

SOUTH EAST ASIAN INSTITUTE OF TECHNOLOGY, INC.

National Highway, Crossing Rubber, Tupi, South Cotabato

**COLLEGE OF INFORMATION AND COMMUNICATION TECHNOLOGY**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**LEARNING MODULE**

**FOR**

**IT 123: WEB DEVELOPMENT 1**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

***WEEK 12***

**COURSE OUTLINE**

**COURSE CODE**  : IT 123

**TITLE**  : Web Development 1

**TARGET POPULATION** : All BS Information Technology Students

**INSTRUCTOR** : MS. CAROL KATE M. ESTACIO

**Overview:**

HTML stands for Hypertext Markup Language, and it is the most widely used language to write Web Pages.

• Hypertext refers to the way in which Web pages (HTML documents) are linked together. Thus, the link available on a webpage is 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 structure it to 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.

Now, HTML is being widely used to format web pages with the help of different tags available in HTML language.

**Content:**

**CSS**

* + **CSS Fonts**
  + **CSS Height/Width**
  + **CSS Links**

**Objectives:**

General Objective

* + Introduce the CSS font properties and its usage.
  + To know and understand the usage of CSS when it comes to height and width.
  + To know how to modify the design of links in your website.

**Instruction to the Learner**

Each chapter in this module contains a major lesson involving the basics of Web page coding and HTML editing tool. The units are characterized by continuity, and are arranged in such a manner that the present unit is related to the next unit. For this reason, you are advised to read this module. After each unit, there are exercises to be given. Submission of task given will be every Monday during your scheduled class hour.

**Specifying Typefaces**

*font-family*

The font family of a text is set with the font-family property. The font-family property should hold several font names as a "fallback" system. If the browser does not support the first font, it tries the next font, and so on. Start with the font you want, and end with a generic family, to let the browser pick a similar font in the generic family, if no other fonts are available. More than one font family is specified in a comma-separated list:

p {  
    font-family: "Times New Roman", Times, serif;  
}

*Note: If the name of a font family is more than one word, it must be in quotation marks, like: "Times New Roman".*

**Example**

<!DOCTYPE html>

<html>

<head>

<title>Font Family</title>

<style type="text/css">

body {

font-family: Georgia, Times, serif;}

h1, h2 {

font-family: Arial, Verdana, sans-serif;}

.credits {

font-family: "Courier New", Courier,

monospace;}

</style>

</head>

<body>

<h1>Briards</h1>

<p class="credits">by Ivy Duckett</p>

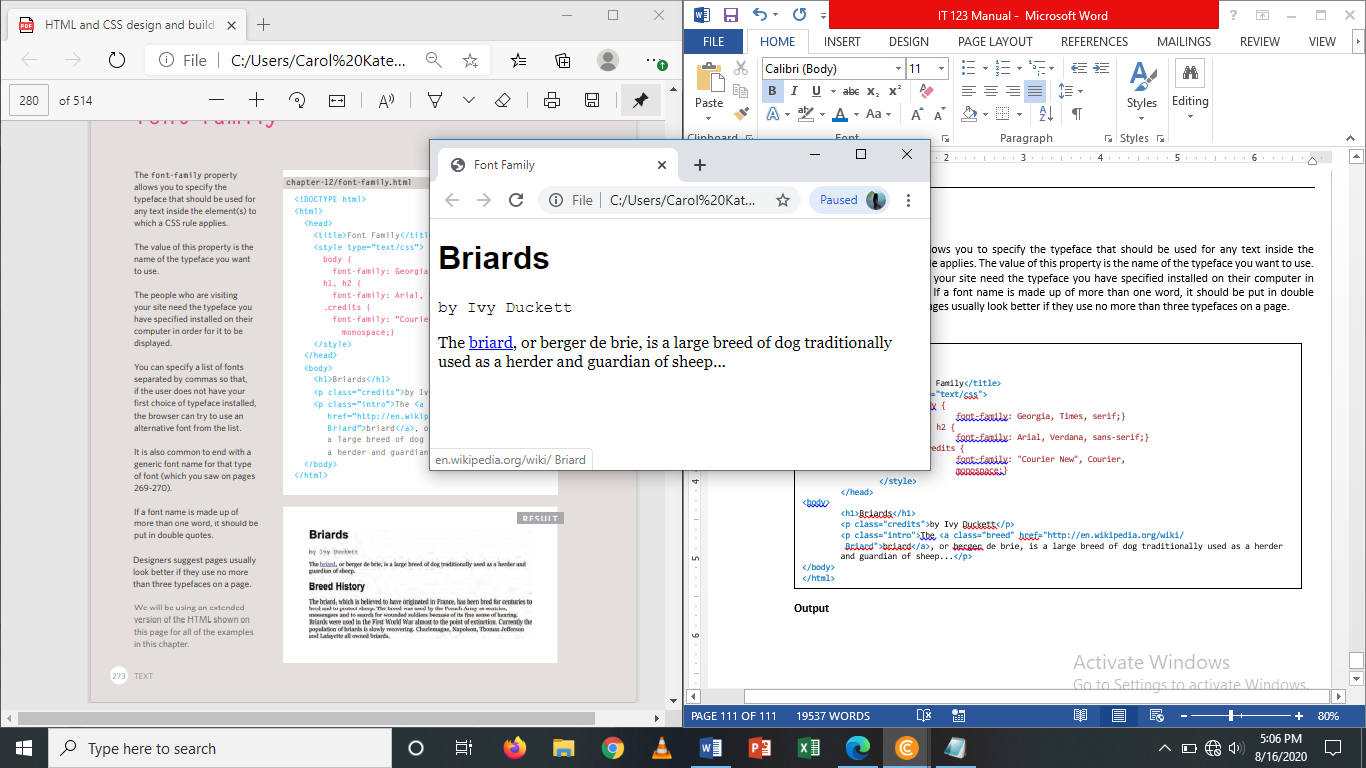
<p class="intro">The <a class="breed" href="http://en.wikipedia.org/wiki/

Briard">briard</a>, or berger de brie, is a large breed of dog traditionally used as a herder and guardian of sheep...</p>

</body>

</html>

**Output**



**Font Style**

The font-style property is mostly used to specify italic text. This property has three values:

* Normal **–** The text is shown normally.
* Italic – The text is shown in italics.
* Oblique – The text is “learning” (oblique is very similar to italic, but less supported)

**Example**

p.normal {  
    font-style: normal;  
}  
  
p.italic {  
    font-style: italic;  
}  
  
p.oblique {  
    font-style: oblique;  
}

**Size of type**

*font-size*

The font-size property sets the size of the text. Being able to manage the text size is important in web design. However, you should not use font size adjustments to make paragraphs look like headings, or headings look like paragraphs. Always use the proper HTML tags, like <h1> - <h6> for headings and <p> for paragraphs. The font-size value can be an absolute, or relative size.

