**What is CSS?**

**C**ascading **S**tyle **S**heets, fondly referred to as CSS,

CSS handles the look and feel part of a web page.

We can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, as well as a variety of other effects.

**Advantages of CSS**

**CSS saves time** - You can write CSS once and then reuse the same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many web pages as you want.

**Pages load faster** - If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So, less code means faster download times.

**Easy maintenance** - To make a global change, simply change the style, and all the elements in all the web pages will be updated automatically.

**Superior styles to HTML** - CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.

**Multiple Device Compatibility** - Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, different versions of a website can be presented for handheld devices such as PDAs and cell phones or for printing.

**Global web standards** – Now HTML attributes are being deprecated and it is being recommended to use CSS. So it’s a good idea to start using CSS in all the HTML pages to make them compatible with future browsers.

**CSS ─ Syntax**

A style rule is made of three parts:

**Selector:** A selector is an HTML tag at which a style will

be applied, like <h1> or <table> etc.

**Property:** A property is a type of attribute of HTML tag.

Like *color*, *border*, etc.

**Value:** Values are assigned to properties.

For example, *color* property can have the value either *red* or

*#F1F1F1* etc.

You can put CSS Style Rule Syntax as follows:

selector { property: value }

**Example:** You can define a table border as follows:

table{ border :1px solid #C00; }

to apply styles we have to add style tag in head tag

ex: <html>

<head>

<style type=”text/css”>

p

{

Background-color:lightgray;

Color:blue;

Text-align:right;

Border:4px groove green;

}

</style>

</head>

<body>

<p>it is para1</p>

<p> it is para2</p>

<p> it is para3</p>

</body>

</html>

**Types of selectors**

**The Type Selectors**

h1 {

color: #36CFFF;

}

**The Universal Selectors**

Rather than selecting elements of a specific type, the universal selector quite simply

matches the name of any element type:

\* {

color: #000000;

}

This rule renders the content of every element in our document in black.

**The Descendant Selectors**

To apply a style rule to a particular element only when it lies inside a

particular element.

in the following example, the style rule will apply to <em>

element only when it lies inside the <ul> tag.

ul em {

color: #000000;

}

**The Class Selectors**

You can define style rules based on the class attribute of the elements. All the elements

having that class will be formatted according to the defined rule.

.black {

color: #000000;

}

This rule renders the content in black for every element with class attribute set

to *black* in our document. You can make it a bit more particular. For example:

h1.black {

color: #000000;

}

This rule renders the content in black for only <h1> elements with class attribute set to

*black*.

You can apply more than one class selectors to a given element. Consider the following

example:

<p class="center bold">

This para will be styled by the classes center and bold.

</p>

**The ID Selectors**

You can define style rules based on the *id* attribute of the elements. All the elements

having that *id* will be formatted according to the defined rule.

#black {

color: #000000;

}

This rule renders the content in black for every element with *id* attribute set to *black* in

our document. You can make it a bit more particular. For example:

h1#black {

color: #000000;

}

This rule renders the content in black for only <h1> elements with *id* attribute set

to *black*.

The true power of *id* selectors is when they are used as the foundation for descendant

selectors. For example:

#black h2 {

color: #000000;

}

In this example, all level 2 headings will be displayed in black color when those headings

will lie within tags having *id* attribute set to *black*.

**The Child Selectors:** To apply a set of styles rules when if a tag is direct child of another tag

You have seen the descendant selectors. There is one more type of selector, which is

very similar to descendants but have different functionality. Consider the following

example:

body > p {

color: #000000;

}

This rule will render all the paragraphs in black if they are a direct child of the <body>

element. Other paragraphs put inside other elements like <div> or <td> would not have

any effect of this rule.

**The Attribute Selectors**

You can also apply styles to HTML elements with particular attributes. The style rule

below will match all the input elements having a type attribute with a value of *text*:

input[type="text"]{

color: #000000;

}

The advantage to this method is that the <input type="submit" /> element is

unaffected, and the color applied only to the desired text fields.

There are following rules applied to attribute selector.

**p[lang]** - Selects all paragraph elements with a *lang* attribute.

**p[lang="fr"]** - Selects all paragraph elements whose *lang* attribute has a value

of exactly "fr".

