



Chapter 3

CSS

3.1 Introduction of CSS

CSS stands for Cascading Style Sheets. It is a simple design language intended to simplify the process of making web pages presentable. CSS handles the look and feel part of a web page. Using CSS, you 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.

CSS works with HTML and other Markup Languages (such as XHTML and XML) to control the way the content is presented. Cascading Style Sheets is a means to separate the appearance of a webpage from the content of a webpage.

3.1.1 Definition

Cascading Style Sheets (CSS) is a simple mechanism used to format the layout of Web Pages and adding **style** (e.g., fonts, colors, spacing...) to web documents that previously could only be defined in a page's HTML. **CSS** describes how HTML elements are to be displayed on screen, paper, or in other media. It can control the layout of multiple web pages all at once.

3.1.2 Advantages

The advantages of CSS are:

- ❖ **CSS saves time** - You can write CSS once and then reuse the same sheet in multiple HTML pages.
- ❖ **Pages load faster** – Increases Download Speed
- ❖ **Easy maintenance** - To make a global change, all the elements in all the web pages will be updated automatically.
- ❖ **Superior styles to HTML** – It is 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.
- ❖ **Global web standards** - Now HTML attributes are being deprecated and it is being recommended to use CSS

3.1.3 What is the “Cascade” part of CSS?

The cascade part of CSS means that more than one style sheet can be attached to a document, and all of them can influence the presentation. For example, a designer can have a global style sheet for the whole site, but a local one for say, controlling the link color and background of a specific page. Or, a user can use own style sheet if s/he has problems seeing the page, or if s/he just prefers a certain look.

3.2 CSS Syntax

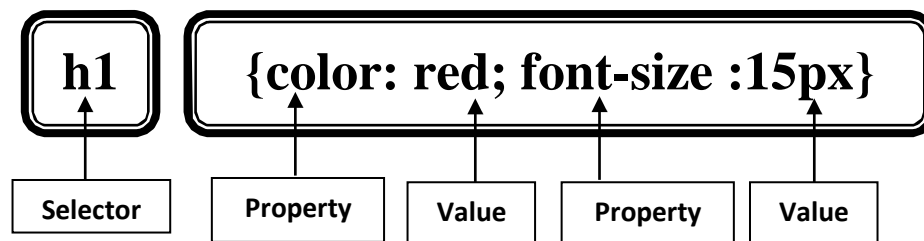
A CSS style rule is made of three parts:

1. **Selector:** A selector is an HTML tag at which a style will be applied. This could be any tag like <h1>, <p> or <table> etc.
2. **Property:** A property is a type of attribute of HTML tag. Put simply, all the HTML attributes are converted into CSS properties. They could be *color*, *border*, *bgcolor* etc.
3. **Value:** Values are assigned to properties. For example, *color* property can have the value either *red* or *#F1F1F1* etc.

The format or syntax of CSS is:

Selector {property:

Example: You can define a heading as follows:



Here **h1** is a selector , **color and font-size** are properties and the given value **red, and 15px** are the value of that property.

- ✎ The selector is normally the HTML element you want to style.
- ✎ Each declaration consists of a property and a value.
- ✎ The property is the style attribute you want to change. Each property has a value.

3.2.1 Rules/ Principle of CSS

1. Every statement must have a selector and a declaration. The declaration comes immediately after the selector and is contained in a pair of curly braces.
2. The declaration is one or more properties separated by semicolons.
3. Each property has a property name followed by a colon and then the value for that property. There are many different types of values, but any given property can only take certain values as set down in the specification.
4. Sometimes a property can take a number of values, as in the font-family. The values in the list should be separated by a comma and a space.
5. Sometimes a value will have a unit as well as the actual value, as in the 1.3em. You must not put a space between the value and its unit.
6. As with HTML, white space can be used to make your style sheet easier to read and write.

3.3 CSS Selectors

You can define selectors in various simple ways based on your comfort. Let me put these selectors one by one. Three types of CSS Selectors

1. The Element selectors
2. The ID Selectors
3. The Class Selectors

3.3.1 The Element selectors

A CSS declaration always ends with a semicolon, and declaration groups are surrounded by curly brackets: example -

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

To make the CSS more readable, you can put one declaration on each line, like this:

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

3.3.2 The ID selectors

The id selector is used to specify a style for a single, unique element. The id selector uses the id attribute of the HTML element, and is defined with a "#". Example –

Imagine within the body element of our html page, we have the following paragraph element

```
<p id="welcome">welcome to the 1st CSS
Document </p>
```

We can then create a CSS rule with the id selector:

```
#welcome
{
color:red;
text-align:center;
}
```

3.3.3 The Class selectors

The class selector is used to specify a style for a group of elements. Unlike the id selector, the class selector is most often used on several elements. This allows you to set a particular style for many HTML elements with the same class. The class selector uses the HTML class attribute, and is defined with a ".". In the example below, all HTML elements with class="center" will be center-aligned:

Imagine within the body element of our html page, we have the following header element

```
<h2 class="center">Summary</h2>
```

We can then create a CSS rule with the class selector:

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

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

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

Some of the other selectors are used in CSS, they are :

3.3.3.1 **Universal selector**

An asterisk (*) is the universal selector for CSS. It matches a single element of any type. Omitting the asterisk with simple selectors has the same effect. For instance, *.warning and . warning are considered equal. Rather than selecting elements of a specific type, the universal selector quite simply matches the name of any element type: Example-

```
*
{
color:#000000;
}
```

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

3.3.3.2 **Attribute Selector**

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-".

3.4 Ways to insert CSS

There are three ways of inserting a style sheet:

1. External style sheet
2. Internal style sheet
3. Inline style

3.4.1 External style sheet

An external style sheet is ideal when the style is applied to many pages. With an external style sheet, you can change the look of an entire Web site by changing one file. Each page must link to the style sheet using the <link> tag. The <link> tag goes inside the head section:

```
<head>
```

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

```
</head>
```

An external style sheet can be written in any text editor. The file should not contain any html tags. Your style sheet should be saved with a .css extension. An example of a style sheet file is shown below:

```
hr {color:sienna;}  
p {margin-left:20px;}  
body {background-image:url("images/back40.gif");}
```

Notes : Do not leave spaces between the property value and the units! "margin-left:20 px" (instead of "margin-left:20px") will work in IE, but not in Firefox or Opera.

3.4.2 Internal style sheet

An internal style sheet should be used when a single document has a unique style. You define internal styles in the head section of an HTML page, by using the <style> tag, like this:

```
<head>
```

```
<style type="text/css">
```

```
hr {color:sienna;}  
p {margin-left:20px;}  
body {background-image:url("images/back1.gif");}
```

```
</style>
```

```
</head>
```

3.4.3 Inline style sheet

An inline style loses many of the advantages of style sheets by mixing content with presentation. Use this method sparingly! To use inline styles you use the style attribute in the relevant tag. The style attribute can contain any CSS property. The example shows how to change the color and the left margin of a paragraph:

```
<p style="color:sienna;margin-left:20px">This is a  
paragraph.</p>
```

3.5 Background image handling

The background-image property specifies an image to use as the background of an element. By default, the image is repeated so it covers the entire element. The background image for a page can be set like this:body {background-image:url('paper.gif');}

Example

```
<html>  
<head>  
  <Title>This is my Internal css page</Title>  
  <style type="text/css">  
    body  
    {  
      background-image:url  
      ("C:/Users/SAI/Desktop/Desktop/100MSDCF/11.  
      jpg");  
    }  
  </style>  
</head>  
<body>  
  Background Image  
</body>  
</html>
```

The output of the above example is :



You can set the following background properties 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 shorthand to specify a number of other background properties.