* Absolute size:
  1. Sets the text to a specified size
  2. Does not allow a user to change the text size in all browsers (bad for accessibility reasons)
  3. Absolute size is useful when the physical size of the output is known.
* Relative size:
  1. Sets the size relative to surrounding elements
  2. Allows a user to change the text size in browsers.

There are several ways to specify the size of a font. The most common are:

* **Pixels**

Pixels are commonly used because they allow web designers very precise control over how much space their text takes up. The number of pixels is followed by the letters px.

* **Percentage**

The default size of text in browsers is 16px. So a size of 75% would be the equivalent of 12px, and 200% would be 32px.

* **EMS**

An em is equivalent to the width of a letter m.

**Example**

body {

font-family: Arial, Verdana, sans-serif;

font-size: 12px;}

h1 {

font-size: 200%;}

h2 {

font-size: 1.3em;}

*Note: If you do not specify a font size, the default size for normal text, like paragraphs, is 16px (16px=1em).*

**Font Weight**

The *font-weight* property specifies the weight of a font:

p.normal {  
    font-weight: normal;  
}  
  
p.thick {  
    font-weight: bold;  
}

**Font Variant**

The font-variant property specifies whether or not a text should be displayed in a small-caps font. In a small-caps font, all lowercase letters are converted to uppercase letters. However, the converted uppercase letters appears in a smaller font size than the original uppercase letters in the text.

p.normal {  
    font-variant: normal;  
}  
  
p.small {  
    font-variant: small-caps;  
}

**All CSS Font-Properties**

|  |  |
| --- | --- |
| Property | Description |
| font | Sets all the font properties in one declaration |
| font-family | Specifies the font family for text |
| font-size | Specifies the font size of text |
| font-style | Specifies the font style for text |
| font-variant | Specifies whether or not a text should be displayed in a small-caps font |
| font-weight | Specifies the weight of a font |

**CSS Height and Width Dimensions**

Setting height and width

The height and width properties are used to set the height and width of an element. The height and width can be set to auto (this is default. Means that the browser calculates the height and width), or be specified in length values, like px, cm, etc., or in percent (%) of the containing block.

**Example**

<!DOCTYPE html>

<html>

<head>

<style>

div {

width: 500px;

height: 100px;

border: 3px solid #73AD21;

}

</style>

</head>

<body>

<h2>Set height and width of an Element:</h2>

<div>

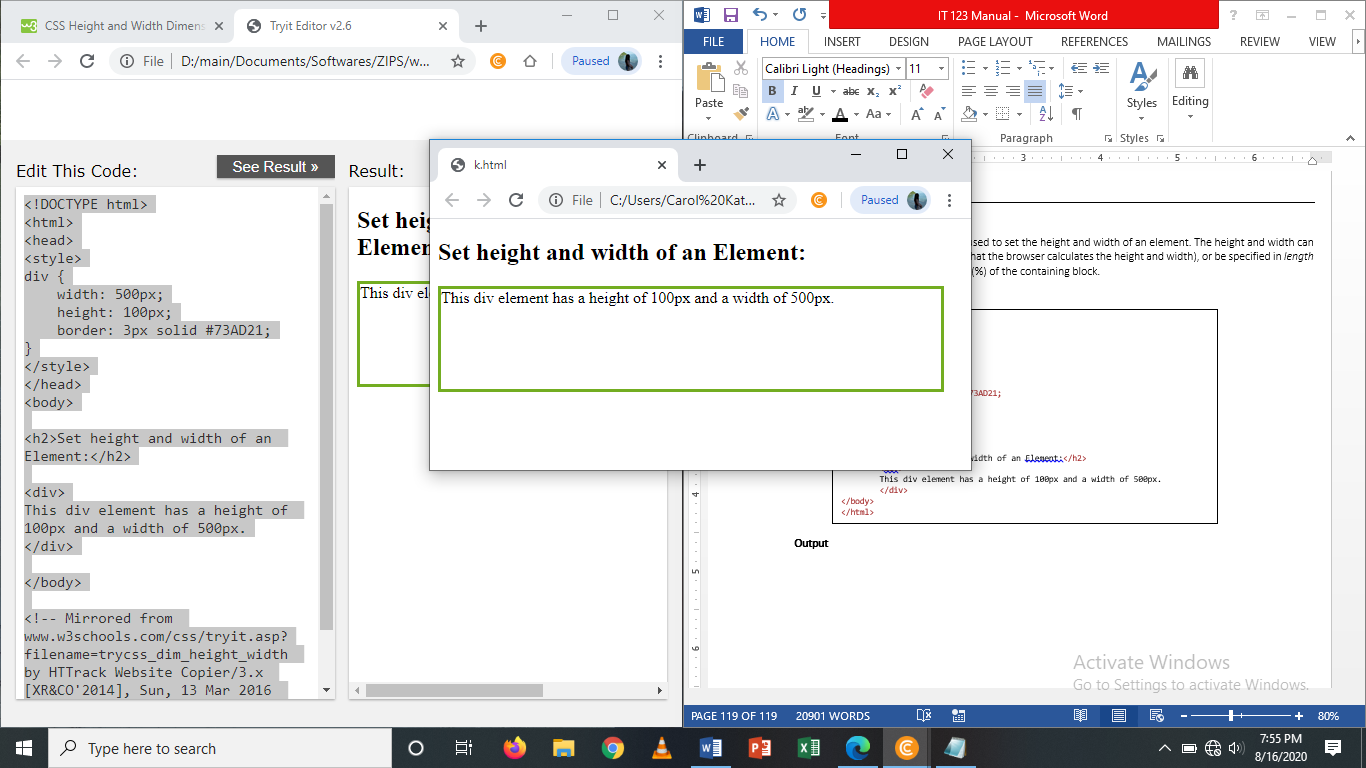
This div element has a height of 100px and a width of 500px.

</div>

</body>

</html>

**Output**



**Setting max-width**

The max-width property is used to set the maximum width of an element. The max-width can be specified in length values, like px, cm, etc., or in percent (%) of the containing block, or set to none (this is default. Means that there is no maximum width).