**p[lang~="fr"]** - Selects all paragraph elements whose *lang* attribute contains

the word "fr".

**p[lang|="en"]** - Selects all paragraph elements whose *lang* attribute contains

values that are exactly "en", or begin with "en-".

**Multiple Style Rules**

You may need to define multiple style rules for a single element. You can define these

rules to combine multiple properties and corresponding values into a single block as

defined in the following example:

h1 {

color: #36C;

font-weight: normal;

letter-spacing: .4em;

margin-bottom: 1em;

text-transform: lowercase;

}

Here all the property and value pairs are separated by a **semicolon (;)**. You can keep

them in a single line or multiple lines. For better readability, we keep them in separate

lines.

For a while, don't bother about the properties mentioned in the above block. These

properties will be explained in the coming chapters and you can find the complete detail

about properties in CSS References.

**Grouping Selectors**

You can apply a style to many selectors if you like. Just separate the selectors with a

comma, as given in the following example:

h1, h2, h3 {

color: #36C;

font-weight: normal;

letter-spacing: .4em;

margin-bottom: 1em;

text-transform: lowercase;

}

This define style rule will be applicable to h1, h2 and h3 element as well. The order of

the list is irrelevant. All the elements in the selector will have the corresponding

declarations applied to them.

You can combine the various *class* selectors together as shown below:

#content, #footer, #supplement {

position: absolute;

left: 510px;

width: 200px;

}

**3. CSS ─ Inclusion**

There are four ways to associate styles with your HTML document. Most commonly used

methods are inline CSS and External CSS.

**Embedded CSS - The <style> Element**

You can put your CSS rules into an HTML document using the <style> element. This tag

is placed inside the <head>...</head> tags. Rules defined using this syntax will be

applied to all the elements available in the document. Here is the generic syntax:

<head>

<style type="text/css" media="...">

Style Rules

............

</style>

</head>

Attributes associated with <style> elements are:

****

**Example**

Following is an example of embed CSS based on the above syntax:

<head>

<style type="text/css" media="all">

h1{

color: #36C;

}

</style>

</head>

**Inline CSS - The *style* Attribute**

You can use *style* attribute of any HTML element to define style rules. These rules will be

applied to that element only. Here is the generic syntax:

<element style="...style rules....">

**Attributes**

**Attribute Value Description**

style style

rules

The value of *style* attribute is a combination of style

declarations separated by semicolon (;).

**Example**

Following is the example of inline CSS based on the above syntax:

<h1 style ="color:#36C;"> This is inline CSS </h1>

**External CSS - The <link> Element**

The <link> element can be used to include an external stylesheet file in your HTML

document.

An external style sheet is a separate text file with **.css** extension. You define all the Style

rules within this text file and then you can include this file in any HTML document using

<link> element.

Here is the generic syntax of including external CSS file:

<head>

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

</head>

**Attributes**

Attributes associated with <style> elements are:

**Attribute Value Description**

type text/css Specifies the style sheet language as a content-type (MIME

type). This attribute is required.

href URL Specifies the style sheet file having Style rules. This attribute

is a required.

media screen

tty

tv

projection

handheld

print

braille

aural

all

Specifies the device the document will be displayed on.

Default value is *all*. This is an optional attribute.

**Example**

Consider a simple style sheet file with a name *mystyle.css* having the following rules:

h1, h2, h3 {

color: #36C;

font-weight: normal;

letter-spacing: .4em;

margin-bottom: 1em;

text-transform: lowercase;

}

Now you can include this file *mystyle.css* in any HTML document as follows:

<head>

<link type="text/css" href="mystyle.css" media="all" />

</head>

**Imported CSS - @import Rule**

@import is used to import an external stylesheet in a manner similar to the <link>

element. Here is the generic syntax of @import rule.

<head>

<@import "URL";

</head>

Here URL is the URL of the style sheet file having style rules. You can use another syntax

as well:

<head>

<@import url("URL");

</head>

**Example**

Following is the example showing you how to import a style sheet file into an HTML

document:

<head>

@import "mystyle.css";

</head>

**CSS Rules Overriding**

We have discussed four ways to include style sheet rules in an HTML document. Here is

the rule to override any Style Sheet Rule.