3.5.1 Repeat the Background Image

The following example demonstrates how to repeat the background image if an image is small. You can use no-repeat value for the background-repeat property if you don't want to repeat an image. In this case, the image will display only once. By default, the background-repeat property will have a repeat value.

```
<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>
```

3.5.2 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>
```

3.6 Background colour Management using CSS

The background-color property specifies the background color of an element. The background color of a page is defined in the body selector: Example

```
body {background-color:#b0c4de;}  
<p style="background-color:yellow;">  
This text has a yellow background color. </p>
```

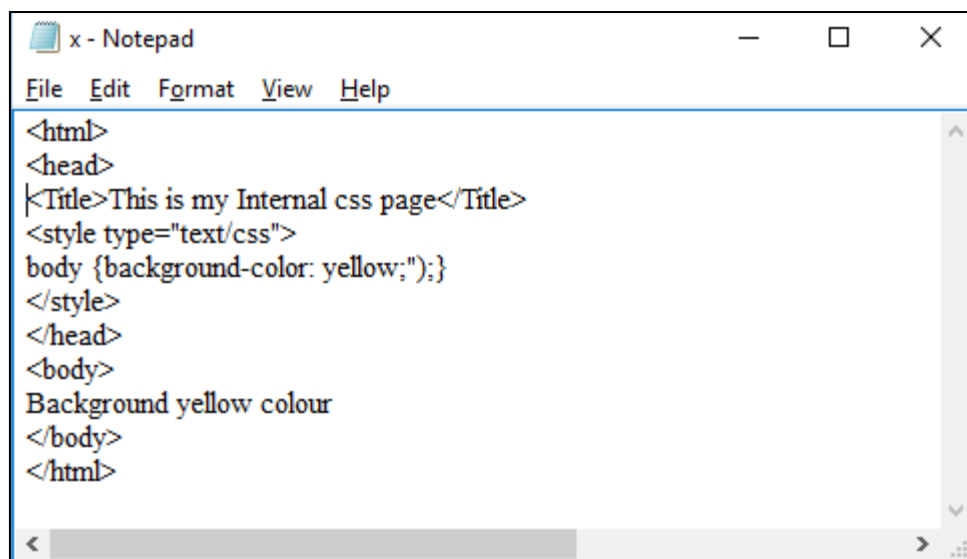
With CSS, a color is most often specified by:

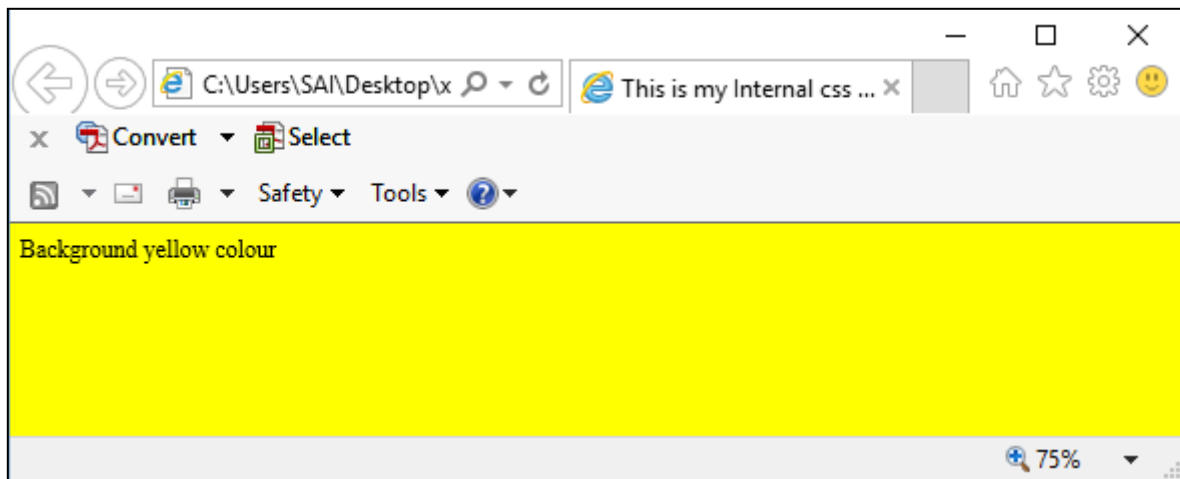
1. a HEX value - like "#ff0000"
2. an RGB value - like "rgb(255,0,0)"
3. a color name - like "red"

In the example below, the h1, p, and div elements have different background colors:

```
h1 {background-color:#6495ed;}  
p {background-color:#e0ffff;}  
div {background-color:#b0c4de;}
```

Example










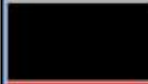







3.6.1 CSS Colors - Hex Codes

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). Each hexadecimal code will be preceded by a pound or hash sign '#'. Following are the examples to use Hexadecimal notation.

3.6.2 CSS Colors - Short Hex Codes







This is a shorter form of the six-digit notation. In this format, each digit is replicated to arrive at an equivalent six-digit value. For example: #6A7 becomes #66AA77.

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

Color	Color HEX
	#000
	#F00
	#0F0
	#0FF
	#FF0
	#0FF
	#F0F
	#FFF

3.6.3 CSS 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.

Color	Color RGB
	<code>rgb(0,0,0)</code>
	<code>rgb(255,0,0)</code>
	<code>rgb(0,255,0)</code>
	<code>rgb(0,0,255)</code>
	<code>rgb(255,255,0)</code>
	<code>rgb(0,255,255)</code>

3.7 Text Management using CSS

CSS is a language that describes the style of an HTML document. You can set the following text properties of an element:

- 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.

3.7.1 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>
```

3.7.2 Set the Text Direction

The following example demonstrates how to set the direction of a text. Possible values are ltr or rtl.

```
<p style="direction:rtl;">This text will be rendered from right to left </p>
```

3.7.3. Set the Space between Characters

The following example demonstrates how to set the space between characters. Possible values are normal or a number specifying space.

```
<p style="letter-spacing:5px;">This text is having space between letters. </p>
```

3.7.4 Set the 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>
```

3.7.5 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>
```