The problem with the <div> above occurs when the browser window is smaller than the width of the element (500px). The browser then adds a horizontal scrollbar to the page. Using max-width instead, in this situation, will improve the browser's handling of small windows. The following example shows a <div> element with a height of 100 pixels and a max-width of 500 pixels:

**Example**

<!DOCTYPE html>

<html>

<head>

<style>

div {

max-width: 500px;

height: 100px;

border: 3px solid #73AD21;

}

</style>

</head>

<body>

<h2>Set height and width of an Element:</h2>

<div>

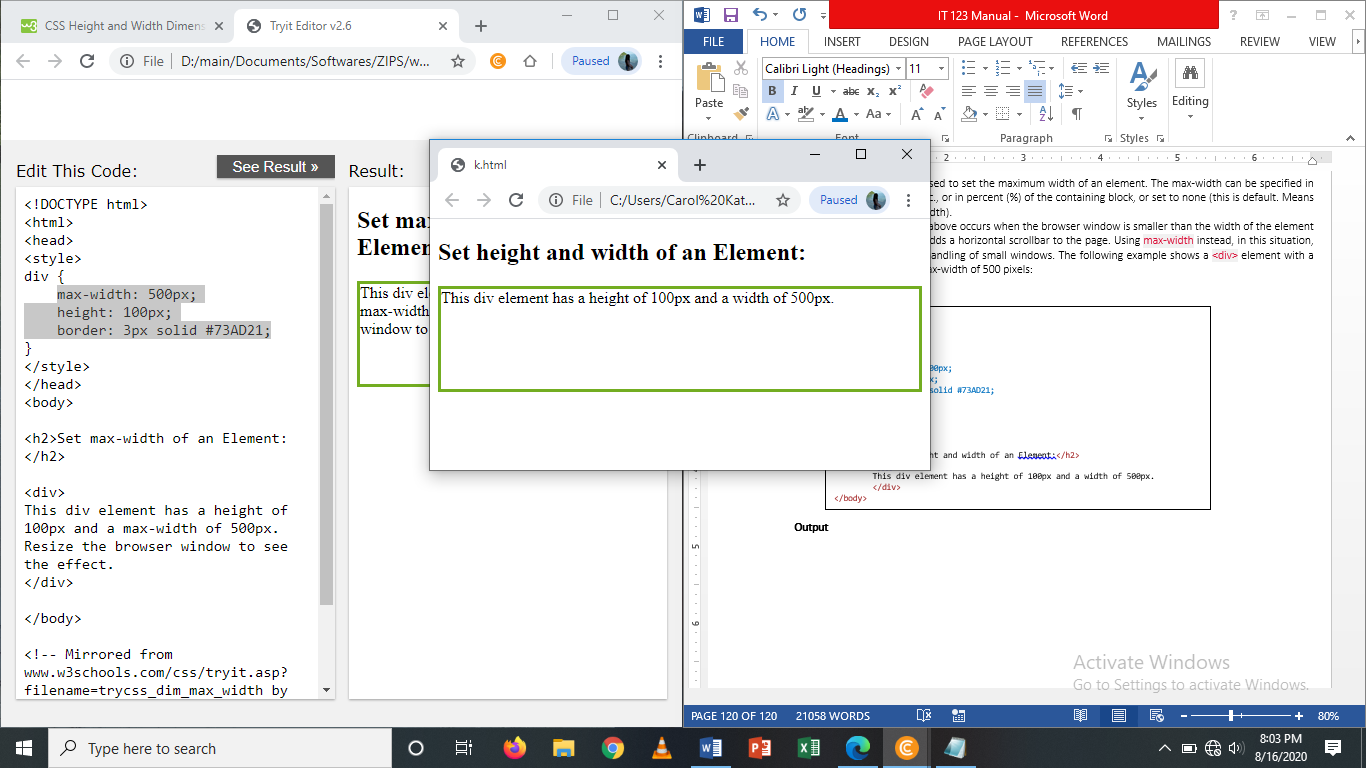
This div element has a height of 100px and a width of 500px.

</div>

</body>

</html>

**Output**



**All CSS Dimension Properties**

|  |  |
| --- | --- |
| Property | Description |
| height | Sets the height of an element |
| max-height | Sets the maximum height of an element |
| max-width | Sets the maximum width of an element |
| min-height | Sets the minimum height of an element |
| min-width | Sets the minimum width of an element |
| width | Sets the width of an element |

**CSS Styling Links**

**Styling Links**

Links can be styled with any CSS property (e.g. *color*, *font-family, background*, etc).

a {

color:hotpink;

}

Example

<!DOCTYPE html>

<html>

<head>

<style>

a {

color: hotpink;