Any inline style sheet takes the highest priority. So, it will override any rule

defined in <style>...</style> tags or the rules defined in any external style sheet

file.

Any rule defined in <style>...</style> tags will override the rules defined in any

external style sheet file.

Any rule defined in the external style sheet file takes the lowest priority, and the

rules defined in this file will be applied only when the above two rules are not

applicable.

**Handling Old Browsers**

There are still many old browsers who do not support CSS. So, we should take care

while writing our Embedded CSS in an HTML document. The following snippet shows how

to use comment tags to hide CSS from older browsers:

<style type="text/css">

<!--

body, td {

color: blue;

}

-->

</style>

**CSS Comments**

Many times, you may need to put additional comments in your style sheet blocks. So, it

is very easy to comment any part in the style sheet. You can simply put your comments

inside /\*.....this is a comment in style sheet.....\*/.

You can use /\* ....\*/ to comment multi-line blocks in similar way you do in C and C++

programming languages.

**Example**

/\* This is an external style sheet file \*/

h1, h2, h3 {

color: #36C;

font-weight: normal;

letter-spacing: .4em;

margin-bottom: 1em;

text-transform: lowercase;

}

/\* end of style rules. \*/

**CSS ─ Measurement Units**

CSS supports a number of measurements including absolute units

such as inches, centimeters, points, and so on, as well as relative measures such as

percentages and em units. You need these values while specifying various

measurements in your Style rules e.g. **border="1px solid red"**.

We have listed out all the CSS Measurement Units along with proper Examples:

****

**CSS ─ Colors**

****

**CSS ─ Background**

The **background-color** property is used to set the background color of an

element.

The **background-image** property is used to set the background image of an

element.

The **background-repeat** property is used to control the repetition of an image in

the background.

The **background-position** property is used to control the position of an image in

the background.

The **background-attachment** property is used to control the scrolling of an

image in the background.

The **background** property is used as a shorthand to specify a number of other

background properties.

**Set the Background Color**

Following is the example, which demonstrates how to set the background color for an

element.

<p style="background-color:yellow;">

This text has a yellow background color.

</p>

**Set the Background Image**

<table style="background-image:url(/images/pattern1.gif);">

<tr><td>

This table has background image set.

</td></tr>

</table>

**Repeat the Background Image**

<table style="background-image:url(/images/pattern1.gif);

background-repeat: repeat;">

<tr><td>

This table has background image which repeats multiple times.

</td></tr>

</table>

**The following example which demonstrates how to repeat the background image**

**vertically**.

<table style="background-image:url(/images/pattern1.gif);

background-repeat: repeat-y;">

<tr><td>

This table has background image set which will repeat vertically.

</td></tr>

</table>

**The following example demonstrates how to repeat the background image horizontally.**

<table style="background-image:url(/images/pattern1.gif);

background-repeat: repeat-x;">

<tr><td>

This table has background image set which will repeat horizontally.

</td></tr>

</table>

**Set the Background Image Position**

The following example demonstrates how to set the background image position 100

pixels away from the left side.

<table style="background-image:url(/images/pattern1.gif);

background-position:100px;">

<tr><td>

Background image positioned 100 pixels away from the left.

</td></tr>

</table>

**The following example demonstrates how to set the background image position 100pixels away from the left side and 200 pixels down from the top.**

<table style="background-image:url(/images/pattern1.gif);

background-position:100px 200px;">

<tr><td>

This table has background image positioned 100

pixels away from the left and 200 pixels from the top.

</td></tr>

</table>

**Set the Background Attachment**

Background attachment determines whether a background image is fixed or scrolls with

the rest of the page.

The following example demonstrates how to set the fixed background image.

<p style="background-image:url(/images/pattern1.gif);

background-attachment:fixed;">

This parapgraph has fixed background image.

</p>

The following example demonstrates how to set the scrolling background image.

<p style="background-image:url(/images/pattern1.gif);

background-attachment:scroll;">

This parapgraph has scrolling background image.

</p>

**Shorthand Property**

You can use the *background* property to set all the background properties at once. For

example:

<p style="background:url(/images/pattern1.gif) repeat fixed;">

This parapgraph has fixed repeated background image.

</p>

**CSS ─ Fonts**

The **font-family** property is used to change the face of a font.

The **font-style** property is used to make a font italic or oblique.

