The pages we have created so far do not have any formatting. To add formatting we need to use style sheets. When HTML was first released it was viewed on text based browsers that did not have the capability to display different colours or images. Now days web pages are full of images, colour, sound, movies. HTML grew to include elements and attributes that formatted the web pages. The problem with these elements is that HTML was never meant to be used that way. HTML is meant to describe the structure of a web page not the formatting. The W3C decided that content should be separated from presentation and came up with style sheets.

What is CSS

CSS stands for cascading Style Sheets. It is a language that allows you to change the formatting of the elements such as text, size, colour, bolding, background colours, position on the page and much more.

Inline styling

The style attribute is added inside the tag. The style attribute can contain one or more declarations between its quotation marks. A declaration is made up of two parts a property and its value.

Inline styling example:

```
<!DOCTYPE html>
<html lang="en">
        <head>
                <title>In line Style</title>
                <meta charset="UTF-8" />
        </head>

        <body>
                <h1>Inline Styling</h1>
                <p style="color:green; font-family: 'Courier New', Courier, monospace; ">This
paragraph appears green because in line styling is applied</p>

        </body>
</html>
```

Inline Styling

This paragraph appears green because in line styling is applied

Fonts

In the above example we used

```
font-family: 'Courier New', Courier, monospace;
```
This tells the browser to use courier new if it is available on the user's computer. If courier new is not available then it uses courier, if that is unavailable it uses any monospace font. A monospace font is a font where all characters are the same width. Fonts which are defined using 2 words need to be surrounded by quotes.

Embedded style

Inline styles are good at adding styling to a small section but they are not the best way to apply styling to your web pages. They produce extra mark-up and are hard to maintain. Embedded styles are a bit better but should not be used either.

Embedded example:

```
<!DOCTYPE html>
<html lang="en">
        <head>
                <title>Embedded Style</title>
                <meta charset="UTF-8" />

                <style type="text/css">
                p
                {
                        font-family: Arial, Helvetica, sans-serif;
                        color: blue;
                }

                h1
                {
                        font-family: Geneva, Arial, Helvetica, sans-serif;
                        color: orange;
                }

                </style>
        </head>

        <body>
                <h1>Embedded Styling</h1>
                <p>This paragraph appears blue because in embedded styling is applied</p>

        </body>
</html>
```

Embedded Styling

This paragraph appears green because in embedded styling is applied

The anatomy of a rule

```
        p

        {

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

                color: blue;

        }
```

The whole thing is called a rule. p is a selector. Curly braces surround the declarations. A declaration (color: blue;) is made up of a property and a value.

External style sheets – The right way to do it

Although embedded styles are better than inline styles they are still not the best. If you have more than one page in your website, and you will. It makes more sense to define the format of your site in one file and link all pages of the site to that file.

CSS gives you the power to set styling rules in one place. When you want to make changes to your web site you change it in one place.

Activity 1.   Create a style sheet

Create a new file and name it style.css make sure you save it with a .css extension it is not a web page.

```
/* CSS Document */

body
{
        font-family: Verdana, Arial, Helvetica, sans-serif;
}
```

The first line (/* CSS Document */) is a comment. The start /* identifies the start of the comment and the end */ indicates the end of the comment. Comments help people understand the styles sheet.

Notice the style is applied to the body element. The body element surrounds all other elements so all elements will have the font changed.

Linking CSS to a page

To apply the style sheet to your web pages you need to link the style sheet.

Linking the style sheet

Activity 2.   Linking the style sheet

Open home.html and add the following line in the head section of the document:

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

The link element is most commonly used for the purpose of linking to one or more style sheets for the document. The rel attribute defines the relationship that the linked resource has to the document from which it's referenced. In most cases, this resource will simply be "stylesheet", which means, not surprisingly, "the referenced document is a stylesheet."

Formatting headings, paragraphs and font

After viewing your home page you most likely noticed that the page already looks neater. This is because we have replaced the times new roman font with verdana. Verdana is a sans serif font. Sans serif fonts are easier to read on screens whereas serif fonts are easier to read on print.

