhi	Sd	LANG_SINDHI	0x59
Sinhalese	Si	(no constant defined)	0x5b
Siswati	Ss		
Slovak	Sk	LANG_SLOVAK	0x1b
Slovenian	Sl	LANG_SLOVENIA N	0x24
Somali	So	(no constant defined)	0x77
Spanish	Es	LANG_SPANISH	0x0a
Sundanese	Su		
Swahili (Kiswahili)	Sw	LANG_SWAHILI	0x41
Swedish	Sv	LANG_SWEDISH	0x1d
Syriac		LANG_SYRIAC	0x5a
Tagalog	Tl	(no constant defined)	0x64
Tajik	Tg	(no constant defined)	0x28
Tamazight		(no constant defined)	0x5f
Tamil	Ta	LANG_TAMIL	0x49

Tatar	Tt	LANG_TATAR	0x44
Telugu	Te	LANG_TELUGU	0x4a
Thai	th	LANG_THAI	0x1e
Tibetan	bo	(no constant defined)	0x51
Tigrinya	ti	(no constant defined)	0x73
Tonga	to		
Tsonga	ts	(no constant defined)	0x31
Turkish	tr	LANG_TURKISH	0x1f
Turkmen	tk	(no constant defined)	0x42
Twi	Tw		
Uighur	Ug		
Ukrainian	Uk	LANG_UKRAINIA N	0x22
Urdu	Ur	LANG_URDU	0x20
Uzbek	Uz	LANG_UZBEK	0x43
Venda		(no constant defined)	0x33
Vietnamese	Vi	LANG_VIETNAM ESE	0x2a
Volap?k	Vo		
Welsh	Cy	(no constant defined)	0x52
Wolof	W o		
Xhosa	Xh	(no constant defined)	0x34

Yi		(no constant defined)	0x78
Yiddish	yi, ji*	(no constant defined)	0x3d

Yoruba	Yo	(no constant defined)	0x6a
Zulu	Zu	(no constant defined)	0x35

### Language Codes: ISO 639, Macintosh

Language	ISO Code	Mac Name	Mac Code
Abkhazian	Ab		
Afar	Aa		
Afrikaans	Af	langAfricaans	141
Albanian	Sq	langAlbanian	36
Amharic	am	langAmharic	85
Arabic	Ar	langArabic	12
Armenian	Hy	langArmenian	51
Assamese	As	langAssamese	68
Aymara	Ay	langAymara	134
Azerbaijani	Az	langAzerbaijani(Cyrillic), langAzerbaijanAr(Arabic)	49(C), 50(A)
Bashkir	ba		
Basque	Eu	langBasque	129
Bengali (Bangla)	bn	langBengali	67

Bhutani	Dz	langDzongkha	137
Bihari	bh		
Bislama	Bi		
Breton	Br	langBreton	142
Bulgarian	bg	langBulgarian	44
Burmese	my	langBurmese	77
Byelorussian (Belarusian)	be	langByelorussian	46
Cambodian	km	langKhmer	78
Catalan	Ca	langCatalan	130
Cherokee			
Chewa		langChewa	92
Chinese (Simplified)	zh	langSimpChinese	33
Chinese (Traditional)	zh	langTradChinese	19
Corsican	co		
Croatian	hr	langCroatian	18

Czech	cs	langCzech	38
Danish	da	langDanish	7
Divehi			
Dutch	nl	langDutch	4
Edo			
English	en	langEnglish	0
Esperanto	eo	langEsperanto	94
Estonian	et	langEstonian	27
Faeroese	fo	langFaeroese	30
Farsi	fa	langFarsi, langPersian	31
Fiji	fj		
Finnish	fi	langFinnish	13
Flemish		langFlemish	34
French	fr	langFrench	1
Frisian	fy		
Fulfulde			
Galician	gl	langGalician	140
Gaelic (Scottish)	gd	langScottishGaelic	144
Gaelic (Manx)	gv	langManxGaelic	145
Georgian	ka	langGeorgian	52
German	de	langGerman	2
Greek	El	langGreek (monotonic), langGreekPoly (polytonic)	14(m), 148(p)

Greenlandic	Kl		
Guarani	Gn	langGuarani	133
Gujarati	Gu	langGujarati	69
Hausa	ha		
Hawaiian			
Hebrew	he, iw*	langHebrew	10
Hindi	Hi	langHindi	21
Hungarian	hu	langHungarian	26
Ibibio			
Icelandic	Is	langIcelandic	15
Igbo			
Indonesian	id, in*	langIndonesian	81
Interlingua	Ia		
Interlingue	Ie		
Inuktitut	Iu	langInuktitut	143
Inupiak	Ik		
Irish	Ga	langIrishGaelic (normal), langIrishGaelicScr (dots above)	35, 146
Italian	It	langItalian	3
Japanese	ja	langJapanese	11
Javanese	lv	langJavaneseRom	138
Kannada	kn	langKannada	73

Kanuri			
Kashmiri	ks	langKashmiri	61
Kazakh	kk	langKazakh	48
Kinyarwanda (Ruanda)	rw	langKinyarwanda (langRuanda)	90
Kirghiz	ky	langKirghiz	54
Kirundi (Rundi)	rn	langRundi	91
Konkani			
Korean	ko	langKorean	23
Kurdish	ku	langKurdish	60
Laotian	lo	langLao	79
Latin	la	langLatin	131
Latvian (Lettish)	lv	langLatvian	28
Limburgish (Limburger)	li		
Lingala	ln		
Lithuanian	lt	langLithuanian	24
Macedonian	mk	langMacedonian	43
Malagasy	mg	langMalagasy	93
Malay	ms	langMalayRoman(Latin), langMalayArabic(Arabic)	83(L), 84(A)

Malayalam	ml	langMalayalam	72
Maltese	mt	langMaltese	16
Maori	mi		
Marathi	mr	langMarathi	66
Moldavian	mo	langMoldavian	53
Mongolian	mn	langMongolian(Mongolian), langMongolianCyr(Cyrillic)	57(M), 58(C)
Nauru	na		
Nepali	ne	langNepali	64
Norwegian	no	langNorwegian	9
Occitan	Oc		
Oriya	Or	langOriya	71
Oromo (Afan, Galla)	om	langOromo (langGalla)	87
Papiamentu			
Pashto (Pushto)	Ps	langPashto	59
Polish	Pl	langPolish	25
Portuguese	Pt	langPortuguese	8
Punjabi	pa	langPunjabi	70
Quechua	qu	langQuechua	132

Rhaeto-Romance	rm		
Romanian	ro	langRomanian	37
Russian	ru	langRussian	32
Sami (Lappish)		langSami (langLappish)	29
Samoan	sm		
Sangro	sg		
Sanskrit	sa	langSanskrit	65
Serbian	sr	langSerbian	42
Serbo-Croatian	sh		
Sesotho	st		
Setswana	tn		
Shona	sn		
Sindhi	sd	langSindhi	62
Sinhalese	si	langSinhalese	76
Siswati	ss		
Slovak	sk	langSlovak	39
Slovenian	sl	langSlovenian	40
Somali	so	langSomali	88
Spanish	es	langSpanish	6
Sundanese	su	langSundaneseRom	139
Swahili (Kiswahili)	sw	langSwahili	89
Swedish	sv	langSwedish	5

Syriac			
Tagalog	Tl	langTagalog	82
Tajik	Tg	langTajiki	55
Tamazight			
Tamil	Ta	langTamil	74
Tatar	Tt	langTatar	135
Telugu	Te	langTelugu	75
Thai	Th	langThai	22
Tibetan	bo	langTibetan	63
Tigrinya	Ti	langTigrinya	86
Tonga	To	langTongan	147
Tsonga	Ts		
Turkish	Tr	langTurkish	17
Turkmen	Tk	langTurkmen	56
Twi	Tw		
Uighur	ug	langUighur	136
Ukrainian	uk	langUkrainian	45
Urdu	ur	langUrdu	20
Uzbek	uz	langUzbek	47
Venda			
Vietnamese	vi	langVietnamese	80
Volapük	vo		
Welsh	cy	langWelsh	128
Wolof	wo		
Xhosa	xh		
Yi			

Yiddish	yi, ji*	langYiddish	41
---------	------------	-------------	----

Yoruba	yo		
Zulu	zu		

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>English Language Page</title>
</head>
<body>
This page is using English Language
</body>
</html>
```

### Lang Attribute

The *xml:lang* attribute is the XHTML replacement for the *lang* attribute. The value of the *xml:lang* attribute should be an ISO-639 country code,

### Generic Attributes

Here's a table of some other attributes that are readily usable with many of the HTML tags.

Attribute	Options	Function
Align	right, left, center	Horizontally aligns tags
Valign	top, middle, bottom	Vertically aligns tags within an HTML element.
Bgcolor	numeric, hexadecimal, RGB values	Places a background color behind an element
background	URL	Places a background image behind an element
Id	User Defined	Names an element for use with Cascading Style Sheets.
Class	User Defined	Classifies an element for use with Cascading Style Sheets.
Width	Numeric Value	Specifies the width of tables, images, or table cells.
Height	Numeric Value	Specifies the height of tables, images, or table cells.
Title	User Defined	"Pop-up" title of the elements.

We will see related examples as we will proceed to study other HTML tags. A complete list of HTML Tags and related attributes are as:-

<b>Ag</b>	<b>Description</b>	<b>Browser</b>
<u>&lt;!--...--&gt;</u>	Specifies a comment	
<u>&lt;!DOCTYPE&gt;</u>	Specifies the document type	
<u>&lt;a&gt;</u>	Specifies an anchor	
<u>&lt;abbr&gt;</u>	Specifies an abbreviation	
<u>&lt;acronym&gt;</u>	Specifies an acronym	
<u>&lt;address&gt;</u>	Specifies an address element	
<u>&lt;applet&gt;</u>	Deprecated. Specifies an applet	
<u>&lt;area&gt;</u>	Specifies an area inside an image map	
<u>&lt;b&gt;</u>	Specifies bold text	
<u>&lt;base&gt;</u>	Specifies a base URL for all the links in a page	
<u>&lt;basefont&gt;</u>	Deprecated. Specifies a base font	
<u>&lt;bdo&gt;</u>	Specifies the direction of text display	
<u>&lt;bgsound&gt;</u>	Specifies the background music	IE
<u>&lt;big&gt;</u>	Specifies big text	
<u>&lt;blink&gt;</u>	Specifies a text which blinks	NS
<u>&lt;blockquote&gt;</u>	Specifies a long quotation	
<u>&lt;body&gt;</u>	Specifies the body element	
<u>&lt;br&gt;</u>	Inserts a single line break	
<u>&lt;button&gt;</u>	Specifies a push button	
<u>&lt;caption&gt;</u>	Specifies a table caption	
<u>&lt;center&gt;</u>	Deprecated. Specifies centered text	

<u>&lt;cite&gt;</u>	Specifies a citation	
<u>&lt;code&gt;</u>	Specifies computer code text	
<u>&lt;col&gt;</u>	Specifies attributes for table columns	
<u>&lt;colgroup&gt;</u>	Specifies groups of table columns	
<u>&lt;comment&gt;</u>	Puts a comment in the document	IE
<u>&lt;dd&gt;</u>	Specifies a definition description	
<u>&lt;del&gt;</u>	Specifies deleted text	
<u>&lt;dfn&gt;</u>	Specifies a definition term	
<u>&lt;dir&gt;</u>	Deprecated. Specifies a directory list	
<u>&lt;div&gt;</u>	Specifies a section in a document	
<u>&lt;dl&gt;</u>	Specifies a definition list	
<u>&lt;dt&gt;</u>	Specifies a definition term	
<u>&lt;em&gt;</u>	Specifies emphasized text	
<u>&lt;embed&gt;</u>	Deprecated. Embeds an application in a document	IE & NS
<u>&lt;fieldset&gt;</u>	Specifies a fieldset	
<u>&lt;font&gt;</u>	Deprecated. Specifies text font, size, and color	
<u>&lt;form&gt;</u>	Specifies a form	
<u>&lt;frame&gt;</u>	Specifies a sub window (a frame)	
<u>&lt;frameset&gt;</u>	Specifies a set of frames	
<u>&lt;h1&gt; to &lt;h6&gt;</u>	Specifies header 1 to header 6	
<u>&lt;head&gt;</u>	Specifies information about the document	
<u>&lt;hr&gt;</u>	Specifies a horizontal rule	

<u>&lt;html&gt;</u>	Specifies an html document	
<u>&lt;i&gt;</u>	Specifies italic text	
<u>&lt;iframe&gt;</u>	Specifies an inline sub window (frame)	
<u>&lt;ilayer&gt;</u>	Specifies an inline layer	NS
<u>&lt;img&gt;</u>	Specifies an image	
<u>&lt;input&gt;</u>	Specifies an input field	
<u>&lt;ins&gt;</u>	Specifies inserted text	
<u>&lt;isindex&gt;</u>	Deprecated. Specifies a single-line input field	
<u>&lt;kbd&gt;</u>	Specifies keyboard text	
<u>&lt;keygen&gt;</u>	Generate key information in a form	
<u>&lt;label&gt;</u>	Specifies a label for a form control	
<u>&lt;layer&gt;</u>	Specifies a layer	NS
<u>&lt;legend&gt;</u>	Specifies a title in a fieldset	
<u>&lt;li&gt;</u>	Specifies a list item	
<u>&lt;link&gt;</u>	Specifies a resource reference	
<u>&lt;map&gt;</u>	Specifies an image map	
<u>&lt;marquee&gt;</u>	Create a scrolling-text marquee	IE
<u>&lt;menu&gt;</u>	Deprecated. Specifies a menu list	
<u>&lt;meta&gt;</u>	Specifies meta information	
<u>&lt;multicol&gt;</u>	Specifies a multicolumn text flow	NS
<u>&lt;nobr&gt;</u>	No breaks allowed in the enclosed text	
<u>&lt;noembed&gt;</u>	Specifies content to be presented by browsers that do not support the <embed> tag	NS

<u>&lt;noframes&gt;</u>	Specifies a noframe section	
<u>&lt;noscript&gt;</u>	Specifies a noscript section	
<u>&lt;object&gt;</u>	Specifies an embedded object	
<u>&lt;ol&gt;</u>	Specifies an ordered list	
<u>&lt;optgroup&gt;</u>	Specifies an option group	
<u>&lt;option&gt;</u>	Specifies an option in a drop-down list	
<u>&lt;p&gt;</u>	Specifies a paragraph	
<u>&lt;param&gt;</u>	Specifies a parameter for an object	
<u>&lt;plaintext&gt;</u>	Deprecated. Render the remainder of the document as preformatted plain text	
<u>&lt;pre&gt;</u>	Specifies preformatted text	
<u>&lt;q&gt;</u>	Specifies a short quotation	
<u>&lt;s&gt;</u>	Deprecated. Specifies strikethrough text	
<u>&lt;samp&gt;</u>	Specifies sample computer code	
<u>&lt;script&gt;</u>	Specifies a script	
<u>&lt;select&gt;</u>	Specifies a selectable list	
<u>&lt;spacer&gt;</u>	Specifies a white space	NS
<u>&lt;small&gt;</u>	Specifies small text	
<u>&lt;span&gt;</u>	Specifies a section in a document	
<u>&lt;strike&gt;</u>	Deprecated. Specifies strikethrough text	
<u>&lt;strong&gt;</u>	Specifies strong text	
<u>&lt;style&gt;</u>	Specifies a style definition	
<u>&lt;sub&gt;</u>	Specifies subscripted text	

<u>&lt;sup&gt;</u>	Specifies superscripted text	
<u>&lt;table&gt;</u>	Specifies a table	
<u>&lt;tbody&gt;</u>	Specifies a table body	
<u>&lt;td&gt;</u>	Specifies a table cell	
<u>&lt;textarea&gt;</u>	Specifies a text area	
<u>&lt;tfoot&gt;</u>	Specifies a table footer	
<u>&lt;th&gt;</u>	Specifies a table header	
<u>&lt;thead&gt;</u>	Specifies a table header	
<u>&lt;title&gt;</u>	Specifies the document title	
<u>&lt;tr&gt;</u>	Specifies a table row	
<u>&lt;tt&gt;</u>	Specifies teletype text	
<u>&lt;u&gt;</u>	Deprecated. Specifies underlined text	
<u>&lt;ul&gt;</u>	Specifies an unordered list	
<u>&lt;var&gt;</u>	Specifies a variable	
<u>&lt;wbr&gt;</u>	Indicate a potential word break point within a <nobr> section	IE & NS
<u>&lt;xmp&gt;</u>	Deprecated. Specifies preformatted text	

A complete list of deprecated HTML tags and attributes are given here. All the tags have been ordered alphabetically along with their equivalent tag or alternate CSS option.

<b>Tag</b>	<b>Description</b>	<b>Alternate</b>
<u>&lt;applet&gt;</u>	Deprecated. Specifies an applet	<object>
<u>&lt;basefont&gt;</u>	Deprecated. Specifies a base font	
<u>&lt;center&gt;</u>	Deprecated. Specifies centered text	text-align
<u>&lt;dir&gt;</u>	Deprecated. Specifies a directory list	
<u>&lt;embed&gt;</u>	Deprecated. Embeds an application in a document	<object>

<code>&lt;font&gt;</code>	Deprecated. Specifies text font, size, and color	font-family, font-size
<code>&lt;isindex&gt;</code>	Deprecated. Specifies a single-line input field	
<code>&lt;listing&gt;</code>	Deprecated. Specifies listing of items	<code>&lt;pre&gt;</code>
<code>&lt;menu&gt;</code>	Deprecated. Specifies a menu list	
<code>&lt;plaintext&gt;</code>	Deprecated. Specifies plaintext	<code>&lt;pre&gt;</code>
<code>&lt;s&gt;</code>	Deprecated. Specifies strikethrough text	text-decoration
<code>&lt;strike&gt;</code>	Deprecated. Specifies strikethrough text	text-decoration
<code>&lt;u&gt;</code>	Deprecated. Specifies underlined text	text-decoration
<code>&lt;xmp&gt;</code>	Deprecated. Specifies preformatted text	<code>&lt;pre&gt;</code>

## CHAPTER-4

### HTML DECPRECATED ATTRIBUTES

Following is the list of deprecated HTML attributes and alternative CSS options available.

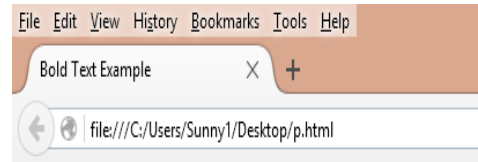
<b>Attribute</b>	<b>Description</b>	<b>Alternate</b>
Align	Specifies positioning of an element	text-align, float & vertical-align
Alink	Specifies the color of an active link or selected link	active
background	Specifies background image	background-image
Bgcolor	Specifies background color	background-color
Border	Specifies a border width of any element	border-width
Clear	Indicates how the browser should display the line after the   element	clear
Height	Specifies height of body and other elements	height
Hspace	Specifies the amount of whitespace or padding that should appear left or right an element	padding
Language	Specifies scripting language being used	type
Link	Specifies the default color of all links in the document	link
Nowrap	Prevents the text from wrapping within that table cell	white-space
Start	Indicates the number at which a browser should start numbering a list	counter-reset
Text	Specifies color of body text	color
Type	Specifies the type of list in <li> tag	list-style-type
Vlink	Specifies the color of visited links	visited
Vspace	Specifies the amount of whitespace or padding that should appear above or below an element	padding
Width	Specifies width of body and other elements	width

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 `<b>...</b>` element, is displayed in bold as shown below:

```
<!DOCTYPE html>
<html>
<head>
<title>Bold Text Example</title>
</head>
<body>
<p>India is <b>great</b> country.</p>
</body>
</html>
```

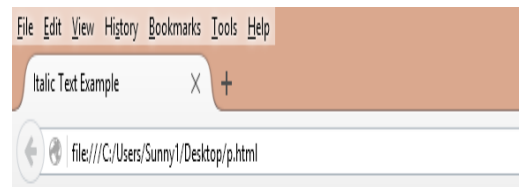


India is **great** country.

### **Italic Text**

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

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



The following word uses a *italicized* typeface.

### **Underlined Text**

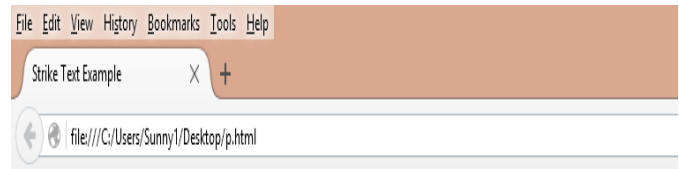
Anything that appears within `<u>...</u>` element, is displayed with underline as shown below:

```
}
```

### **Strike Text**

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

```
<!DOCTYPE html>
<html>
<head>
<title>Strike Text Example</title>
</head>
<body>
<p>The strikethrough tag is used<strike> to
strikethrough the text</strike>
typeface.</p>
</body>
</html>
```



The strikethrough tag is used to strikethrough the text typeface.

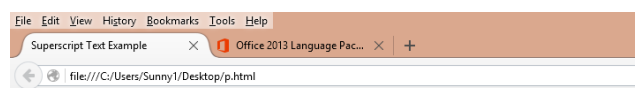
## Monospaced Font

The content of a `<tt>...</tt>` element is written in monospaced (equal width) font. Most of the fonts are known as variable-width fonts because different letters are of different widths (for example, the letter 'm' is wider than the letter 'i'). In a monospaced font, however, each letter has the same width.

## Superscript Text

The content of a `<sup>...</sup>` 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.

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



The following word uses a <sup>superscript</sup> typeface.

## Subscript Text

The content of a `<sub>...</sub>` element is written in subscript; the font size used is the same as the characters surrounding it, but is displayed half a character's height beneath the other characters.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Subscript Text Example</title>
```

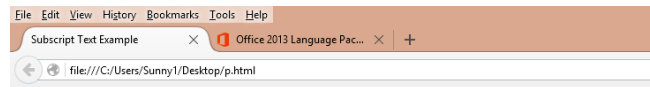
```
</head>
```

```
<body>
```

```
<p>The following word uses a <sub>subscript</sub> typeface.</p>
```

```
</body>
```

```
</html>
```



The following word uses a subscript typeface.

## Inserted Text

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

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Inserted Text Example</title>
```

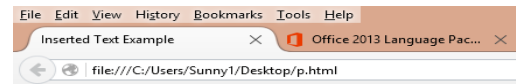
```
</head>
```

```
<body>
```

```
<p>I want to drink <del>Coke</del> <ins>Milk</ins></p>
```

```
</body>
```

```
</html>
```



I want to drink ~~Coke~~ Milk

## 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:

```
<!DOCTYPE html>
```

```
<html>
```

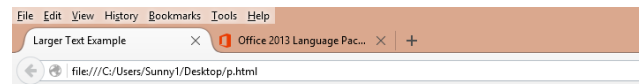
```
<head>
```

```
<title>Larger Text Example</title>
```

```
</head>
```

```
<body>
```

```
<p>The following word uses a <big>big</big> typeface.</p>
```



The following word uses a big typeface.

</body>

</html>

## Smaller Text

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

<!DOCTYPE html>

<html>

<head>

<title>Smaller Text Example</title>

</head>

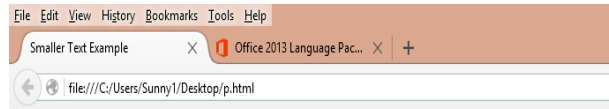
<body>

<p>The following word uses a

<small>small</small> typeface.</p>

</body>

</html>



The following word uses a small typeface.

## Grouping Content

The `<div>` and `<span>` 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.

<!DOCTYPE html>

<html>

<head>

<title>Div Tag Example</title>

</head>

<body>

<div id="menu" align="middle" >

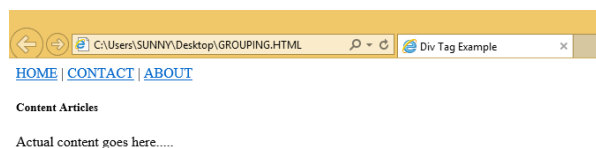
<a href="http://www.jnvmahendragarh.gov.in">HOME</a> |

<a href="http://www.jnvmahendragarh.gov.in/innerpage.php?pagename=contactus">

CONTACT</a> |

<a href="/about/index.htm">ABOUT</a>

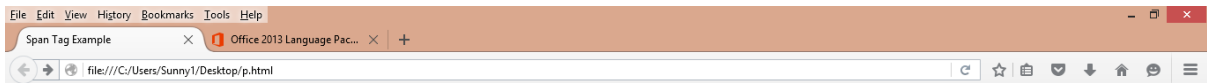
</div>



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

The `<span>` element, on the other hand, can be used to group inline elements only. So, if you have a part of a sentence or paragraph which you want to group together, you could use the `<span>` element as follows

```
<!DOCTYPE html>
<html>
<head>
<title>Span Tag Example</title>
</head>
<body>
<p>This is the example of <span style="color:green">span tag</span> and the <span
style="color:red">div tag</span> alongwith CSS</p>
</body>
</html>
```



This is the example of `span tag` and the `div tag` alongwith CSS

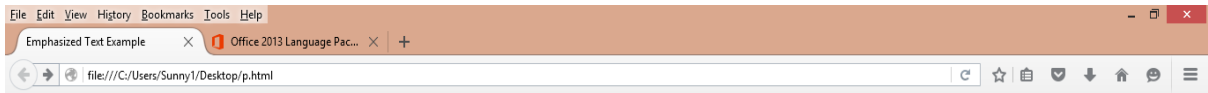
These tags are commonly used with CSS to allow you to attach a style to a section of a page. The phrase tags have been designed for specific purposes, though they are displayed in a similar way as other basic tags like `<b>`, `<i>`, `<pre>`, and `<tt>`, you have seen in previous chapter. This chapter will take you through all the important phrase tags, so let's start seeing them one by one.

## Emphasized Text

Anything that appears within `<em>...</em>` element is displayed as emphasized text.

```
<!DOCTYPE html>
<html>
<head>
```

```
<title>Emphasized Text Example</title>
</head>
<body>
<p>The following word uses a <em>emphasized</em> typeface.</p>
</body>
</html>
```

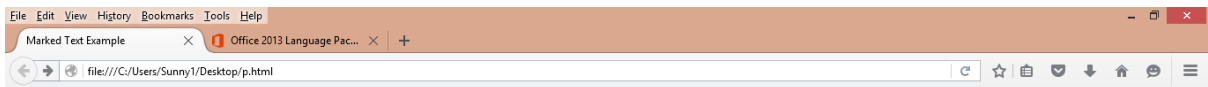


The following word uses a *emphasized* typeface.

## Marked Text

Anything that appears with-in **<mark>...</mark>** element, is displayed as marked with yellow ink.

```
<!DOCTYPE html>
<html>
<head>
<title>Marked Text Example</title>
</head>
<body>
<p>The following word has been <mark>marked</mark> with yellow</p>
</body>
</html>
```



The following word has been **marked** with yellow

## Strong Text

Anything that appears within **<strong>...</strong>** element is displayed as important text.

```
<!DOCTYPE html>
<html>
<head>
<title>Strong Text Example</title>
</head>
```

```
<body>
<p>The following word uses a <strong>strong</strong> typeface.</p>
</body>
</html>
```

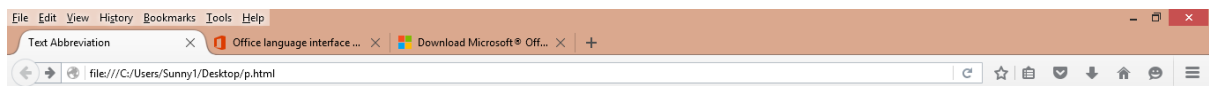
This will produce following result:

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

## Text Abbreviation

You can abbreviate a text by putting it inside opening **<abbr>** and closing **</abbr>** tags. If present, the title attribute must contain this full description and nothing else.

```
<!DOCTYPE html>
<html>
<head>
<title>Text Abbreviation</title>
</head>
<body>
<p>My best friend's name is <abbr title="Abhishek">Abhy</abbr>.</p>
</body>
</html>
```



My best friend's name is Abhy.

## Acronym Element

The **<acronym>** element allows you to indicate that the text between **<acronym>** and **</acronym>** tags is an acronym.

At present, the major browsers do not change the appearance of the content of the **<acronym>** element.

```
<!DOCTYPE html>
<html>
<head>
<title>Acronym Example</title>
</head>
<body>
```

```
<p>This chapter covers marking up text in <acronym>XHTML</acronym>.</p>
</body>
</html>
```

This will produce following result:

This chapter covers marking up text in XHTML.

### Text Direction

The **<bdo>...</bdo>** element stands for Bi-Directional Override and it is used to override the current text direction.

```
<!DOCTYPE html>
<html>
<head>
<title>Text Direction Example</title>
</head>
<body>
<p>This text will go left to right.</p>
<p><bdo dir="rtl">This text will go right to left.</bdo></p>
</body>
</html>
```

This will produce following result:

This text will go left to right.

This text will go right to left.

### Special Terms

The **<dfn>...</dfn>** element (or HTML Definition Element) allows you to specify that you are introducing a special term. It's usage is similar to italic words in the midst of a paragraph. Typically, you would use the **<dfn>** element the first time you introduce a key term. Most recent browsers render the content of a **<dfn>** element in an italic font.

```
<!DOCTYPE html>
<html>
<head>
<title>Special Terms Example</title>
</head>
<body>
<p>The following word is a <dfn>special</dfn> term.</p>
```

```
</body>
```

```
</html>
```

This will produce following result:

The following word is a *special* term.

### Quoting Text

When you want to quote a passage from another source, you should put it in between **<blockquote>...</blockquote>** tags.

Text inside a `<blockquote>` element is usually indented from the left and right edges of the surrounding text, and sometimes uses an italicized font.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Blockquote Example</title>
```

```
</head>
```

```
<body>
```

```
<p>The following description of XHTML is taken from the W3C Web site:</p>
```

```
<blockquote>XHTML 1.0 is the W3C's first Recommendation for XHTML, following on  
from earlier work on HTML 4.01, HTML 4.0, HTML 3.2 and HTML 2.0.</blockquote>
```

```
</body>
```

```
</html>
```

This will produce following result:

The following description of XHTML is taken from the W3C Web site:

XHTML 1.0 is the W3C's first Recommendation for XHTML, following on from earlier work on HTML 4.01, HTML 4.0, HTML 3.2 and HTML 2.0.

### Short Quotations

The **<q>...</q>** element is used when you want to add a double quote within a sentence.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Double Quote Example</title>
```

```
</head>
```

```
<body>
```

```
<p>Amit is in Spain, <q>I think I am wrong</q>.</p>
```

```
</body>
```

```
</html>
```

This will produce following result:

Amit is in Spain, I think I am wrong.

## Text Citations

If you are quoting a text, you can indicate the source placing it between an opening `<cite>` tag and closing `</cite>` tag

As you would expect in a print publication, the content of the `<cite>` element is rendered in italicized text by default.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Citations Example</title>
```

```
</head>
```

```
<body>
```

```
<p>This HTML is derived from <cite>W3 Standard for HTML</cite>.</p>
```

```
</body>
```

```
</html>
```

This will produce following result:

This HTML is derived from *W3 Standard for HTML*.

## Computer Code

Any programming code to appear on a Web page should be placed inside `<code>...</code>` tags. Usually the content of the `<code>` element is presented in a monospaced font, just like the code in most programming books.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Computer Code Example</title>
```

```
</head>
```

```
<body>
```

```
<p>Regular text. <code>This is code.</code> Regular text.</p>
```

```
</body>
```

```
</html>
```

This will produce following result:

Regular text. This is code. Regular text.

### **Keyboard Text**

When you are talking about computers, if you want to tell a reader to enter some text, you can use the `<kbd>...</kbd>` element to indicate what should be typed in, as in this example.

```
<!DOCTYPE html>
<html>
<head>
<title>Keyboard Text Example</title>
</head>
<body>
<p>Regular text. <kbd>This is inside kbd element</kbd> Regular text.</p>
</body>
</html>
```

This will produce following result:

Regular text. This is inside kbd element Regular text.

### **Setting Font Face**

You can set font face using *face* attribute but be aware that if the user viewing the page doesn't have the font installed, they will not be able to see it. Instead user will see the default font face applicable to the user's computer.

Example

```
<!DOCTYPE html>
<html>
<head>
<title>Font Face</title>
</head>
<body>
<font face="Times New Roman" size="5">Times New Roman</font><br />
<font face="Verdana" size="5">Verdana</font><br />
<font face="Comic sans MS" size="5">Comic Sans MS</font><br />
<font face="WildWest" size="5">WildWest</font><br />
<font face="Bedrock" size="5">Bedrock</font><br />
</body>
```

</html>

This will produce following result:



### Specify alternate font faces

A visitor will only be able to see your font if they have that font installed on their computer. So, it is possible to specify two or more font face alternatives by listing the font face names, separated by a comma.