The **font-variant** property is used to create a small-caps effect.

The **font-weight** property is used to increase or decrease how bold or light a font

appears.

The **font-size** property is used to increase or decrease the size of a font.

The **font** property is used as shorthand to specify a number of other font

properties.

**Set the Font Family**

Following is the example, which demonstrates how to set the font family of an element.

Possible value could be any font family name.

<p style="font-family:georgia,garamond,serif;">

This text is rendered in either georgia, garamond, or the default

serif font depending on which font you have at your system.

</p>

**Set the Font Style**

The following example demonstrates how to set the font style of an element. Possible

values are *normal, italic and oblique*.

<p style="font-style:italic;">

This text will be rendered in italic style

</p>

**Set the Font Variant**

The following example demonstrates how to set the font variant of an element. Possible

values are *normal and small-caps*.

<p style="font-variant:small-caps;">

This text will be rendered as small caps

</p>

**Set the Font Weight**

The following example demonstrates how to set the font weight of an element. The fontweight

property provides the functionality to specify how bold a font is. Possible values

could be *normal, bold, bolder, lighter, 100, 200, 300, 400, 500, 600, 700, 800, 900*.

<p style="font-weight:bold;">

This font is bold.

</p>

<p style="font-weight:bolder;">

This font is bolder.

</p>

<p style="font-weight:900;">

This font is 900 weight.

</p>

**Set the Font Size**

The following example demonstrates how to set the font size of an element. The fontsize

property is used to control the size of fonts. Possible values could be *xx-small, xsmall,*

*small, medium, large, x-large, xx-large, smaller, larger, size in pixels or in %.*

<p style="font-size:20px;">

This font size is 20 pixels

</p>

<p style="font-size:small;">

This font size is small

</p>

<p style="font-size:large;">

This font size is large

</p>

to set the font size adjust of an element

<p style="font-size-adjust:0.61;">

This text is using a font-size-adjust value.

</p>

**Set the Font Stretch**

The following example demonstrates how to set the font stretch of an element. This

property relies on the user's computer to have an expanded or condensed version of the

font being used.

Possible values could be *normal, wider, narrower, ultra-condensed, extra-condensed,*

*condensed, semi-condensed, semi-expanded, expanded, extra-expanded, ultraexpanded*.

<p style="font-stretch:ultra-expanded;">

If this doesn't appear to work, it is likely that

your computer doesn't have a condensed or expanded

version of the font being used.

</p>

**Shorthand Property**

You can use the *font* property to set all the font properties at once. For example:

<p style="font:italic small-caps bold 15px georgia;">

Applying all the properties on the text at once.

</p>

**CSS ─ Text**

The **color** property is used to set the color of a text.

The **direction** property is used to set the text direction.

The **letter-spacing** property is used to add or subtract space between the letters

that make up a word.

The **word-spacing** property is used to add or subtract space between the words

of a sentence.

The **text-indent** property is used to indent the text of a paragraph.

The **text-align** property is used to align the text of a document.

The **text-decoration** property is used to underline, overline, and strikethrough

text.

The **text-transform** property is used to capitalize text or convert text to

uppercase or lowercase letters.

The **white-space** property is used to control the flow and formatting of text.

The **text-shadow** property is used to set the text shadow around a text.

**Set the Text Color**

The following example demonstrates how to set the text color. Possible value could be

any color name in any valid format.

<p style="color:red;">

This text will be written in red.

</p>

**Set the Text Direction**

EX:

<p style="direction:rtl;">

This text will be renedered from right to left

</p>

**seting the space between characters**

<p style="letter-spacing:5px;">

This text is having space between letters.

</p>

It will produce the following result:

T h i s t e x t i s h a v i n g s p a c e b e t w e e n l e t t e r s .

**Space between Words**

The following example demonstrates how to set the space between words. Possible

values are *normal or a number specifying space.*

<p style="word-spacing:5px;">

This text is having space between words.

</p>

**Set the Text Indent**

The following example demonstrates how to indent the first line of a paragraph. Possible

values are *% or a number specifying indent space.*

<p style="text-indent:1cm;">

This text will have first line indented by 1cm

and this line will remain at its actual position

this is done by CSS text-indent property.