Activity 3.   Formatting text

Modify your style sheet so that it contains these extra rules:

```
body
{
        font-family: Verdana, Arial, Helvetica, sans-serif;
        font-size: 12px;
}

h1
{
        color: #333333;
        font-size: 150%;
}
h2
{
        color: #333333;
```

```
        font-size: 120%;
}
```

Change your home page HTML to contain extra headings and paragraphs:

```
<!DOCTYPE html>
<html lang="en">
      <head>
              <title>HTML and CSS reference</title>
              <meta charset="UTF-8" />
              <link href="style.css" rel="stylesheet" type="text/css" />
      </head>

      <body>
              <h1>Home</h1>
              <ul>
                      <li><a href="whatIsHTML.html">What is HTML</a></li>
                      <li><a href="headingsAndParagraphs.html">Headings and Paragraphs</a>
                      <li><a href="lists.html">Lists</a></li>
                      <li><a href="markup.html">Markup</a></li>

              </ul>

              <h2>HTML</h2>
              <p>This will be replaced with other text later</p>

              <h3>CSS</h3>
              <p>This will be replaced with other text later</p>

      </body>
</html>
```
Refresh your browser and you should see those changes applied.

Home

- What is HTML
- Headings and Paragraphs
- Lists
- Markup

HTML

This will be replaced with other text later

CSS

This will be replaced with other text later

Typography
http://reference.sitepoint.com/css/typography

Fonts

Font-Family
The property font-family sets a prioritized list of font family names and/or generic family names to be used to display a given element's text content. A user agent (browser) will use the first family that's available. Since there's no guarantee that any particular font will be available, a generic family name should always be the last value in the list.

Note that the values are separated by commas, not the spaces that are used in most other CSS properties. Comma separators are used because the values are alternatives—at most, one of them will be used.

Font-size

This property specifies the font size to be applied to the text content of an element. In the above style sheet a font-size of 12px means it will take up 12 pixels on the screen. A font size of 150% means it will be 150% larger than the default font size (which we set to 12px).

There are other measurements that can be used.

An absolute size (sometimes referred to as a T-shirt size) is specified using one of the following keywords:

- xx-small
- x-small
- small
- medium
- large
- x-large
- xx-large

The exact sizes to which those keywords map aren't defined, but each one in the list above must be larger than or equal to the keyword that precedes it.

A relative size is specified using one of the following keywords:

- smaller
- larger

Those keywords will make the font size smaller or larger than the inherited value.

Font-weight

This property sets the font weight that's applied to the text content of an element. Note that many common computer fonts are only available in a limited number of weights (often, the only options are normal and bold).

Example:

```
p
{
    font-weight: bold;
}
```

Font-style

This property sets the font style to be applied for the text content of an element.

Value

italic - This value specifies a font that's labeled "italic" in the user agent's font database. If such a font isn't available, it will use one labeled "oblique."

normal - This value specifies a font classified as "normal" in the user agent's font database. This is typically a Roman (upright) font for Latin characters.

oblique - This value specifies a font labeled "oblique" in the user agent's font database. This may not be a true oblique font, but may be generated by slanting a Roman font.

Example:

```
p
{
```

```
        font-style: italic;
}
```

Adding colour

Colour can be added to the text (foreground) or background. Colour in HTML is written as a hexadecimal colour specification. When you count in hexadecimal you count to 16 instead of ten.

Base 10	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Base 16	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

So to represent the number 255 in hexadecimal it would be written as FF. This means 15 lots of 16 and 15 lots of 1. Ie. (15 x 16) + (15 x 1)

In our style sheet we added a colour of #333333 this is actual 3 groups of hexadecimals representing the portions of red, green and blue mixed to produce the colour.

You can also use keywords to represent colour these include: aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, and yellow.

Activity 4.   Background colour
Modify your style sheet to change the background colour applied to the body tag:

```
body
{
        font-family: Verdana, Arial, Helvetica, sans-serif;
        font-size: 12px;
        background-color: #EFEFEF;
}
```


Grouping styles

You can group styles together. For example all our headings have the same colour.

Activity 5.   Grouping
Change your style sheet to include the following:

```
h1, h2, h3
{
        color: #333333;
}

h1
{
```

```
        font-size: 150%;
}

h2
{
        font-size: 120%;
}

h3
{
        font-size: 110%;
}
```

Home

- What is HTML
- Headings and Paragraphs
- Lists
- Markup

HTML

This will be replaced with other text later

CSS

This will be replaced with other text later

Order matters

In our above example we could have applied a size in the grouped rule:

```
h1, h2, h3
{
        color: #333333;
        font-size: 150%;
}

h2
{
        font-size: 120%;
}

h3
{
        font-size: 110%;
}
```
The level 2 and 3 headings will not be displayed with a size of 150%. The rule which is declared last wins. Swap these around:

```
h2
{
        font-size: 120%;
}

h3
{
        font-size: 110%;
}
```

```
h1, h2, h3
{
        color: #333333;
        font-size: 150%;
}
```
All h1, h2, h3 headings will have a size of 150%.