}

</style>

</head>

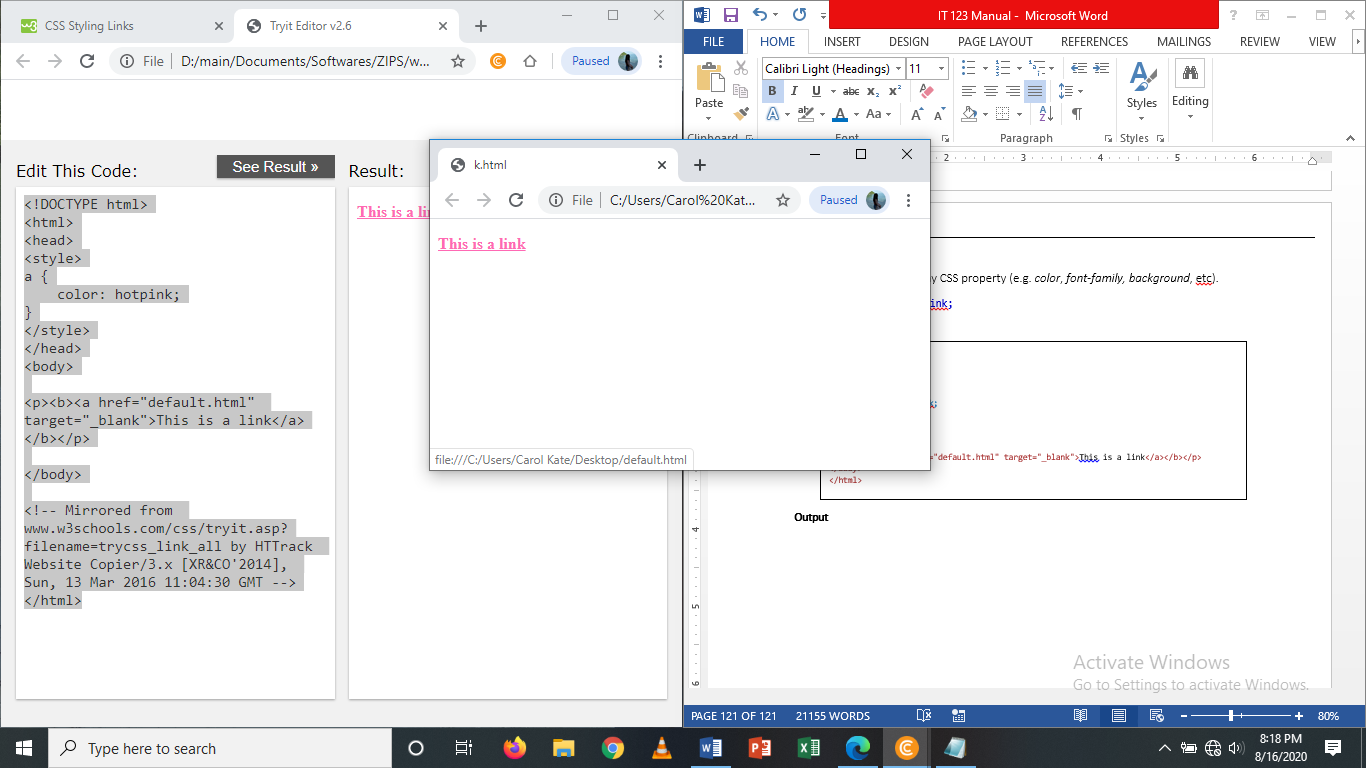
<body>

<p><b><a href="default.html" target="\_blank">This is a link</a></b></p>

</body>

</html>

**Output**



In addition, links can be styled differently depending on what**state** they are in.

The four links states are:

* a**:**link –a normal, universal link
* a:visited – a link the user has visited
* a:hover – a link when the user hover the mouse over it
* a:active – a link the moment it is clicked

/\* unvisited link \*/  
a:link {  
    color: red;  
}  
  
/\* visited link \*/  
a:visited {  
    color: green;  
}  
  
/\* mouse over link \*/  
a:hover {  
    color: hotpink;  
}  
  
/\* selected link \*/  
a:active {  
    color: blue;  
}

**Example**

<!DOCTYPE html>

<html>

<head>

<style>

/\* unvisited link \*/

a:link {

color: red;

}

/\* visited link \*/

a:visited {

color: green;

}

/\* mouse over link \*/

a:hover {

color: hotpink;

}

/\* selected link \*/

a:active {

color: blue;

}

</style>

</head>

<body>

<p><b><a href="default.html" target="\_blank">This is a link</a></b></p>

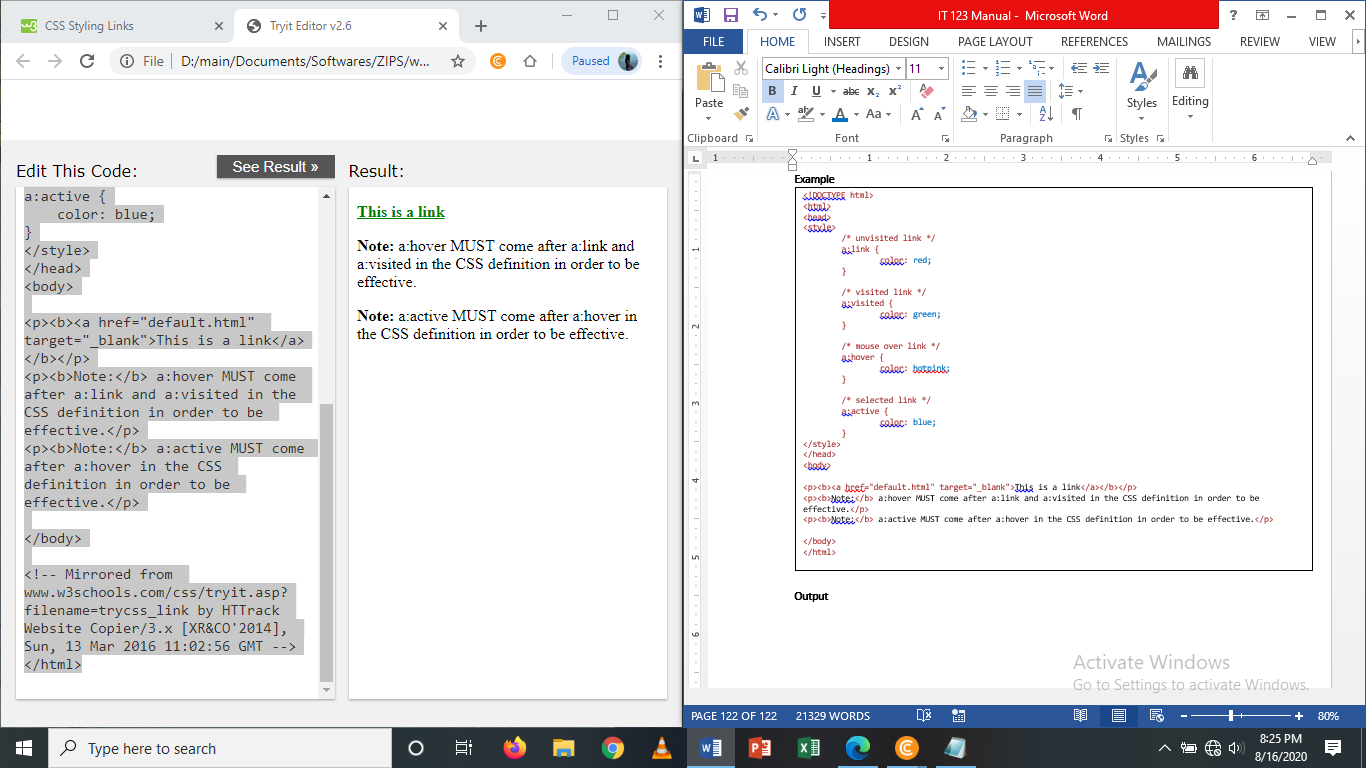
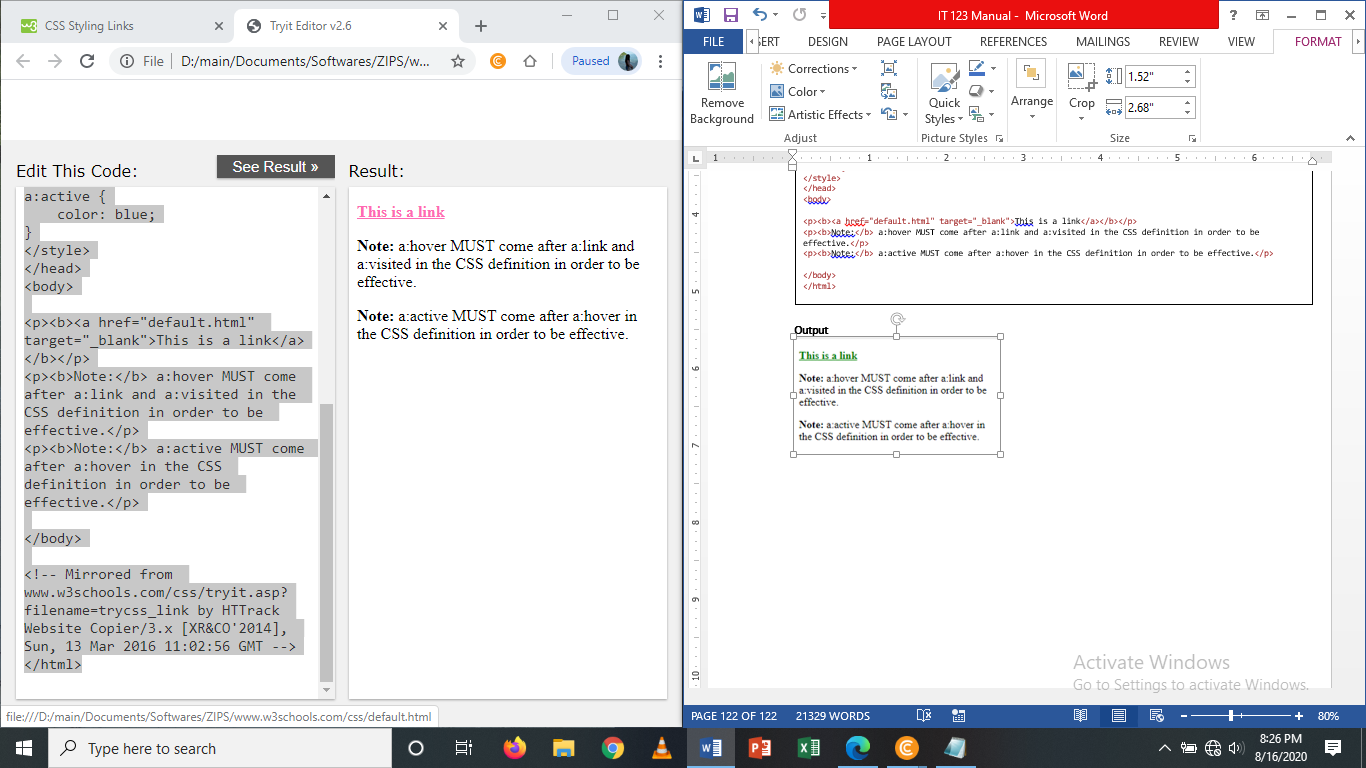
<p><b>Note:</b> a:hover MUST come after a:link and a:visited in the CSS definition in order to be effective.</p>