3.7.6 Set the 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>
```

3.7.7 Decorating the Text

The following example demonstrates how to decorate a text. Possible values are none, underline, overline, line-through, blink.

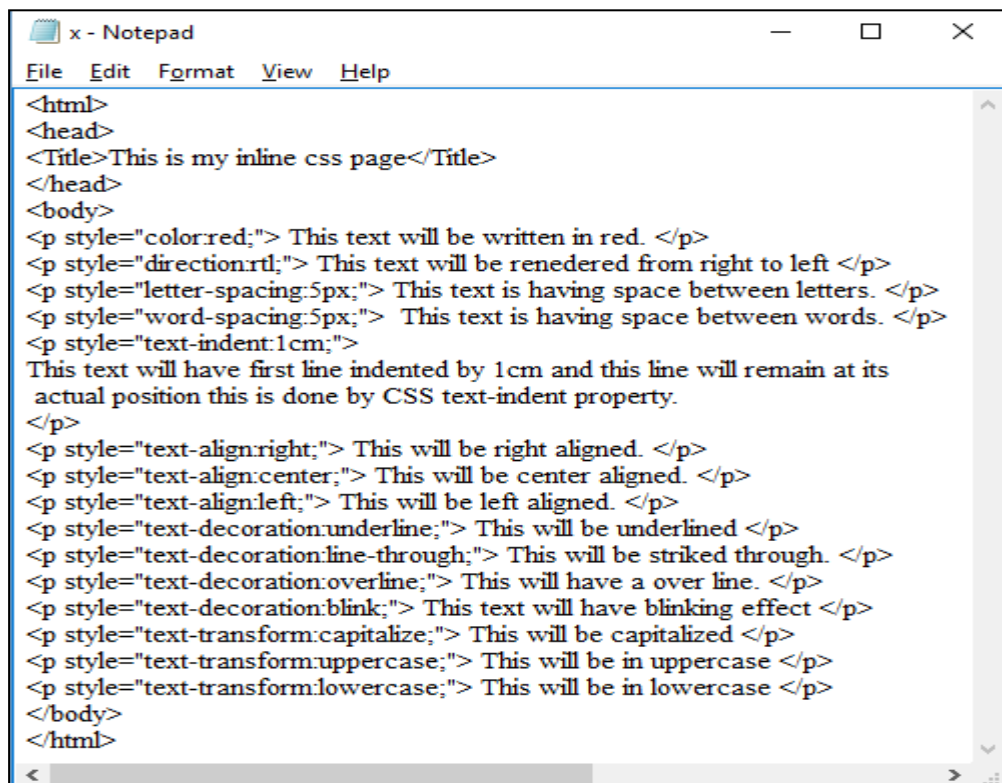
```
<p style="text-decoration:underline;"> This will be underlined </p>
<p style="text-decoration:line-through;"> This will be striked through.
</p>
<p style="text-decoration:overline;"> This will have a over line. </p>
<p style="text-decoration:blink;">This text will have blinking effect </p>
```

3.7.8 Set the Text Cases

The following example demonstrates how to set the cases for a text. Possible values are none, capitalize, uppercase, lowercase.

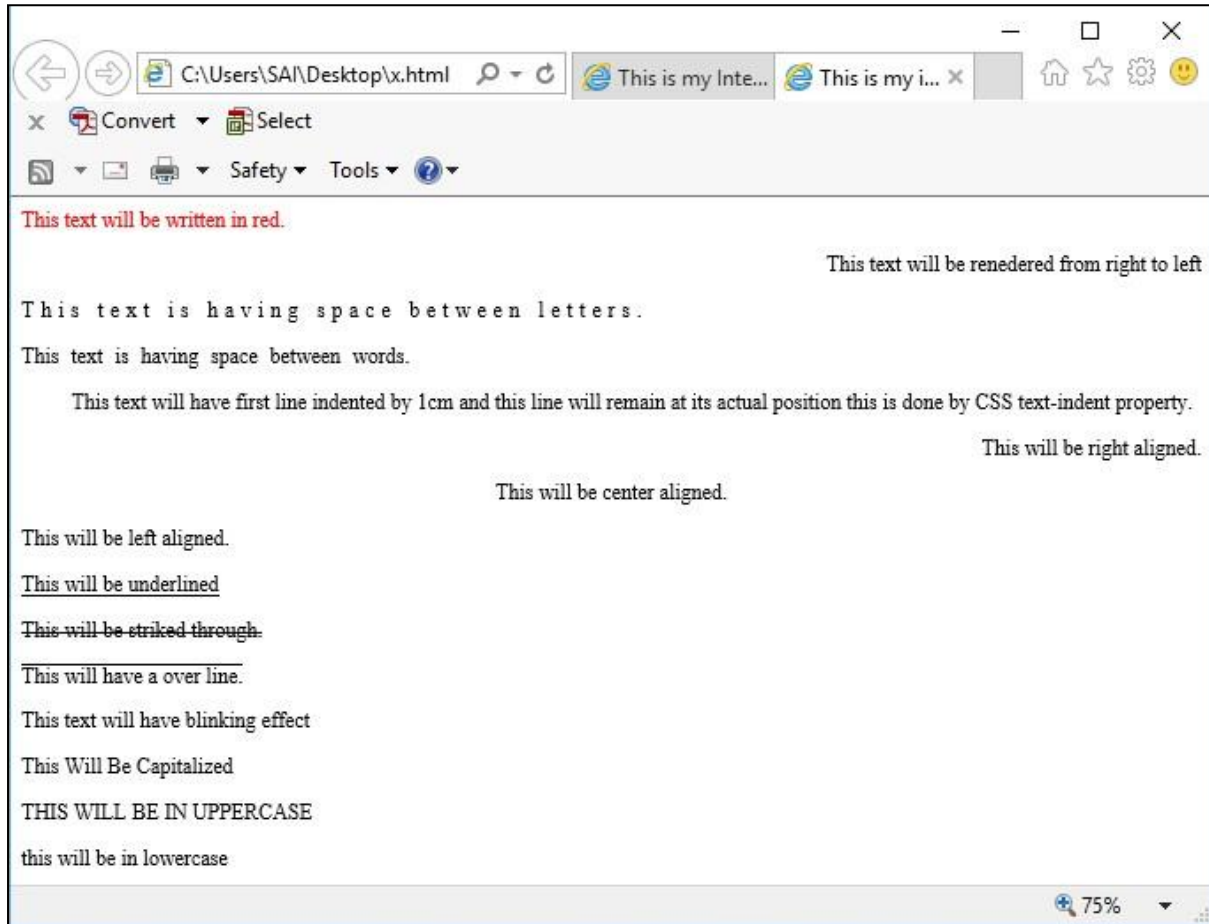
```
<p style="text-transform:capitalize;"> This will be capitalized </p>
<p style="text-transform:uppercase;"> This will be in uppercase </p>
<p style="text-transform:lowercase;"> This will be in lowercase </p>
```

Example
:

A screenshot of a Notepad window titled 'x - Notepad'. The window contains HTML code for text styling. The code includes a head section with a title 'This is my inline css page' and a body section with several paragraphs demonstrating various CSS text properties. The paragraphs show text in red, right-to-left direction, letter and word spacing, text indentation, right, center, and left alignment, underlined, striked through, overlined, blinking, capitalized, uppercase, and lowercase text transformations. The code is as follows:

```
<html>
<head>
<Title>This is my inline css page</Title>
</head>
<body>
<p style="color:red;"> This text will be written in red. </p>
<p style="direction:rtl;"> This text will be renedered from right to left </p>
<p style="letter-spacing:5px;"> This text is having space between letters. </p>
<p style="word-spacing:5px;"> This text is having space between words. </p>
<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>
<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>
<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 a over line. </p>
<p style="text-decoration:blink;"> This text will have blinking effect </p>
<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>
</body>
</html>