Styling links

By default browsers display links as:

Blue   Unvisited

Purple   Visited

Red   Active – you have clicked on it but the next page is not yet displayed

With CSS we can change this.

Activity 6.   Styling links

Modify your style sheet to include:

```
a:link
{
        color: #000066;
}

a:visited
{
        color: #999999;
}

a:hover
{
        text-decoration: none;
        color: #FFFFFF;
        background-color: #666666;
}

a:active
{
        color: #FF9900;
}
```
Again order matters so you should try and remember link, visited, hover, active (LVHA). I think of LoVe HAte.

Class selectors

So far we have seen how to format all the elements in a document. What if we want all paragraphs to be styled a particular way but one or two to be styled differently?

We can create class selectors to define styling rules. You can then apply the class selectors to the regions of the web pages you want displayed in that style.

A class selector can be used as many times as you like.

Activity 7.   Class selectors

Create the following web page:

Style sheets

What is CSS

CSS stands for cascading style sheets

Anatomy of a rule

```
p
{
font-family: Arial, Helvetica, sans-serif;
color: blue;
}
```

The whole thing is called a rule. p is a selector. Curly braces surround the declarations. A declaration (color: blue;) is made up of a property and a value.

Linking CSS to a page

To apply the style sheet to your web pages you need to link the style sheet.

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

```
<!DOCTYPE html>
<html lang="en">
      <head>
            <title>HTML and CSS reference</title>
            <meta charset="UTF-8" />
            <link href="style.css" rel="stylesheet" type="text/css" />
      </head>

      <body>
            <h1>Style sheets</h1>


            <h2>What is CSS</h2>
            <p>CSS stands for cascading style sheets</p>

            <h2>Anatomy of a rule</h2>
            <p class="code">p<br />
            {<br />
            font-family:  Arial, Helvetica, sans-serif;<br />
            color:  blue;<br />
            } </p>

            <p>The whole thing is called a rule. p is a  selector. Curly braces surround the
declarations. A declaration (color:  blue;) is made up of a property and a  value. </p>

            <h2>Linking CSS to a page</h2>
            <p>To apply the style sheet to your web pages  you need to link the style sheet.
</p>
```

```
                        <p class="code">&lt;link  href=&quot;style.css&quot; rel=&quot;stylesheet&quot;
type=&quot;text/css&quot;  /&gt;</p>

            </body>
</html>
```
Notice the class="code" added to the paragraph tags.

Add the following to the style sheet:

```
.code
{
        font-family:"Courier New", Courier, monospace;
        color: #FF6600;
}
```
This sets up a rule that will display all sections of the page which have a style class named "code" in courier new and orange.

You can apply the class to any element. Change one of the h2 tags to include the class:

```
<h2 class="code">Linking CSS to a page</a></h2>
```
If you only want paragraphs to have the class applied you can modify your style sheet and prefix the class with the element.

```
p.code
{
        font-family:"Courier New", Courier, monospace;
        color: #FF6600;
}
```
Now even if you have the class in a h2 tag it will not be applied.

Span
If you want to apply the class to a small section of text. You can use the <span> tag to surround the text you want applied.

Activity 8.   Span
Add the following style rule to your style sheet:

```
.example
{
        font-family:"Courier New", Courier, monospace;
        color: #FFCC33;
        background-color:#333333;
}
```

span.html

Create the following web page:

Span

Span offers no semantic information about the content inside it. It allows you to mark a section of text so it can be styled.

```
<!DOCTYPE html>
<html lang="en">
        <head>
                <title>HTML and CSS reference</title>
                <meta charset="UTF-8" />
                <link href="style.css" rel="stylesheet" type="text/css" />
```

```
        </head>

        <body>
                <h1>Span</h1>
                <p>Span offers no semantic information about the content inside it. It allows you
to <span class="example">mark a section of text</span> so it can be styled.

        </body>
</html>
```
Activity 9.   Linking your site
Create links from the home page to all the newly created web pages. Add a link to all your pages so far to link back to the home page. Add the style sheet to all your pages.

Once all pages are linked we now have a website.

References
Build your own website the right way using HTML & CSS By Ian LLOYD Published by Sitepoint. Read chapter 3

HTML complete Reference by Sitepoint.