<p><b>Note:</b> a:active MUST come after a:hover in the CSS definition in order to be effective.</p>

</body>

</html>

**Output**



When setting the style for several link states, there are some order rules:

* a:hover MUST come after a:link and a:visited
* a:active MUST come after a:hover

**Text Decoration**

The *text-decoration* property is mostly used to remove underlines from links:

a:link {  
    text-decoration: none;}  
  
a:visited {  
    text-decoration: none;}  
  
a:hover {  
    text-decoration: underline;}  
  
a:active {  
    text-decoration: underline;}

**Example**

<!DOCTYPE html>

<html>

<head>

<style>

a:link {

text-decoration: none;

}

a:visited {

text-decoration: none;

}

a:hover {

text-decoration: underline;

}

a:active {

text-decoration: underline;

}

</style>

</head>

<body>

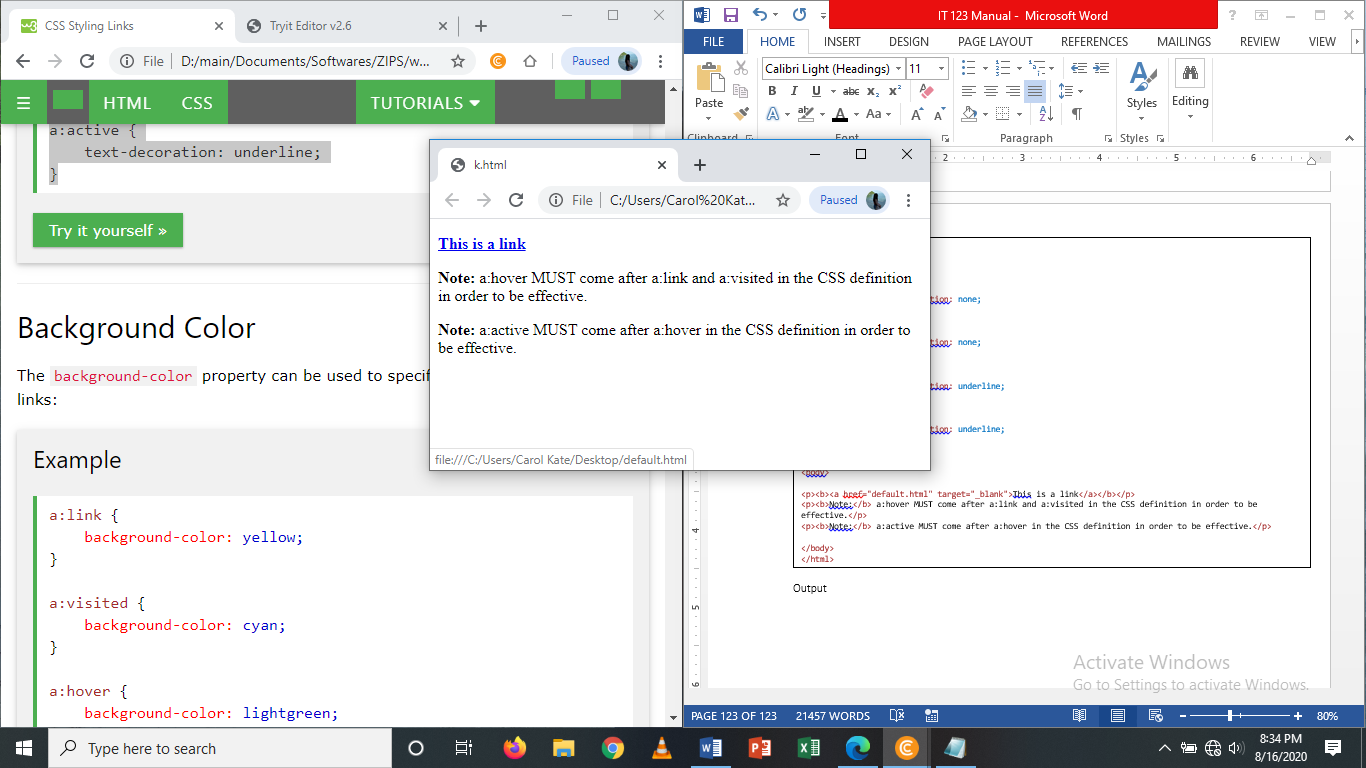
<p><b><a href="default.html" target="\_blank">This is a link</a></b></p>

