CSCI 1120-Introduction to Computing

CSS: Cascading Style Sheets

Xin Yang

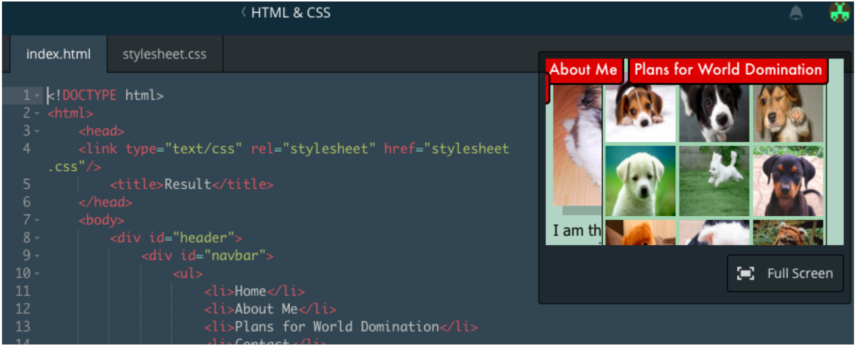
Jan 9th, 2019

What is CSS?

- CSS stands for Cascading Style Sheets.
- CSS defines how HTML elements are to be displayed.
- CSS saves a lot of work. It can control the layout of multiple web pages all at once.

HTML&CSS example

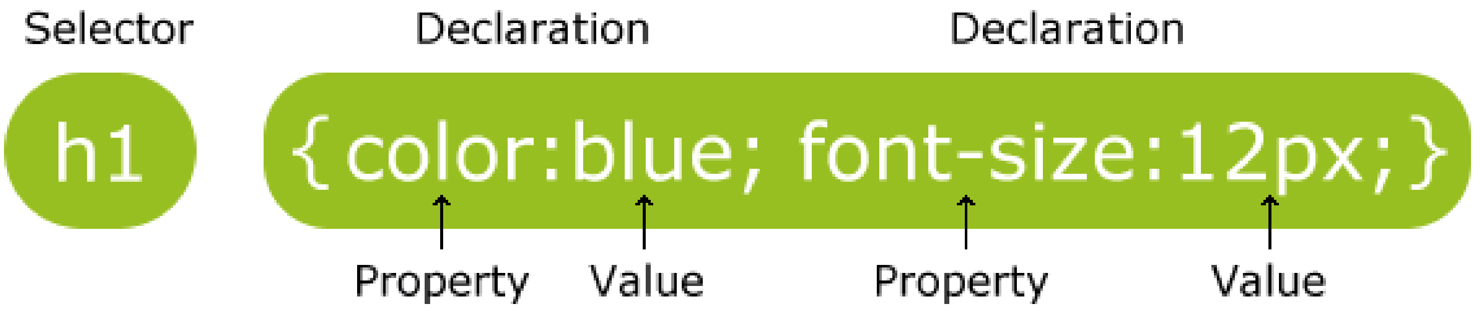
- CSS contains all styling information: where HTML elements should go, what color they should be, how big they should be, and more.



```
<!DOCTYPE html>
<html>
    <head>
    <link type="text/css" rel="stylesheet" href="stylesheet.css"/>
        <title>Result</title>
    </head>
    <body>
        <div id="header">
            <div id="navbar">
                <ul>
                    <li>Home</li>
                    <li>About Me</li>
                    <li>Plans for World Domination</li>
                    <li>Contact</li>
```

CSS Syntax

A CSS rule set consists of a selector and a declaration block:



```
h1 {color:blue; font-size:12px;}
```

CSS Syntax

- The selector points to the HTML element you want to style.
- The declaration block contains one or more declarations separated by semicolons.

The importance of semicolons

- As you start adding more and more property-value pairs for each CSS selector, it's important to remember to put a semicolon (;) at the end of each line.
- The semicolon tells CSS that one property-value pair is over and it's time to move on to the next one.
- Without semicolons, it'll become confused and your page won't look right.
- Also, don't forget: all property-value pairs for a selector are surrounded by curly braces ().

CSS Syntax

- Each declaration includes a CSS property name and a value, separated by a colon.
- A CSS declaration always ends with a semicolon, and declaration blocks are surrounded by curly braces.

CSS How To ...

When a browser reads a style sheet, it will format the HTML document according to the information in the style sheet.

There are three ways of inserting a style sheet:

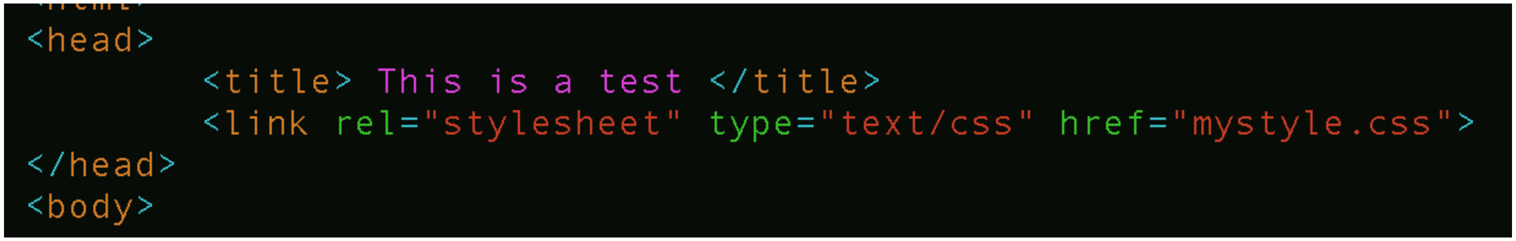
- External style sheet.
- Internal style sheet.
- Inline style.

External Style Sheet

- With an external style sheet, you can change the look of an entire website by changing just one file!

- Each page must include a reference to the external style sheet file inside the <link> element. The <link> element goes inside the <head> section:

External Style Sheet



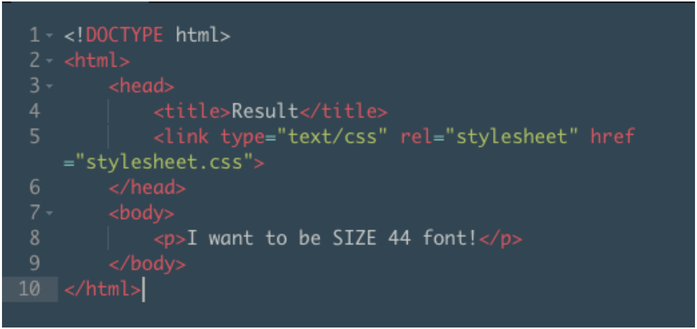
```
<html>
<head>
        <title> This is a test </title>
        <link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
<body>
```

How to make sure HTML file can see the external CSS information?

You do this by putting a <**link**> tag between the <head>...</head> tags of the HTML page. Your <link> tag needs three attributes:

- A **type** attribute that should be always be equal to **"text/css"**.
- A **rel** attribute that should be always be equal to **"stylesheet"**.
- A **href** attribute that should point to the web address of your CSS file.