</p>

**Text Alignment**

The following example demonstrates how to align a text. Possible values are *left, right,*

*center, justify.*

<p style="text-align:right;">

This will be right aligned.

</p>

<p style="text-align:center;">

This will be center aligned.

</p>

<p style="text-align:left;">

This will be left aligned.

</p>

**Decorating the Text**

The following example demonstrates how to decorate a text. Possible values are *none,*

*underline, overline, line-through, blink.*

<p style="text-decoration:underline;">

This will be underlined

</p>

<p style="text-decoration:line-through;">

This will be striked through.

</p>

<p style="text-decoration:overline;">

This will have an over line.

</p>

<p style="text-decoration:blink;">

This text will have blinking effect

</p>

**Set the Text Cases**

The following example demonstrates how to set the cases for a text. Possible values

are *none, capitalize, uppercase, lowercase.*

<p style="text-transform:capitalize;">

This will be capitalized

</p>

<p style="text-transform:uppercase;">

This will be in uppercase

</p>

<p style="text-transform:lowercase;">

This will be in lowercase

</p>

**Set the White Space between Text**

The following example demonstrates how white space inside an element is handled.

Possible values are *normal, pre, nowrap*.

<p style="white-space:pre;">This text has a line break

and the white-space pre setting tells the browser to honor it

just like the HTML pre tag.</p>

It will produce the following result:

This text has a line break

and the white-space pre setting tells the browser to honor it

just like the HTML pre tag.

**Set the Text Shadow**

The following example demonstrates how to set the shadow around a text. This may not

be supported by all the browsers.

<p style="text-shadow:4px 4px 8px blue;">

If your browser supports the CSS text-shadow property,

this text will have a blue shadow.</p>

**CSS ─ Images**

The **border** property is used to set the width of an image border.

The **height** property is used to set the height of an image.

The **width** property is used to set the width of an image.

The **opacity** property is used to set the opacity of an image.

**The Image Border Property**

The *border* property of an image is used to set the width of an image border. This

property can have a value in length or in %.

A width of zero pixels means no border.

Here is an example:

<img style="border:0px;" src="/images/css.gif" />

<br />

<img style="border:3px dashed red;" src="/images/css.gif" />

**Image height property**

<img style="border:1px solid red; height:100px;"

src="/images/css.gif" />

<br />

<img style="border:1px solid red; height:50%;"

src="/images/css.gif" />

**Image height property**

<img style="border:1px solid red; width:100px;"

src="/images/css.gif" />

<br />

<img style="border:1px solid red; width:100%;"

src="/images/css.gif" />

**The opacity Property**

The *opacity* property of an image is used to set the opacity of an image. This

property is used to create a transparent image.

can be a value from 0.0 - 1.0. A lower value makes the element more transparent A lower value makes the element more transparent.

EX

<img style="border:1px solid red;opacity:0.4" src="/images/css.gif" />

**CSS ─ Links**

The **:link** signifies unvisited hyperlinks.

The **:visited** signifies visited hyperlinks.

The **:hover** signifies an element that currently has the user's mouse pointer

hovering over it.

The **:active** signifies an element on which the user is currently clicking.

Usually, all these properties are kept in the header part of the HTML document.

Remember a:hover MUST come after a:link and a:visited in the CSS definition in order

to be effective. Also, a:active MUST come after a:hover in the CSS definition as follows:

<style type="text/css">

a:link {color: #000000}

a:visited {color: #006600}

a:hover {color: #FFCC00}

a:active {color: #FF00CC}

</style>

Now, we will see how to use these properties to give different effects to hyperlinks.

**Set the Color of Links**

The following example demonstrates how to set the link color. Possible values could be

any color name in any valid format.

<style type="text/css">

a:link {color:#000000}

</style>

<a href="/html/index.htm">Black Link</a>

**Set the Color of Visited Links**

The following example demonstrates how to set the color of the visited links. Possible

values could be any color name in any valid format.

<style type="text/css">

a:visited {color: #006600}

</style>

<a href="/html/index.htm">Click this link</a>

It will produce the following link. Once you click this link, it will change its color to green.

Click this link

**Change the Color of Links when Mouse is Over**

The following example demonstrates how to change the color of links when we bring a

mouse pointer over that link. Possible values could be any color name in any valid

format.