```
<font face="arial,helvetica">
```

```
<font face="Lucida Calligraphy,Comic Sans MS,Lucida Console">
```

When your page is loaded, their browser will display the first font face available. If none of the given fonts are installed, then it will display the default font face *Times New Roman*.

**Full List of Fonts is as below:-**

#### Microsoft OS and Browsers Fonts:

Font	Font	Font
Andale Mono	Arial	Arial Bold
Arial Italic	Arial Bold Italic	Arial Black
Comic Sans MS	Comic Sans MS Bold	Courier New
Courier New Bold	Courier New Italic	Courier New Bold Italic
Georgia	Georgia Bold	Georgia Italic
Georgia Bold Italic	Impact	Lucida Console
Lucida Sans Unicode	Marlett	Minion Web
Symbol	Times New Roman	Times New Roman Bold

Times New Roman Italic	Times New Roman Bold Italic	Tahoma
Trebuchet MS	Trebuchet MS Bold	Trebuchet MS Italic
Trebuchet MS Bold Italic	Verdana	Verdana Bold
Verdana Italic	Verdana Bold Italic	Webdings

### Macintosh Systems Fonts:

Following is the list of fonts supported by Macintosh System 7 and higher versions

Font	Font	Font
American Typewriter	Andale Mono	Apple Chancery
Arial	Arial Black	Brush Script
Baskerville	Big Caslon	Comic Sans MS
Copperplate	Courier New	Gill Sans
Futura	Herculanum	Impact
Lucida Grande	Marker Felt	Optima
Trebuchet MS	Verdana	Webdings
Palatino	Symbol	Times
Osaka	Papyrus	Times New Roman
Textile	Zapf Dingbats	Zapfino
Techno	Hoefler Text	Skia
Hoefler Text Ornaments	Capitals	Charcoal
Gadget	Sand	

### Unix Systems Fonts:



Following is the list of fonts supported by most Unix System variants

Font	Font	Font
Charter	Clean	Courier
Fixed	Helvetica	Lucida
Lucida bright	Lucida Typewriter	New Century Schoolbook
Symbol	Terminal	Times
Utopia		

### Setting Font Color

You can set any font color you like using *color* attribute. You can specify the color that you want by either the color name or hexadecimal code for that color. The complete list of font color is shown below:

The following table shows the 16 color names that were introduced in HTML 3.2:






Color Name	Hex Value	Color
aqua	#00ffff	
black	#000000	
blue	#0000ff	
fuchsia	#ff00ff	
green	#008000	
gray	#808080	
lime	#00ff00	

maroon	#800000	
navy	#000080	
olive	#808000	
purple	#800080	
Red	#ff0000	
silver	#c0c0c0	
teal	#008080	
white	#ffffff	
yellow	#ffff00	

























There are other colors which are not part of HTML or XHTML but they are supported by most of the versions of major browsers.

Color Name	Hex Value	Color
aliceblue	#f0f8ff	
antiquewhite	#faebd7	


aliceblue	#f0f8ff	
antiquewhite	#faebd7	


aquamarine	#7fffd4	
azure	#f0ffff	
beige	#f5f5dc	
bisque	#ffe4c4	
blanchedalmond	#ffeacd	
blueviolet	#8a2be2	
brown	#a52a2a	
burlywood	#deb887	
cadetblue	#5f9ea0	
chartreuse	#7fff00	
chocolate	#d2691e	
coral	#ff7f50	
cornflowerblue	#6495ed	
cornsilk	#fff8dc	
crimson	#dc143c	
cyan	#00ffff	
darkblue	#00008b	
darkcyan	#008b8b	
darkgoldenrod	#b8860b	
darkgray	#a9a9a9	
darkgreen	#006400	
darkkhaki	#bdb76b	
darkmagenta	#8b008b	
darkolivegreen	#556b2f	

darkorange	#ff8c00	
darkorchid	#9932cc	
darkred	#8b0000	
darksalmon	#e9967a	
darkseagreen	#8fbc8f	
darkslateblue	#483d8b	
darkslategray	#2f4f4f	
darkturquoise	#00ced1	
darkviolet	#9400d3	
deeppink	#ff1493	
deepskyblue	#00bfff	
dimgray	#696969	
dodgerblue	#1e90ff	
firebrick	#b22222	
floralwhite	#fffaf0	
forestgreen	#228b22	
gainsboro	#dcdcdc	
ghostwhite	#f8f8ff	
gold	#ffd700	
goldenrod	#daa520	
gray	#808080	
greenyellow	#adff2f	
honeydew	#f0ffff	
hotpink	#ff69b4	

indianred	#cd5c5c	
indigo	#4b0082	
ivory	#ffff0	
khaki	#f0e68c	
lavender	#e6e6fa	
lavenderblush	#fff0f5	
lawngreen	#7cfc00	
lemonchiffon	#fffacd	
lightblue	#add8e6	
lightcoral	#f08080	
lightcyan	#e0ffff	
lightgoldenrodyellow	#fafad2	
lightgreen	#90ee90	
lightgrey	#d3d3d3	
lightpink	#ffb6c1	
lightsalmon	#ffa07a	
lightseagreen	#20b2aa	
lightskyblue	#87cefa	
lightslategray	#778899	
lightsteelblue	#b0c4de	
lightyellow	#ffffe0	
limegreen	#32cd32	
linen	#faf0e6	
magenta	#ff00ff	

mediumblue	#0000cd	
mediumorchid	#ba55d3	
mediumpurple	#9370db	
midnightblue	#191970	
mistyrose	#ffe4e1	
moccasin	#ffe4b5	
oldlace	#fdf5e6	
orange	#ffa500	
orchid	#da70d6	
peachpuff	#ffdab9	
peru	#cd853f	
pink	#ffc0cb	
plum	#dda0dd	
purple	#800080	
rosybrown	#bc8f8f	
royalblue	#4169e1	
salmon	#fa8072	
sandybrown	#f4a460	
seagreen	#2e8b57	
sienna	#a0522d	
skyblue	#87ceeb	
slateblue	#6a5acd	
steelblue	#4682b4	
tan	#d2b48c	

thistle	#d8bfd8	
tomato	#ff6347	
violet	#ee82ee	
wheat	#f5deb3	

whitesmoke	#f5f5f5	
yellow	#ffff00	
yellowgreen	#9acd32	

```
<!DOCTYPE html>
<html>
<head>
<title>Setting Font Color</title>
</head>
<body>
<font color="#FF00FF">This text is in pink</font><br />
<font color="red">This text is red</font>
</body>
</html>
```

This will produce following result:

This text is in pink

This text is red

### The <basefont> Element:

The <basefont> element is supposed to set a default font size, color, and typeface for any parts of the document that are not otherwise contained within a <font> tag. You can use the <font> elements to override the <basefont> settings.

The <basefont> tag also takes color, size and face attributes and it will support relative font setting by giving size a value of +1 for a size larger or -2 for two sizes smaller.

```
<!DOCTYPE html>
<html>
<head>
<title>Setting Basefont Color</title>
</head>
<body>
<basefont face="arial, verdana, sans-serif" size="2" color="#ff0000">
<p>This is the page's default font.</p>
```

`<h2>Example of the &lt;basefont&gt; Element</h2>`

`<p><font size="+2" color="darkgray">`

This is darkgray text with two sizes larger

`</font></p>`

`<p><font face="courier" size="-1" color="#000000">`

It is a courier font, a size smaller and black in color.

`</font></p>`

`</body>`

`</html>`

This will produce following result:



This is the page's default font.

## **Example of the `<basefont>` Element**

This is darkgray text with two sizes larger

It is a courier font, a size smaller and black in color.

## CHAPTER-5

### PROGRAMMING VARIABLE

This element is usually used in conjunction with the `<pre>` and `<code>` elements to indicate that the content of that element is a variable.

```
<!DOCTYPE html>
<html>
<head>
<title>Variable Text Example</title>
</head>
<body>
<p><code>document.write("<var>user-name</var>")</code></p>
</body>
</html>
```

This will produce following result:

```
document.write("user-name")
```

#### Program Output

The `<samp>...</samp>` element indicates sample output from a program, and script etc. Again, it is mainly used when documenting programming or coding concepts.

```
<!DOCTYPE html>
<html>
<head>
<title>Program Output Example</title>
</head>
<body>
<p>Result produced by the program is <samp>Hello World!</samp></p>
</body>
</html>
```

This will produce following result:

```
Result produced by the program is Hello World!
```

#### Address Text

The `<address>...</address>` element is used to contain any address.

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

```

<head>
<title>Address Example</title>
</head>
<body>
<address>388A, Road No 22, Jubilee Hills - Hyderabad</address>
</body>
</html>

```

This will produce following result:

*388A, Road No 22, Jubilee Hills - Hyderabad*

HTML lets you specify metadata - additional important information about a document in a variety of ways. The META elements can be used to include name/value pairs describing properties of the HTML document, such as author, expiry date, a list of keywords, document author etc. The **<meta>** tag is used to provide such additional information. This tag is an empty element and so does not have a closing tag but it carries information within its attributes. You can include one or more meta tags in your document based on what information you want to keep in your document but in general, meta tags do not impact physical appearance of the document so from appearance point of view, it does not matter if you include them or not.

### **Adding Meta Tags to Your Documents**

You can add metadata to your web pages by placing **<meta>** tags inside the header of the document which is represented by **<head>** and **</head>** tags. A meta tag can have following attributes in addition to core attributes:

<b>Attribute</b>	<b>Description</b>
Name	Name for the property. Can be anything. Examples include, keywords, description, author, revised, generator etc.
content	Specifies the property's value.
scheme	Specifies a scheme to interpret the property's value (as declared in the content attribute).
http-equiv	Used for http response message headers. For example http-equiv can be used to refresh the page or to set a cookie. Values include content-type, expires, refresh and set-cookie.

### **Specifying Keywords**

You can use <meta> tag to specify important keywords related to the document and later these keywords are used by the search engines while indexing your webpage for searching purpose. Following is an example where we are adding HTML, Meta Tags, Metadata as important keywords about the document.

```
<!DOCTYPE html>
<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
</head>
<body>
<p>Hello HTML.....</p>
</body>
</html>
```

This will produce following result:

Hello HTML.....

### **Document Description**

You can use <meta> tag to give a short description about the document. This again can be used by various search engines while indexing your webpage for searching purpose.

```
<!DOCTYPE html>
<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
</head>
<body>
<p>Hello HTML5!</p>
</body>
</html>
```

### **Document Revision Date**

You can use <meta> tag to give information about when last time the document was updated. This information can be used by various web browsers while refreshing your webpage.

```
<!DOCTYPE html>
<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
<meta name="revised" content=" spoint, 3/7/2014" />
</head>
<body>
<p>Hello HTML5!</p>
</body>
</html>
```

### **Document Refreshing**

A <meta> tag can be used to specify a duration after which your web page will keep refreshing automatically.

If you want your page keep refreshing after every 5 seconds then use the following syntax.

```
<!DOCTYPE html>
<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
<meta name="revised" content=" spoint, 3/7/2014" />
<meta http-equiv="refresh" content="5" />
</head>
<body>
<p>Hello HTML5!</p>
</body>
</html>
```

### **Page Redirection**

You can use <meta> tag to redirect your page to any other webpage. You can also specify a duration if you want to redirect the page after a certain number of seconds.

Following is an example of redirecting current page to another page after 5 seconds. If you want to redirect page immediately then do not specify *content* attribute.

```
<!DOCTYPE html>
<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
<meta name="revised" content=" spoint, 3/7/2014" />
<meta http-equiv="refresh" content="5; url=http://www. spoint.com" />
</head>
<body>
<p>Hello HTML5!</p>
</body>
</html>
```

## CHAPTER-6

### SETTING OF COOKIES

Cookies are data, stored in small text files on your computer and it is exchanged between web browser and web server to keep track of various information based on your web application need. You can use <meta> tag to store cookies on client side and later this information can be used by the Web Server to track a site visitor.

Following is an example of redirecting current page to another page after 5 seconds. If you want to redirect page immediately then do not specify *content* attribute.

```
<!DOCTYPE html>
<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
<meta name="revised" content=" spoint, 3/7/2014" />
<meta http-equiv="cookie" content="userid=xyz; expires=Wednesday, 08-Aug-15 23:59:59
GMT;" />
</head>
<body>
<p>Hello HTML5!</p>
</body>
</html>
```

If you do not include the expiration date and time, the cookie is considered a session cookie and will be deleted when the user exits the browser.

#### Setting Cookies with PHP

PHP provided **setcookie()** function to set a cookie. This function requires upto six arguments and should be called before <html> tag. For each cookie this function has to be called separately.

```
setcookie(name, value, expire, path, domain, security);
```

Here is the detail of all the arguments –

- **Name** – This sets the name of the cookie and is stored in an environment variable called `HTTP_COOKIE_VARS`. This variable is used while accessing cookies.

- **Value** – This sets the value of the named variable and is the content that you actually want to store.
- **Expiry** – This specifies a future time in seconds since 00:00:00 GMT on 1st Jan 1970. After this time cookie will become inaccessible. If this parameter is not set then cookie will automatically expire when the Web Browser is closed.
- **Path** – This specifies the directories for which the cookie is valid. A single forward slash character permits the cookie to be valid for all directories.
- **Domain** – This can be used to specify the domain name in very large domains and must contain at least two periods to be valid. All cookies are only valid for the host and domain which created them.
- **Security** – This can be set to 1 to specify that the cookie should only be sent by secure transmission using HTTPS otherwise set to 0 which mean cookie can be sent by regular HTTP.

### Setting Author Name

You can set an author name in a web page using meta tag. See an example below:

```
<!DOCTYPE html>
<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
<meta name="author" content="Mahnaz Mohtashim" />
</head>
<body>
<p>Hello HTML5!</p>
</body>
</html>
```

### Specify Character Set

You can use <meta> tag to specify character set used within the webpage.

By default, Web servers and Web browsers use ISO-8859-1 (Latin1) encoding to process Web pages. Following is an example to set UTF-8 encoding:

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

```

<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
<meta name="author" content="Mahnaz Mohtashim" />
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
</head>
<body>
<p>Hello HTML5!</p>
</body>
</html>

```

To serve the static page with traditional Chinese characters, the webpage must contain a <meta> tag to set Big5 encoding:

```

<!DOCTYPE html>
<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
<meta name="author" content="Mahnaz Mohtashim" />
<meta http-equiv="Content-Type" content="text/html; charset=Big5" />
</head>
<body>
<p>Hello HTML5!</p>
</body>
</html>

```

Comment is a piece of code which is ignored by any web browser. It is a good practice to add comments into your HTML code, especially in complex documents, to indicate sections of a document, and any other notes to anyone looking at the code. Comments help you and others understand your code and increases code readability.

HTML comments are placed in between <!-- ... --> tags. So any content placed with-in <!-- ... --> tags will be treated as comment and will be completely ignored by the browser.

```

<!DOCTYPE html>
<html>

```

```
<head> <!-- Document Header Starts -->
<title>This is document title</title>
</head> <!-- Document Header Ends -->
<body>
<p>Document content goes here.....</p>
</body>
</html>
```

This will produce following result without displaying the content given as a part of comments:  
Document content goes here.....

### Comments

Comments do not nest which means a comment cannot be put inside another comment. Second the double-dash sequence "--" may not appear inside a comment except as part of the closing -> tag. You must also make sure that there are no spaces in the start-of-comment string.

Here given comment is a valid comment and will be wiped off by the browser.

```
<!DOCTYPE html>
<html>
<head>
<title>Valid Comment Example</title>
</head>
<body>
<!-- This is valid comment -->
<p>Document content goes here.....</p>
</body>
</html>
```

But following line is not a valid comment and will be displayed by the browser. This is because there is a space between the left angle bracket and the exclamation mark.

```
<!DOCTYPE html>
<html>
<head>
<title>Invalid Comment Example</title>
</head>
<body>
<!-- This is not a valid comment -->
```

```
<p>Document content goes here.....</p>
```

```
</body>
```

```
</html>
```

This will produce following result:

```
<!-- This is not a valid comment -->
```

Document content goes here.....

### **Multiline Comments**

So far we have seen single line comments, but HTML supports multi-line comments as well. You can comment multiple lines by the special beginning tag `<!--` and ending tag `-->` placed before the first line and end of the last line as shown in the given example below.

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

```
<head>
```

```
<title>Multiline Comments</title>
```

```
</head>
```

```
<body>
```

```
<!--
```

This is a multiline comment and it can span through as many as lines you like.

```
-->
```

```
<p>Document content goes here.....</p>
```

```
</body>
```

```
</html>
```

This will produce following result:

Document content goes here.....

### **Conditional Comments**

Conditional comments only work in Internet Explorer (IE) on Windows but they are ignored by other browsers. They are supported from Explorer 5 onwards, and you can use them to give conditional instructions to different versions of IE.

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

```
<head>
```

```
<title>Conditional Comments</title>
```

```
<!--[if IE 6]>
```

Special instructions for IE 6 here

```
<![endif]-->
</head>
<body>
<p>Document content goes here.....</p>
</body>
</html>
```

You will come across a situation where you will need to apply a different style sheet based on different versions of Internet Explorer, in such situation conditional comments will be helpful.

### **Using Comment Tag**

There are few browsers that support <comment> tag to comment a part of HTML code.

```
<!DOCTYPE html><html>
<head>
<title>Using Comment Tag</title>
</head>
<body>
<p>This is <comment>not</comment> Internet Explorer.</p>
</body>
</html>
```

If you are using IE then it will produce following result:

This is Internet Explorer.

But if you are not using IE, then it will produce following result:

This is Internet Explorer.

### **Commenting Script Code**

Though you will learn JavaScript with HTML, in a separate, but here you must make a note that if you are using Java Script or VB Script in your HTML code then it is recommended to put that script code inside proper HTML comments so that old browsers can work properly.

```
<!DOCTYPE html><html>
<head>
<title>Commenting Script Code</title>
<script>
<!--
    document.write("Hello World!")
//-->
```

```
</script>
</head>
<body>
<p>Hello , World!</p>
</body>
</html>
```

This will produce following result:

Hello World!

Hello , World!

### **Commenting Style Sheets**

Though you will learn using style sheets with HTML in a separate, but here you must make a note that if you are using Cascading Style Sheet (CSS) in your HTML code then it is recommended to put that style sheet code inside proper HTML comments so that old browsers can work properly.