How to make sure HTML file can see the external CSS information?



```
<!DOCTYPE html>
<html>
    <head>
        <title>Result</title>
        <link type="text/css" rel="stylesheet" href="stylesheet.css">
    </head>
    <body>
        <p>I want to be SIZE 44 font!</p>
    </body>
</html>
```

External Style Sheet

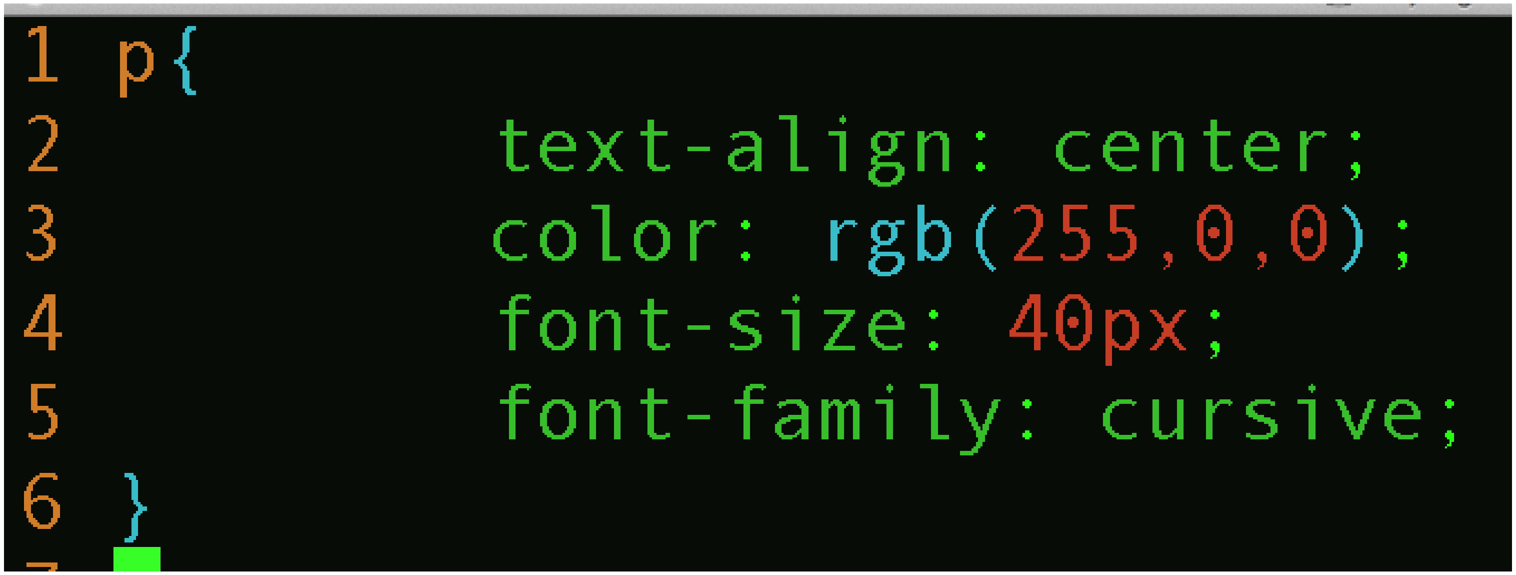
- An external style sheet can be written in any text editor.

- The file should not contain any html tags.

- The style sheet file must be saved with a .css extension.

External Style Sheet

- Here is how the ”mystyle.css” looks:

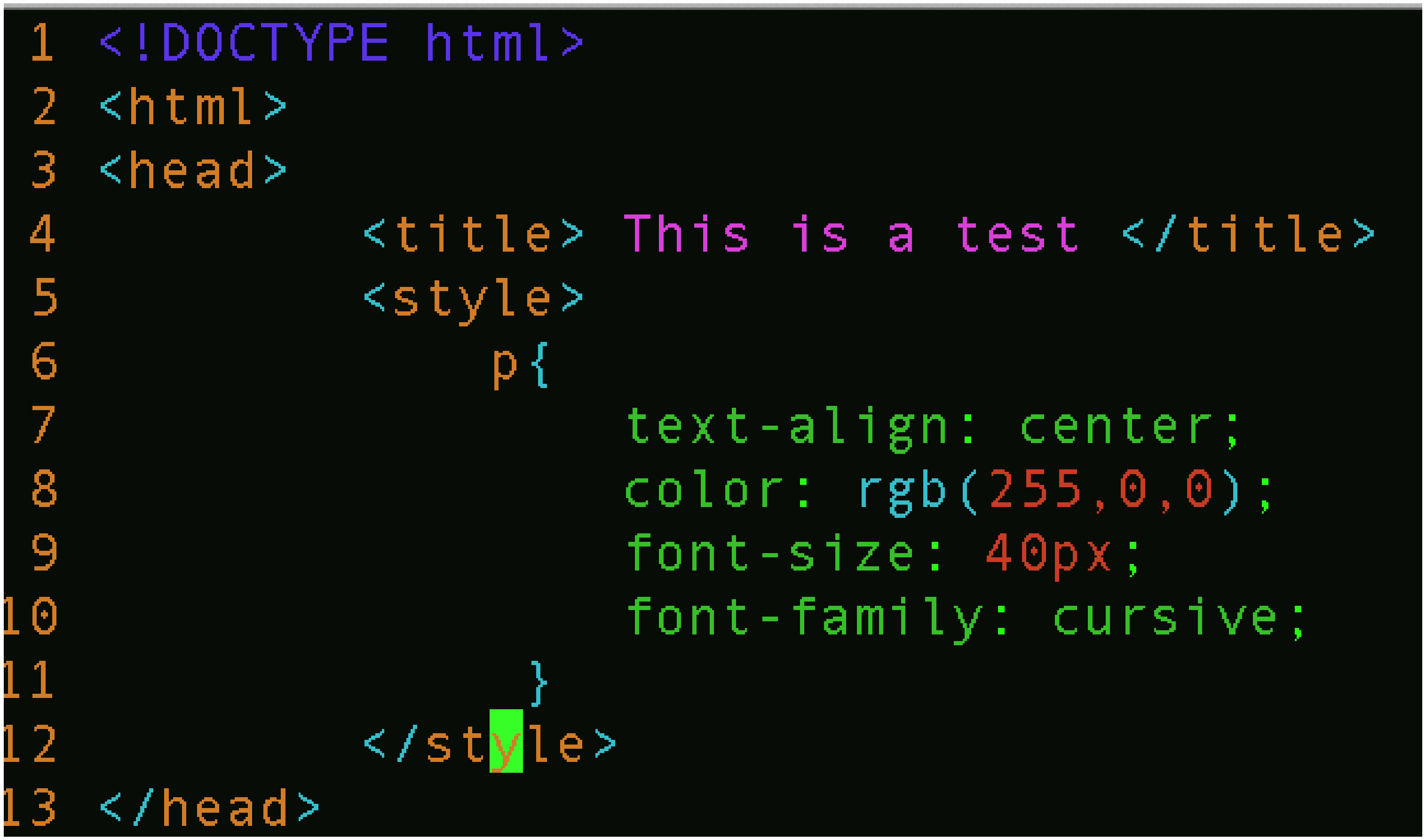


```css
p{
        text-align: center;
        color: rgb(255,0,0);
        font-size: 40px;
        font-family: cursive;
}
```

Internal Style Sheet

- An internal style sheet may be used if one single page has a unique style.
- Internal styles are defined within the <style> element, inside the <head> section of an HTML page:

Internal Style Sheet



```
<!DOCTYPE html>
<html>
<head>
        <title> This is a test </title>
        <style>
            p{
                text-align: center;
                color: rgb(255,0,0);
                font-size: 40px;
                font-family: cursive;
             }
        </style>
</head>
```

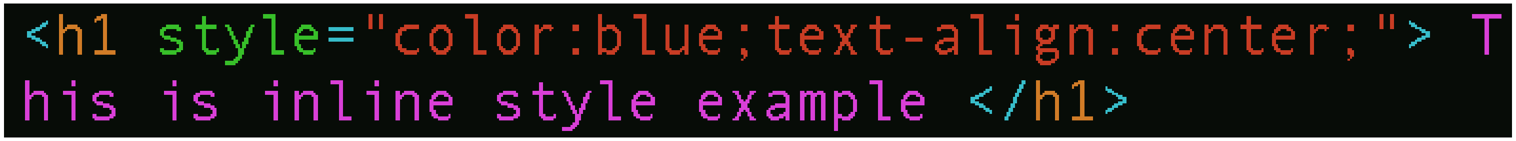
Inline style

- An inline style may be used to apply a unique style for a single element.