```

The output of the above program is :



3.8 Font Management using CSS

A **font** is the combination of typeface and other qualities, such as size, pitch, and spacing. For example, Times Roman is a typeface that defines the shape of each character. Within Times Roman, however, there are many **fonts** to choose from -- different sizes, italic, bold, and so on. You can set the following font properties of an element:

- 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.

3.8.1 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>

3.8.2 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>
```

3.8.3 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>
```

3.8.4 Set the Font Weight

The following example demonstrates how to set the font weight of an element. The font-weight 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>
```

3.8.5 Set the Font Size

The following example demonstrates how to set the font size of an element. The font-size property is used to control the size of fonts. Possible values could be xx-small, x-small, 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>
```

3.8.6 Set the Font Size Adjust

The following example demonstrates how to set the font size adjust of an element. This property enables you to adjust the x-height to make fonts more legible. Possible value could be any number.

```
<p style="font-size-adjust:0.61;">This text is using a font-size-adjust value. </p>
```

3.8.7 Set the Font Stretch

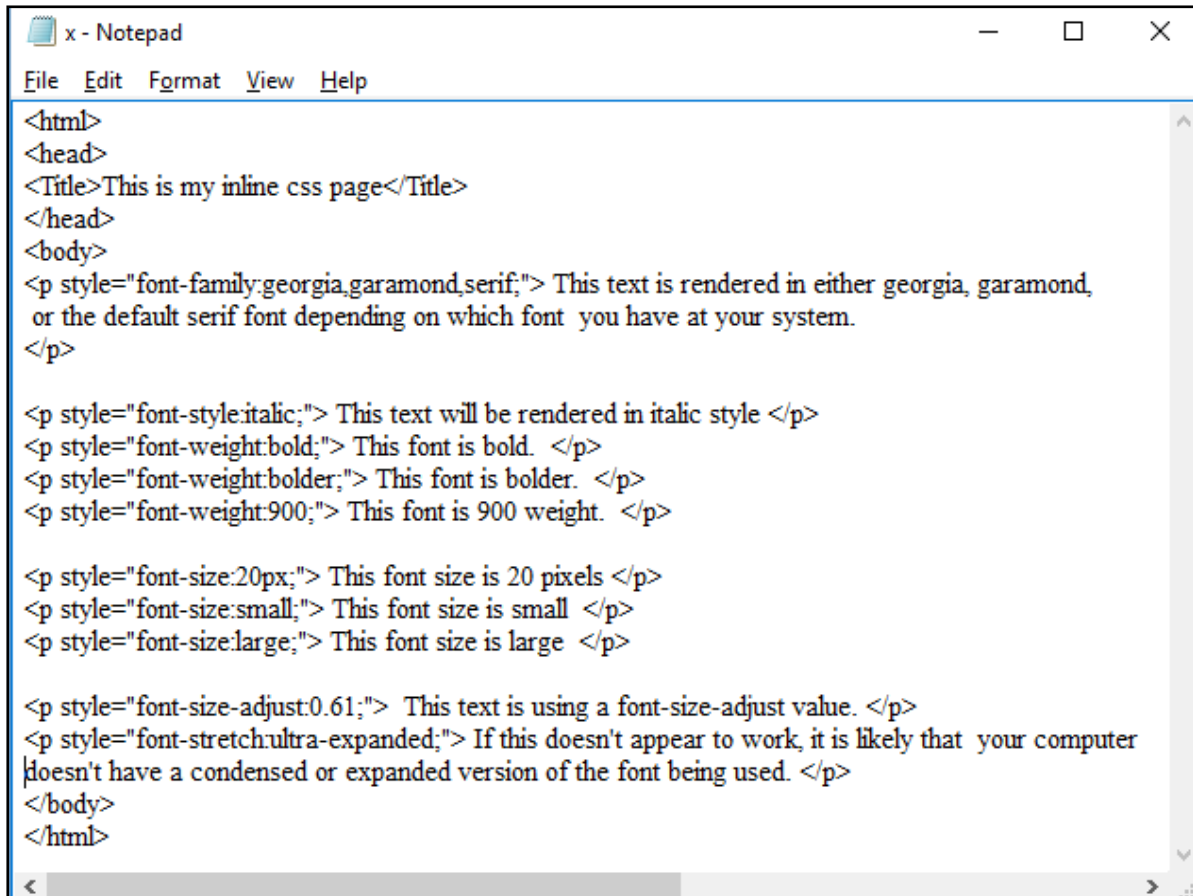
The following example demonstrates how to set the font stretch of an element. This property relies on the user's computer to have an expanded or condensed version of the font being used.

Possible values could be normal, wider, narrower, ultra-condensed, extra-condensed, condensed, semi-condensed, semi-expanded, expanded, extra-expanded, ultra-expanded.

`<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>`

Example:-



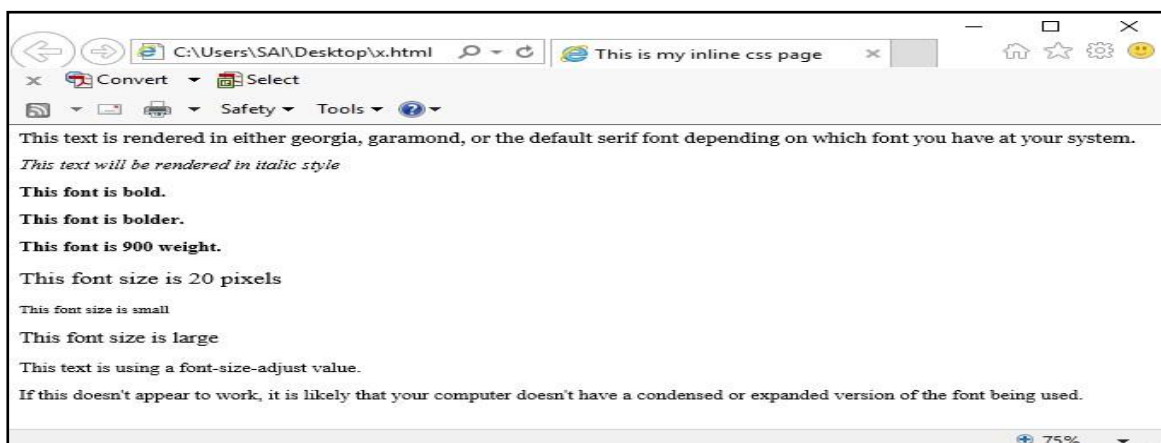
```
<html>
<head>
<Title>This is my inline css page</Title>
</head>
<body>
<p style="font-family:georgia,garamond,serif;"> This text is rendered in either georgia, garamond,
or the default serif font depending on which font you have at your system.
</p>