<p><b>Note:</b> a:hover MUST come after a:link and a:visited in the CSS definition in order to be effective.</p>

<p><b>Note:</b> a:active MUST come after a:hover in the CSS definition in order to be effective.</p>

</body>

</html>

**Output**

**Background Color**

The *background-color* property can be used to specify a background color for links:

a:link {  
    background-color: yellow;  
}  
  
a:visited {  
    background-color: cyan;  
}  
  
a:hover {  
    background-color: lightgreen;  
}  
  
a:active {  
    background-color: hotpink;  
}

Example

<!DOCTYPE html>

<html>

<head>

<style>

a:link {

background-color: lightblue;

}

a:visited {

background-color: cyan;

}

a:hover {

background-color: lightgreen;

}

a:active {

background-color: hotpink;

}

</style>

</head>

<body>

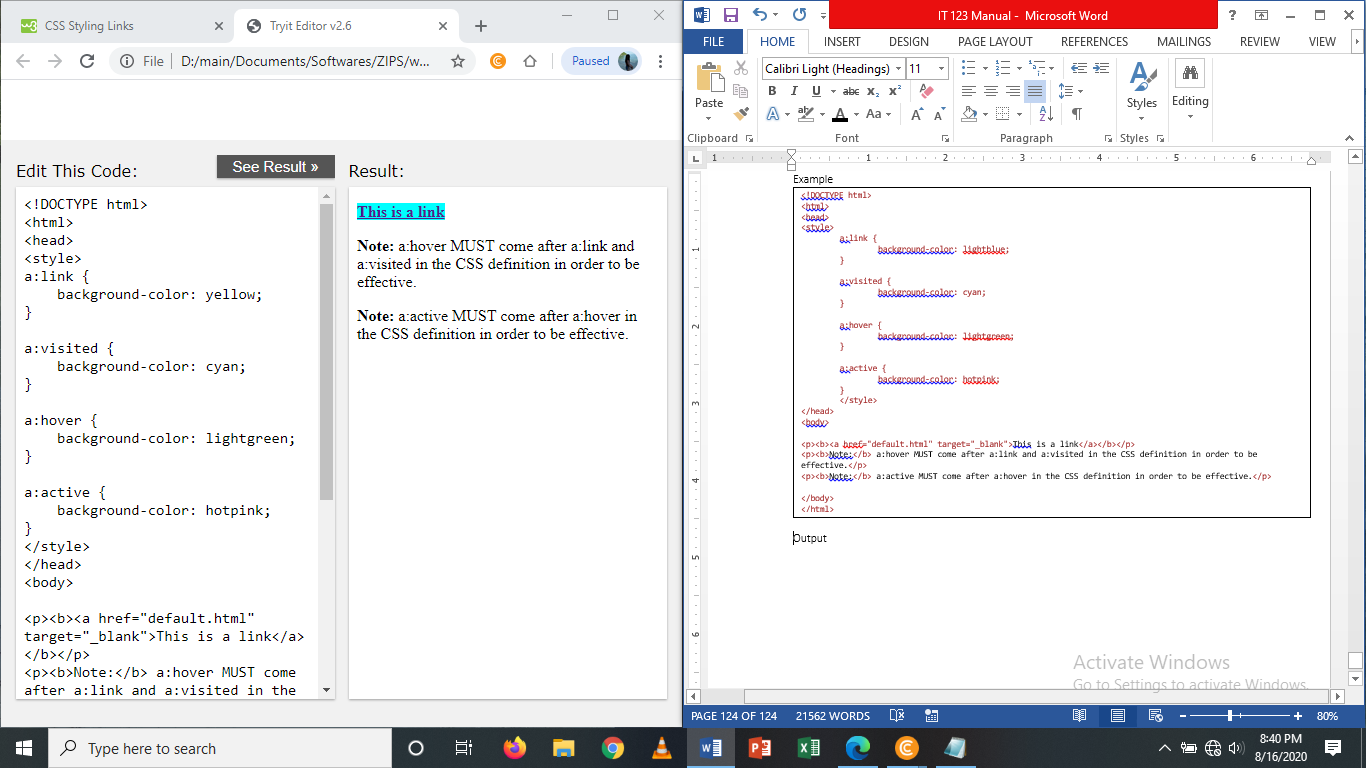
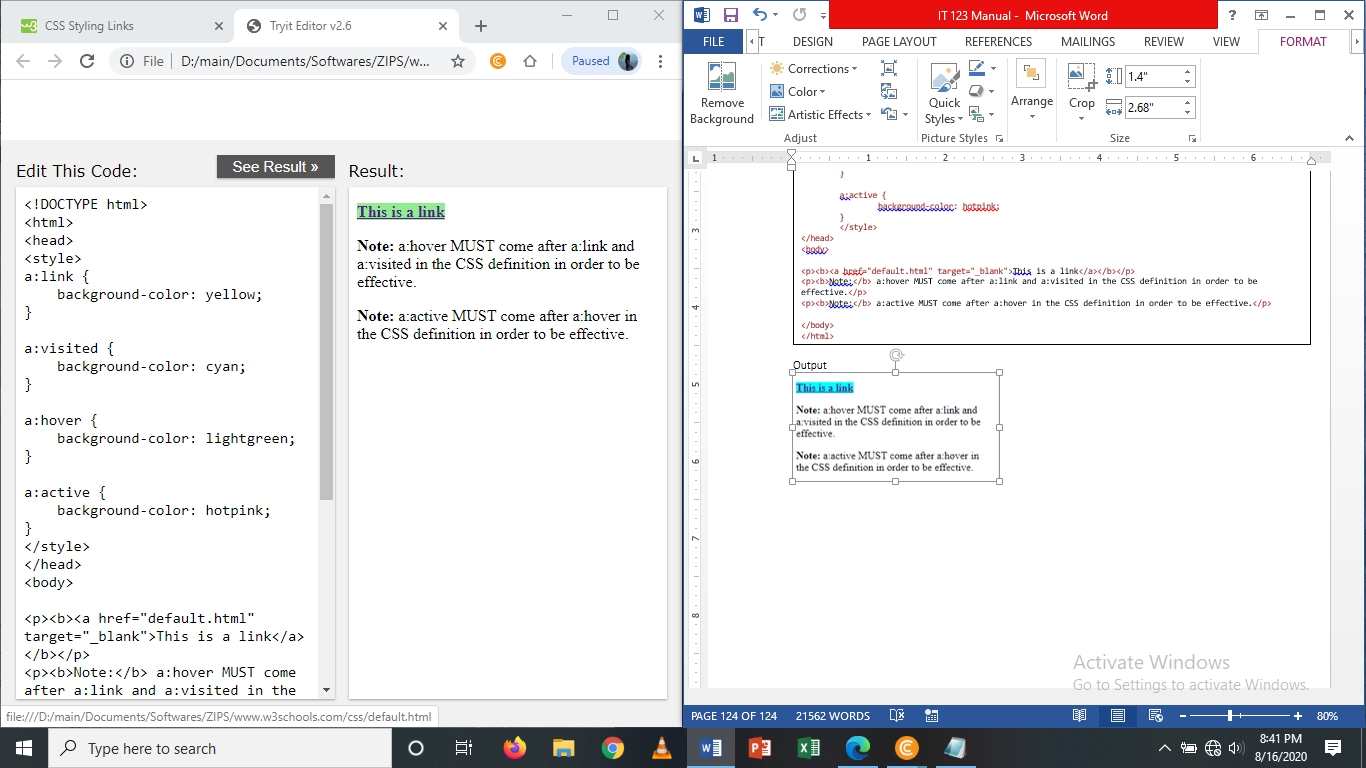
<p><b><a href="default.html" target="\_blank">This is a link</a></b></p>