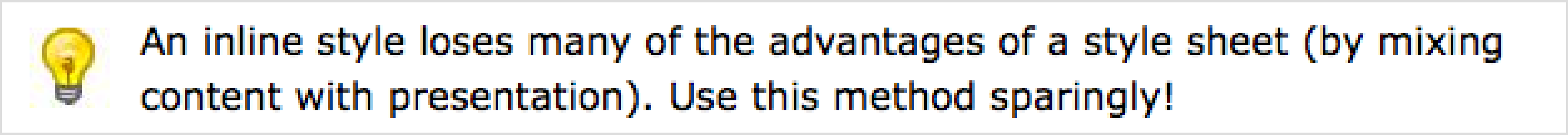
- To use inline styles, add the style attribute to the relevant tag.

- The style attribute can contain any CSS property.

Inline style



```
<h1 style="color:blue;text-align:center;"> This is inline style example </h1>
```



An inline style loses many of the advantages of a style sheet (by mixing content with presentation). Use this method sparingly!

Cascading Order

- What style will be used when there is more than one style specified for an HTML element?

Cascading Order

- Generally speaking we can say that all the styles will "cascade" into a new "virtual" style sheet by the following rules, where number one has the highest priority:
  - 1. Inline style (inside an HTML element).
  - 2. External and internal style sheets (in the head section)
  - 3. Browser default

Cascading Order

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

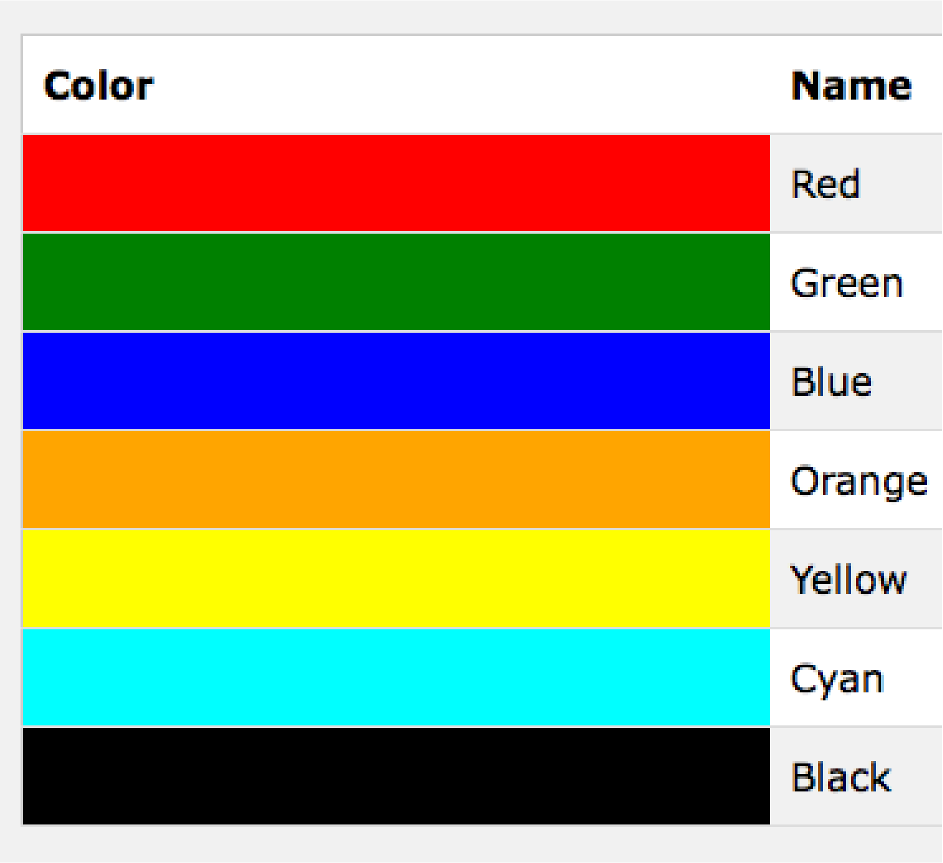
CSS Colors

Colors in CSS are most often specified by:

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

Color Names

Colors set by using color names:

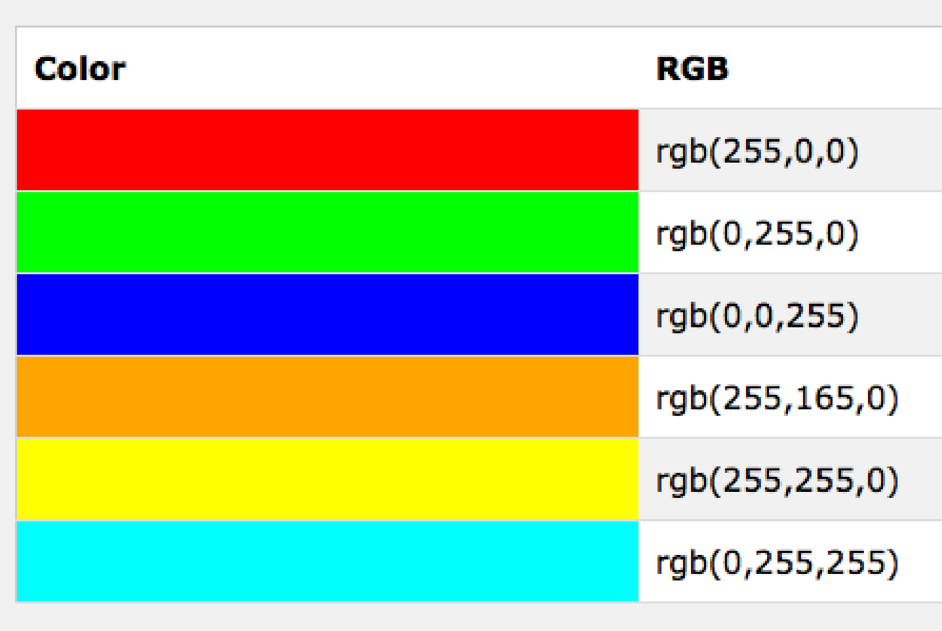


Color	Name
🟥	Red
🟩	Green
🟦	Blue
🟧	Orange
🟨	Yellow
🟦	Cyan
⬛	Black

HTML and CSS supports 140 standard color names.