<style type="text/css">

a:hover {color: #FFCC00}

</style>

<a href="/html/index.htm">Bring Mouse Here</a>

It will produce the following link. Now, you bring your mouse over this link and you will

see that it changes its color to yellow.

Bring Mouse Here

**Change the Color of Active Links**

The following example demonstrates how to change the color of active links. Possible

values could be any color name in any valid format.

<style type="text/css">

a:active {color: #FF00CC}

</style>

<a href="/html/index.htm">Click This Link</a>

**CSS ─ Tables**

The **border-collapse** specifies whether the browser should control the

appearance of the adjacent borders that touch each other or whether each cell

should maintain its style.

The **border-spacing** specifies the width that should appear between table cells.

The **caption-side** captions are presented in the <caption> element. By default,

these are rendered above the table in the document. You use the *caption-side*

property to control the placement of the table caption.

The **empty-cells** specifies whether the border should be shown if a cell is empty.

The **table-layout** allows browsers to speed up the layout of a table by using the

first width properties it comes across for the rest of a column rather than having

to load the whole table before rendering it.

Now, we will see how to use these properties with examples.

**The border-collapse Property**

This property can have two values *collapse* and *separate*. The following example uses

both the values:

<style type="text/css">

table.one {border-collapse:collapse;}

table.two {border-collapse:separate;}

td.a {

border-style:dotted;

border-width:3px;

border-color:#000000;

padding: 10px;

}

td.b {border-style:solid;

border-width:3px;

border-color:#333333;

padding:10px;

}

</style>

<table class="one">

<caption>Collapse Border Example</caption>

<tr><td class="a"> Cell A Collapse Example</td></tr>

<tr><td class="b"> Cell B Collapse Example</td></tr>

</table>

<br />

<table class="two">

<caption>Separate Border Example</caption>

<tr><td class="a"> Cell A Separate Example</td></tr>

<tr><td class="b"> Cell B Separate Example</td></tr>

</table>

**The border-spacing Property**

**NOTE:** Unfortunately, this property does not work in Netscape 7 or IE 6.

Ex:

<style type="text/css">

/\* If you provide one value \*/

table.example {border-spacing:10px;}

/\* This is how you can provide two values \*/

table.example {border-spacing:10px; 15px;}

</style>

program

<style type="text/css">

table.one {

border-collapse:collapse;

width:400px;

border-spacing:10px;

}

table.two {

border-collapse:separate;

width:400px;

border-spacing:20px 50px;

}

</style>

<table class="one" border="1">

<caption>Separate Border Example with border-spacing</caption>

<tr><td> Cell A Collapse Example</td></tr>

<tr><td> Cell B Collapse Example</td></tr>

</table>

<br />

<table class="two" border="1">

<caption>Separate Border Example with border-spacing</caption>

<tr><td> Cell A Separate Example</td></tr>

<tr><td> Cell B Separate Example</td></tr>

</table>

**The caption-side Property**

The caption-side property allows you to specify where the content of a <caption>

element should be placed in relationship to the table. The table that follows lists the

possible values.

This property can have one of the four values *top, bottom, left*, or *right*. The following

example uses each value.

**NOTE:** These properties may not work with your IE Browser.

<style type="text/css">

caption.top {caption-side:top}

caption.bottom {caption-side:bottom}

caption.left {caption-side:left}

caption.right {caption-side:right}

</style>

<table style="width:400px; border:1px solid black;">

<caption class="top">

This caption will appear at the top

</caption>

<tr><td > Cell A</td></tr>

<tr><td > Cell B</td></tr>

</table>

<br />

<table style="width:400px; border:1px solid black;">

<caption class="bottom">

This caption will appear at the bottom

</caption>

<tr><td > Cell A</td></tr>

<tr><td > Cell B</td></tr>

</table>

<br />

<table style="width:400px; border:1px solid black;">

<caption class="left">

This caption will appear at the left

</caption>

<tr><td > Cell A</td></tr>

<tr><td > Cell B</td></tr>

</table>

<br />

<table border=1 style="width:400px; solid black;">

<caption class="right">

This caption will appear at the right

</caption>

<tr><td > Cell A</td></tr>

<tr><td > Cell B</td></tr>

</table>

**The empty-cells Property**

The empty-cells property indicates whether a cell without any content should have a

border displayed.