```
<!DOCTYPE html><html>
<head>
<title>Commenting Style Sheets</title>
<style>
<!--
.example {
  border:1px solid #4a7d49;
}
/-->
</style>
</head>
<body>
<div class="example">Hello , World!</div>
</body>
</html>
```

This will produce following result:

Hello , World!

## CHAPTER- 7

### IMAGES

Images are very important to beautify as well as to depict many complex concepts in simple way on your web page. This 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 **<img>** tag. Following is the simple syntax to use this tag.

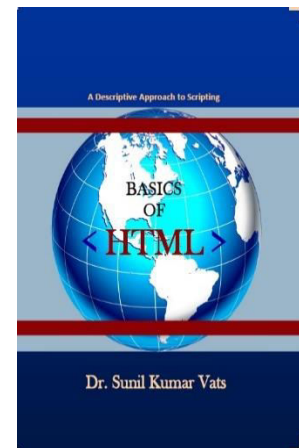
```

```

The **<img>** 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 test.htm and image file test.png in the same directory:

```
<!DOCTYPE html>
<html>
<head>
<title>Using Image in Webpage</title>
</head>
<body>
<p>Simple Image Insert</p>
</img></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 a mandatory attribute which specifies an alternate text for an image, if the image cannot be displayed.

#### Image Location

Usually we keep our all the images in a separate directory. So let's keep HTML file test.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 `"/html/image/test.png"`, try the following example:

```
<!DOCTYPE html>
```

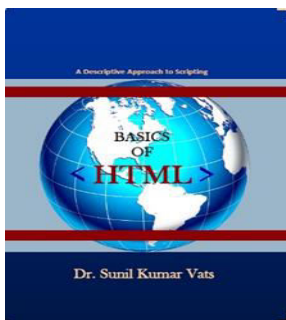
```

<html>
<head>
<title>Using Image in Webpage</title>
</head>
<body>
<p>Simple Image Insert</p>
</img></body>
</html>

```

This will produce following result:

Simple Image Insert



## 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.

```

<!DOCTYPE html>
<html>
<head>
<title>Set Image Width and Height</title>
</head>
<body>
<p>Setting image width and height</p>
</img></body>
</html>

```

## 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 0 means, no border around the picture.

```

<!DOCTYPE html>

```

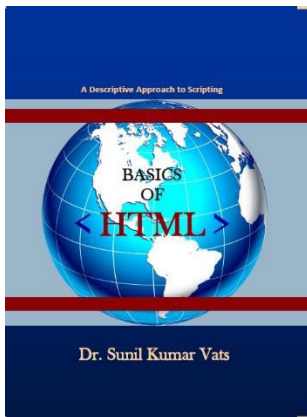
```

<html>
<head>
<title>Set Image Border</title>
</head>
<body>
<p>Setting image Border</p>
</img>
</body>
</html>

```

This will produce following result:

Setting image Border



## Image Alignment

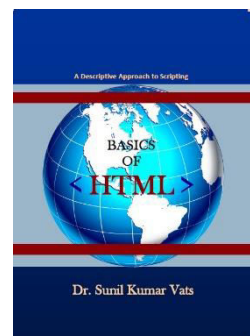
By default image will align at the left side of the page, but you can use **align** attribute to set it in the center or right.

```

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

</body>
</html>

```



This will produce following result:

## CHAPTER-8

### INSERTING TABLE

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.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Tables</title>
</head>
<body>
<table border="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.

```

<!DOCTYPE html>
<html>
<head>
<title>HTML Table Header</title>
</head>
<body>
<table border="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

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.

```

<!DOCTYPE html>
<html>

```

```

<head>
<title>HTML Table Cellpadding</title>
</head>
<body>
<table border="1" cellpadding="5" cellspacing="5">
<tr>
<th>Name</th>
<th>Qualification</th>
</tr>
<tr>
<td>Dr S.K.Vats</td>
<td>M.Sc.,MCA, Ph.D.</td>
</tr>
<tr>
<td>Abhishek</td>
<td>M.Sc.</td>
</tr>
</table>
</body>
</html>

```

This will produce following result:

Name	Salary
Ramesh Raman	5000
Shabbir Hussein	7000

## Colspan And Rowspan

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.

```

<!DOCTYPE html>
<html>
<head>
<title>HTML Table Colspan/Rowspan</title>

```

```

</head>
<body>
<table border="1">
<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		

## 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.

```

<!DOCTYPE html>
<html>
<head>
<title>HTML Table Background</title>
</head>
<body>
<table border="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		

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

```

<!DOCTYPE html>
<html>
<head>
<title>HTML Table Background</title>
</head>
<body>
<table border="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		

## 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.

```
<!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 deprecated in newer version of HTML/XHTML.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Table Caption</title>
</head>
<body>
<table border="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

## 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>`

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Table</title>
</head>
<body>
<table border="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

## Nested

You can use one table inside another table. Not only tables you can use almost all the tags inside table data tag <td>.

Following is the example of using another table and other tags inside a table cell.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Table</title>
</head>
<body>
<table border="1" width="100%">
<tr>
<td>
<table border="1" width="100%">
<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>
</td>
</tr>
```

```
</table>
```

```
</body>
```

```
</html>
```

This will produce following result:

Name	Salary
Ramesh Raman	5000
Shabbir Hussein	7000

The **bgcolor** attribute is used to control the background of an HTML element, specifically page body and table backgrounds. Following is the syntax to use bgcolor attribute with any HTML tag.

```
<tagname bgcolor="color_value"...>
```

This color\_value can be given in any of the following formats:

```
<!-- Format 1 - Use color name -->
```

```
<table bgcolor="lime" >
```

```
<!-- Format 2 - Use hex value -->
```

```
<table bgcolor="#f1f1f1" >
```

```
<!-- Format 3 - Use color value in RGB terms -->
```

```
<table bgcolor="rgb(0,0,120)" >
```

Example

Here are the examples to set background of an HTML tag:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>HTML Background Colors</title>
```

```
</head>
```

```
<body>
```

```
<!-- Format 1 - Use color name -->
```

```
<table bgcolor="yellow" width="100%">
```

```
<tr><td>
```

This background is yellow

```
</td></tr>
```

```
</table>
```

```
<!-- Format 2 - Use hex value -->
```

```
<table bgcolor="#6666FF" width="100%">
```

```
<tr><td>
```

This background is sky blue

```
</td></tr>
```

```
</table>
```

```
<!-- Format 3 - Use color value in RGB terms -->
```

```
<table bgcolor="rgb(255,0,255)" width="100%">
```

```
<tr><td>
```

This background is green

```
</td></tr>
```

```
</table>
```

```
</body>
```

```
</html>
```

This will produce following result:

This background is yellow

This background is sky blue

This background is green

## Background with Images

The **background** attribute can also be used to control the background of an HTML element, specifically page body and table backgrounds. You can specify an image to set background of your HTML page or table. Following is the syntax to use background attribute with any HTML tag.

**Note:** The *background* attribute is deprecated and it is recommended to use Style Sheet for background setting.

```
<tagname background="Image URL"...>
```

The most frequently used image formats are JPEG, GIF and PNG images.

Here are the examples to set background images of a table.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Background Images</title>
</head>
<body>
<!-- Set table background -->
<table background="/images/html.gif" width="100%" height="100">
<tr><td>
This background is filled up with HTML image.
</td></tr>
</table>
</body>
</html>
```

This will produce following result:

This background is filled up with HTML image.

### **Patterned & Transparent Backgrounds**

You might have seen many pattern or transparent backgrounds on various websites. This simply can be achieved by using patterned image or transparent image in the background. It is suggested that while creating patterns or transparent GIF or PNG images, use the smallest dimensions possible even as small as 1x1 to avoid slow loading. Here are the examples to set background pattern of a table:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Background Images</title>
</head>
<body>
<!-- Set a table background using pattern -->
<table background="/images/pattern1.gif" width="100%" height="100">
```

```
<tr><td>
```

This background is filled up with a pattern image.

```
</td></tr>
```

```
</table>
```

```
<!-- Another example on table background using pattrenn -->
```

```
<table background="/images/pattern2.gif" width="100%" height="100">
```

```
<tr><td>
```

This background is filled up with a pattern image.

```
</td></tr>
```

```
</table>
```

```
</body>
```

```
</html>
```

This will produce following result:

This background is filled up with a pattern image.

This background is filled up with a pattern image.

Colors are very important to give a good look and feel to your website. You can specify colors on page level using `<body>` tag or you can set colors for individual tags using **bgcolor** attribute.

The `<body>` tag has following attributes which can be used to set different colors:

- **bgcolor** - sets a color for the background of the page.
- **text** - sets a color for the body text.
- **alink** - sets a color for active links or selected links.
- **link** - sets a color for linked text.
- **vlink** - sets a color for *visited links* - that is, for linked text that you have already clicked on.

## Color Coding Methods

There are following three different methods to set colors in your web page:

- **Color names** - You can specify color names directly like green, blue or red.
- **Hex codes** - A six-digit code representing the amount of red, green, and blue that makes up the color.
- **Color decimal or percentage values** - This value is specified using the `rgb( )` property.

Now we will see these coloring schemes one by one.

## HTML Colors - Color Names

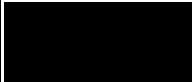

You can specify directly a color name to set text or background color. W3C has listed 16 basic color names that will validate with an HTML validator but there are over 200 different color names supported by major browsers.

Here are the examples to set background of an HTML tag by color name:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Colors by Name</title>
</head>
<body text="blue" bgcolor="green">
<p>Use different color names for for body and table and see the result.</p>
<table bgcolor="black">
<tr>
<td>
<font color="white">This text will appear white on black background.</font>
</td>
</tr>
</table>
</body>
</html>
```

## HTML Colors - Hex Codes

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). A hexadecimal value can be taken from any graphics software like Adobe Photoshop, Paintshop Pro or MS Paint. Each hexadecimal code will be preceded by a pound or hash sign #. Following is a list of few colors using hexadecimal notation.

Color	Color HEX
	#000000
	#FF0000

	#00FF00
	#0000FF
	#FFFF00

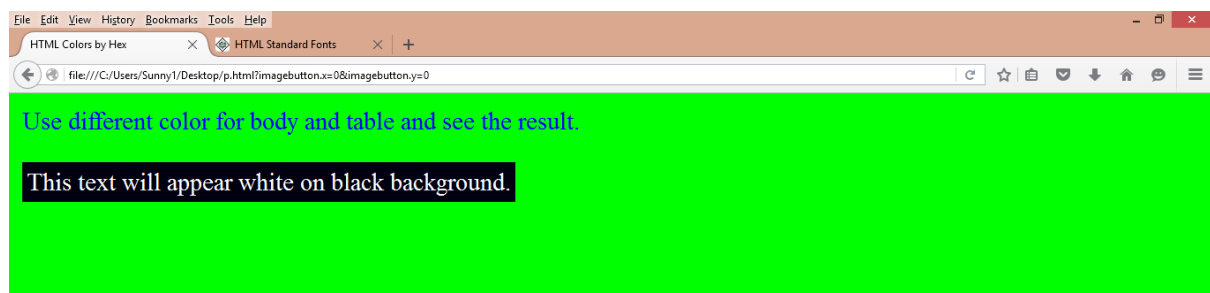
	#00FFFF
	#FF00FF

	#C0C0C0
	#FFFFFF

Here are the examples to set background of an HTML tag by color code in hexadecimal:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Colors by Hex</title>
</head>
<body text="#0000FF" bgcolor="#00FF00">
<p>Use different color for body and table and see the result.</p>
<table bgcolor="#000011">
<tr>
<td>
<font color="#FFFFFF">This text will appear white on black background.</font>
</td>
</tr>
</table>
</body>
</html>
```

The Output will look like as:


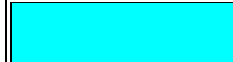





## HTML Colors - RGB Values

This color value is specified using the **rgb( )** property. This property takes three values, one each for red, green, and blue. The value can be an integer between 0 and 255 or a percentage.

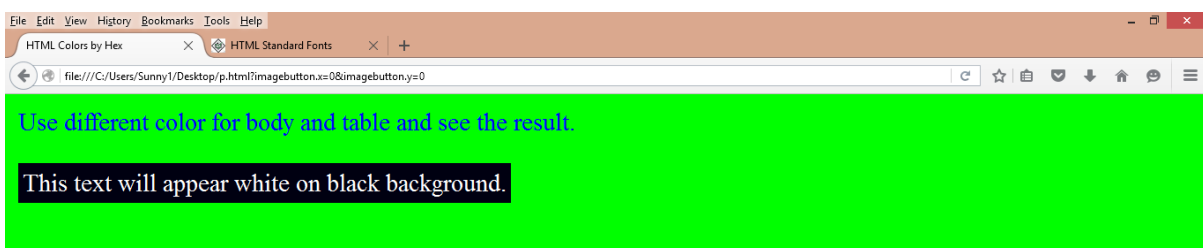
**Note:** All the browsers does not support rgb() property of color so it is recommended not to use it. Following is a list to show few colors using RGB values.

Color	Color RGB
	rgb(0,0,0)
	rgb(255,0,0)
	rgb(0,255,0)
	rgb(0,0,255)

	rgb(255,255,0)
	rgb(0,255,255)
	rgb(255,0,255)
	rgb(192,192,192)
	rgb(255,255,255)

Here are the examples to set background of an HTML tag by color code using rgb() values:

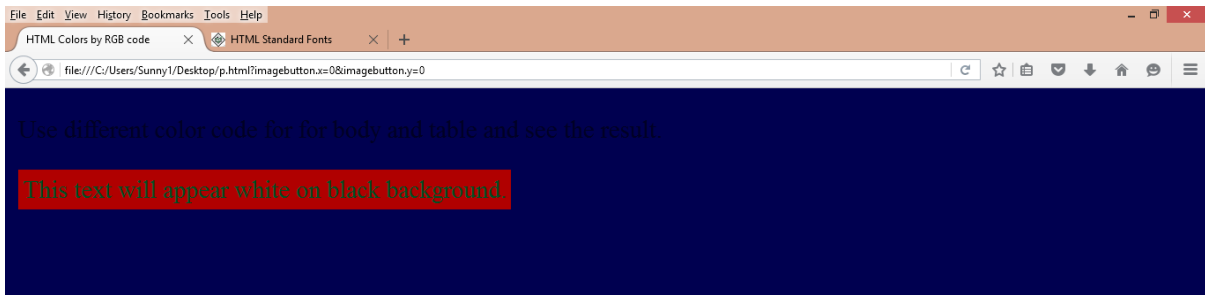
```
<!DOCTYPE html>
<html>
<head>
<title>HTML Colors by RGB code</title>
</head>
<body text="rgb(0,0,255)" bgcolor="rgb(0,255,0)">
<p>Use different color code for for body and table and see the result.</p>
<table bgcolor="rgb(0,0,0)">
<tr>
<td>
<font color="rgb(255,255,255)">This text will appear white on black background.</font>
```



```
</td>
</tr>
</table>
</body>
```

</html>

The Screen look like this:-



### Browser Safe Colors

Here is the list of 216 colors which are supposed to be safest and computer independent colors. These colors vary from hexa code 000000 to FFFFFFFF and they will be supported by all the computers having 256 color palette.

000000	000033	000066	000099	0000CC	0000FF
003300	003333	003366	003399	0033CC	0033FF
006600	006633	006666	006699	0066CC	0066FF
009900	009933	009966	009999	0099CC	0099FF
00CC00	00CC33	00CC66	00CC99	00CCCC	00CCFF
00FF00	00FF33	00FF66	00FF99	00FFCC	00FFFF
330000	330033	330066	330099	3300CC	3300FF
333300	333333	333366	333399	3333CC	3333FF
336600	336633	336666	336699	3366CC	3366FF
339900	339933	339966	339999	3399CC	3399FF
33CC00	33CC33	33CC66	33CC99	33CCCC	33CCFF
33FF00	33FF33	33FF66	33FF99	33FFCC	33FFFF
660000	660033	660066	660099	6600CC	6600FF
663300	663333	663366	663399	6633CC	6633FF
666600	666633	666666	666699	6666CC	6666FF
669900	669933	669966	669999	6699CC	6699FF
66CC00	66CC33	66CC66	66CC99	66CCCC	66CCFF

66FF00	66FF33	66FF66	66FF99	66FFCC	66FFFF
990000	990033	990066	990099	9900CC	9900FF
993300	993333	993366	993399	9933CC	9933FF
996600	996633	996666	996699	9966CC	9966FF
999900	999933	999966	999999	9999CC	9999FF
99CC00	99CC33	99CC66	99CC99	99CCCC	99CCFF
99FF00	99FF33	99FF66	99FF99	99FFCC	99FFFF
CC0000	CC0033	CC0066	CC0099	CC00CC	CC00FF
CC3300	CC3333	CC3366	CC3399	CC33CC	CC33FF
CC6600	CC6633	CC6666	CC6699	CC66CC	CC66FF
CC9900	CC9933	CC9966	CC9999	CC99CC	CC99FF
CCCC00	CCCC33	CCCC66	CCCC99	CCCCCC	CCCCFF
CCFF00	CCFF33	CCFF66	CCFF99	CCFFCC	CCFFFF
FF0000	FF0033	FF0066	FF0099	FF00CC	FF00FF
FF3300	FF3333	FF3366	FF3399	FF33CC	FF33FF
FF6600	FF6633	FF6666	FF6699	FF66CC	FF66FF
FF9900	FF9933	FF9966	FF9999	FF99CC	FF99FF
FFCC00	FFCC33	FFCC66	FFCC99	FFCCCC	FFCCFF
FFFF00	FFFF33	FFFF66	FFFF99	FFFFCC	FFFFFF

Fonts play very important role in making a website more user friendly and increasing content readability. Font face and color depends entirely on the computer and browser that is being used to view your page but you can use HTML **<font>** tag to add style, size, and color to the text on your website. You can use a **<basefont>** tag to set all of your text to the same size, face, and color.

The font tag is having three attributes called **size**, **color**, and **face** to customize your fonts. To change any of the font attributes at any time within your webpage, simply use the **<font>** tag. The text that follows will remain changed until you close with the **</font>** tag. You can change one or all of the font attributes within one **<font>** tag.

**Note:** The font and basefont tags are deprecated and it is supposed to be removed in a future version of HTML. So they should not be used rather, it's suggested to use CSS styles to

manipulate your fonts. But still for learning purpose, this chapter will explain font and basefont tags in detail.

## Definition Lists

HTML and XHTML support a list style which is called **definition lists** where entries are listed like in a dictionary or encyclopaedia. The definition list is the ideal way to present a glossary, list of terms, or other name/value list.

Definition List makes use of following three tags.

- `<dl>` - Defines the start of the list
- `<dt>` - A term
- `<dd>` - Term definition
- `</dl>` - Defines the end of the list

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Definition List</title>
</head>
<body>
<dl>
<dt><b>HTML</b></dt>
<dd>This stands for Hyper Text Markup Language</dd>
<dt><b>HTTP</b></dt>
<dd>This stands for Hyper Text Transfer Protocol</dd>
</dl>
</body>
</html>
```

This will produce following result:

### **HTML**

This stands for Hyper Text Markup Language

### **HTTP**

This stands for Hyper Text Transfer Protocol

## CHAPTER-9

### LISTS

HTML offers web authors three ways for specifying lists of information. All lists must contain one or more list elements. Lists may contain:

- **<ul>** - An unordered list. This will list items using plain bullets.
- **<ol>** - An ordered list. This will use different schemes of numbers to list your items.
- **<dl>** - A definition list. This arranges your items in the same way as they are arranged in a dictionary.