<p style="font-style:italic;"> This text will be rendered in italic style </p>
<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>

<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>

<p style="font-size-adjust:0.61;"> This text is using a font-size-adjust value. </p>
<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>
</body>
</html>
```

The output of the above program is :



The border-spacing Property

The border-spacing property specifies the distance that separates the adjacent cells' borders. It can take either one or two values; these should be units of length. If you provide one value, it applies to both vertical and horizontal borders. Or you can specify two values, in which case, the first refers to the horizontal spacing and the second to the vertical spacing:

NOTE: Unfortunately, this property does not work in Netscape 7 or IE 6. Now let's modify the previous example and see the effect:

```
<style type="text/css">
table.one
{
border-collapse:separate; width:400px;
border-spacing:10px;
}
table.two
{
border-collapse:separate; width:400px;
border-spacing:10px 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>
```

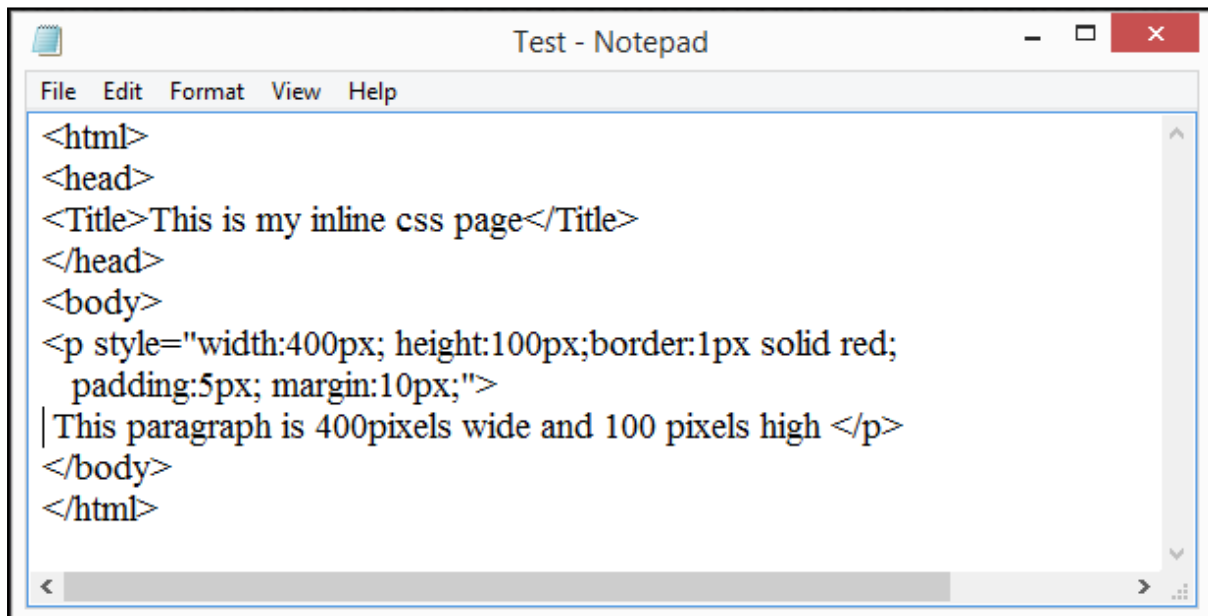
3.9 Working with Box Model

You have seen the border that surrounds every box i.e. element, the padding that can appear inside each box, and the margin that can go around them. In this chapter, we will learn how to change the dimensions of boxes. We have the following properties that allow you to control the dimensions of a box.

- The **height** property is used to set the height of a box.
- The **width** property is used to set the width of a box.
- The **line-height** property is used to set the height of a line of text.
- The **max-height** property is used to set a maximum height that a box can be.
- The **min-height** property is used to set the minimum height that a box can be.
- The **max-width** property is used to set the maximum width that a box can be.
- The **min-width** property is used to set the minimum width that a box can be.

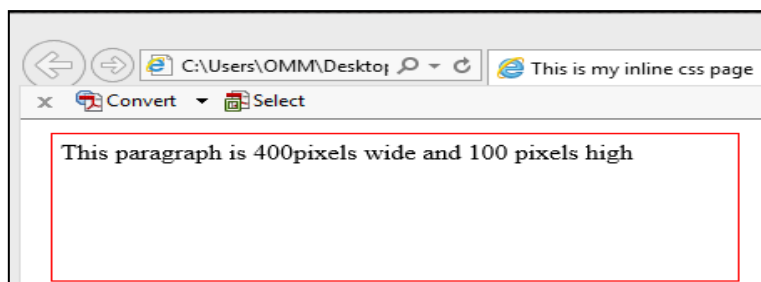
3.9.1 The Height and Width Properties

The height and width properties allow you to set the height and width for boxes. They can take values of a length, a percentage, or the keyword auto. Here is an example:



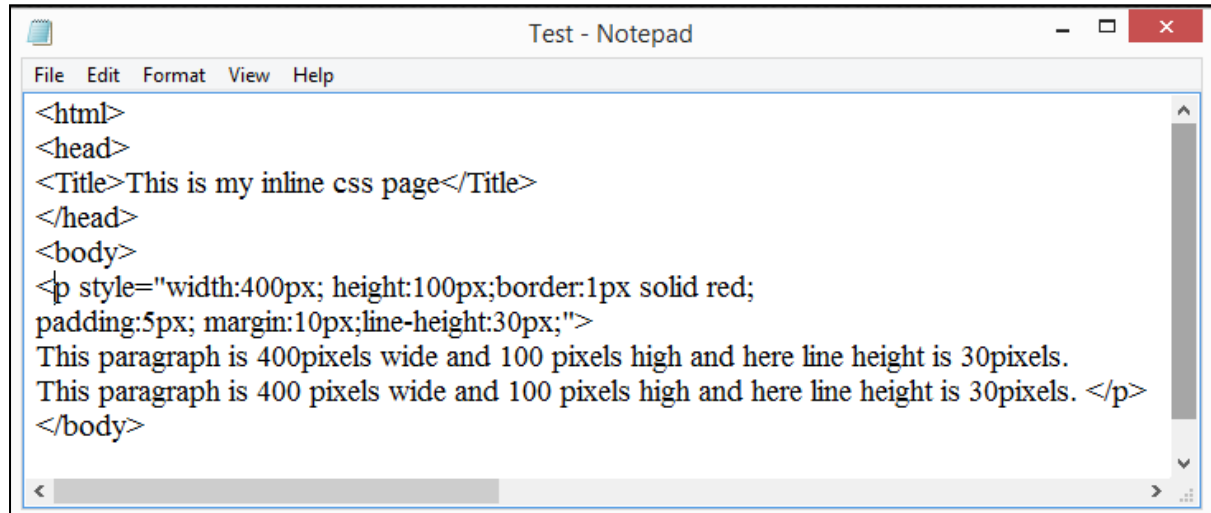
```
<html>
<head>
<Title>This is my inline css page</Title>
</head>
<body>
<p style="width:400px; height:100px;border:1px solid red;
padding:5px; margin:10px;">
| This paragraph is 400pixels wide and 100 pixels high </p>
</body>
</html>
```

The output of the above program is:

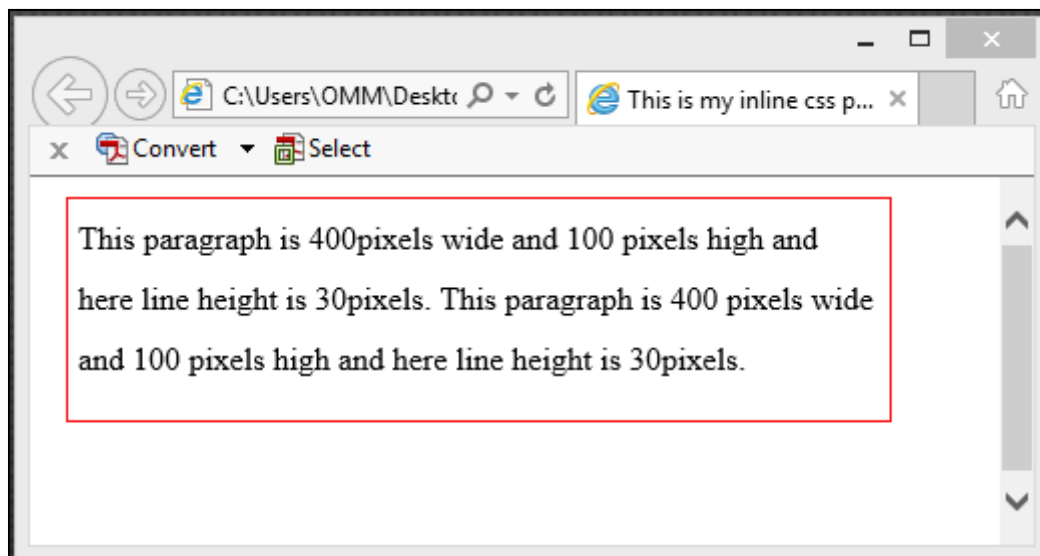


3.9.2 The line-height Property

The line-height property allows you to increase the space between lines of text. The value of the line-height property can be a number, a length, or a percentage. Here is an example:



The output of the above program is



3.10 Designing borders using CSS

The *border* properties allow you to specify how the border of the box representing an element should look. There are three properties of a border you can change:

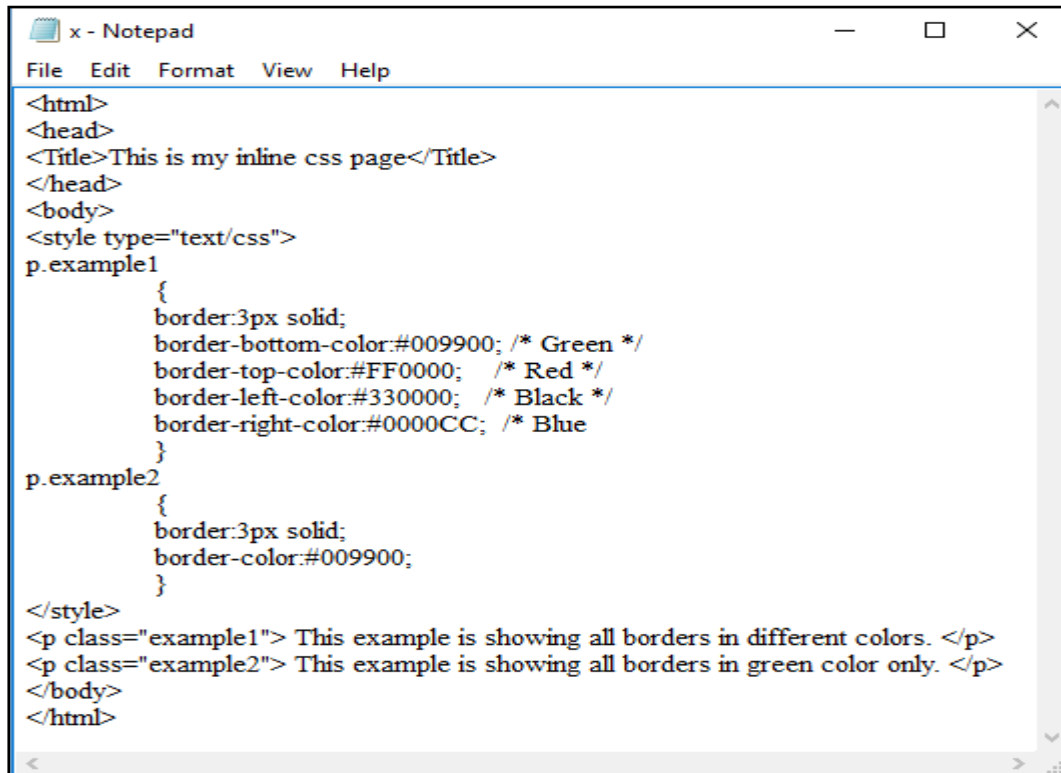
- The **border-color** specifies the color of a border.
- The **border-style** specifies whether a border should be solid, dashed line, double line, or one of the other possible values.
- The **border-width** specifies the width of a border.

3.10.1 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:

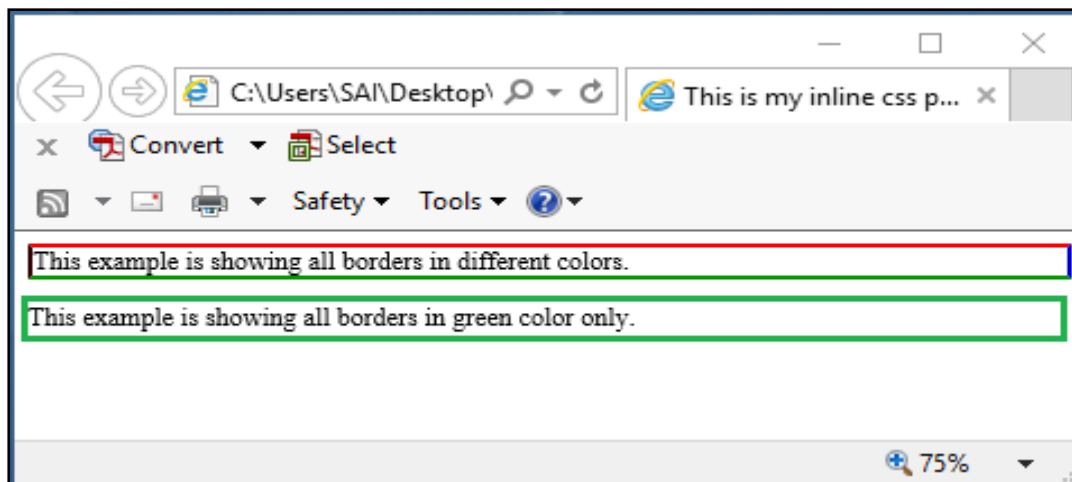
- border-bottom-color changes the color of bottom border.
- border-top-color changes the color of top border.
- border-left-color changes the color of left border.
- border-right-color changes the color of right border.

The following example shows the effect of all these properties:



```
<html>
<head>
<Title>This is my inline css page</Title>
</head>
<body>
<style type="text/css">
p.example1
{
border:3px solid;
border-bottom-color:#009900; /* Green */
border-top-color:#FF0000; /* Red */
border-left-color:#330000; /* Black */
border-right-color:#0000CC; /* Blue
}
p.example2
{
border:3px solid;
border-color:#009900;
}
</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>
</body>
</html>
```

The output of the above program is :



3.10.2 The border-style Property

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

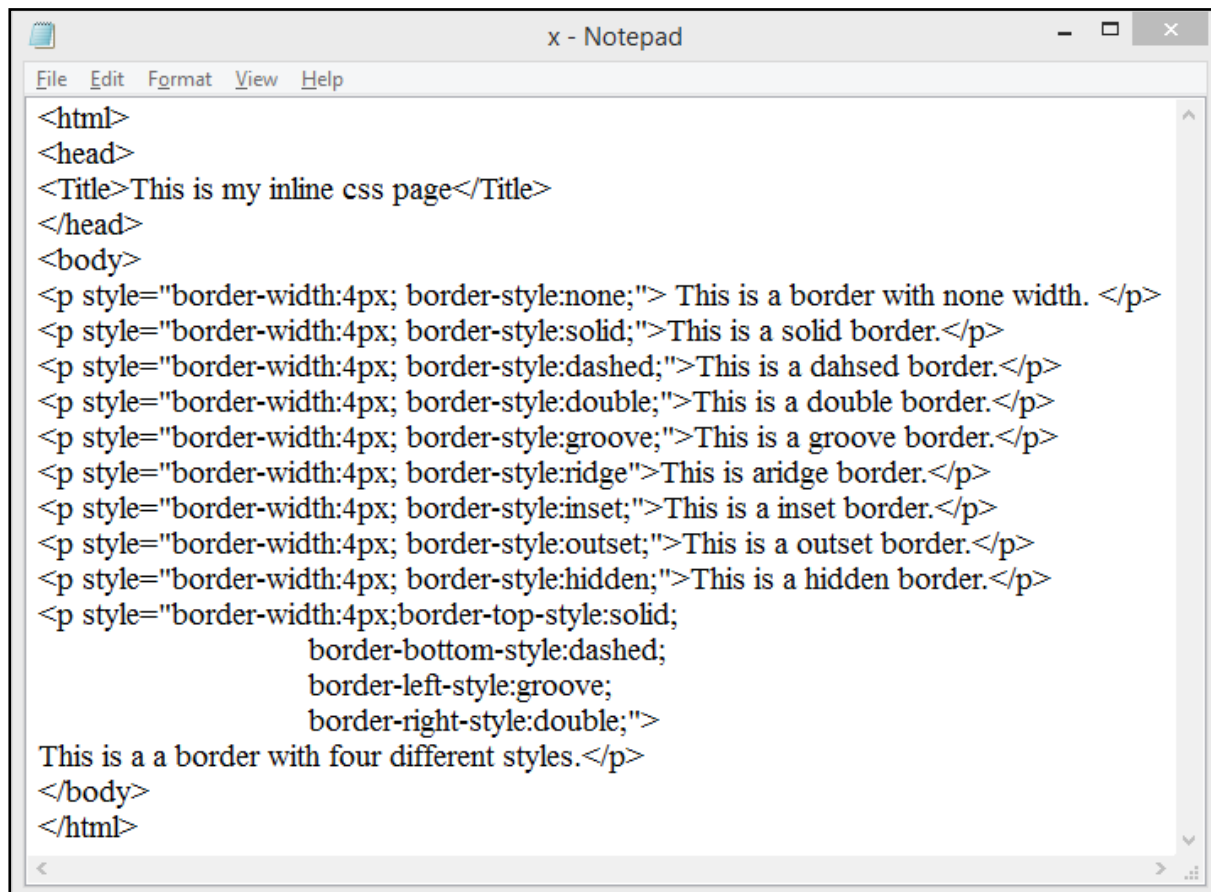
- **none**: No border. (Equivalent of border-width:0;)
- **solid**: Border is a single solid line.
- **dotted**: Border is a series of dots.
- **dashed**: Border is a series of short lines.
- **double**: Border is two solid lines.

- **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:

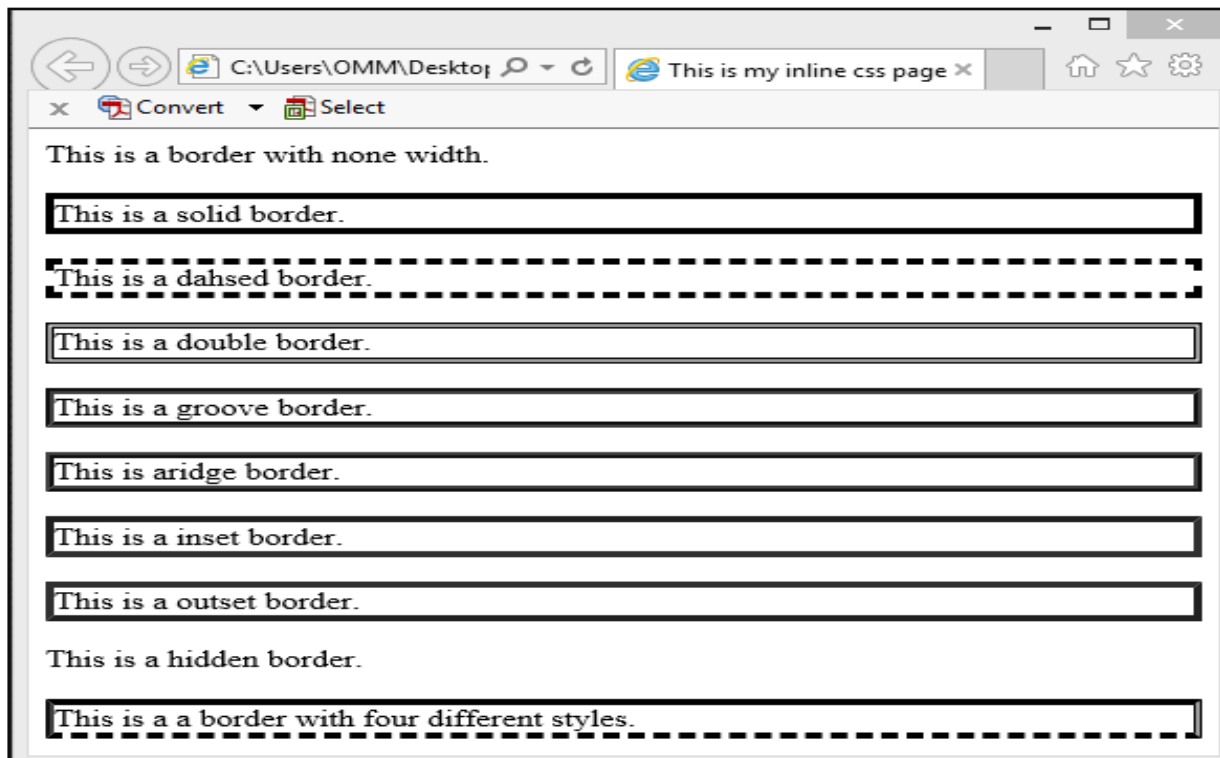
- **border-bottom-style** changes the style of bottom border.
- **border-top-style** changes the style of top border.
- **border-left-style** changes the style of left border.
- **border-right-style** changes the style of right border.

The following example shows all these border styles:



```
x - Notepad
File Edit Format View Help
<html>
<head>
<Title>This is my inline css page</Title>
</head>
<body>
<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 a ridge 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 a border with four different styles.</p>
</body>
</html>
```

It will produce the following result:



3.11 Setting Page Margin using CSS

The margin property defines the space around an HTML element. It is possible to use negative values to overlap content. The values of the margin property are not inherited by the child elements. Remember that the adjacent vertical margins (top and bottom margins) will collapse into each other so that the distance between the blocks is not the sum of the margins, but only the greater of the two margins or the same size as one margin if both are equal.

We have the following properties to set an element margin.

- 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.

3.11.1 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}
```