RGB (Red, Green, Blue)

RGB color values can be specified using this formula: rgb(red, green, blue):

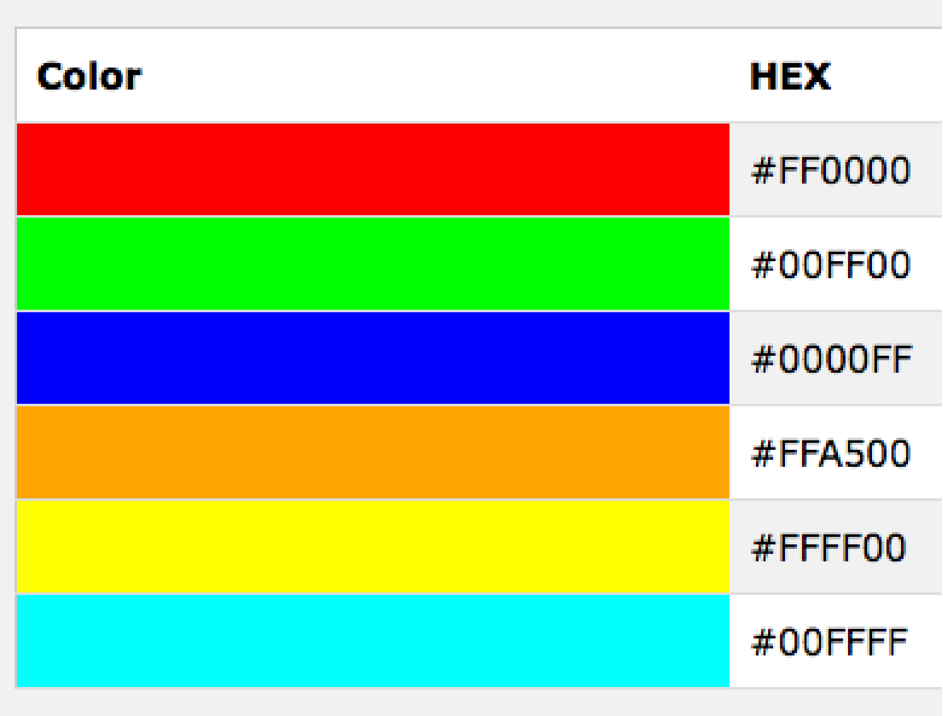


Color	RGB
■	rgb(255,0,0)
■	rgb(0,255,0)
■	rgb(0,0,255)
■	rgb(255,165,0)
■	rgb(255,255,0)
■	rgb(0,255,255)

HTML Color Picker.

Hexadecimal Colors

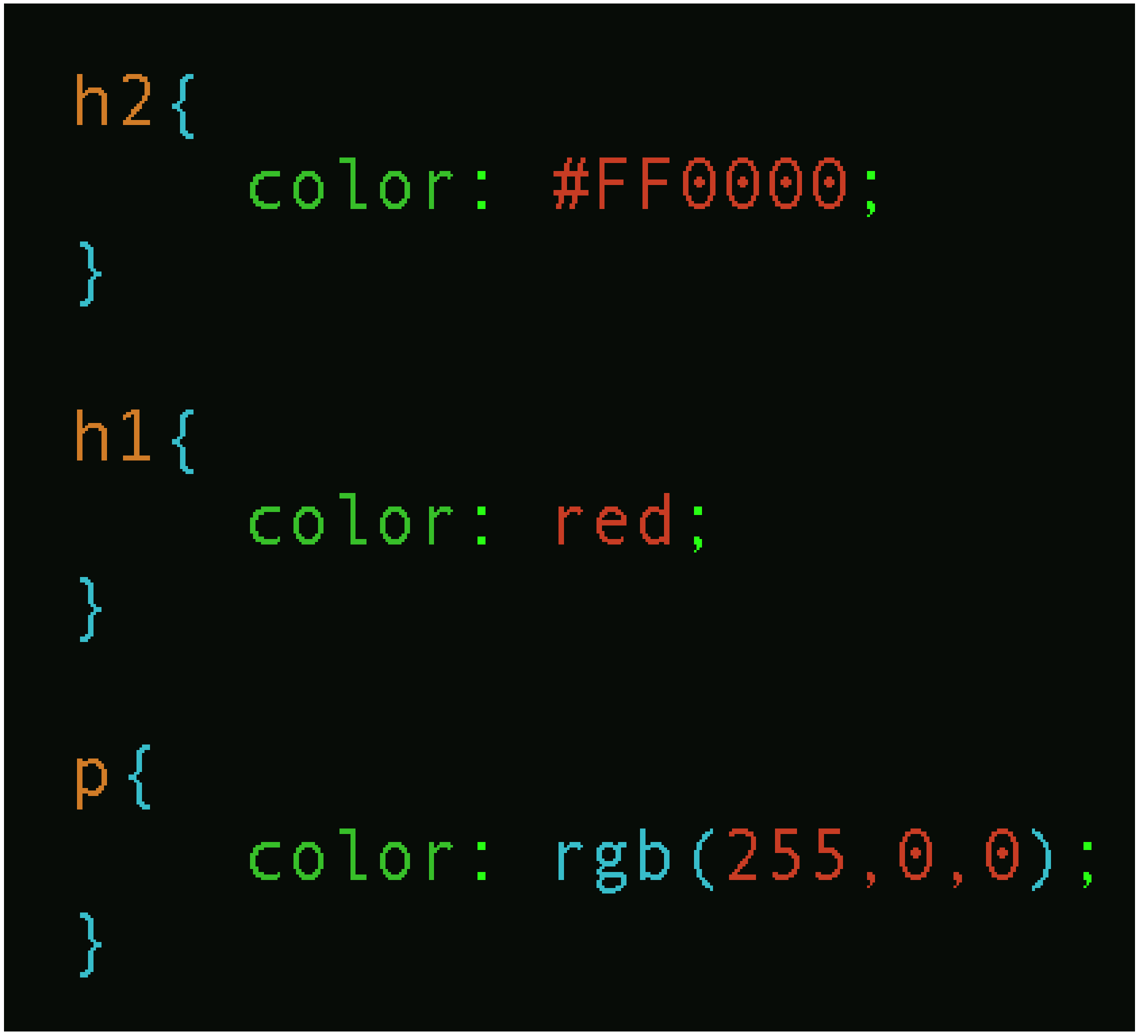
RGB values can also be specified using hexadecimal color values in the form: #RRGGBB, where RR (red), GG (green) and BB (blue) are hexadecimal values between 00 and FF (same as decimal 0-255):



Color	HEX
🟥	#FF0000
🟩	#00FF00
🟦	#0000FF
🟧	#FFA500
🟨	#FFFF00
■	#00FFFF

HTML Color Picker.

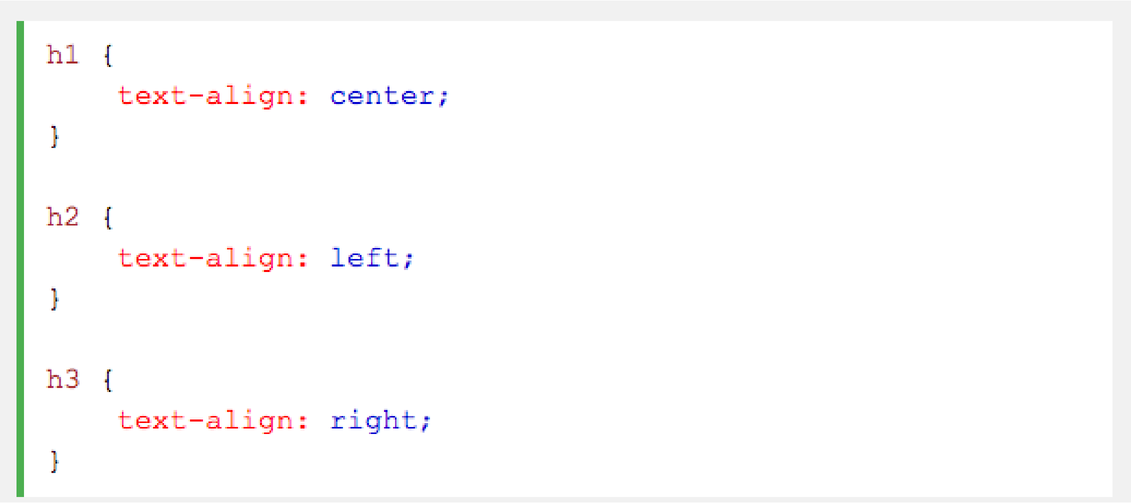
Text Colors



```
h2{
    color: #FF0000;
}

h1{
    color: red;
}

p{
    color: rgb(255,0,0);
}
```