<p><b>Note:</b> a:hover MUST come after a:link and a:visited in the CSS definition in order to be effective.</p>

<p><b>Note:</b> a:active MUST come after a:hover in the CSS definition in order to be effective.</p>

</body>

</html>

**Output**

This example demonstrate a more advanced example where we combine several CSS properties to display links as boxes/buttons:

Example

<!DOCTYPE html>

<html>

<head>

<style>

a:link, a:visited {

background-color: #f44336;

color: white;

padding: 14px 25px;

text-align: center;

text-decoration: none;

display: inline-block;

}

a:hover, a:active {

background-color: lightblue;

}

</style>

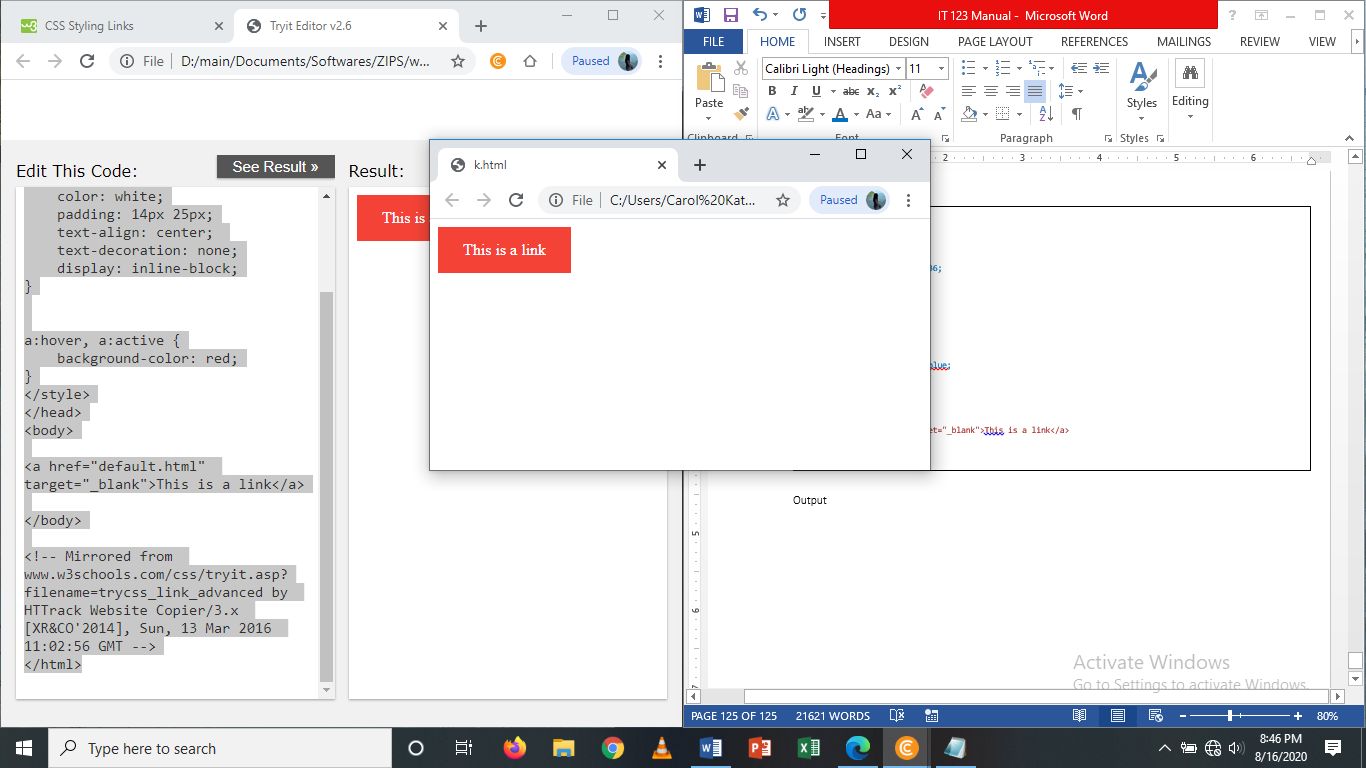
</head>

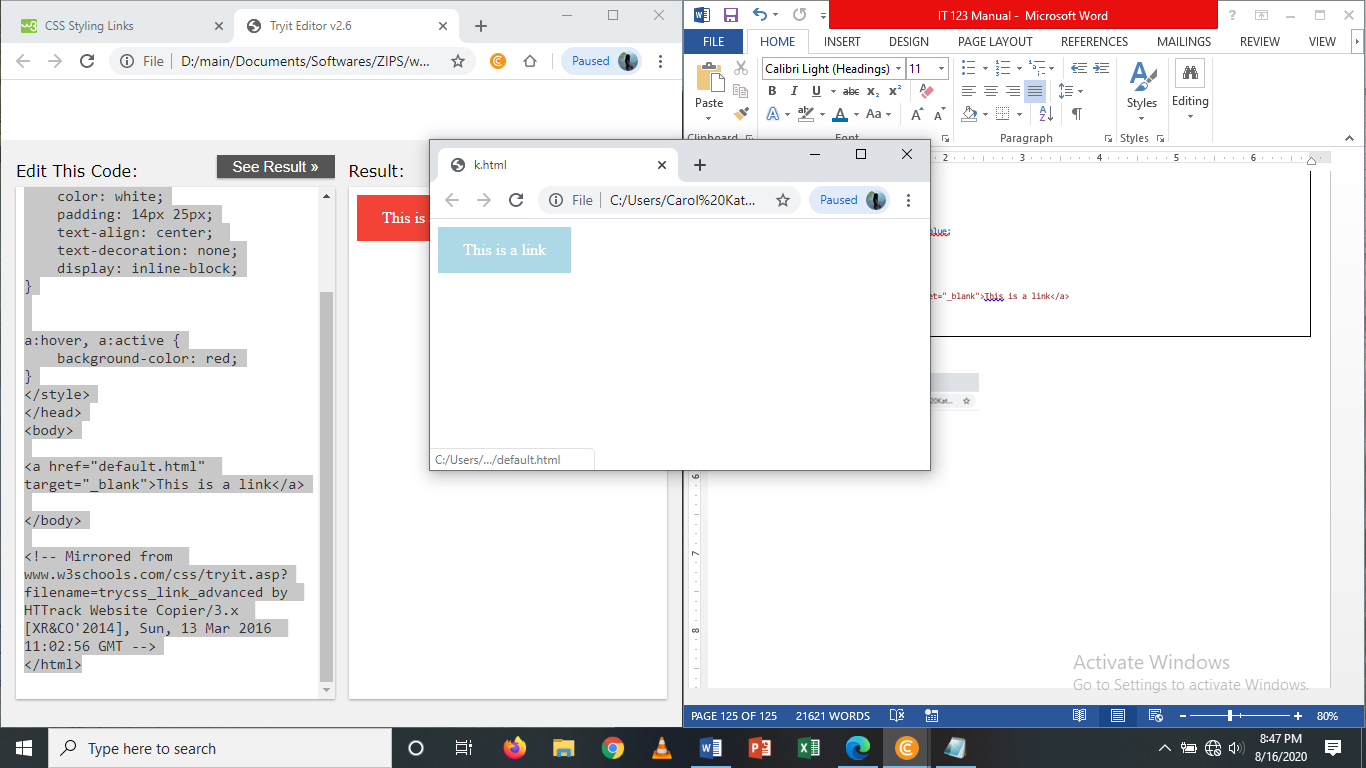
<body>

<a href="default.html" target="\_blank">This is a link</a>

</body>

</html>

Output



**Test Yourself**

1. Set the font family for the page to “Courier New”, and the font family for <h1> to “Verdana”