This property can have one of the three values - *show, hide*, or *inherit*.

Here is the empty-cells property used to hide borders of empty cells in the <table>

element.

<style type="text/css">

table.empty{

width:350px;

border-collapse:separate;

empty-cells:hide;

}

td.empty{

padding:5px;

border-style:solid;

border-width:1px;

border-color:#999999;

}

</style>

<table class="empty">

<tr>

<th></th>

<th>Title one</th>

<th>Title two</th>

</tr>

<tr>

<th>Row Title</th>

<td class="empty">value</td>

<td class="empty">value</td>

</tr>

<tr>

<th>Row Title</th>

<td class="empty">value</td>

<td class="empty"></td>

</tr>

</table>

**The table-layout Property**

The table-layout property is supposed to help you control how a browser should render

or lay out a table.

This property can have one of the three values: *fixed, auto*, or *inherit*.

The following example shows the difference between these properties.

**NOTE:** This property is not supported by many browsers, so do not rely on this property.

<style type="text/css">

table.auto

{

table-layout: auto

}

table.fixed

{

table-layout: fixed

}

</style>

<table class="auto" border="1" width="100%">

<tr>

<td width="20%">1000000000000000000000000000</td>

<td width="40%">10000000</td>

<td width="40%">100</td>

</tr>

</table>

<br />

<table class="fixed" border="1" width="100%">

<tr>

<td width="20%">1000000000000000000000000000</td>

<td width="40%">10000000</td>

<td width="40%">100</td>

</tr>

</table>

**CSS ─ Borders**

The **border-color** specifies the color of a border.

The **border-style** specifies whether a border should be solid, dashed line, double

line, or one of the other possible values.

The **border-width** specifies the width of a border.

Now, we will see how to use these properties with examples.

**The border-color Property**

The border-color property allows you to change the color of the border surrounding an

element. You can individually change the color of the bottom, left, top and right sides of

an element's border using the properties:

**border-bottom-color** changes the color of bottom border.

**border-top-color** changes the color of top border.

**border-left-color** changes the color of left border.

**border-right-color** changes the color of right border.

The following example shows the effect of all these properties:

<style type="text/css">

p.example1{

border:1px solid;

border-bottom-color:#009900; /\* Green \*/

border-top-color:#FF0000; /\* Red \*/

border-left-color:#330000; /\* Black \*/

border-right-color:#0000CC; /\* Blue \*/

}

p.example2{

border:1px solid;

border-color:#009900; /\* Green \*/

}

</style>

<p class="example1">

This example is showing all borders in different colors.

</p>

<p class="example2">

This example is showing all borders in green color only.

</p>

**The border-style Property**

The border-style property allows you to select one of the following styles of border:

**none:** No border. (Equivalent of border-width:0;)

**solid:** Border is a single solid line.

**dotted:** Border is a series of dots.

**dashed:** Border is a series of short lines.

**double:** Border is two solid lines.

**groove:** Border looks as though it is carved into the page.

**ridge:** Border looks the opposite of groove.

**inset:** Border makes the box look like it is embedded in the page.

**outset:** Border makes the box look like it is coming out of the canvas.

**hidden:** Same as none, except in terms of border-conflict resolution for table

elements.

You can individually change the style of the bottom, left, top, and right borders of an

element using the following properties:

**border-bottom-style** changes the style of bottom border.

**border-top-style** changes the style of top border.

**border-left-style** changes the style of left border.

**border-right-style** changes the style of right border.

The following example shows all these border styles:

<p style="border-width:4px; border-style:none;">

This is a border with none width.

</p>

<p style="border-width:4px; border-style:solid;">

This is a solid border.

</p>

<p style="border-width:4px; border-style:dashed;">

This is a dahsed border.

</p>

<p style="border-width:4px; border-style:double;">

This is a double border.

</p>

<p style="border-width:4px; border-style:groove;">

This is a groove border.

</p>

<p style="border-width:4px; border-style:ridge">

This is aridge border.

</p>

<p style="border-width:4px; border-style:inset;">

This is a inset border.

</p>

<p style="border-width:4px; border-style:outset;">

This is a outset border.

</p>

<p style="border-width:4px; border-style:hidden;">

This is a hidden border.