Text Alignment

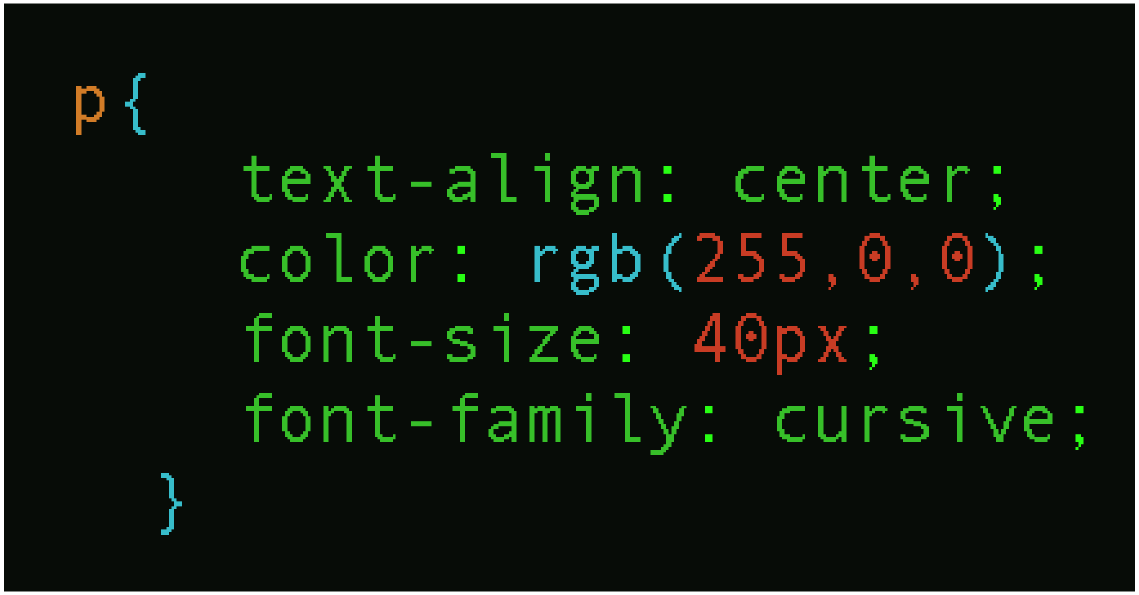
- The text-align property is used to set the horizontal alignment of a text.
- A text can be left or right aligned, centered, or justified.
- The following example shows center aligned, and left and right aligned text:



```
h1 {
    text-align: center;
}

h2 {
    text-align: left;
}

h3 {
    text-align: right;
}
```

Font Size

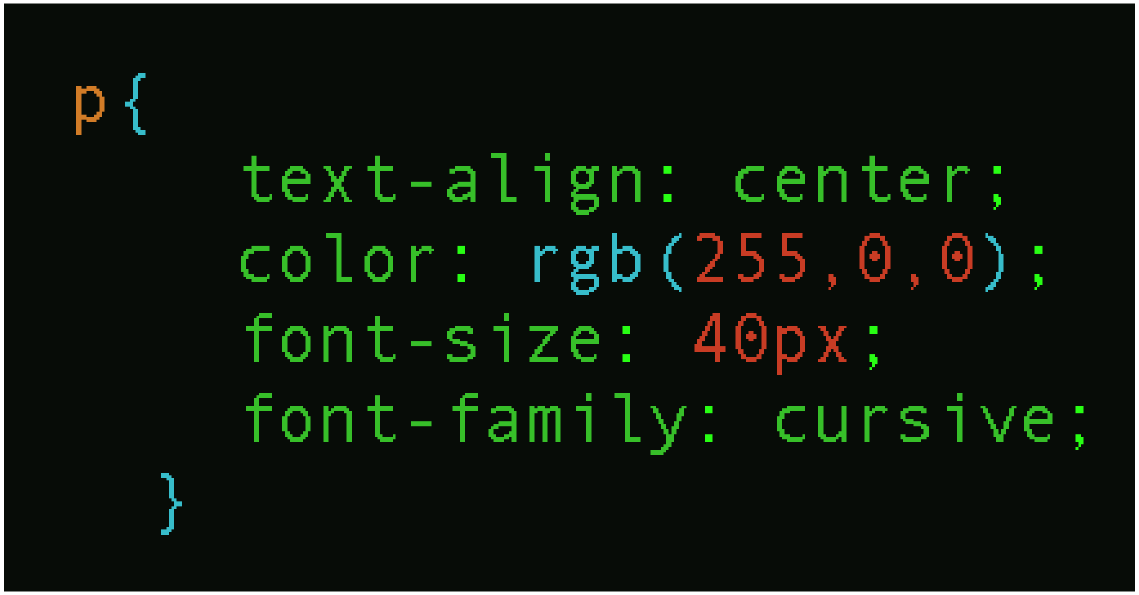
- The font-size property sets the size of the text.



```
p{
    text-align: center;
    color: rgb(255,0,0);
    font-size: 40px;
    font-family: cursive;
  }
```

Font Family

- The font-family property sets the font family of the text.



```
p{
    text-align: center;
    color: rgb(255,0,0);
    font-size: 40px;
    font-family: cursive;
  }
```

font-family

- Most computers will understand popular fonts like Verdana, Courier, and Garamond, but each individual computer has different fonts installed on it.
- CSS has some built-in defaults meant to ensure your users see what you intend. They are:
  - **serif**: A font with little decorative bits on the ends of the strokes that make up letters. Do a search on "serif" to see what we mean.
  - **sans-serif**: A plain-looking font, like this one. It doesn't have the little doohickies on the ends of letters like a serif font does.
  - **cursive**:A scripty font! It looks like cursive writing.

font-family backup values

- You don't have to jump straight to a default value like cursive or sans-serif: you can tell CSS to try several, going from one to the next if the one you want isn't available.

- For example, if you write:
  p{ font-family: Tahoma, Verdana, sans-serif; }

- CSS will first try to apply Tahoma to your paragraphs. If the user's computer doesn't have that font, it will try Verdana next, and if that doesn't work, it will show a default sans-serif font.

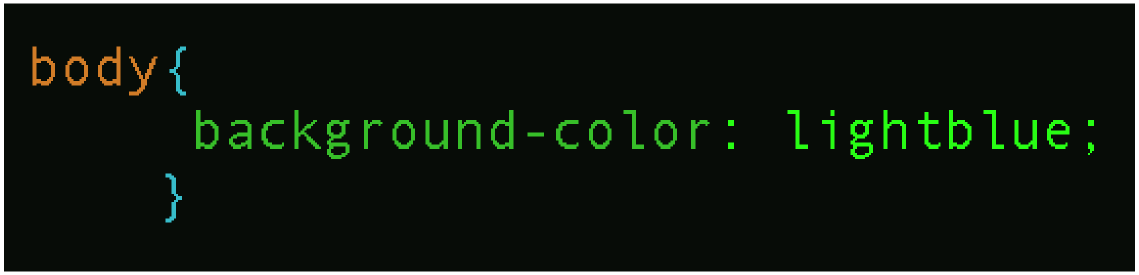
CSS Backgrounds

- The CSS background properties are used to define the background effects for elements.
- CSS background properties:
  - background-color
  - background-image
  - background-repeat
  - background-attachment
  - background-position

Background Color

- The background-color property specifies the background color of an element.