#### Unordered Lists

An unordered list is a collection of related items that have no special order or sequence. This list is created by using HTML **<ul>** tag. Each item in the list is marked with a bullet.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Unordered List</title>
</head>
<body>
<ul>
<li>Beetroot</li>
<li>Ginger</li>
<li>Potato</li>
<li>Radish</li>
</ul>
</body>
</html>
```

This will produce following result:

- Beetroot
- Ginger
- Potato
- Radish

#### Type

You can use **type** attribute for **<ul>** tag to specify the type of bullet you like. By default it is a disc. Following are the possible options:

```
<ul type="square">
```

```
<ul type="disc">
```

```
<ul type="circle">
```

Following is an example where we used `<ul type="square">`

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>HTML Unordered List</title>
```

```
</head>
```

```
<body>
```

```
  <ul type="square">
```

```
    <li>Beetroot</li>
```

```
    <li>Ginger</li>
```

```
    <li>Potato</li>
```

```
    <li>Radish</li>
```

```
  </ul>
```

```
</body>
```

```
</html>
```

This will produce following result:

- Beetroot
- Ginger
- Potato
- Radish

Following is an example where we used `<ul type="disc">` :

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>HTML Unordered List</title>
```

```
</head>
```

```
<body>
```

```
  <ul type="disc">
```

```
    <li>Beetroot</li>
```

```
    <li>Ginger</li>
```

```
    <li>Potato</li>
```

```
<li>Radish</li>
</ul>
</body>
</html>
```

This will produce following result:

- Beetroot
- Ginger
- Potato
- Radish

Following is an example where we used `<ul type="circle">` :

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Unordered List</title>
</head>
<body>
  <ul type="circle">
    <li>Beetroot</li>
    <li>Ginger</li>
    <li>Potato</li>
    <li>Radish</li>
  </ul>
</body>
</html>
```

This will produce following result:

- Beetroot
- Ginger
- Potato
- Radish

## Ordered Lists

If you are required to put your items in a numbered list instead of bulleted then HTML ordered list will be used. This list is created by using `<ol>` tag. The numbering starts at one and is incremented by one for each successive ordered list element tagged with `<li>`.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Ordered List</title>
</head>
<body>
<ol>
<li>Beetroot</li>
<li>Ginger</li>
<li>Potato</li>
<li>Radish</li>
</ol>
</body>
</html>
```

This will produce following result:

1. Beetroot
2. Ginger
3. Potato
4. Radish

You can use **type** attribute for `<ol>` tag to specify the type of numbering you like. By default it is a number. Following are the possible options:

`<ol type="1">` - Default-Case Numerals.

`<ol type="I">` - Upper-Case Numerals.

`<ol type="i">` - Lower-Case Numerals.

`<ol type="a">` - Lower-Case Letters.

`<ol type="A">` - Upper-Case Letters.

Following is an example where we used `<ol type="1">`

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Ordered List</title>
</head>
<body>
```

```
<ol type="1">
<li>Beetroot</li>
<li>Ginger</li>
<li>Potato</li>
<li>Radish</li>
</ol>
</body>
</html>
```

This will produce following result:

1. Beetroot
2. Ginger
3. Potato
4. Radish

Following is an example where we used `<ol type="I">`

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Ordered List</title>
</head>
<body>
<ol type="I">
<li>Beetroot</li>
<li>Ginger</li>
<li>Potato</li>
<li>Radish</li>
</ol>
</body>
</html>
```

This will produce following result:

- I. Beetroot
- II. Ginger
- III. Potato
- IV. Radish

Following is an example where we used `<ol type="i">`

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Ordered List</title>
</head>
<body>
  <ol type="i">
    <li>Beetroot</li>
    <li>Ginger</li>
    <li>Potato</li>
    <li>Radish</li>
  </ol>
</body>
</html>
```

This will produce following result:

- i. Beetroot
- ii. Ginger
- iii. Potato
- iv. Radish

Following is an example where we used <ol type="A">

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Ordered List</title>
</head>
<body>
  <ol type="A">
    <li>Beetroot</li>
    <li>Ginger</li>
    <li>Potato</li>
    <li>Radish</li>
  </ol>
</body>
</html>
```

This will produce following result:

- A. Beetroot
- B. Ginger
- C. Potato
- D. Radish

Following is an example where we used `<ol type="a">`

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Ordered List</title>
</head>
<body>
  <ol type="a">
    <li>Beetroot</li>
    <li>Ginger</li>
    <li>Potato</li>
    <li>Radish</li>
  </ol>
</body>
</html>
```

This will produce following result:

- a. Beetroot
- b. Ginger
- c. Potato
- d. Radish

### **Start Attribute**

You can use **start** attribute for `<ol>` tag to specify the starting point of numbering you need.

Following are the possible options:

- `<ol type="1" start="4">` - Numerals starts with 4.
- `<ol type="I" start="4">` - Numerals starts with IV.
- `<ol type="i" start="4">` - Numerals starts with iv.
- `<ol type="a" start="4">` - Letters starts with d.
- `<ol type="A" start="4">` - Letters starts with D.

Following is an example where we used `<ol type="i" start="4" >`

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Ordered List</title>
</head>
<body>
  <ol type="i" start="4">
    <li>Beetroot</li>
    <li>Ginger</li>
    <li>Potato</li>
    <li>Radish</li>
  </ol>
</body>
</html>
```

This will produce following result:

- iv. Beetroot
- v. Ginger
- vi. Potato
- vii. Radish

## CHAPTER-10

### HYPERLINK

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.

A link is specified using HTML tag `<a>`. This tag is called **anchor tag** and anything between the opening `<a>` tag and the closing `</a>` 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.

```
<a href="web page full Path">Link Text</a>
```

Let's try following example which links `http://www.jnvmahendragarh.gov.in` at your page:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Hyperlink Example</title>
```

```
</head>
```

```
<body>
```

```
<p>Click following link to Visit JNV Mohindergarh</p>
```

```
<a href="http://www.jnvmahendragarh.gov.in"> JNV Mohinndergarh</a>
```

```
</body>
```

```
</html>
```

This will produce following result.

#### Target Attribute

We can also use **target** attribute with hyperlink after full path. This attribute is used to specify the location where linked document is opened. Following are possible options:

Option	Description
<code>_blank</code>	Opens the linked document in a new window or tab.
<code>_self</code>	Opens the linked document in the same frame.
<code>_parent</code>	Opens the linked document in the parent frame.
<code>_top</code>	Opens the linked document in the full body of the window.
<code>targetframe</code>	Opens the linked document in a named <i>targetframe</i> .

Try following example to understand basic difference in few options given for target attribute.

```

<!DOCTYPE html>
<html>
<head>
<title>Hyperlink Example</title>
<base href="http://www. spoint.com/">
</head>
<body>
<p>Click any of the following links</p>
<a href="http://cbse.nic.in/e-cbse/index.html="_blank">Opens in New</a>
<a href="http://cbse.nic.in/e-cbse/index.html="_self">Opens in Self</a>
<a href="http://cbse.nic.in/e-cbse/index.html="_parent">Opens in Parent</a>
<a href="http://cbse.nic.in/e-cbse/index.html="_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

[Opens in New](#) | [Opens in Self](#) | [Opens in Parent](#) | [Opens in Body](#)

### 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. Following example makes use of **<base>** tag to specify base URL and later we can use relative path to all the links instead of giving complete URL for every link.

```

<!DOCTYPE html>
<html>
<head>
<title>Hyperlink Example</title>
<base href="http://www. spoint.com/">
</head>
<body>
<p>Click following link</p>

```

```
<a href="/html/index.htm" target="_blank">HTML </a>
```

```
</body>
```

```
</html>
```

This will produce following result, where you can click on the link generated **HTML** to reach to the HTML. Now given URL `<a href="/html/index.htm"` is being considered as `<a href="http://59.179.16.89/cbse/2016/regn/cbse.aspx"`.

Click following link

HTML

### 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 with-in a webpage and name it using `<a...>` tag as follows:

```
<h1>HTML Text Links <a name="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 .

Go to the Top

### Setting Link Colors

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

Save the following in test.htm and open it in any web browser to see how **link**, **alink** and **vlink** attributes work.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Hyperlink Example</title>
```

```
<base href="http://www. spoint.com/">
```

```
</head>
```

```
<body alink="#54A250" link="#040404" vlink="#F40633">
```

```
<p>Click following link</p>
```

```
<a href="/html/index.htm" target="_blank" >HTML </a>
```

```
</body>
```

```
</html>
```

This will produce following result. 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.

Click following link

HTML

### 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:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Hyperlink Example</title>
```

```
</head>
```

```
<a href="http://www.spoin.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

### Download Dialog Box

Sometimes it is desired that you want to give an option where a user will click a link and it will pop up a "File Download" box to the user in stead of displaying actual content. This is very easy and can be achieved using an HTTP header in your HTTP response. For example, if you want make a **FileName** file downloadable from a given link then its syntax will be as follows.

```
#!/usr/bin/perl
```

```
# Additional HTTP Header
```

```
print "Content-Type:application/octet-stream; name=\"FileName\\r\\n\";
```

```
print "Content-Disposition: attachment; filename=\"FileName\\r\\n\\n\";
```

```
# Open the target file and list down its content as follows
```

```
open( FILE, "<FileName" );
```

```
while(read(FILE, $buffer, 100)){
```

```
    print("$buffer");
```

```
}
```

We have seen how to create hypertext link using text and we also learnt how to use images in our webpages. Now we will learn how to use images to create hyperlinks. It's simple to use an image as hyperlink. We just need to use an image inside hyperlink at the place of text as shown below:

```
<!DOCTYPE html>
<html>
<head>
<title>Image Hyperlink Example</title>
</head>
<body>
<p>Click following link</p>
<a href="http://59.179.16.89/cbse/2016/regn/cbse.aspx=_self">
  
</a>
</body>
</html>
```

This will produce following result, where you can click on the images to reach to the home page of Point. Click following link



HTML `<a>` tag provides you option to specify an email address to send an email. While using `<a>` tag as an email tag, you will use **mailto:email address** along with *href* attribute. Following is the syntax of using **mailto** instead of using http.

```
<a href= "mailto:abc@example.com">Send Email</a>
```

This code will generate following link which you can use to send email.

Send\_Email

Now if a user clicks this link, it launches one Email Client (like Lotus Notes, Outlook Express etc.) installed on your user's computer. There is another risk to use this option to send email because if user do not have email client installed on their computer then it would not be possible to send email.

### **Default Settings**

You can specify a default *email subject* and *email body* along with your email address.

Following is the example to use default subject and body.

```
<a href="mailto:abc@example.com?subject=Feedback&body=Message">
```

Send Feedback

```
</a>
```

This code will generate following link which you can use to send email.

Send\_Feedback

## CHAPTER-11

### FRAMES

HTML frames are used to divide your browser window into multiple sections where each section can load a separate HTML document. A collection of frames in the browser window is known as a frameset. The window is divided into frames in a similar way the tables are organized: into rows and columns.

#### Disadvantages of Frames

There are few drawbacks with using frames, so it's never recommended to use frames in your webpages:

- Some smaller devices cannot cope with frames often because their screen is not big enough to be divided up.
- Sometimes your page will be displayed differently on different computers due to different screen resolution.
- The browser's *back button* might not work as the user hopes.
- There are still few browsers that do not support frame technology.

#### Creating Frames

To use frames on a page we use `<frameset>` tag instead of `<body>` tag. The `<frameset>` tag defines how to divide the window into frames. The **rows** attribute of `<frameset>` tag defines horizontal frames and **cols** attribute defines vertical frames. Each frame is indicated by `<frame>` tag and it defines which HTML document shall open into the frame.

Example

Following is the example to create three horizontal frames:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Frames</title>
</head>
<frameset rows="10%,80%,10%">
  <frame name="top" src="/html/top_frame.htm" />
  <frame name="main" src="/html/main_frame.htm" />
  <frame name="bottom" src="/html/bottom_frame.htm" />
</frameset>
<body>
```

Your browser does not support frames.

```
</body>
</noframes>
</frameset>
</html>
```

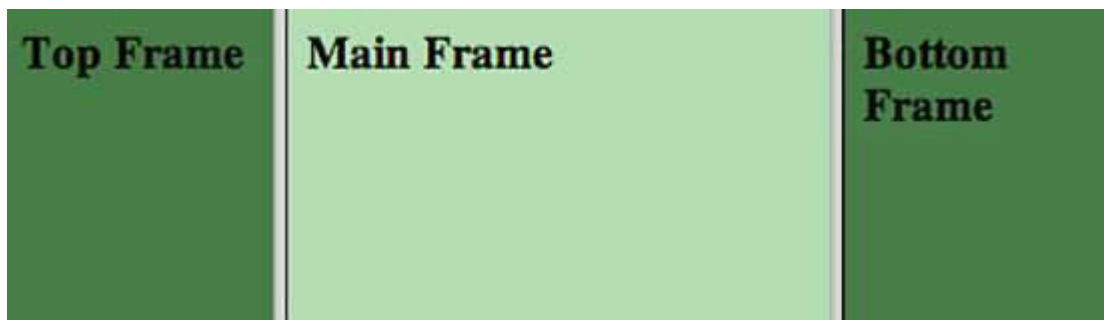
This will produce following result:



Let's put above example as follows, here we replaced rows attribute by cols and changed their width. This will create all the three frames vertically:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Frames</title>
</head>
<frameset cols="25%,50%,25%">
  <frame name="left" src="/html/top_frame.htm" />
  <frame name="center" src="/html/main_frame.htm" />
  <frame name="right" src="/html/bottom_frame.htm" />
</noframes>
<body>
  Your browser does not support frames.
</body>
</noframes>
</frameset>
</html>
```

This will produce following result:



## Frameset Attributes

Following are important attributes of the <frameset> tag:

Attribute	Description
Cols	<p>specifies how many columns are contained in the frameset and the size of each column. You can specify the width of each column in one of four ways:</p> <ul style="list-style-type: none"> <li>• Absolute values in pixels. For example to create three vertical frames, use <code>cols="100, 500,100"</code>.</li> <li>• A percentage of the browser window. For example to create three vertical frames, use <code>cols="10%, 80%,10%"</code>.</li> <li>• Using a wildcard symbol. For example to create three vertical frames, use <code>cols="10%, *,10%"</code>. In this case wildcard takes remainder of the window.</li> <li>• As relative widths of the browser window. For example to create three vertical frames, use <code>cols="3*,2*,1*"</code>. This is an alternative to percentages. You can use relative widths of the browser window. Here the window is divided into sixths: the first column takes up half of the window, the second takes one third, and the third takes one sixth.</li> </ul>
Rows	<p>This attribute works just like the cols attribute and takes the same values, but it is used to specify the rows in the frameset. For example to create two horizontal frames, use <code>rows="10%, 90%"</code>. You can specify the height of each row in the same way as explained above for columns.</p>
Border	<p>This attribute specifies the width of the border of each frame in pixels. For example <code>border="5"</code>. A value of zero means no border.</p>

frameborder	This attribute specifies whether a three-dimensional border should be displayed between frames. This attribute takes value either 1 (yes) or 0 (no). For example frameborder="0" specifies no border.
framespacing	This attribute specifies the amount of space between frames in a frameset. This can take any integer value. For example framespacing="10" means there should be 10 pixels spacing between each frames.

## Frame Attributes

Following are important attributes of <frame> tag:

Attribute	Description
src	This attribute is used to give the file name that should be loaded in the frame. Its value can be any URL. For example, src="/html/top_frame.htm" will load an HTML file available in html directory.
name	This attribute allows you to give a name to a frame. It is used to indicate which frame a document should be loaded into. This is especially important when you want to create links in one frame that load pages into an another frame, in which case the second frame needs a name to identify itself as the target of the link.
frameborder	This attribute specifies whether or not the borders of that frame are shown; it overrides the value given in the frameborder attribute on the <frameset> tag if one is given, and this can take values either 1 (yes) or 0 (no).
marginwidth	This attribute allows you to specify the width of the space between the left and right of the frame's borders and the frame's content. The value is given in pixels. For example marginwidth="10".
marginheight	This attribute allows you to specify the height of the space between the top and bottom of the frame's borders and its contents. The value is given in pixels. For example marginheight="10".
noresize	By default you can resize any frame by clicking and dragging on the borders of a frame. The noresize attribute prevents a user from being able to resize the frame. For example noresize="noresize".

scrolling	This attribute controls the appearance of the scrollbars that appear on the frame. This takes values either "yes", "no" or "auto". For example scrolling="no" means it should not have scroll bars.
longdesc	This attribute allows you to provide a link to another page containing a long description of the contents of the frame. For example longdesc="framedescription.htm"

## Browser Support for Frames

If a user is using any old browser or any browser which does not support frames then <noframes> element should be displayed to the user.

So you must place a <body> element inside the <noframes> element because the <frameset> element is supposed to replace the <body> element, but if a browser does not understand <frameset> element then it should understand what is inside the <body> element which is contained in a <noframes> element.

You can put some nice message for your user having old browsers. For example *Sorry!! your browser does not support frames.* as shown in the above example.

## Frame's name and target attributes

One of the most popular uses of frames is to place navigation bars in one frame and then load main pages into a separate frame.

Let's see following example where a test.htm file has following code:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Target Frames</title>
</head>
<frameset cols="200, *">
  <frame src="/html/menu.htm" name="menu_page" />
  <frame src="/html/main.htm" name="main_page" />
</frameset>
<body>
  Your browser does not support frames.
</body>
</noframes>
```

```
</frameset>
```

```
</html>
```

Here we have created two columns to fill with two frames. The first frame is 200 pixels wide and will contain the navigation menu bar implemented by **menu.htm** file. The second column fills in remaining space and will contain the main part of the page and it is implemented by **main.htm** file. For all the three links available in menu bar, we have mentioned target frame as **main\_page**, so whenever you click any of the links in menubar, available link will open in **main\_page**.

Following is the content of menu.htm file

```
<!DOCTYPE html>
```

```
<html>
```

```
<body bgcolor="#4a7d49">
```

```
<a href="http://www.google.com" target="main_page">Google</a>
```

```
<br /><br />
```

```
<a href="http://www.microsoft.com" target="main_page">Microsoft</a>
```

```
<br /><br />
```

```
<a href="http://news.bbc.co.uk" target="main_page">BBC News</a>
```

```
</body>
```

```
</html>
```

Following is the content of main.htm file:

```
<!DOCTYPE html>
```

```
<html>
```

```
<body bgcolor="#b5dcb3">
```

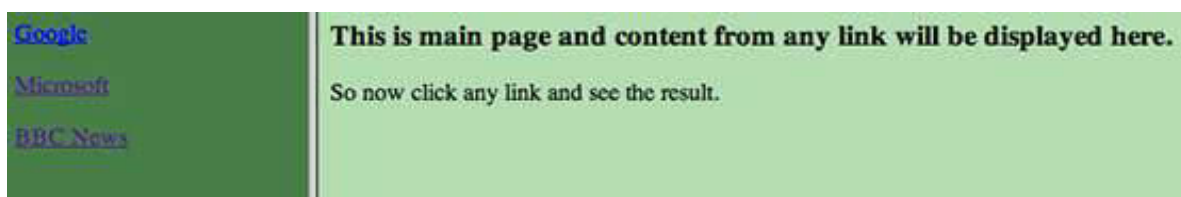
```
<h3>This is main page and content from any link will be displayed here.</h3>
```

```
<p>So now click any link and see the result.</p>
```

```
</body>
```

```
</html>
```

When we load **test.htm** file, it produces following result:



Now you can try to click links available in the left panel and see the result. The *target* attribute can also take one of the following values:

Option	Description
_self	Loads the page into the current frame.
_blank	Loads a page into a new browser window.opening a new window.
_parent	Loads the page into the parent window, which in the case of a single frameset is the main browser window.
_top	Loads the page into the browser window, replacing any current frames.
Target frame	Loads the page into a named target frame.

You can define an inline frame with HTML tag **<iframe>**. The **<iframe>** tag is not somehow related to **<frameset>** tag, instead, it can appear anywhere in your document. The **<iframe>** tag defines a rectangular region within the document in which the browser can display a separate document, including scrollbars and borders.

The **src** attribute is used to specify the URL of the document that occupies the inline frame.

Following is the example to show how to use the **<iframe>**:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Iframes</title>
</head>
<body>
<p>Document content goes here...</p>
<iframe src="/html/menu.htm" width="555" height="200">
  Sorry your browser does not support inline frames.
</iframe>
<p>Document content also go here...</p>
</body>
</html>
```

This will produce following result:

Document content goes here...

Sorry your browser does not support inline frames.

Document content can also go here...

### **Iframe Tag Attributes**

Most of the attributes of the <iframe> tag, including *name*, *class*, *frameborder*, *id*, *longdesc*, *marginheight*, *marginwidth*, *name*, *scrolling*, *style*, and *title* behave exactly like the corresponding attributes for the <frame> tag.

Attribute	Description
src	This attribute is used to give the file name that should be loaded in the frame. Its value can be any URL. For example, src="/html/top_frame.htm" will load an HTML file available in html directory.
name	This attribute allows you to give a name to a frame. It is used to indicate which frame a document should be loaded into. This is especially important when you want to create links in one frame that load pages into an another frame, in which case the second frame needs a name to identify itself as the target of the link.
frameborder	This attribute specifies whether or not the borders of that frame are shown; it overrides the value given in the frame border attribute on the <frameset> tag if one is given, and this can take values either 1 (yes) or 0 (no).
marginwidth	This attribute allows you to specify the width of the space between the left and right of the frame's borders and the frame's content. The value is given in pixels. For example marginwidth="10".
marginheight	This attribute allows you to specify the height of the space between the top and bottom of the frame's borders and its contents. The value is given in pixels. For example marginheight="10".
noresize	By default you can resize any frame by clicking and dragging on the borders of a frame. The noresize attribute prevents a user from being able to resize the frame. For example noresize="noresize".
scrolling	This attribute controls the appearance of the scrollbars that appear on the frame. This takes values either "yes", "no" or "auto". For example scrolling="no" means it should not have scroll bars.
longdesc	This attribute allows you to provide a link to another page containing a long description of the contents of the frame. For example longdesc="framedescription.htm"

All the HTML elements can be categorized into two categories (a) Block Level Elements (b)

Inline Elements

**Block Elements**

Block elements appear on the screen as if they have a line break before and after them. For example the <p>, <h1>, <h2>, <h3>, <h4>, <h5>, <h6>, <ul>, <ol>, <dl>, <pre>, <hr />, <blockquote>, and <address> elements are all block level elements. They all start on their own new line, and anything that follows them appears on its own new line.

### **Inline Elements**

Inline elements, on the other hand, can appear within sentences and do not have to appear on a new line of their own. The <b>, <i>, <u>, <em>, <strong>, <sup>, <sub>, <big>, <small>, <li>, <ins>, <del>, <code>, <cite>, <dfn>, <kbd>, and <var> elements are all inline elements.

### **Grouping HTML Elements**

There are two important tags which we use very frequently to group various other HTML tags (i) <div> tag and (ii) <span> tag

#### **The Div Tag**

This is the very important block level tag which plays a big role in grouping various other HTML tags and applying CSS on group of elements. Even now <div> tag can be used to create webpage layout where we define different parts ( Left, Right, Top etc) of the page using <div> tag. This tag does not provide any visual change on the block but this has more meaning when it is used with CSS.

Following is a simple example of <div> tag. We will learn Cascading Style Sheet (CSS) in a separate chapter but we used it here to show the usage of <div> tag:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML div Tag</title>
</head>
<body>
<!-- First group of tags -->
<div style="color:red">
  <h4>This is first group</h4>
  <p>Following is a list of vegetables</p>
  <ul>
    <li>Beetroot</li>
    <li>Ginger</li>
    <li>Potato</li>
```

```

    <li>Radish</li>
  </ul>
</div>
<!-- Second group of tags -->
<div style="color:green">
  <h4>This is second group</h4>
  <p>Following is a list of fruits</p>
  <ul>
    <li>Apple</li>
    <li>Banana</li>
    <li>Mango</li>
    <li>Strawberry</li>
  </ul>
</div>
</body>
</html>

```

This will produce following result:

*This is first group*

Following is a list of vegetables

- Beetroot
- Ginger
- Potato
- Radish

*This is second group*

Following is a list of fruits

- Apple
- Banana
- Mango
- Strawberry

## **The Span Tag**

The HTML `<span>` is an inline element and it can be used to group inline-elements in an HTML document. This tag also does not provide any visual change on the block but has more meaning when it is used with CSS. The difference between the `<span>` tag and the `<div>` tag is that the

<span> tag is used with inline elements where as the <div> tag is used with block-level elements. Following is a simple example of <span> tag. We will learn Cascading Style Sheet (CSS) in a separate chapter but we used it here to show the usage of <span> tag:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML span Tag</title>
</head>
<body>
<p>This is <span style="color:red">red</span> and this is <span
style="color:green">green</span></p>
</body>
</html>
```

This will produce following result:

This is red, and this is green

By default, your webpage background is white in color. You may not like it, but no worries. HTML provides you following two good ways to decorate your webpage background.

- Html Background with Colors
- Html Background with Images

Now let's see both the approaches one by one using appropriate examples.

### TEXT AREA ATTRIBUTE

A Text Area is the important Attribute in HTML, which is most of the time used to create Application or Feedback form. The important attribute of Text Area is as follow:-

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

### 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**.

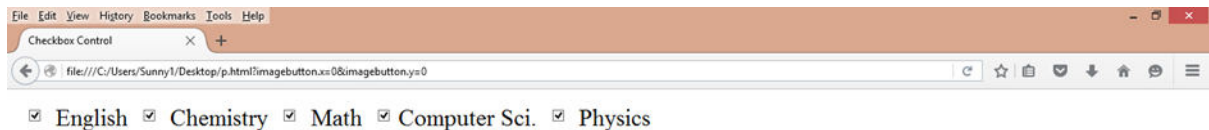
Here is an example HTML code for a form with two checkboxes:

```

<!DOCTYPE html>
<html>
<head>
<title>Checkbox Control</title>
</head>
<body>
<form>
<input type="checkbox" name="English" value="on"> English
<input type="checkbox" name="Chemistry" value="on"> Chemistry
<input type="checkbox" name="Math" value="on"> Math
<input type="checkbox" name="Computer Sci." value="on">Computer Sci.
<input type="checkbox" name="physics" value="OFF"> Physics
</form>
</body>
</html>

```

This will produce following result:



## 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 <b>checkbox</b> .
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.

## Radio Button Control

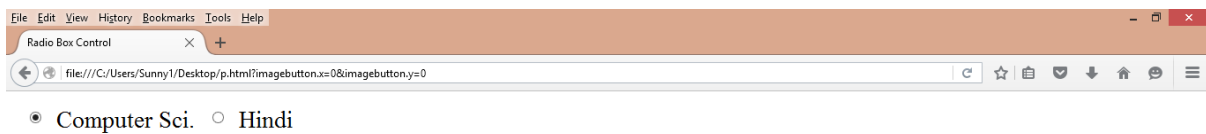
Radio buttons are used when out of many options, just one option is required to be selected.

They are also created using HTML `<input>` tag but type attribute is set to **radio**.

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="Computer Sci."> Computer Sci.
<input type="radio" name="subject" value="Hindi"> Hindi
</form>
</body>
</html>
```

This will produce following result:



## 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 <b>radio</b> .
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.

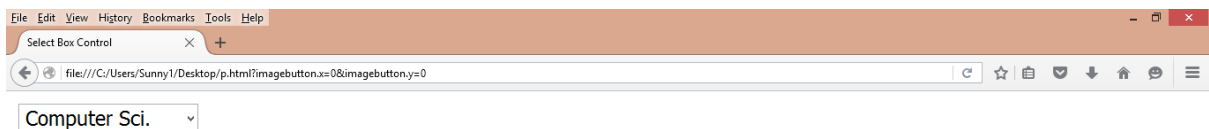
## Select (Combo) Box Control

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.

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>
<select name="dropdown">
<option value="Hindi" selected>Hindi</option>
<option value="English" selected>English</option>
<option value="Computer Sci." selected>Computer Sci.</option>
<option value="Informatics Prac." selected>Informatics Prac.</option>
<option value="Maths" selected>Maths</option>
<option value="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

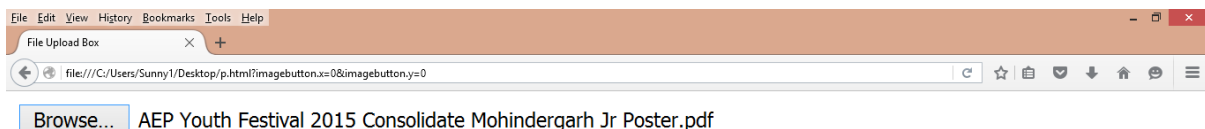
## File Upload Box

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**.

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>
```

This will produce following result:



## 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.

### Button Controls

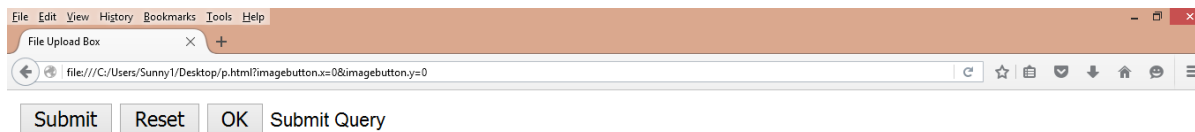
There are various ways in HTML to create clickable buttons. You can also create a clickable button using `<input>` tag by setting its type attribute to **button**. The type attribute can take the following values:

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.

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" /> &nbsp; Submit Query
</form>
</body>
/html>
```

This will produce following result:



## Hidden Form Controls

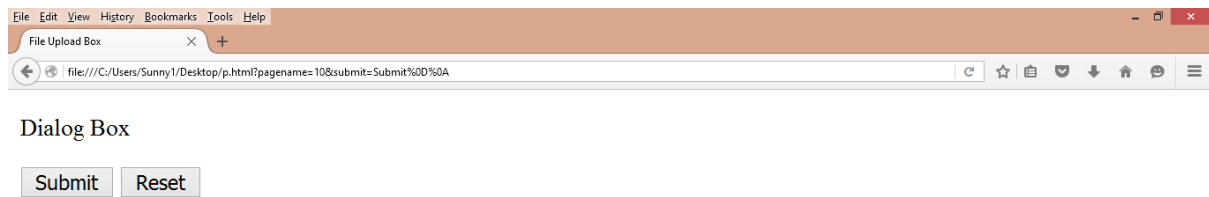
Hidden form controls are used to hide data inside the page which later on can be pushed to the server. This control hides inside the code and does not appear on the actual page. For example, following hidden form is being used to keep current page number. When a user will click next page then the value of hidden control will be sent to the web server and there it will decide which page has be displayed next based on the passed current page.