<!DOCTYPE html>  
<html>

<head>

<style>

</style>

</head>  
<body>

<h1>This is a Heading</h1>

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

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

</body>  
</html>

1. With the font property: set the <p> to “italic”, “20px” and “Verdana”.

<!DOCTYPE html>  
<html>

<head>

</head>  
<body>

<h1>This is a Heading</h1>

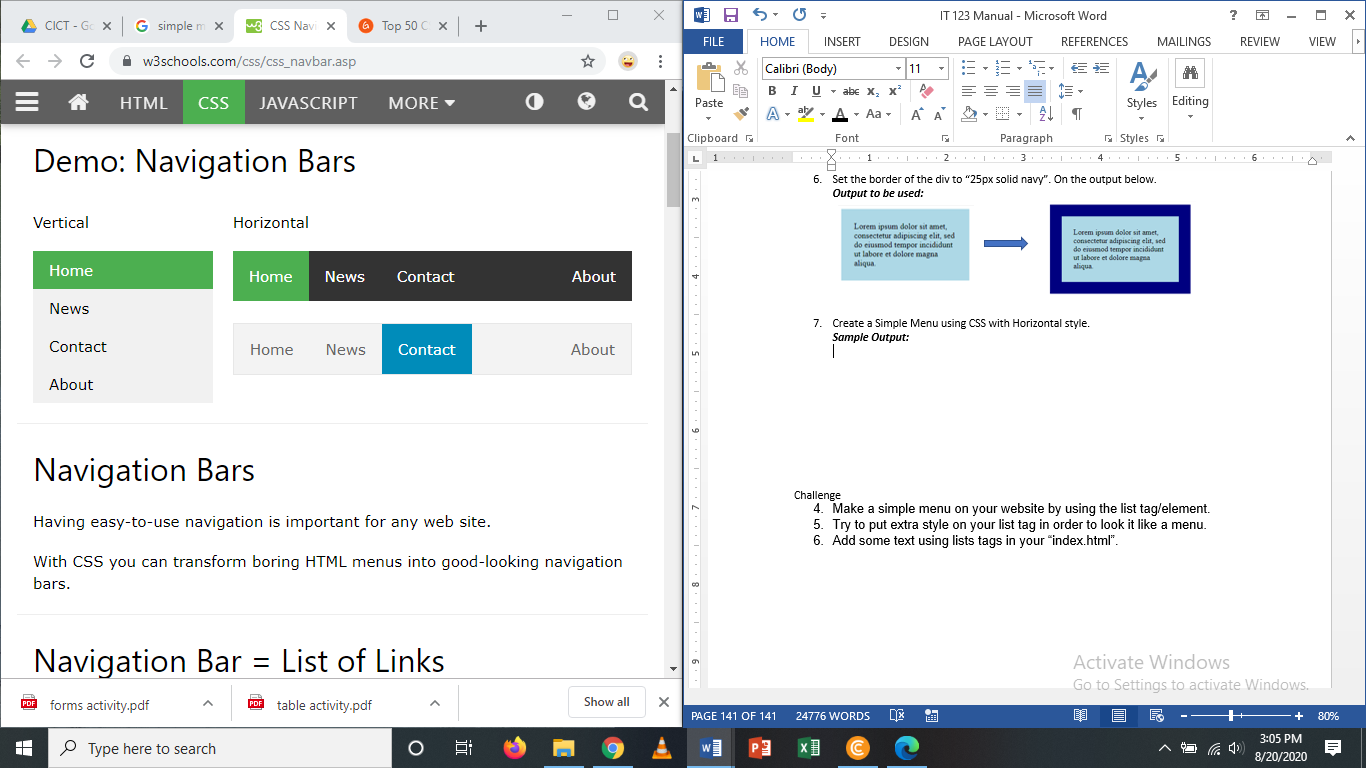
<p>This is a paragraph.</p>

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

</body>  
</html>

1. Create a Simple Menu using CSS with Horizontal style.

***Sample Output:***



**Challenge**

1. Open your “index.html” file.
2. On the said file put a menu using list.
3. After creating a horizontal list add some designs using CSS in order to make it as a real Menu Link containing the following list (Home, Gallery (Multimedia), About Us, Contacts, Others) you can add additional list if you would want to.