- The background color of a page is set like this:

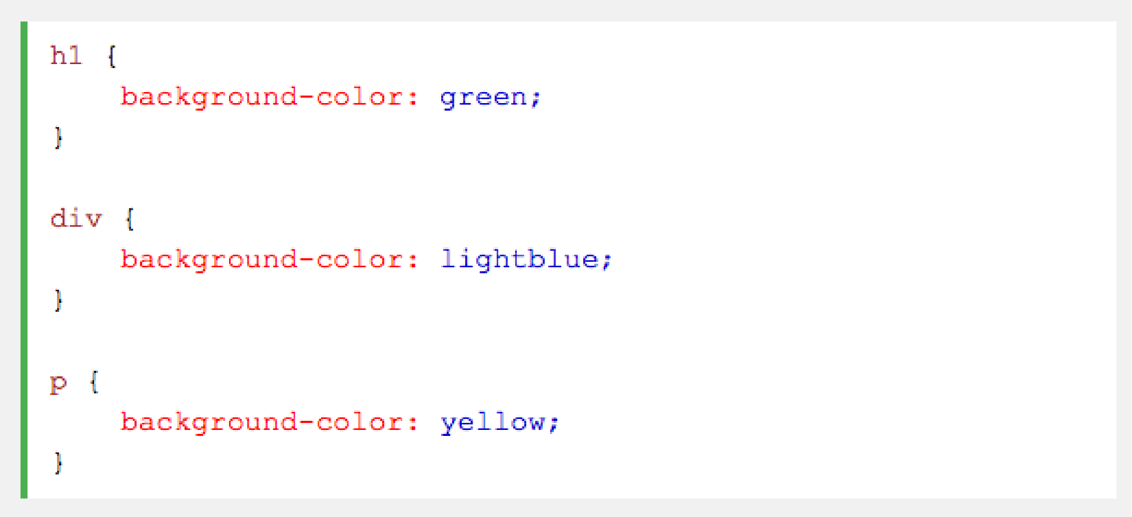


```
body{
     background-color: lightblue;
    }
```

Background Color

- The background-color property specifies the background color of an element.

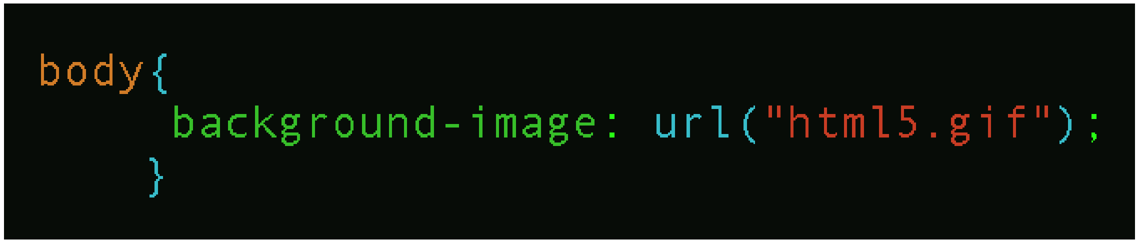
- In the example below, the <h1>, <p>, and <div> elements have different background colors:



```
h1 {
    background-color: green;
}

div {
    background-color: lightblue;
}

p {
    background-color: yellow;
}
```

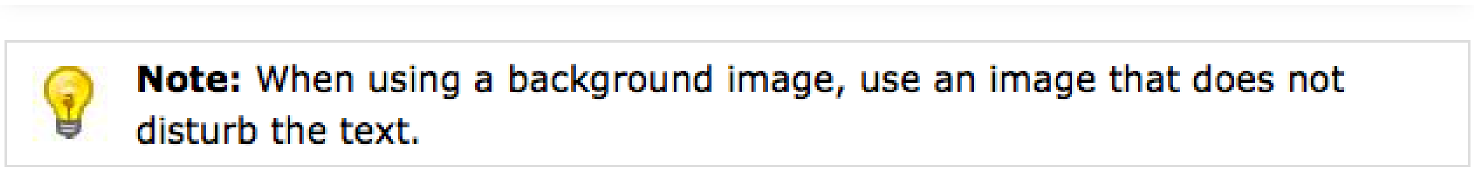
Background Image

- 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("html5.gif");
   }
```

Background Image

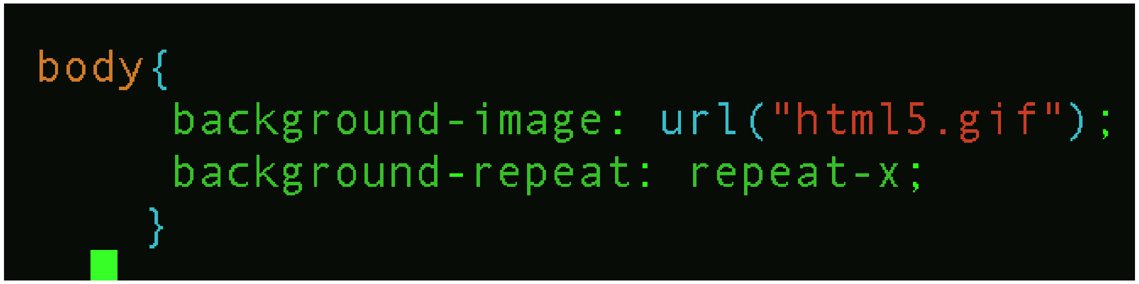


Note: When using a background image, use an image that does not disturb the text.

Background Image - Repeat Horizontally

- By default, the background-image property repeats an image both horizontally and vertically.

- To repeat an image horizontally set background-repeat: repeat-x:

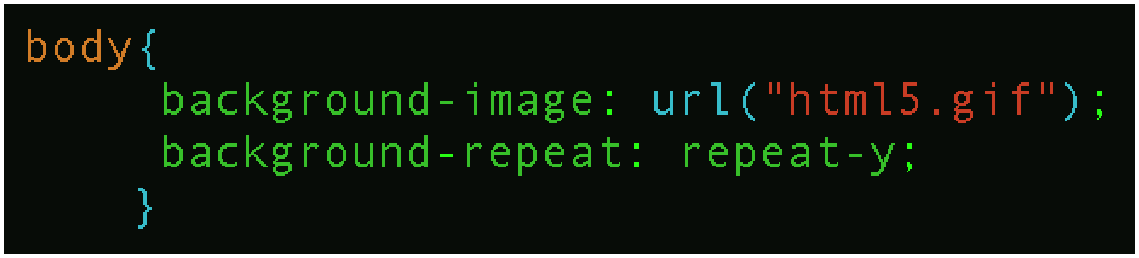


```
body{
     background-image: url("html5.gif");
     background-repeat: repeat-x;
    }
```

Background Image - Repeat Vertically

- By default, the background-image property repeats an image both horizontally and vertically.

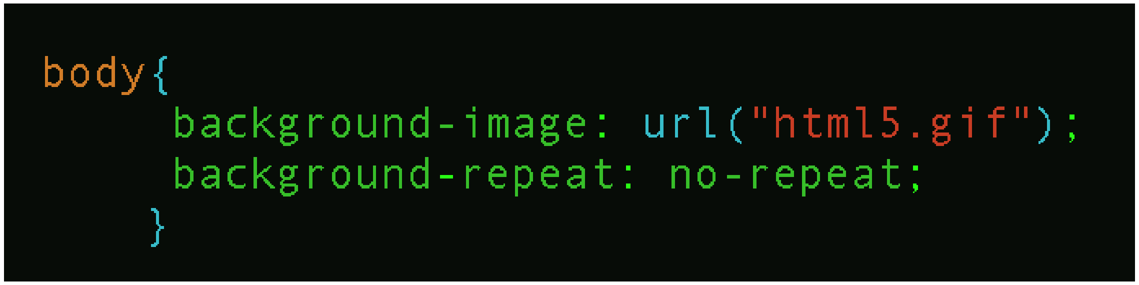
- To repeat an image vertically set background-repeat: repeat-y:



```
body{
     background-image: url("html5.gif");
     background-repeat: repeat-y;
    }
```

Background Image - No Repeat

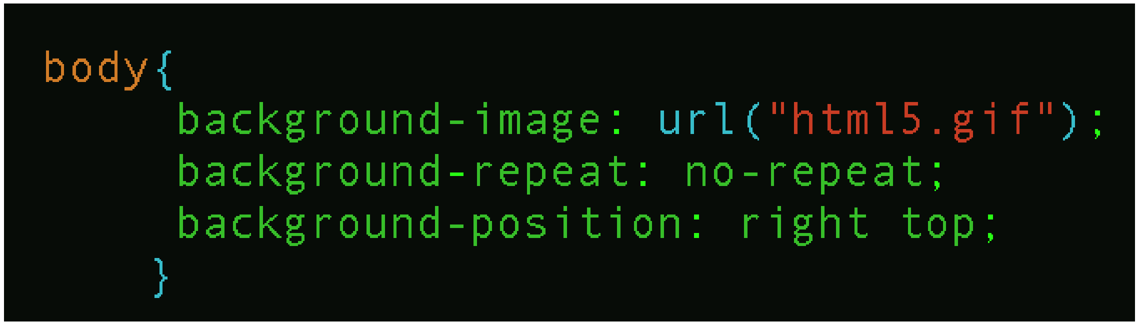
- Showing the background image only once is also specified by the background-repeat property:



```
body{
     background-image: url("html5.gif");
     background-repeat: no-repeat;
    }
```

Background Image - Set Position

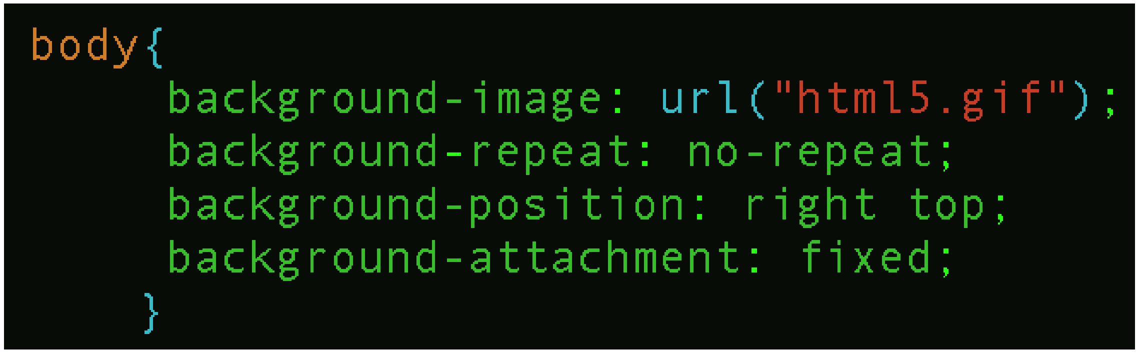
- The position of the image is specified by the background-position property:



```
body{
     background-image: url("html5.gif");
     background-repeat: no-repeat;
     background-position: right top;
    }
```

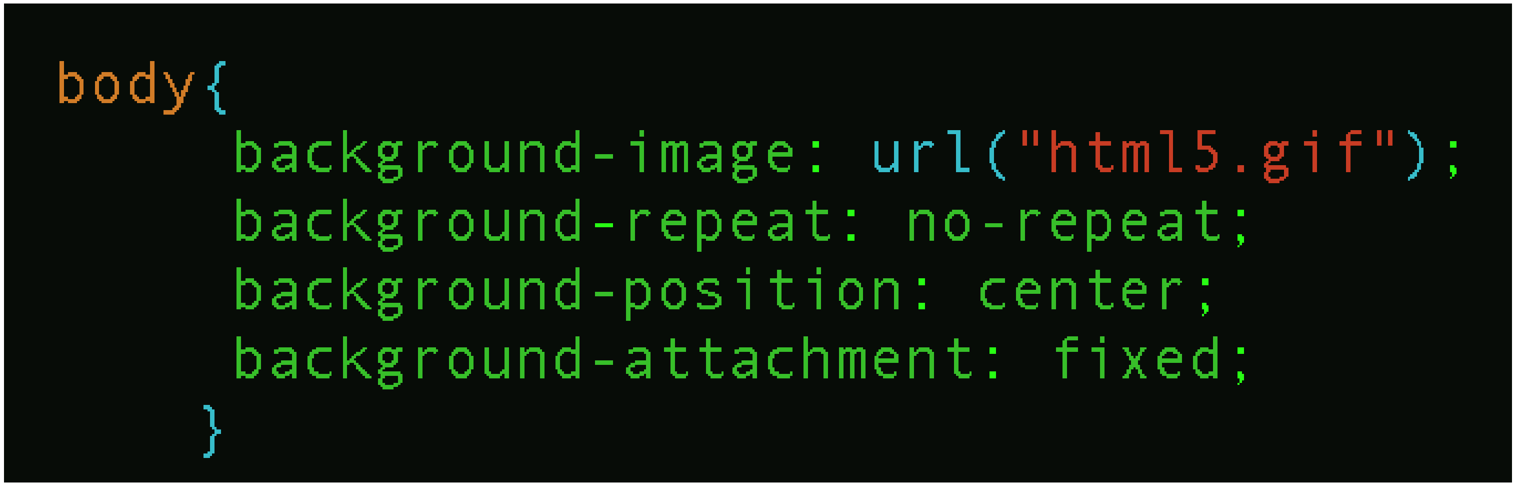
Background Image - Fixed Position

- To specify that the background image should be fixed (will not scroll with the rest of the page), use the background-attachment property:



```
body{
     background-image: url("html5.gif");
     background-repeat: no-repeat;
     background-position: right top;
     background-attachment: fixed;
    }
```

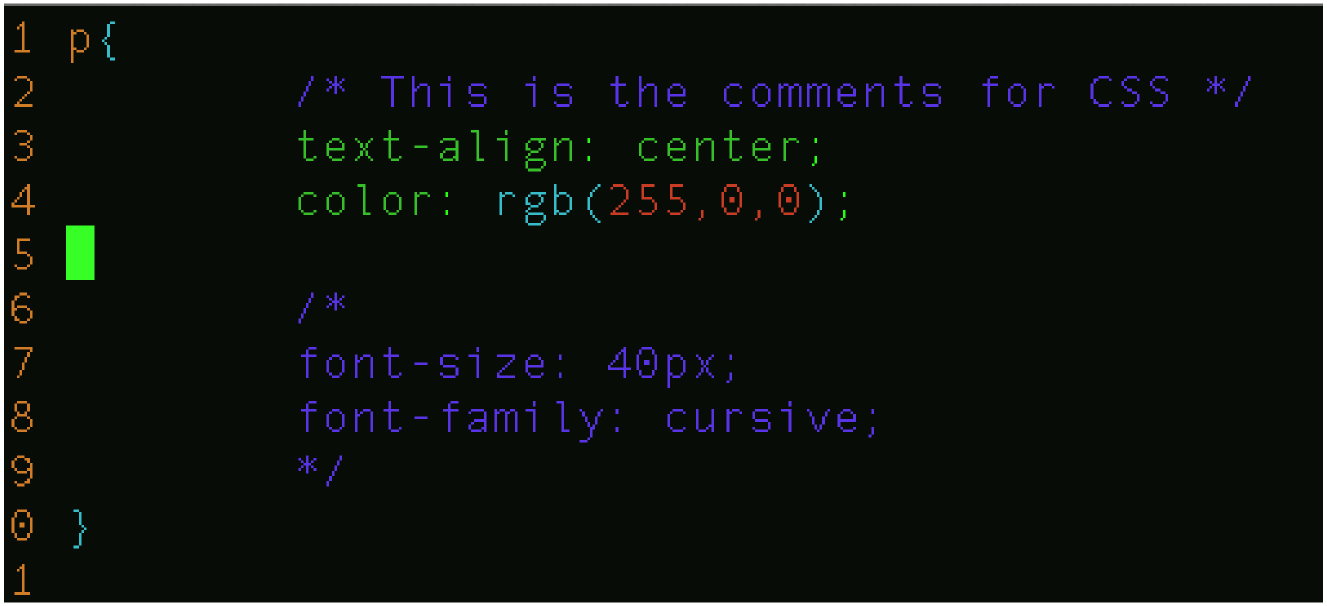
Background Image - Fixed Center



```
body{
     background-image: url("html5.gif");
     background-repeat: no-repeat;
     background-position: center;
     background-attachment: fixed;
    }
```

CSS Comment

- Comments are used to explain the code, and may help when you edit the source code at a later date.
- Comments are ignored by browsers.
- A CSS comment starts with /* and ends with */. Comments can also span multiple lines:



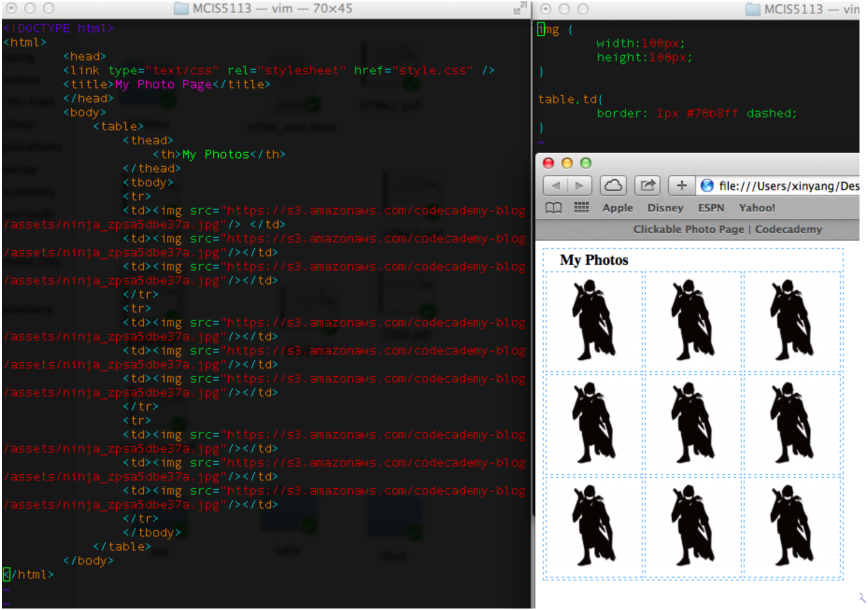
```
p{
        /* This is the comments for CSS */
        text-align: center;
        color: rgb(255,0,0);

        /*
        font-size: 40px;
        font-family: cursive;
        */
}

```

Bordering property

- Many HTML elements support the **border** property. This can be especially useful with tables.

- The border property in turn supports several values. For example, for a border 2 pixels thick, solid, and red, you'd type: selector{ border: 2px solid red; }

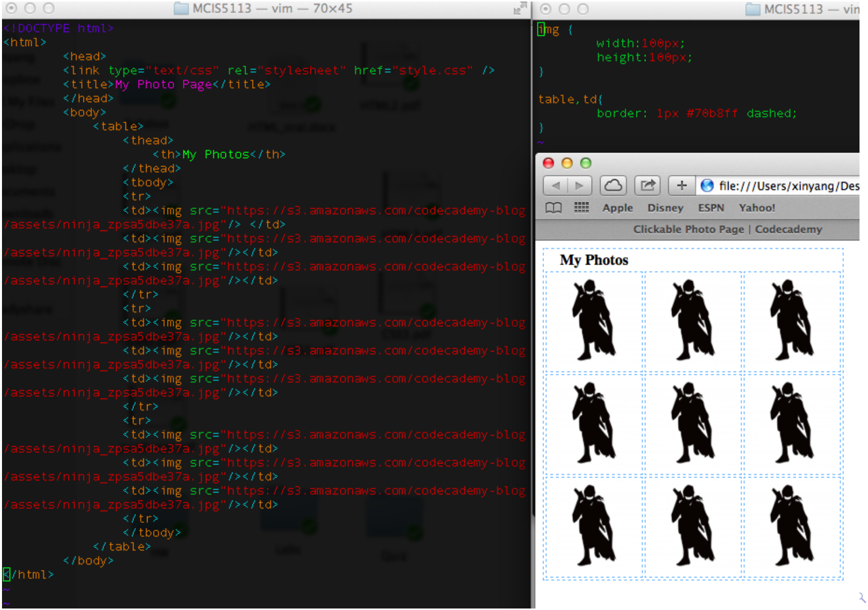
HTML&CSS example



```
<!DOCTYPE html>
<html>
        <head>
        <link type="text/css" rel="stylesheet" href="style.css" />
        <title>My Photo Page</title>
        </head>
        <body>
            <table>
                <thead>
                    <th>My Photos</th>
                </thead>
                <tbody>
                <tr>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/> </td>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/></td>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/></td>
                </tr>
                <tr>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/></td>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/></td>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/></td>
                </tr>
                <tr>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/></td>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/></td>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/></td>
                </tr>
                </tbody>
            </table>
        </body>
</html>
~
~
```

```
img {
        width:100px;
        height:100px;
}

table,td{
        border: 1px #70b8ff dashed;
}
~
```

HTML&CSS example



```
<!DOCTYPE html>
<html>
        <head>
        <link type="text/css" rel="stylesheet" href="style.css" />
        <title>My Photo Page</title>
        </head>
        <body>
            <table>
                <thead>
                    <th>My Photos</th>
                </thead>
                <tbody>
                <tr>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/> </td>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/></td>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/></td>
                </tr>
                <tr>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/></td>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/></td>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/></td>
                </tr>
                <tr>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/></td>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/></td>
                <td><img src="https://s3.amazonaws.com/codecademy-blog/assets/ninja_zpsa5dbe37a.jpg"/></td>
                </tr>
                </tbody>
            </table>
        </body>
</html>
```

```
img {
        width:100px;
        height:100px;
}

table,td{
        border: 1px #70b8ff dashed;
}
```

My Photos

References

1 http://www.w3schools.com