Here is example HTML code to show the usage of hidden control:

```
<!DOCTYPE html>

<html>
<head>
<title>File Upload Box</title>
</head>
<body>
<form>
<p>Dialog Box</p>
<input type="hidden" name="pagename" value="10" />
<input type="Submit" name="Submit" value="Submit
" />
<input type="Reset" name="Reset" value="Reset" />
</form>
</body>
</html>
```

This will produce following result:



---

A general form of the above tags is look like as:

```
<!DOCTYPE html>
<html>
<head>
<title>Select Box Control</title>
</head>
<body>
<h1><font color="red"><center><b>Application Form</b></center></font></h1><hr>
<form>
Name <INPUT TYPE="FieldType" NAME="Field Text" VALUE="Enter Your Name
Here">
<Br><br><br>Father Name <INPUT TYPE="FieldType" FATHER NAME="Field
Text" VALUE="Enter Your Father Name">
<br><br><br>Mother Name <INPUT TYPE="FieldType" MOTHER NAME="Field
Text" VALUE="Enter Your Mother Name">
<br><br><br>Date of Birth <INPUT TYPE="FieldType" DOB="Field
Text" VALUE="DD/MM/YYYY">
<br><br>Attach File <input type="file" name="fileupload" accept="image/*" /> File size
should be lestt then 50 KB
<br><br>Sex <input type="radio" name="Sex" value="Male."> Male.
<input type="radio" name="Sex" value="Female"> Female
<br><br>
```

```

Choose Subject: <select name="dropdown">
<option value="Hindi" selected>Hindi</option>
<option value="English" selected>English</option>
<option value="Computer Sci." selected>Computer Sci.</option>
<option value="Informatics Prac." selected>Informatics Prac.</option>
<option value="Maths" selected>Maths</option>
<option value="Physics">Physics</option>
</select>

<br><br>Documents Attached<br><input type="checkbox" name="Matric" value="on">
Matric

<br><input type="checkbox" name="Senior Secondary" value="on"> Senior Secondary
<br><input type="checkbox" name="Batchelor" value="on"> Batchelor
<br><input type="checkbox" name="Master Degree." value="on">Master Degree.
<br><input type="checkbox" name="Other" value="on"> Other
<br><br><input type="submit" name="submit" value="Submit" />
<input type="reset" name="reset" value="Reset" />
<input type="button" name="ok" value="OK" />&nbsp;Submit Query

```

The screenshot shows a web browser window with the title 'Application Form'. The form contains the following elements:

- Name:** A text input field with the placeholder text 'Enter Your Name Here'.
- Father Name:** A text input field with the placeholder text 'Enter Your Father Nam'.
- Mother Name:** A text input field with the placeholder text 'Enter Your Mother Nam'.
- Date of Birth:** A text input field with the placeholder text 'DD/MM/YYYY'.
- Attach File:** A file selection button labeled 'Browse...' followed by the text 'File size should be lesst then 50 KB'.
- Sex:** Radio buttons for 'Male' and 'Female'.
- Choose Subject:** A dropdown menu currently showing 'Maths'.
- Documents Attached:** A list of checkboxes for 'Matric', 'Senior Secondary', 'Batchelor', 'Master Degree.', and 'Other'.
- Buttons:** 'Submit', 'Reset', and 'OK' buttons, followed by the text 'Submit Query'.

## CHAPTER-12

### F.A.Q. (HTML)

Q. Is the end tag for other commands simply the begin tag with the added slash?

A. Yes.

Q. Will the tags show up on my page?

A. No. As long as your commands are inside the < and > marks, the tag is used to alter the text, but the actual code is hidden from the viewer.

Q. Do I use capitals or lower case? I've seen people using both.

A. In HTML, the browser doesn't care. However, should you move on to XHTML, they will have to be lower case, so you may as well just use lower case--it can't hurt and can only help.

Q. Must everything have a tag to show up on the page?

A. No. If you just type in text, it'll show up. But it will not have any special look.

Q. What if I forget to add the end tag or forget to add the slash to the end tag command?

A. That's trouble, but easy-to-fix trouble. It will be obvious if you've not placed an end tag when you look at the document in your browser. The entire document will be affected after the point where you forgot the end tag. Just go back into the document, add the slash, and reload the document into the browser.

Q. Do all HTML tags require both a begin and end tag, like above?

A. No. There are exceptions to the rule, such as break <BR> tags and image tags <IMG>, but let's stay on the ones that do require both tags to work for now. Moving along...

#### Open and Close Tags

The majority of HTML tags do require both an open and a close tag (a begin and end tag). Most are very easy to understand because the tag is obvious. Here are a few and what they do to text:

Affect	Code	Code Used	What It Does
Strong	Strong	<strong>Bold</strong>	<b>Bold</b>
Emphasis	em	<em>Italic</em>	<i>Italic</i>
Paragraph	p	<p>Paragraph</p>	Plain

Note: the strong and em tags are normally found inside a paragraph. There is a technical reason for this which we'll cover later.

#### Can I Use Two Tags at Once?

Yes. Just make sure to begin and end both. Like so:

<strong><em>Strong and emphasis</em></strong> gives you ***Bold and Italic***

If you do use multiple tags to alter text, make a point of not getting the end tags out of order. Look at this:

```
<strong><em>Strong and emphasis</strong></em>
```

In terms of format, the example above is not correct. The end tags are out of order in relation to the start tags.

Follow this rule:

Always set the beginning and end tags at the same time, always placing them on the farthest end of the item being affected.

Here, again, is the example above in correct form:

```
<strong><em>Strong and emphasis</em></strong>
```

Notice the strong tags are on the far ends. Next in line are the emphasis tags. Just keep setting commands at the farthest ends each time you add them and you'll stay in good form

## CHAPTER-13

### GLOSSARY

---

#### **Applet**

A small Java application that is downloaded by an ActiveX or Java-enabled web browser. Once it has been downloaded, the applet will run on the user's computer. Common applets include financial calculators and web drawing programs.

#### **Application**

Computer software that performs a task or set of tasks, such as word processing or drawing. Applications are also referred to as programs.

#### **ASCII**

American Standard Code for Information Interchange, an encoding system for converting keyboard characters and instructions into the binary number code that the computer understands.

#### **Bandwidth**

The capacity of a networked connection. Bandwidth determines how much data can be sent along the networked wires. Bandwidth is particularly important for Internet connections, since greater bandwidth also means faster downloads.

#### **Binary code**

The most basic language a computer understands, it is composed of a series of 0s and 1s. The computer interprets the code to form numbers, letters, punctuation marks, and symbols.

#### **Bit**

The smallest piece of computer information, either the number 0 or 1. In short they are called binary digits.

#### **Boot**

To start up a computer. Cold boot means restarting computer after the power is turned off. Warm boot means restarting computer without turning off the power.

#### **Browser**

Software used to navigate the Internet. Google Chrome, Firefox, Netscape Navigator and Microsoft Internet Explorer are today's most popular browsers for accessing the World Wide Web.

#### **Bug**

A malfunction due to an error in the program or a defect in the equipment.

#### **Byte**

Most computers use combinations of eight bits, called bytes, to represent one character of data or instructions. For example, the word **cat** has three characters, and it would be represented by three bytes.

### **Cache**

A small data-memory storage area that a computer can use to instantly re-access data instead of re-reading the data from the original source, such as a hard drive. Browsers use a cache to store web pages so that the user may view them again without reconnecting to the Web.

### **CAD-CAM**

**Computer Aided Drawing - Computer Aided Manufacturing.** The instructions stored in a computer that will be translated to very precise operating instructions to a robot, such as for assembling cars or laser-cutting signage.

### **CD-ROM**

**Compact Disc Read-Only Memory**, an optically read disc designed to hold information such as music, reference materials, or computer software. A single CD-ROM can hold around 640 megabytes of data, enough for several encyclopaedias. Most software programs are now delivered on CD-ROMs.

### **CGI**

**Common Gateway Interface**, a programming standard that allows visitors to fill out form fields on a Web page and have that information interact with a

database, possibly coming back to the user as another Web page. CGI may also refer to Computer-Generated Imaging, the process in which sophisticated computer programs create still and animated graphics, such as special effects for movies.

### **Chat**

Typing text into a message box on a screen to engage in dialogue with one or more people via the Internet or other network.

### **Chip**

A tiny wafer of silicon containing miniature electric circuits that can store millions of bits of information.

### **Client**

A single user of a network application that is operated from a server. A client/server architecture allows many people to use the same data simultaneously. The program's main component (the data) resides on a centralized server, with smaller components (user interface) on each client.

### **Cookie**

A text file sent by a Web server that is stored on the hard drive of a computer and relays back to the Web server things about the user, his or her computer, and/or his or her computer activities.

### **CPU**

Central Processing Unit. The brain of the computer.

### **Cracker**

A person who breaks in to a computer through a network, without authorization and with mischievous or destructive intent.

### **Crash**

A hardware or software problem that causes information to be lost or the computer to malfunction. Sometimes a crash can cause permanent damage to a computer.

### **Cursor**

A moving position-indicator displayed on a computer monitor that shows a computer operator where the next action or operation will take place.

### **Cyberspace**

Slang for internet ie. An international conglomeration of interconnected computer networks. Begun in the late 1960s, it was developed in the 1970s to allow government and university researchers to share information. The Internet is not controlled by any single group or organization. Its original focus was research and communications, but it continues to expand, offering a wide array of resources for business and home users.

### **Database**

A collection of similar information stored in a file, such as a database of addresses. This information may be created and stored in a database management system (DBMS).

### **Debug**

Slang. To find and correct equipment defects or program malfunctions.

### **Default**

The pre-defined configuration of a system or an application. In most programs, the defaults can be changed to reflect personal preferences.

### **Desktop**

The main directory of the user interface. Desktops usually contain icons that represent links to the hard drive, a network (if there is one), and a trash or recycling can for files to be deleted. It can also display icons of frequently used applications, as requested by the user.

### **Desktop publishing**

The production of publication-quality documents using a personal computer in combination with text, graphics, and page layout programs.

### **Directory**

A repository where all files are kept on computer.

### **Disk**

Two distinct types. The names refer to the media inside the container:

A hard disc stores vast amounts of data. It is usually inside the computer but can be a separate peripheral on the outside. Hard discs are made up of several rigid coated metal discs. Currently, hard discs can store 15 to 30 Gb (gigabytes).

A floppy disc, 3.5" square, usually inserted into the computer and can store about 1.4 megabytes of data. The 3.5" square floppies have a very thin, flexible disc inside. There is also an intermediate-sized floppy disc,

trademarked Zip discs, which can store 250 megabytes of data.

### **Disk drive**

The equipment that operates a hard or floppy disc.

### **Domain**

Represents an IP (Internet Protocol) address or set of IP addresses that comprise a domain. The domain name appears in URLs to identify web pages or in email addresses. For example, the email address for the First Lady is `first.lady@whitehouse.gov`, `whitehouse.gov`, being the domain name. Each domain name ends with a suffix that indicates what top level domain it belongs to. These are : `.com` for commercial, `.gov` for government, `.org` for organization, `.edu` for educational institution, `.biz` for business, `.info` for information, `.tv` for television, `.ws` for website. Domain suffixes may also indicate the country in which the domain is registered. No two parties can ever hold the same domain name.

### **Domain name**

The name of a network or computer linked to the Internet. Domains are defined by a common IP address or set of similar IP (Internet Protocol) addresses.

### **Download**

The process of transferring information from a web site (or other remote location on a network) to the computer. It is possible to download a file which include text, image, audio, video and many others.

### **DOS**

Disk Operating System. An operating system designed for early IBM-compatible PCs.

### **Drop-down menu**

A menu window that opens vertically on-screen to display context-related options. Also called pop-up menu or pull-down menu.

### **DSL**

**Digital Subscriber Line**, a method of connecting to the Internet via a phone line. A DSL connection uses copper telephone lines but is able to relay data at much higher speeds than modems and does not interfere with telephone use.

### **DVD**

Digital Video Disc. Similar to a CD-ROM, it stores and plays both audio and video.

### **E-book**

An electronic (usually hand-held) reading device that allows a person to view digitally stored reading materials.

### **Email**

Electronic mail; messages, including memos or letters, sent electronically between networked computers that may be across the office or around the world.

### **Emoticon**

A text-based expression of emotion created from ASCII characters that mimics a facial expression when viewed with your head tilted to the left. Here are some examples:

- Smiling

- Frowning
- Winking
- Crying

### **Encryption**

The process of transmitting scrambled data so that only authorized recipients can unscramble it. For instance, encryption is used to scramble credit card information when purchases are made over the Internet.

### **Ethernet**

A type of network.

### **Ethernet card**

A board inside a computer to which a network cable can be attached.

### **File**

A set of data that is stored in the computer.

### **Firewall**

A set of security programs that protect a computer from outside interference or access via the Internet.

### **Folder**

A structure for containing electronic files. In some operating systems, it is called a directory.

### **Fonts**

Sets of typefaces (or characters) that come in different styles and sizes.

### **Freeware**

Software created by people who are willing to give it away for the satisfaction of sharing or knowing they helped to simplify other people's lives. It may be free-standing software, or it may add functionality to existing software.

### **FTP**

**File Transfer Protocol**, a format and set of rules for transferring files from a host to a remote computer.

### **Gigabyte (GB)**

1024 megabytes. Also called gig.

### **Glitch**

The cause of an unexpected malfunction.

### **Gopher**

An Internet search tool that allows users to access textual information through a series of menus, or if using FTP, through downloads.

### **GUI**

**Graphical User Interface**, a system that simplifies selecting computer commands by enabling the user to point to symbols or illustrations (called icons) on the computer screen with a mouse.

### **Groupware**

Software that allows networked individuals to form groups and collaborate on documents, programs, or databases.

### **Hacker**

A person with technical expertise who experiments with computer systems to determine how to develop additional features. Hackers are occasionally requested by system administrators to try and break into systems via a network to test security. The term hacker is sometimes incorrectly used interchangeably with cracker. A hacker is called a white hat and a cracker a black hat.

**Hard copy**

A paper printout of what you have prepared on the computer.

**Hard drive**

Another name for the hard disc that stores information in a computer.

**Hardware**

The physical and mechanical components of a computer system, such as the electronic circuitry, chips, monitor, disks, disk drives, keyboard, modem, and printer.

**Home page**

The main page of a Web site used to greet visitors, provide information about the site, or to direct the viewer to other pages on the site.

**HTML**

**Hypertext Markup Language**, a standard of text markup conventions used for documents on the World Wide Web. Browsers interpret the codes to give the text structure and formatting (such as bold, blue, or italic).

**HTTP**

**Hypertext Transfer Protocol**, a common system used to request and send HTML documents on the World Wide Web. It is the first portion of all URL addresses on the World Wide Web.

**HTTPS**

**Hypertext Transfer Protocol Secure**, often used in intracompany internet sites. Passwords are required to gain access.

**Hyperlink**

Text or an image that is connected by hypertext coding to a different location. By selecting the text or image with a mouse, the computer jumps to (or displays) the linked text.

**Hypermedia**

Integrates audio, graphics, and/or video through links embedded in the main program.

**Hypertext**

A system for organizing text through links, as opposed to a menu-driven hierarchy such as Gopher. Most Web pages include hypertext links to other pages at that site, or to other sites on the World Wide Web.

**Icons**

Symbols or illustrations appearing on the computer screen that indicate program files or other computer functions.

**Input**

Data that goes into a computer device.

**Input device**

A device, such as a keyboard, stylus and tablet, mouse, puck, or microphone, that allows input of information (letters, numbers, sound, video) to a computer.

**Instant messaging (IM)**

A chat application that allows two or more people to communicate over the Internet via real-time keyed-in messages.

**Interface**

The interconnections that allow a device, a program, or a person to interact. Hardware interfaces are the cables that connect the

device to its power source and to other devices. Software interfaces allow the program to communicate with other programs (such as the operating system), and user interfaces allow the user to communicate with the program (e.g., via mouse, menu commands, icons, voice commands, etc.).

### **Internet**

An international conglomeration of interconnected computer networks. Begun in the late 1960s, it was developed in the 1970s to allow government and university researchers to share information. The Internet is not controlled by any single group or organization. Its original focus was research and communications, but it continues to expand, offering a wide array of resources for business and home users.

### **IP (Internet Protocol) address**

An Internet Protocol address is a unique set of numbers used to locate another computer on a network. The format of an IP address is a 32-bit string of four numbers separated by periods. Each number can be from 0 to 255 (i.e., 1.154.10.266). Within a closed network IP addresses may be assigned at random, however, IP addresses of web servers must be registered to avoid duplicates.

### **Java**

An object-oriented programming language designed specifically for programs (particularly multimedia) to be used over

the Internet. Java allows programmers to create small programs or applications (applets) to enhance Web sites.

### **JavaScript/ECMA script**

A programming language used almost exclusively to manipulate content on a web page. Common JavaScript functions include validating forms on a web page, creating dynamic page navigation menus, and image rollovers.

### **Kilobyte (K or KB)**

Equal to 1,024 bytes.

### **Linux**

A UNIX - like, open-source operating system developed primarily by Linus Torvalds. Linux is free and runs on many platforms, including both PCs and Macintoshes. Linux is an open-source operating system, meaning that the source code of the operating system is freely available to the public. Programmers may redistribute and modify the code, as long as they don't collect royalties on their work or deny access to their code. Since development is not restricted to a single corporation more programmers can debug and improve the source code faster.

### **Laptop and notebook**

Small, lightweight, portable battery-powered computers that can fit onto your lap. They each have a thin, flat, liquid crystal display screen.

### **Macro**

A script that operates a series of commands to perform a function. It is set up to automate repetitive tasks.

### **Mac OS**

An operating system with a graphical user interface, developed by Apple for Macintosh computers. Current System X.1.1(10) combines the traditional Mac interface with a strong underlying UNIX. Operating system for increased performance and stability.

### **Megabyte (MB)**

Equal to 1,048,576 bytes, usually rounded off to one million bytes (also called a meg).

### **Memory**

Temporary storage for information, including applications and documents. The information must be stored to a permanent device, such as a hard disc or CD-ROM before the power is turned off, or the information will be lost. Computer memory is measured in terms of the amount of information it can store, commonly in megabytes or gigabytes.

### **Menu**

A context-related list of options that users can choose from.

### **Menu bar**

The horizontal strip across the top of an application's window. Each word on the strip has a context sensitive drop-down menu containing features and actions that are available for the application in use.

### **Merge**

To combine two or more files into a single file.

### **MHz**

An abbreviation for **Megahertz**, or **one million hertz**. One MHz represents one million clock cycles per second and is the measure of a computer microprocessor's speed. For example, a microprocessor that runs at 300 MHz executes 300 million cycles per second. Each instruction a computer receives takes a fixed number of clock cycles to carry out, therefore the more cycles a computer can execute per second, the faster its programs run. Megahertz is also a unit of measure for bandwidth.

### **Microprocessor**

A complete central processing unit (CPU) contained on a single silicon chip.

### **Minimize**

A term used in a GUI operating system that uses windows. It refers to reducing a window to an icon, or a label at the bottom of the screen, allowing another window to be viewed.

### **Modem**

A device that connects two computers together over a telephone or cable line by converting the computer's data into an audio signal. Modem is a contraction for the process it performs: modulate-demodulate.

### **Monitor**

A video display terminal.

## **Mouse**

A small hand-held device, similar to a trackball, used to control the position of the cursor on the video display; movements of the mouse on a desktop correspond to movements of the cursor on the screen.

## **MP3**

Compact audio and video file format. The small size of the files makes them easy to download and e-mail. Format used in portable playback devices.

## **Multimedia**

Software programs that combine text and graphics with sound, video, and animation. A multimedia PC contains the hardware to support these capabilities.

## **MS-DOS**

An early operating system developed by Microsoft Corporation (Microsoft Disc Operating System).

## **Network**

A system of interconnected computers.

## **Open source**

Computer programs whose original source code was revealed to the general public so that it could be developed openly. Software licensed as open source can be freely changed or adapted to new uses, meaning that the source code of the operating system is freely available to the public. Programmers may redistribute and modify the code, as long as they don't collect royalties on their work or deny access to

their code. Since development is not restricted to a single corporation more programmers can debug and improve the source code faster.

## **Operating system**

A set of instructions that tell a computer on how to operate when it is turned on. It sets up a filing system to store files and tells the computer how to display information on a video display. Most PC operating systems are DOS (disc operated system) systems, meaning the instructions are stored on a disc (as opposed to being originally stored in the microprocessors of the computer). Other well-known operating systems include UNIX, Linux, Macintosh, and Windows.

## **Output**

Data that come out of a computer device. For example, information displayed on the monitor, sound from the speakers, and information printed to paper.

## **Palm**

A hand-held computer.

## **PC**

Personal computer. Generally refers to computers running Windows with a Pentium processor.

## **PC board**

**Printed Circuit board**, a board printed or etched with a circuit and processors. Power supplies, information storage devices, or changers are attached.

**PDA**

**Personal Digital Assistant**, a hand-held computer that can store daily appointments, phone numbers, addresses, and other important information. Most PDAs link to a desktop or laptop computer to download or upload information.

**PDF**

**Portable Document Format**, a format presented by Adobe Acrobat that allows documents to be shared over a variety of operating systems. Documents can contain words and pictures and be formatted to have electronic links to other parts of the document or to places on the web.

**Pentium chip**

Intel's fifth generation of sophisticated high-speed microprocessors. Pentium means the fifth element.

**Peripheral**

Any external device attached to a computer to enhance operation. Examples include external hard drive, scanner, printer, speakers, keyboard, mouse, trackball, stylus and tablet, and joystick.

**Personal computer (PC)**

A single-user computer containing a central processing unit (CPU) and one or more memory circuits.

**Petabyte**

A measure of memory or storage capacity and is approximately a thousand terabytes.

**Petaflop**

A theoretical measure of a computer's speed and can be expressed as a thousand-trillion floating-point operations per second.

**Platform**

The operating system, such as UNIX, Macintosh, Windows, on which a computer is based.

**Plug and play**

Computer hardware or peripherals that come set up with necessary software so that when attached to a computer, they are recognized by the computer and are ready to use.

**Pop-up menu**

A menu window that opens vertically or horizontally on-screen to display context-related options. Also called drop-down menu or pull-down menu.

**Power PC**

A competitor of the Pentium chip. It is a new generation of powerful sophisticated microprocessors produced from an Apple-IBM-Motorola alliance.

**Printer**

A mechanical device for printing a computer's output on paper. There are three major types of printer:

- **Dot matrix** - creates individual letters, made up of a series of tiny ink dots, by punching a ribbon with the ends of tiny wires. (This type of printer is most often used in

industrial settings, such as direct mail for labelling.)

- **Ink jet** - sprays tiny droplets of ink particles onto paper.
- **Laser** - uses a beam of light to reproduce the image of each page using a magnetic charge that attracts dry toner that is transferred to paper and sealed with heat.

### **Program**

A precise series of instructions written in a computer language that tells the computer what to do and how to do it. Programs are also called software or applications.

### **Programming language**

A series of instructions written by a programmer according to a given set of rules or conventions (syntax). High-level programming languages are independent of the device on which the application (or program) will eventually run; low-level languages are specific to each program or platform. Programming language instructions are converted into programs in language specific to a particular machine or operating system (machine language). So that the computer can interpret and carry out the instructions. Some common programming languages are BASIC, C, C++, dBASE, FORTRAN, and Perl.

### **Puck**

An input device, like a mouse. It has a magnifying glass with crosshairs on the

front of it that allows the operator to position it precisely when tracing a drawing for use with CAD-CAM software.

### **Pull-down menu**

A menu window that opens vertically on-screen to display context-related options. Also called drop-down menu or pop-up menu.

### **Push technology**

Internet tool that delivers specific information directly to a user's desktop, eliminating the need to surf for it. PointCast, which delivers news in user-defined categories, is a popular example of this technology.

### **QuickTime**

Audio-visual software that allows movie-delivery via the Internet and e-mail. QuickTime images are viewed on a monitor.

### **RAID**

**Redundant Array of Inexpensive Disks**, a method of spreading information across several disks set up to act as a unit, using two different techniques:

- **Disk striping** - storing a bit of information across several discs (instead of storing it all on one disc and hoping that the disc doesn't crash).
- **Disk mirroring** - simultaneously storing a copy of information on another disc so that the information

can be recovered if the main disc crashes.

## **RAM**

**Random Access Memory**, one of two basic types of memory. Portions of programs are stored in RAM when the program is launched so that the program will run faster. Though a PC has a fixed amount of RAM, only portions of it will be accessed by the computer at any given time. Also called memory.

## **Right-click**

Using the right mouse button to open context-sensitive drop-down menus.

## **ROM**

**Read-Only Memory**, one of two basic types of memory. ROM contains only permanent information put there by the manufacturer. Information in ROM cannot be altered, nor can the memory be dynamically allocated by the computer or its operator.

## **Scanner**

An electronic device that uses light-sensing equipment to scan paper images such as text, photos, and illustrations and translate the images into signals that the computer can then store, modify, or distribute.

## **Search engine**

Software that makes it possible to look for and retrieve material on the Internet, particularly the Web. Some popular search

engines are Alta Vista, Google, HotBot, Yahoo!, Web Crawler, and Lycos.

## **Server**

A computer that shares its resources and information with other computers, called clients, on a network.

## **Shareware**

Software created by people who are willing to sell it at low cost or no cost for the gratification of sharing. It may be freestanding software, or it may add functionality to existing software.

## **Software**

Computer programs; also called applications.

## **Spider**

A process search engines use to investigate new pages on a web site and collect the information that needs to be put in their indices.

## **Spreadsheet**

Software that allows one to calculate numbers in a format that is similar to pages in a conventional ledger.

## **Storage**

Devices used to store massive amounts of information so that it can be readily retrieved. Devices include RAIDs, CD-ROMs, DVDs.

## **Streaming**

Taking packets of information (sound or visual) from the Internet and storing it in

temporary files to allow it to play in continuous flow.

### **Stylus and tablet**

An input device similar to a mouse. The stylus is pen shaped. It is used to draw on a tablet (like drawing on paper) and the tablet transfers the information to the computer. The tablet responds to pressure. The firmer the pressure used to draw, the thicker the line appears.

### **Surfing**

Exploring the Internet.

### **Surge protector**

A controller to protect the computer and make up for variances in voltage.

### **Telnet**

A way to communicate with a remote computer over a network.

### **Trackball**

Input device that controls the position of the cursor on the screen; the unit is mounted near the keyboard, and movement is controlled by moving a ball.

### **Terabytes (TB)**

A thousand gigabytes.

### **Teraflop**

A measure of a computer's speed. It can be expressed as a trillion floating-point operations per second.

### **UNIX**

A very powerful operating system used as the basis of many high-end computer applications.

### **Upload**

The process of transferring information from a computer to a web site (or other remote location on a network). To transfer information from a computer to a web site (or other remote location on a network).

### **URL**

Uniform Resource Locator.

- The protocol for identifying a document on the Web.
- A Web address (e.g., www.spoin.com). A URL is unique to each user. See also domain.

### **UPS**

Universal Power Supply or Uninterruptible Power Supply. An electrical power supply that includes a battery to provide enough power to a computer during an outage to back-up data and properly shut down.

### **USB**

A multiple-socket USB connector that allows several USB-compatible devices to be connected to a computer.

### **USENET**

A large unmoderated and unedited bulletin board on the Internet that offers thousands of forums, called newsgroups. These range from newsgroups exchanging information on scientific advances to celebrity fan clubs.

### **User friendly**

A program or device whose use is intuitive to people with a non-technical background.

### **Video conferencing**

A remote "face-to-face chat," when two or more people using a webcam and an Internet telephone connection chat online. The webcam enables both live voice and video.

### **Virtual reality (VR)**

A technology that allows one to experience and interact with images in a simulated three-dimensional environment. For example, you could design a room in a house on your computer and actually feel that you are walking around in it even though it was never built. (The Holodeck in the science-fiction TV series Star Trek : Voyager would be the ultimate virtual reality.) Current technology requires the user to wear a special helmet, viewing goggles, gloves, and other equipment that transmits and receives information from the computer.

### **Virus**

An unauthorized piece of computer code attached to a computer program or portions of a computer system that secretly copies itself from one computer to another by shared discs and over telephone and cable lines. It can destroy information stored on the computer, and in extreme cases, can destroy operability. Computers can be protected from viruses if the operator utilizes good virus prevention software and keeps the virus definitions up to date. Most viruses are not programmed to spread themselves. They have to be sent to another

computer by e-mail, sharing, or applications. The worm is an exception, because it is programmed to replicate itself by sending copies to other computers listed in the e-mail address book in the computer. There are many kinds of viruses, Macro viruses copy their macros to templates and/or other application document files.

### **WAV**

A sound format (pronounced wave) used to reproduce sounds on a computer.

### **Webcam**

A video camera/computer setup that takes live images and sends them to a Web browser.

### **Window**

A portion of a computer display used in a graphical interface that enables users to select commands by pointing to illustrations or symbols with a mouse. "Windows" is also the name Microsoft adopted for its popular operating system.

### **World Wide Web**

A network of servers on the Internet that use hypertext-linked databases and files. It was developed in 1989 by Tim Berners-Lee, a British computer scientist, and is now the primary platform of the Internet. The feature that distinguishes the Web from other Internet applications is its ability to display graphics in addition to text.

### **Word processor**

A computer system or program for setting, editing, revising, correcting, storing, and printing text

### **WYSIWYG**

What You See Is What You Get. When using most word processors, page layout programs (See desktop publishing), and web page design programs, words and images will be displayed on the monitor as they will look on the printed page or web pag

## INTERNET TERMINOLOGY

### 1 Address Box

A narrow, rectangular box in the browser window where you can type in a web address. Typing in the web address in the address box and hitting Enter on the keyboard will take you to a website.

### 2 Back Arrow

This arrow, often green, is found at the top of most browsers. When you click on the back arrow, it takes you back – in order – through all of the web pages you’ve seen. (Sometimes called the back button.)

### Browse

To explore a website or a number of websites by scanning and reading information.

### 4 Browser

Software, such as Microsoft Internet Explorer, used to find information on the Web. The most visible part of a browser sits at the top of the computer screen, above the web page.

### 5 Button

Small box that looks like it’s being depressed when you select it. Buttons can turn on (and turn off) many types of functions on the Internet.

### 6 Clicking

Pressing and releasing a button on a mouse to select or activate the area on the screen where the cursor is pointing. Usually, you click on the left side of the


mouse (called a left click). For more advanced functions, you click on the right side of the mouse (called a right click).



### 7 Computer or CPU (central processing unit)

The main part, or “brains” of a computer. The CPU interprets and carries out program instructions.

### 8 Cursor

A small image on the screen indicating where you are pointing; the mouse controls the movements of the cursor. The cursor can appear in different forms, including:

 An arrow, which indicates where you are positioned on the screen. I  
An I-beam, often blinking, which marks a place on the screen where you can enter or select text.

A pointing  hand, which indicates that you are  hovering over a link. (See **Link**.) An hourglass, which indicates that the computer is doing a task. You must wait until it disappears before you can proceed.

### 9 desktop Screen

The information that appears on the computer soon after the computer is turned on. The desktop contains a number of icons, or images, that you can click on to start programs.

## **10** dialog Box

A special box that appears when the computer needs additional information in order to carry out a task. (You must click on “ok” to have a dialog box disappear.)

## **11** drop down List

A list of items from which you can make selections.

- A. When you first see a box containing a drop down list, the box will be empty or may display only a single item.
- B. To see a list of choices, left click on the arrow in the box and hold. The list of choices will display above or below the box.
- C. Keeping your left index finger pressed on the mouse, move the cursor to the desired choice (In this case, a quantity of 3 booklets).
- D. Release your left index finger from the mouse, and your selection will appear in the box. The full list of choices will disappear.

## **FAQs**

Questions and answers that appear on many websites.

## **1** Forward Arrow

This arrow is present at the top of most browsers. When you click on the forward arrow, it takes you forward to a page you just left and is opposite in direction from pages the back arrow takes you through.

When the arrow

## **14** Go Online

To go on the Internet.

## **15** Hardware

The physical parts of a computer system.

## **16** Home page

The first thing you see when you come to a website, or the opening page of a website. It provides information about the site and directs you to other pages on the site.

## **17** Icon

A small picture or image representing a command (such as print), a file, or a program. When you click on an icon, you start a command, open a file, or launch a program.

## **18** The Internet

A vast, international collection of computer networks that transfers information. A combination of the words international and network. Websites and e-mail are part of the Internet.

## **19** Keyboard

The keys that operate the computer, very much like a typewriter, with extra keys for special functions.

## **20** Link (or hyperlink)

A highlighted or underlined feature on a web page that, when clicked, will take you to another web page. A link most often appears as underlined words or an image.

One sure way to tell if something is a link or not: Whenever your cursor turns into a pointing hand, the image or word you are pointing to is a link.

### 1 Log On

To gain access to a computer system or to a page on a website by entering a password or user ID.

### 2 menu

A list of options, or topics, on a website that users can choose from.

### 3 monitor

The part of a computer system that contains the computer screen, where information is displayed.

### 4 mouse

A small hand-held device that controls the position of the cursor on the computer screen. Movements of the mouse correspond to movements of the cursor. (See **Cursor**.)

### 5 mouse pad

The pad on which you move the mouse.

### 6 Navigate

To move through a website or through various websites.

### 7 scroll

To move text or other information on a computer screen up, down, or sideways, with new information appearing as the old disappears.

### 8 scroll Bar

A narrow, rectangular bar on the right edge and bottom edge of a web page that lets you move the page to see more of the information it contains. The scroll bar on the right

moves the web page up and down, and

### 9 search Box

A small rectangular blank space on a web page where you can type in a word or phrase to look for information. Clicking on the button next to the search box (or hitting the Enter key on the keyboard) will take you to a next word or phrase.

### 1 site map

A list of the contents on a website, similar to an index in a book. A link to the site map is usually found at the top or bottom of the home page.

### 2 software

The instructions that tell the computer and computer networks what to do. Software is installed inside the computer.

### 3 speakers

Devices that allow you to hear sound from the computer.

### Surf the Net

To explore various websites on the Internet.

### 4 Web Address or urL

The address for a website. (URL stands for Uniform Resource Locator.) U.S.-based web addresses usually start with the letters www (for World Wide Web) and end with a dot followed by letters that indicate the type of website it is:

On the Internet, you get to a website by typing in the web address (or URL) into the address box of the browser. For

**.com** = commercial enterprise or business

**.org** = non-profit organization

**.edu** = educational institution

**.gov** = government agency

**.mil** = military agency

**.net** = another ending for a commercial website

example, to get to the website of the National Institute on Aging (NIA), a Federal agency, you would type [www.nia.nih.gov](http://www.nia.nih.gov) in the address box.

## 5 Website

A location on the World Wide Web (and Internet) that contains information about a specific topic. A website usually contains multiple pages with different types of information about the topic.

## 6 Window

A framed area of a computer screen that appears in front of the web page. Sometimes the appearance of a window means that you have entered another website. At other times, it means you may still be on the same website.

## 7 The World Wide Web

Also known as the Web, it is a system that lets you access information on the Internet. People often use the term *Web* to refer to the *Internet*, but they are not exactly the same thing. The World Wide

Web operates *over* the Internet, and it is the most widely used part of the Internet.

**DR. SUNIL VATS**