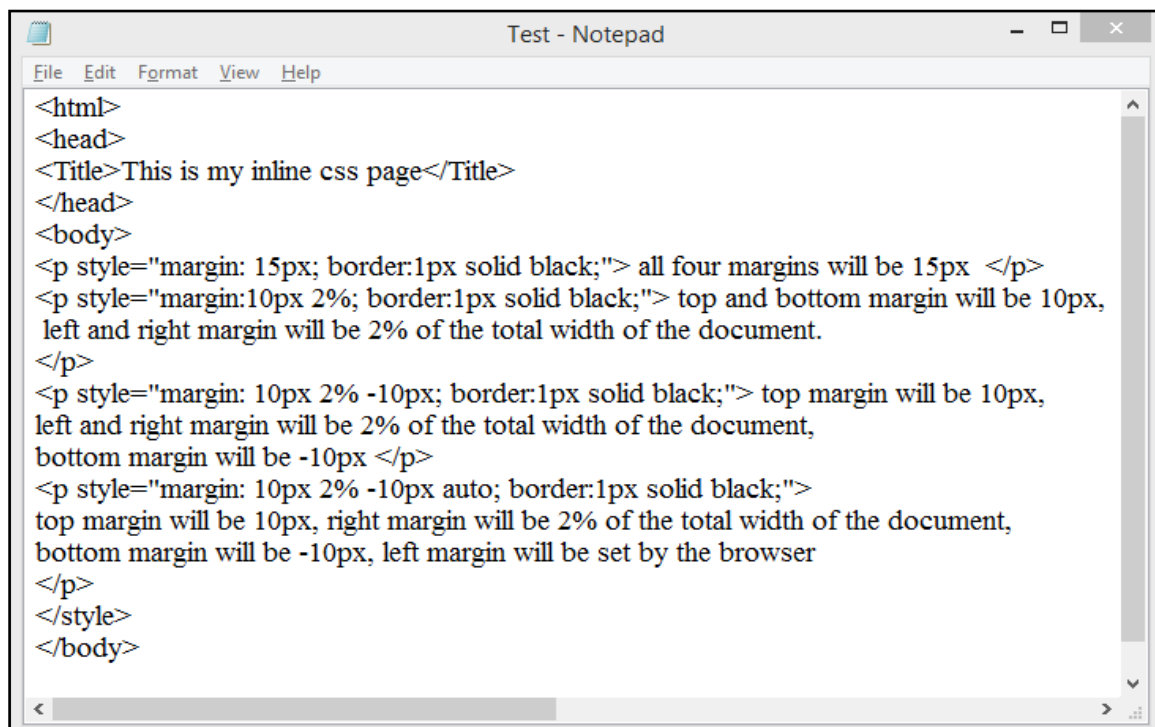
top margin will be 10px, left and right margin will be 2% of the total width of the document, bottom margin will be -10px

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

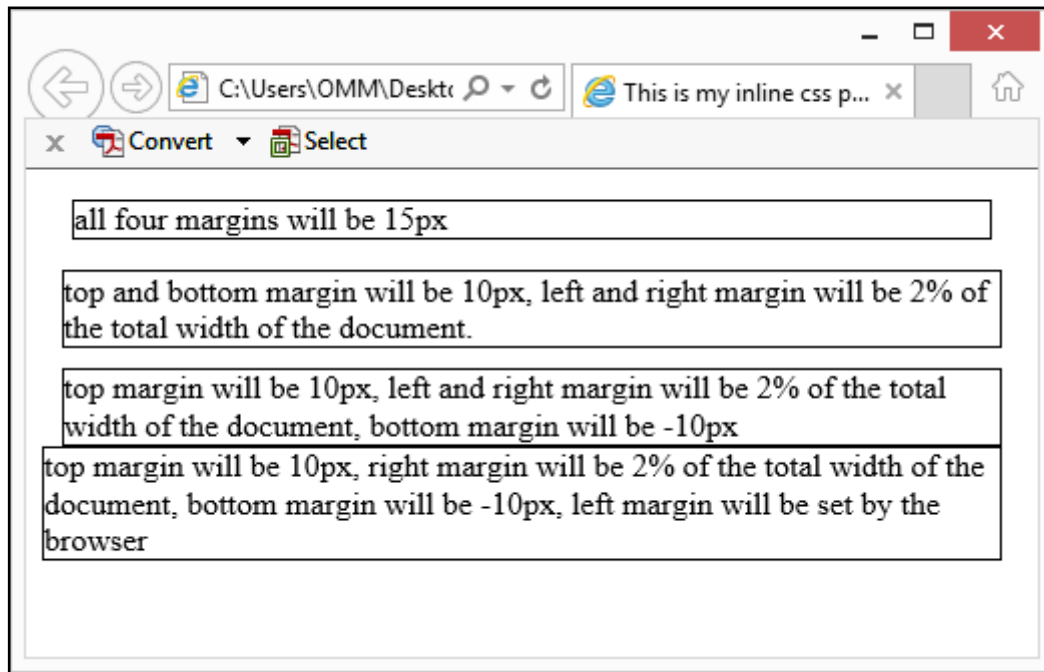
top margin will be 10px, right margin will be 2% of the total width of the document, bottom margin will be -10px, left margin will be set by the browser

```
</style>
```

Here is an example:



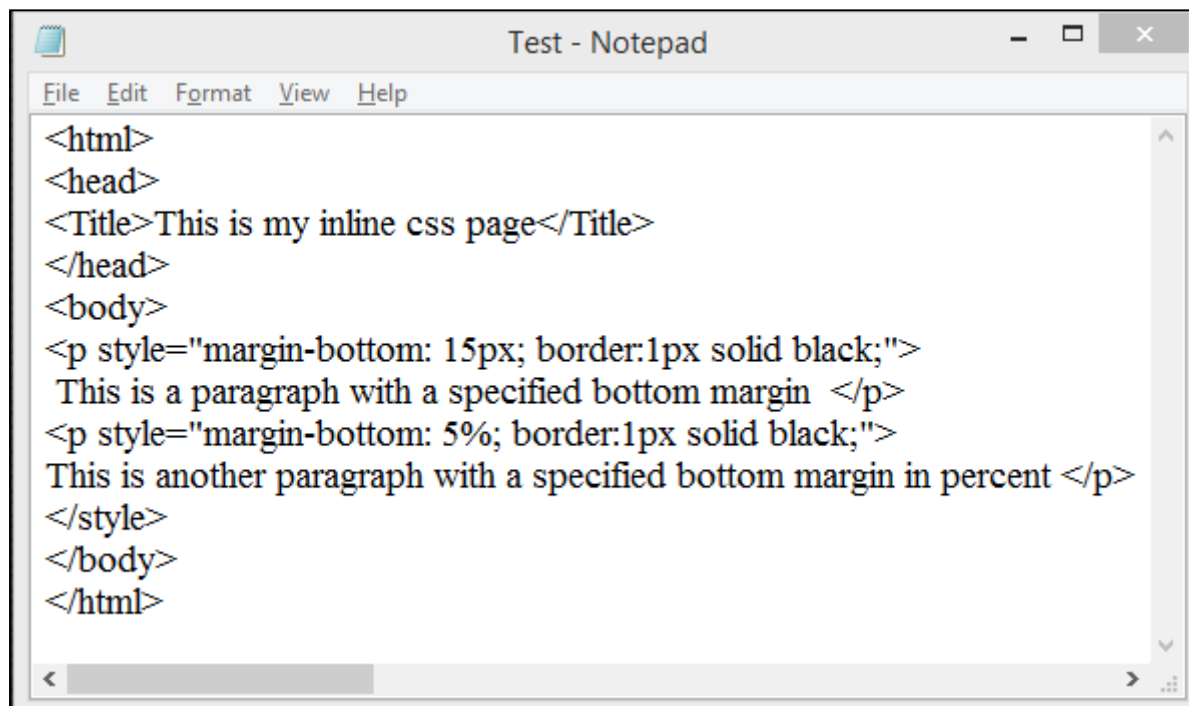
The above program will produce the following result:



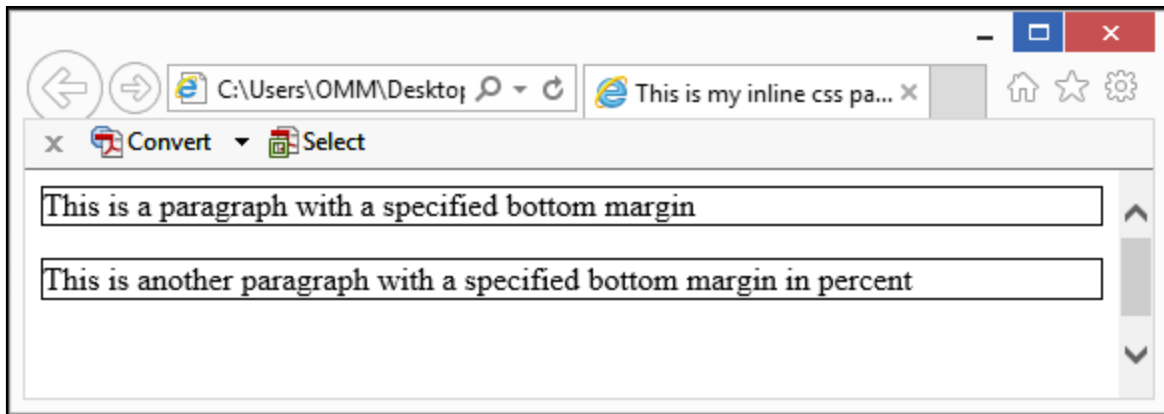
3.11.2 The margin-bottom Property

The margin-bottom property allows you to set the bottom margin of an element. It can have a value in length, %, or auto.