</p>

<p style="border-width:4px;

border-top-style:solid;

border-bottom-style:dashed;

border-left-style:groove;

border-right-style:double;">

This is a border with four different styles.

</p>

**The border-width Property**

The border-width property allows you to set the width of an element borders. The value

of this property could be either a length in px, pt, or cm, or it should be set to *thin,*

*medium*, *or thick.*

You can individually change the width of the bottom, top, left, and right borders of an

element using the following properties:

**border-bottom-width** changes the width of bottom border.

**border-top-width** changes the width of top border.

**border-left-width** changes the width of left border.

**border-right-width** changes the width of right border.

The following example shows all these border width:

<p style="border-width:4px; border-style:solid;">

This is a solid border whose width is 4px.

</p>

<p style="border-width:4pt; border-style:solid;">

This is a solid border whose width is 4pt.

</p>

<p style="border-width:thin; border-style:solid;">

This is a solid border whose width is thin.

</p>

<p style="border-width:medium; border-style:solid;">

This is a solid border whose width is medium;

</p>

<p style="border-width:thick; border-style:solid;">

This is a solid border whose width is thick.

</p>

<p style="border-bottom-width:4px;

border-top-width:10px;

border-left-width: 2px;

border-right-width:15px;

border-style:solid;">

This is a a border with four different width.

</p>

**Border Properties Using Shorthand**

Ex

<p style="border:4px solid red;">

This example is showing shorthand property for border.

</p>

**CSS ─ Margins**

The **margin** specifies a shorthand property for setting the margin properties in

one declaration.

The **margin-bottom** specifies the bottom margin of an element.

The **margin-top** specifies the top margin of an element.

The **margin-left** specifies the left margin of an element.

The **margin-right** specifies the right margin of an element.

Now, we will see how to use these properties with examples.

**The Margin Property**

The margin property allows you to set all of the properties for the four margins in one

declaration. Here is the syntax to set margin around a paragraph:

<style type="text/css">

p {margin: 15px}

all four margins will be 15px

p {margin: 10px 2%}

top and bottom margin will be 10px, left and right margin will be 2% of the

total width of the document.

p {margin: 10px 2% -10px}

top margin will be 10px, left and right margin will be 2% of the total width of

the document, bottom margin will be -10px

p {margin: 10px 2% -10px auto}

top margin will be 10px, right margin will be 2% of the total width of the

document, bottom margin will be -10px, left margin will be set by the browser

</style>

EX:

<p style="margin: 15px; border:1px solid black;">

all four margins will be 15px

</p>

<p style="margin:10px 2%; border:1px solid black;">

top and bottom margin will be 10px, left and right margin will be 2% of the

total width of the document.

</p>

<p style="margin: 10px 2% -10px; border:1px solid black;">

top margin will be 10px, left and right margin will be 2% of the total width of

the document, bottom margin will be -10px

</p>

<p style="margin: 10px 2% -10px auto; border:1px solid black;">

top margin will be 10px, right margin will be 2% of the total width of the

document, bottom margin will be -10px, left margin will be set by the browser

</p>

**The margin-bottom Property**

The margin-bottom property allows you to set the bottom margin of an element. It can

have a value in length, %, or auto.

example:

<p style="margin-bottom: 15px; border:1px solid black;">

This is a paragraph with a specified bottom margin

</p>

<p style="margin-bottom: 5%; border:1px solid black;">

This is another paragraph with a specified bottom margin in percent

</p>

**The margin-top Property**

n example:

<p style="margin-top: 15px; border:1px solid black;">

This is a paragraph with a specified top margin

</p>

<p style="margin-top: 5%; border:1px solid black;">

This is another paragraph with a specified top margin in percent

</p>

**The margin-left Property**

Example:

<p style="margin-left: 15px; border:1px solid black;">

This is a paragraph with a specified left margin

</p>

<p style="margin-left: 5%; border:1px solid black;">

This is another paragraph with a specified top margin in percent

</p>

**The margin-right Property**

example:

<p style="margin-right: 15px; border:1px solid black;">

This is a paragraph with a specified right margin

</p>

<p style="margin-right: 5%; border:1px solid black;">

This is another paragraph with a specified right margin in percent

</p>