Here is an example:



The above program will produce the following result:



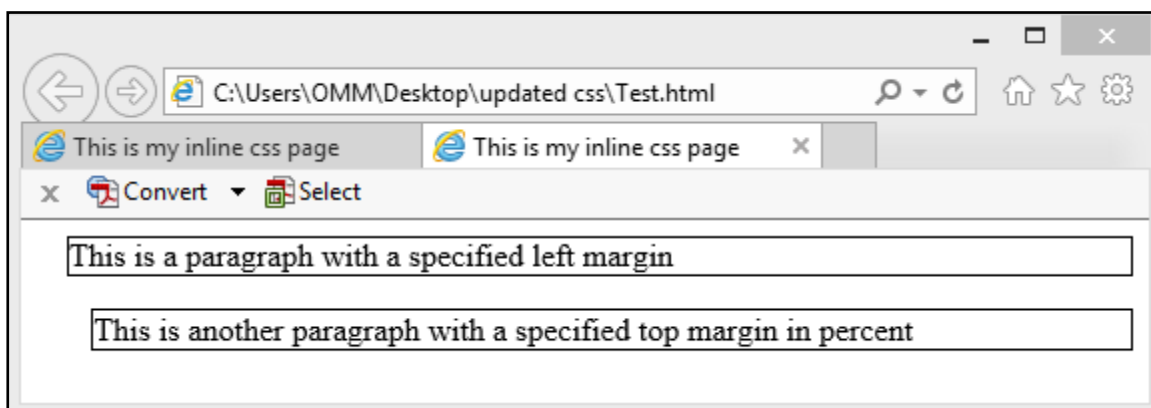
3.11.3 The margin-left Property

The margin-left property allows you to set the left margin of an element. It can have a value in length, %, or auto. Here is an example:

```

Test - Notepad
File Edit Format View Help
<html>
<head>
<Title>This is my inline css page</Title>
</head>
<body>
<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>
</style>
</body>
</html>
  
```

The above program will produce the following result



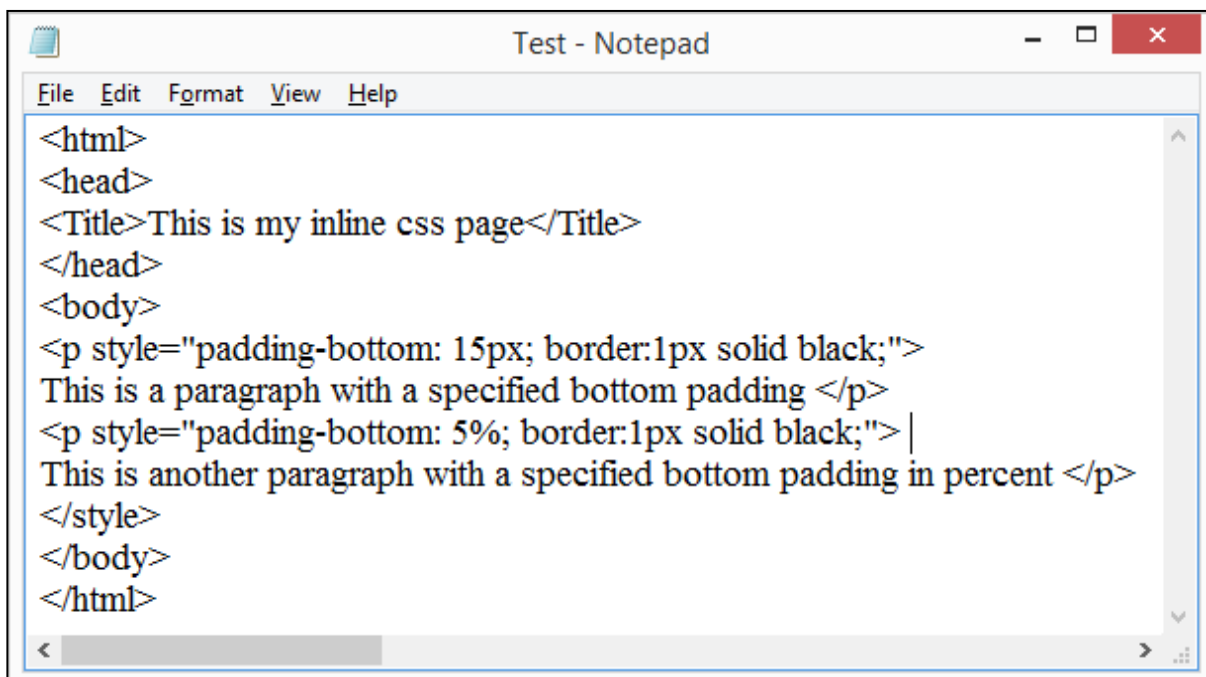
4.1 Padding using CSS

The padding property allows you to specify how much space should appear between the content of an element and its border. The value of this attribute should be either a length, a percentage, or the word inherits. If the value is inherit, it will have the same padding as its parent element. If a percentage is used, the percentage is of the containing box. The following CSS properties can be used to control lists. You can also set different values for the padding on each side of the box using the following properties:

- The padding-bottom specifies the bottom padding of an element.
- The padding-top specifies the top padding of an element.
- The padding-left specifies the left padding of an element.
- The padding-right specifies the right padding of an element.
- The padding serves as shorthand for the preceding properties.

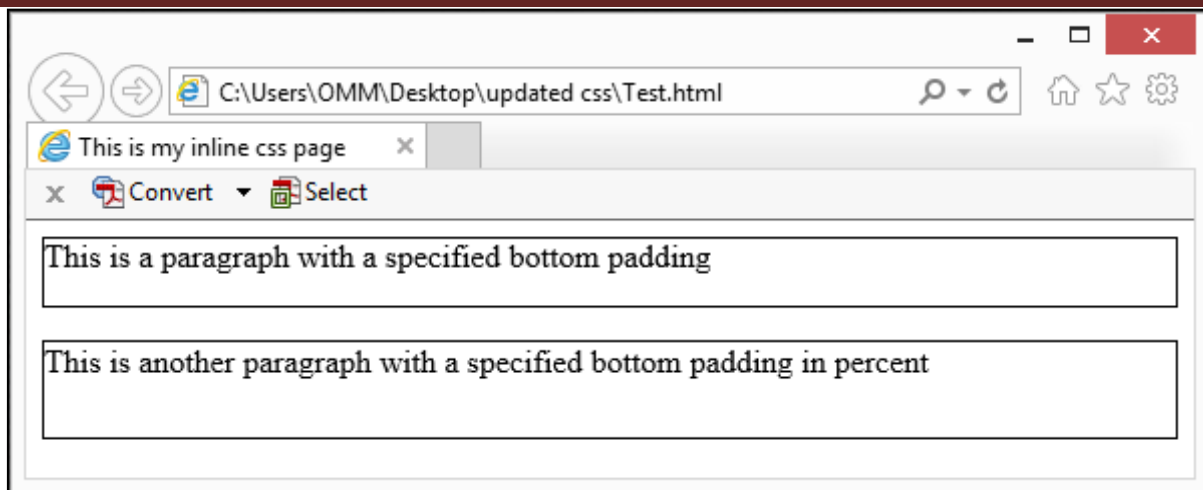
4.1.1 The padding-bottom Property

The padding-bottom property sets the bottom padding (space) of an element. This can take a value in terms of length or %. Here is an example:



```
<html>
<head>
<Title>This is my inline css page</Title>
</head>
<body>
<p style="padding-bottom: 15px; border:1px solid black;">
This is a paragraph with a specified bottom padding </p>
<p style="padding-bottom: 5%; border:1px solid black;"> |
This is another paragraph with a specified bottom padding in percent </p>
</style>
</body>
</html>
```

The above program will produce the following result



4.1.2 The padding-top Property

The padding-top property sets the top padding (space) of an element. This can take a value in terms of length of %. Here is an example:

```

Test - Notepad
File Edit Format View Help
<html>
<head>
<Title>This is my inline css page</Title>
</head>
<body>
<p style="padding-top: 15px; border:1px solid black;">
This is a paragraph with a specified top padding </p>
<p style="padding-top: 5%; border:1px solid black;">
This is another paragraph with a specified top padding in percent </p>
</body>
</html>
  
```

The above program will produce the following result:

