

# Computing Science

Information Systems  
Design and Development

## Cascading Style Sheets & HTML Practical Exercises



Buckhaven High School

Version 1

## Contents

Page 1	How to use this booklet
Page 2	Introduction
Page 8	Selectors, Properties and Values
Page 9	Practical Exercise 1 - Editing a CSS File
Page 11	Practical Exercise 2 - Creating and Linking a CSS File
Page 13	IDs and Classes
Page 14	Practical Exercise 3 - Adding Classes to a Style Sheet
Page 15	Extension Work - Page Layout Using Positioning Properties
Page 19	Practical Extension Exercise - Creating a Page Layout Using Positioning
Page 20	Appendix A - CSS Properties and Values
Page 22	Appendix B - HTML Tag Reference

## How to use this booklet

This booklet has been written to cover the following content in Higher Computing.

	<b>H</b>
<b>Structures and Links (Websites)</b>	<b>Higher</b> <ul style="list-style-type: none"><li>• stylesheets, CSS (cascading style sheets)</li><li>• page structure: head, title, body</li><li>• metadata</li><li>• dynamic web pages</li><li>• interactive web page</li></ul>

For assessment purposes, students should:

- memorise a definition of cascading style sheets
- be able to explain the uses/advantages of cascading style sheets
- be able to write a simple external style to manipulate text, colours, margins & padding.

## Introduction

As you have already learned, the source code in web pages (HTML) is used by browsers to display the pages according to the format detailed in the HTML.

In addition to the web page text itself, the code will contain style details such as:

- what fonts, spacing, justification to use for text
- what colours to use for objects or backgrounds
- where to position objects
- what size objects should be
- which objects are in front or behind other objects etc, etc, etc

There are three ways that styles may be added to HTML code: *in-line*, *internally* or *externally*.

Each of the three examples that follow will use the HTML code shown below. Before styles are added the code produces a white web page with black text. The page contains no formatting.

```
<html>

<head>
<title>An HTML Document</title>
</head>

<body>
My Favourite Chocolates

<table>
<tr>
<td>Galaxy</td>
<td>Mars Bar</td>
<td>Yorkie</td>
</tr>
<tr>
<td>Bournville Dark</td>
<td>Buttons</td>
<td>Wispa</td>
</tr>
</table>

<br><br>
My Favourite Chocolate Quotes
<p>
Charles M. Schulz<br>
All you need is love. But a little
</p>
<p>
Jo Brand<br>
Anything is good if it's made of c
</p>
<p>
Sandra Boynton<br>
The greatest tragedies were written
</p>
<p>
Michael Levine<br>
Chemically speaking, chocolate rea
</p>
</body>

</html>
```

Source Code

Web Page

```
My Favourite Chocolates
Galaxy      Mars Bar Yorkie
Bournville Dark Buttons Wispa

My Favourite Chocolate Quotes
Charles M. Schulz
All you need is love. But a little chocolate now and then doesn't hurt.

Jo Brand
Anything is good if it's made of chocolate.

Sandra Boynton
The greatest tragedies were written by the Greeks and Shakespeare...neither knew chocolate.

Michael Levine
Chemically speaking, chocolate really is the world's perfect food.
```

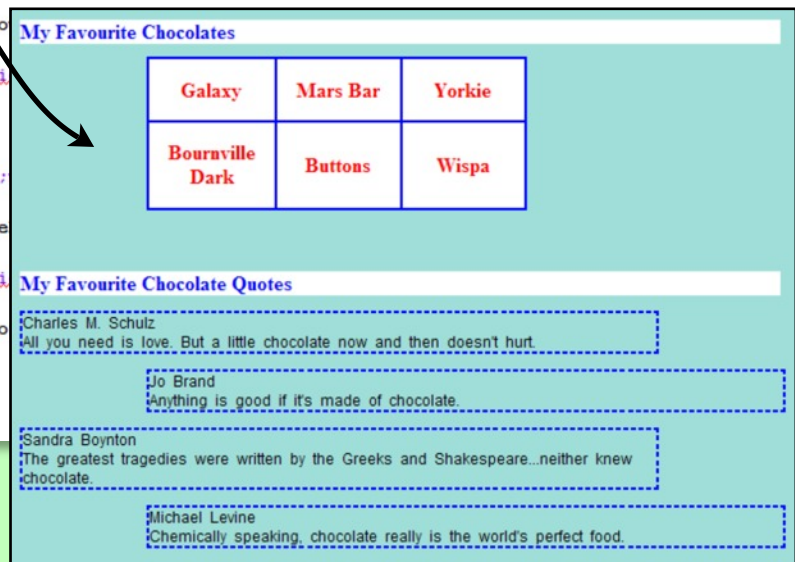
## In-line Styles

Method one involves embedding styles directly, “in-line”, into HTML tags as shown below.

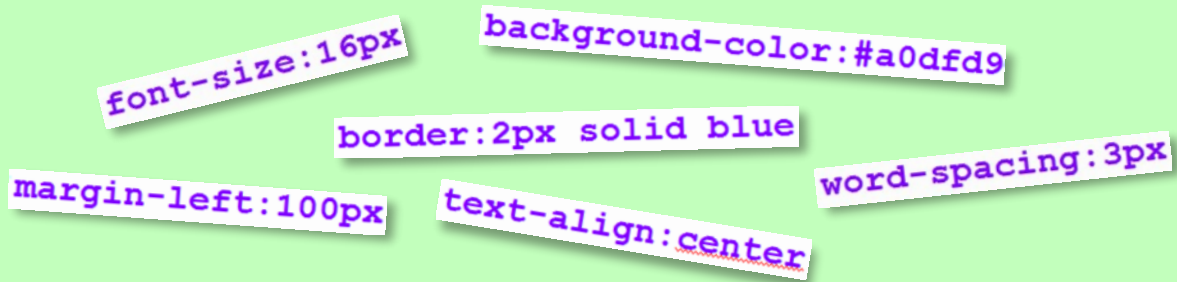
```
<html>
<head>
<title>Example of In-Line Styles Within an HTML Document</title>
</head>

<body style = "background-color:#a0dfd9">
<h1 style="color:blue;text-align:left;font-size:16px;background:#ffffff;width:600px">My Favourite Chocolates</h1>

<table style="margin-left:100px;width:300px;border-collapse:collapse;background-color:white">
<tr style="width:300px">
<td style="width:100px;border:2px solid blue"><h2 style="color:red;text-align:center;font-size:16px">Galaxy</h2></td>
<td style="width:100px;border:2px solid blue"><h2 style="color:red;text-align:center;font-size:16px">Mars Bar</h2></td>
<td style="width:100px;border:2px solid blue"><h2 style="color:red;text-align:center;font-size:16px">Yorkie</h2></td>
</tr>
<tr>
<td style="border:2px solid blue"><h2 style="color:red;text-align:center;font-size:16px">Bournville Dark</h2></td>
<td style="border:2px solid blue"><h2 style="color:red;text-align:center;font-size:16px">Buttons</h2></td>
<td style="border:2px solid blue"><h2 style="color:red;text-align:center;font-size:16px">Wispa</h2></td>
</tr>
</table>
<br><br>
<h1 style="color:blue;text-align:left;font-size:16px;background:#ffffff;width:600px">My Favourite Chocolate Quotes</h1>
<p style="font-family:helvetica;font-size:12px;word-spacing:3px;border:2px dashed blue;width:500px">
Charles M. Schulz<br>
All you need is love. But a little chocolate now
</p>
<p style="margin-left:100px;font-family:helveti
Jo Brand<br>
Anything is good if it's made of chocolate.
</p>
<p style="font-family:helvetica;font-size:12px;
Sandra Boynton<br>
The greatest tragedies were written by the Gree
</p>
<p style="margin-left:100px;font-family:helvetei
Michael Levine<br>
Chemically speaking, chocolate really is the wo
</p>
</body>
</html>
```



Styles have been added to the <body>, <h1>, <h2>, <table>, <tr>, <td> and <p> tags to control margins, colours, fonts, text size, spacing and borders.



When in-line styles have been inserted into a tag they only apply to that tag. You will notice that the above document has a lot of repetition of styles throughout.

## Internal Styles

Method two is highlighted below and involves the use of the `<style>` tag to list all the styles that will be used throughout the source code at the top of the document.

```

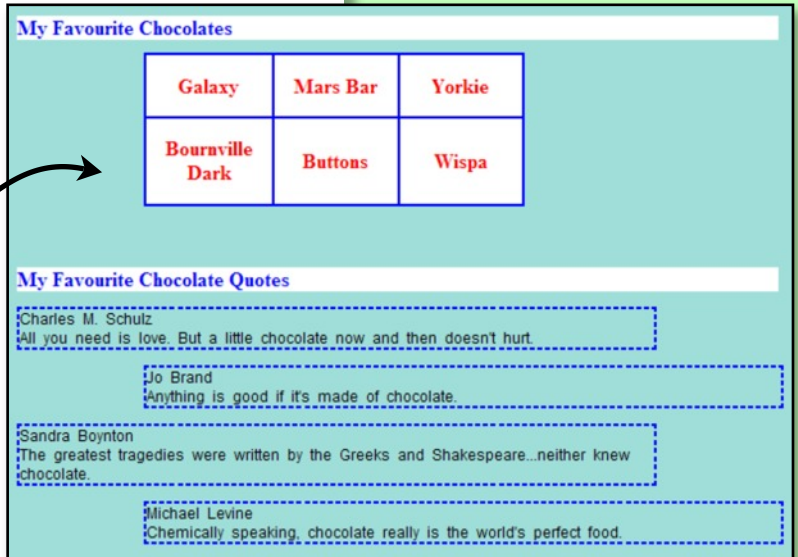
<head>
<title>Example of Internal Style Sheets Within an HTML Document</title>
<style>
body {background-color:#a0dfd9}
h1 {
  color:blue;
  text-align:left;
  font-size:16px;
  background:#ffffff;
  width:600px
}
h2 {
  color:red;
  text-align:center;
  font-size:16px
}
.evenparagraphs {
  font-family:helvetica;
  font-size:12px;
  word-spacing:3px;
  border:2px dashed blue;
  width:500px
}
.oddparagraphs {
  margin-left:100px;
  font-family:helvetica;
  font-size:12px;
  word-spacing:3px;
  border:2px dashed blue;
  width:500px
}
table {
  margin-left:100px;
  width:300px;
  border-collapse:collapse;
  background-color:white
}
tr {width:300px}
td {width:100px;border:2px solid blue}
</style>

</head>

<body>
<h1>My Favourite Chocolates</h1>

<table>
<tr>
<td><h2>Galaxy</h2></td>
<td><h2>Mars Bar</h2></td>
<td><h2>Yorkie</h2></td>

```



Although the code has been dramatically changed the same output is produced.

The styles are written out as lists, defining how each tag should be formatted when the source code is interpreted by a browser.

```

h2 {
  color:red;
  text-align:center;
  font-size:16px
}

```

For example, text enclosed by the `<h2>` `</h2>` tags should now be displayed as red, centred on the page and font size 16.

Note that the HTML tags in the body of the document no longer contain additional style instructions.

When a style is used to define a tag internally in this way, the styles apply to **every** time the defined tag is used in the source code. Although this significantly reduces the repetition of code in the document it creates a problem with larger websites as the same styles must be included in each web page in the site.

## External Styles

Method three removes the styles from the HTML source code and places them in a separate document called a *cascading style sheet*.

The two files are linked by adding the highlighted line below to the HTML source code. The browser interprets this by loading the named style sheet (in this example - "External StyleSheet.css") and applying the style definitions it lists when displaying the web page.

```

<html>
<head>
<title>Example of External Styles Within an HTML Document</title>
<link rel="stylesheet" href="External StyleSheet.css">
</head>
<body>
<h1>My Favourite Chocolates</h1>
<table>
<tr>
<td><h2>Galaxy</h2></td>
<td><h2>Mars Bar</h2></td>
<td><h2>Yorkie</h2></td>
</tr>
<tr>
<td><h2>Bournville Dark</h2></td>
<td><h2>Buttons</h2></td>
<td><h2>Wispa</h2></td>
</tr>
</table>
<br><br>
<h1>My Favourite Chocolate Quotes</h1>
<p class="evenparagraphs">
Charles M. Schulz<br>
All you need is love. But a little chocolate now and then doesn't
</p>
<p class="oddparagraphs">
Jo Brand<br>
Anything is good if it's made of chocolate.
</p>
<p class="evenparagraphs">
Sandra Boynton<br>
The greatest tragedies were written by the Greeks and Shakespeare.
</p>
<p class="oddparagraphs">
Michael Levine<br>
Chemically speaking, chocolate really is the world's perfect food.
</p>
</body>
</html>
    
```

```

body {background-color:white}
h1 {
color:blue;
text-align:left;
font-size:16px;
background:#ffffff;
width:600px
}
h2 {
color:red;
text-align:center;
font-size:16px
}
.evenparagraphs {
font-family:helvetica;
font-size:12px;
word-spacing:3px;
border:2px dashed blue;
width:500px
}
.oddparagraphs {
margin-left:100px;
font-family:helvetica;
font-size:12px;
word-spacing:3px;
border:2px dashed blue;
width:500px
}
table {
margin-left:100px;
width:300px;
border-collapse:collapse;
background-color:white
}
    
```

The HTML source code file and the cascading style sheet file are combined by the browser to produce exactly the same web page as in the previous two methods.

**My Favourite Chocolates**

Galaxy	Mars Bar	Yorkie
Bournville Dark	Buttons	Wispa

**My Favourite Chocolate Quotes**

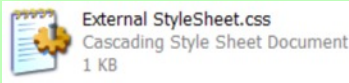
Charles M. Schulz  
All you need is love. But a little chocolate now and then doesn't hurt.

Jo Brand  
Anything is good if it's made of chocolate.

Sandra Boynton  
The greatest tragedies were written by the Greeks and Shakespeare...neither knew chocolate.

Michael Levine  
Chemically speaking, chocolate really is the world's perfect food.

Once a CSS has been created it can be linked to multiple web pages. This ensures that headings, text format, layouts, tables, paragraphs are formatted the same way throughout a website.



```
body {background-color:#a0dfd9}
h1 {
  color:blue;
  text-align:left;
  font-size:16px;
  background:#ffffff;
  width:600px
}
h2 {
  color:red;
  text-align:center;
  font-size:16px
}
.evenparagraphs {
  font-family:helvetica;
  font-size:12px;
  word-spacing:3px;
  border:2px dashed blue;
  width:500px
}
.oddparagraphs {
  margin-left:100px;
  font-family:helvetica;
  font-size:12px;
  word-spacing:3px;
  border:2px dashed blue;
  width:500px
}
table {
  margin-left:100px;
  width:300px;
  border-collapse:collapse;
  background-color:white
}
tr {width:300px}
td {width:100px;border:2px solid blue}
```

**My Favourite Chocolates**

Galaxy	Mars Bar	Yorkie
Bournville Dark	Buttons	Wispa

**My Favourite Chocolate Quotes**

- Charles M. Schulz: All you need is love. But a little chocolate now and then doesn't hurt.
- Jo Brand: Anything is good if it's made of chocolate.
- Sandra Boynton: The greatest tragedies were written by the Greeks and Shakespeare...neither knew chocolate.
- Michael Levine: Chemically speaking, chocolate really is the world's perfect food.

**My Favourite Films**

The Shawshank Redemption	Alien	12 Angry Men
Pulp Fiction	Toy Story	Twelve Monkeys
Return of the Jedi	Stand By Me	Braveheart

**My Favourite Film Quotes**

- Marlon Brando: I'm going to make him an offer he can't refuse.
- Harrison Ford: May the Force be with you.
- Bette Davis: Fasten your seatbelts. It's going to be a bumpy night.
- Tom Hanks: Houston, we have a problem.

**My Favourite Sporting Tournament**

The Rider Cup	The Olympics	The Tour de France
---------------	--------------	--------------------

**My Favourite Silly Sporting Quotes**

- Dion Dublin: Sandro is holding his face. You can tell from that itâ€™s a knee injury.
- Danny Higgenbotham: It was six of a half and one dozen of the other.
- Matt Le Tissier: Who will win the league? It is a toss of a coin between three of them.
- Alan Wright: There is no such thing as a must win game, and this is one of them.

The term “cascading” is used as the style definitions in the style sheet cascade from a single file through the entire website.

For large websites, cascading style sheets are essential as they:

- reduce workload - styles are only created once and are then applied to every page
- ensure consistency - if styles were applied internally every web page would have to be checked to ensure it used exactly the same styles as the others
- aid maintenance - a change can be applied to an entire website by simply changing one line of code in the CSS.

```
body {background-color:#ffcafb}
h1 {
  color:red;
  text-align:left;
  font-size:18px;
  background:#caf8ff;
  width:420px
}
h2 {
  color:blue;
  text-align:center;
  font-size:14px
}
.evenparagraphs {
  font-family:helvetica;
  font-size:12px;
  word-spacing:3px;
  border:1px;
  width:300px
}
.oddparagraphs {
  margin-left:20px;
  font-family:helvetica;
  font-size:12px;
  word-spacing:3px;
  border:1px;
  width:300px
}
table {
  margin-left:20px;
  width:360px;
  border-collapse:collapse;
  background-color:caf8ff
}
tr {width:360px}
td {width:120px;border:2px dashed green}
```

Edited  
CSS  
File

**My Favourite Chocolates**

Galaxy	Mars Bar	Yorkie
Bournville Dark	Buttons	Wispa

**My Favourite Chocolate Quotes**

Charles M. Schulz  
All you need is love. But a little chocolate now and then doesn't hurt.

Jo Brand  
Anything is good if it's made of chocolate.

Sandra Boynton  
The greatest tragedies were written by the Greeks and Shakespeare...neither knew chocolate.

Michael Levine  
Chemically speaking, chocolate is the perfect food.

**My Favourite Sporting Tournaments**

The Rider Cup	The Olympics	The Tour de France
---------------	--------------	--------------------

**My Favourite Silly Sporting Quotes**

Dion Dublin  
Sandro is holding his face. You can tell from that it is a knee injury.

Danny Higgenbotham  
It was six of a half and one dozen of the other.

Matt Le Tissier  
Who will win the league? It is a toss of a coin between three of them.

Alan Wright  
There is no such thing as a must win game, and this is one of them.

**My Favourite Films**

The Shawshank Redemption	Alien	12 Angry Men
Pulp Fiction	Toy Story	Twelve Monkeys
Return of the Jedi	Stand By Me	Braveheart

**My Favourite Film Quotes**

Marlon Brando  
I'm going to make him an offer he can't refuse.

Harrison Ford  
May the Force be with you.

Bette Davis  
Fasten your seatbelts. It's going to be a bumpy night.

Tom Hanks  
Houston, we have a problem.

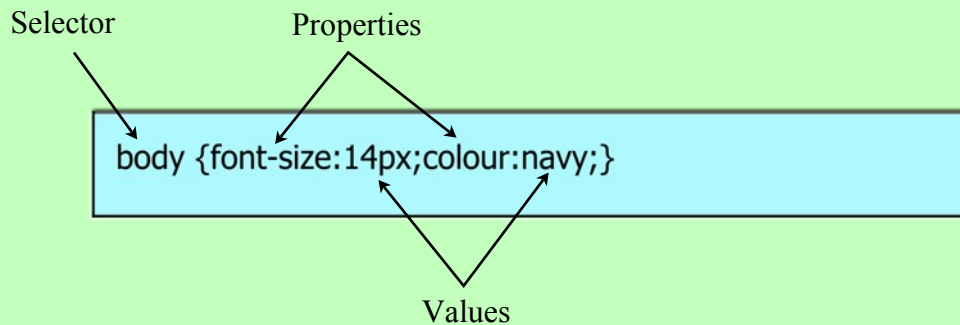
In this unit you will work through a variety of practical tasks during which you will learn to create cascading style sheets and use them to manipulate the appearance of a website.



## Selectors, Properties and Values

CSS files are highly structured and their use is very similar to learning a new programming language.

The files are written in blocks called *selectors*. Often these selectors are simply HTML tags but it is possible to create your own. Each selector contains *properties* within two { } brackets. Each property is then given a *value*.



Note:

- each property is separated from the value with a colon
- each property is separated from the next property with a semi-colon

Selectors with a large number of properties become harder to read if the properties are all squashed together in a single line. For this reason it is common to split the properties over several lines as shown below.

```
body {  
    font-size:14px;  
    colour:navy;  
}
```

Once you know how a CSS file is structured, the next step is to learn how the different properties can be applied to each selector. Unless you are professional web developer it's unlikely that you will memorise the extensive list of properties and values that are available.

At the back of this booklet you will find an Appendix of commonly used properties and values. Use Appendix A when completing each practical exercise.

If you wish to use additional properties and values the Internet has many, many websites devoted to the use of CSS files in website development.

## Practical Exercise 1 - Editing a CSS file

**Step 1** Locate the folder called CSS Task 1 and copy it into your own network folder.



**Step 2** Using a suitable editor open both the CSS file.



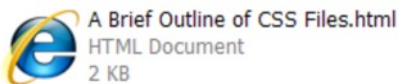
You can see that the CSS code contains selectors, properties and values for 6 of the HTML tags used in the Task 1 web page.

Read through the code and use Appendix A at the back of this booklet to make sense of the properties and values assigned to the selectors.

```

*F:\Higher\ISDD\Cascading Style Sheets and HTML\Pupil Files\Task 1\Outline of CSS File.css
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
A Brief Outline of CSS Files.html Outline of CSS File.css My Film Web Page.html
1  body {background-color:#a0dfd9}
2
3  h1 {
4      color:black;
5      text-align:left;
6      font-size:18px;
7  }
8
9  h2 {
10     color:black;
11     text-align:left;
12     font-size:14px
13 }
14
15 p {
16     font-family:helvetica;
17     font-size:12px;
18     word-spacing:3px;
19     text-align:left;
20     margin-left:20px;
21     margin-right:20px;
22 }
23
24 table {
25     margin-left:20px;
26     width:600px;
27     border-collapse:collapse;
28     background-color:dark blue
29 }
30
31 td {
32     border:2px solid black}
    
```

**Step 3** Open the HTML file in the CSS Task 1 folder in a browser of your choice.



**Step 4** Edit the CSS file so that the <h1> tag will display:

- white text
- centered text
- size 24px text

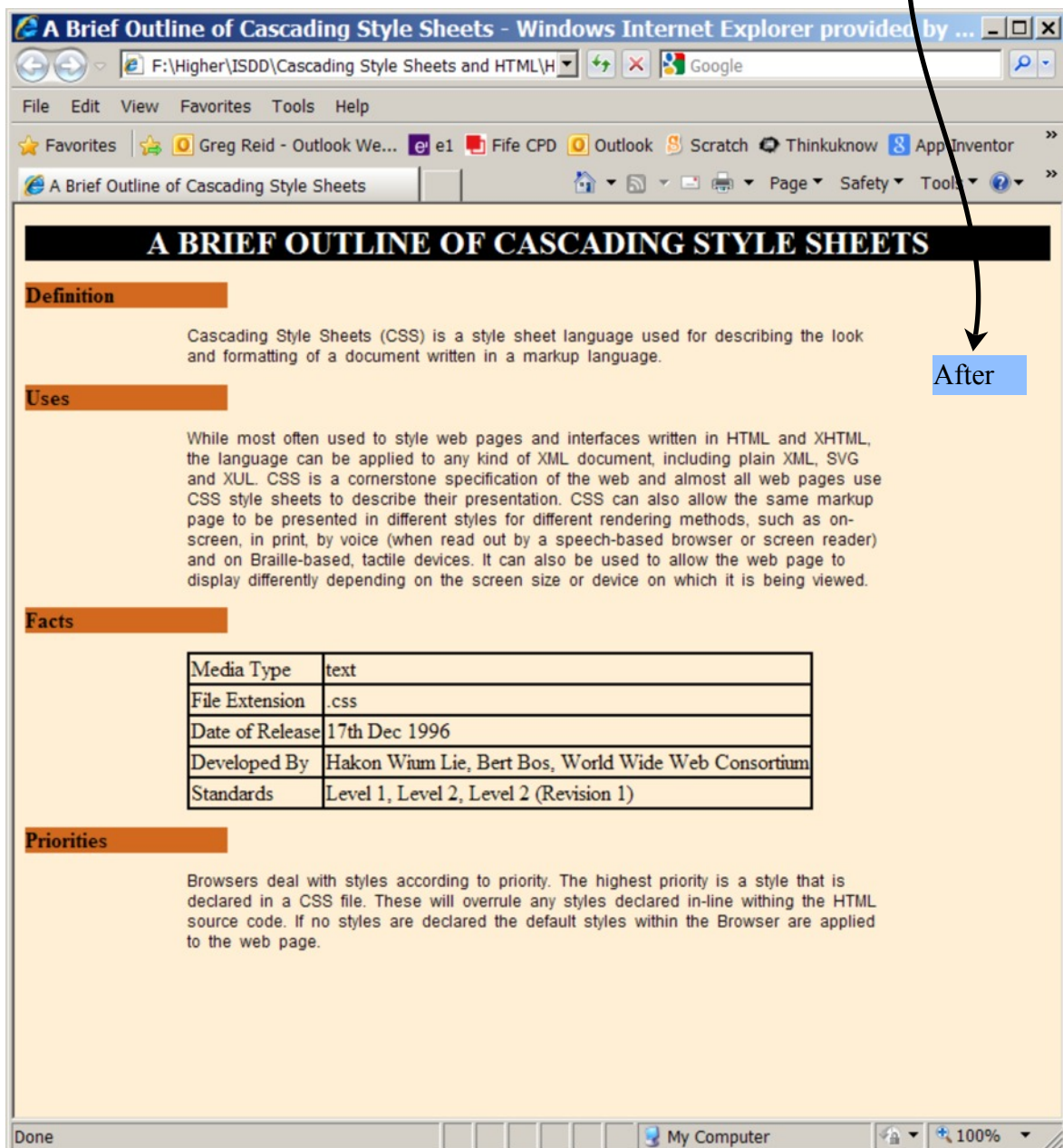
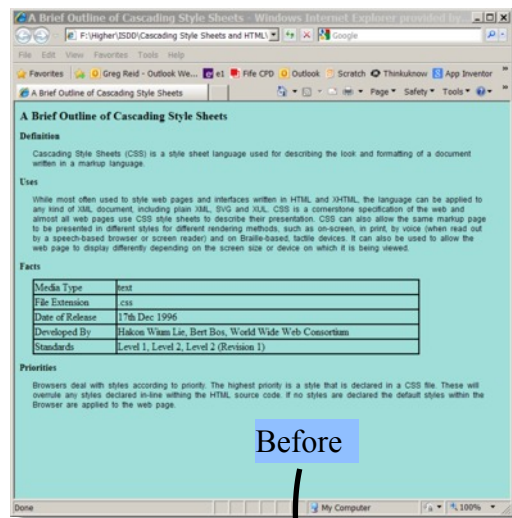
Save the CSS file to update the changes you have made and then refresh the web page in your browser. The web page should change to reflect the newly edited CSS file.

**Step 6** Using Appendix A for reference, continue to make changes to the cascading style sheet.

You could:

- change values
- remove existing properties
- add new properties

By editing only the style sheet, try and make the final web page look as different from the original as you possibly can.



## Practical Exercise 2 - Creating and Linking a CSS file

**Step 1** Locate the folder called CSS Task 2 and copy it into your own network folder.



**Step 2** Using a suitable editor, create a new blank document and save it as a .css file to your own copy of the CSS Task 2 folder. Choose a suitable name for the file.

**Step 3** Using a suitable editor open the HTML file called “Standard Algorithms.html”.



**Step 4** Also open the “Standard Algorithms” file in a browser and look carefully at the web page you are about to format.

### Standard Algorithms

#### Introduction

In programming there are a variety of algorithms that programmers make use of over and over again. These are called Standard Algorithms.

Higher Computing requires that you are familiar with 5 standard algorithms.

- Input Validation - Check that data is input correctly by the user.
- Find Max - Find the maximum value in a list of numbers.
- Find Min - Find the minimum value in a list of numbers.
- Count Occurrences - Find the number of times a value occurs in a list of numbers.
- Linear Search - Find the position of a value in a list.

#### Input Validation

The pseudocode below shows an input validation for a temperature reading between -25 and 40.

**Line 1** SEND "Enter a temperature reading between -25 and 40 degrees centigrade." TO DISPLAY

**Line 2** REPEAT

**Line 3** RECEIVE temperatureReading FROM (REAL) KEYBOARD

You can see from the above web page that there is a main heading, several sub-headings, a bullet-point list, paragraphs and several tables containing pseudocode.

These were created using the following HTML tags:

- |                       |                                         |
|-----------------------|-----------------------------------------|
| • <h1>, <h2> & <h3>   | three levels of headings                |
| • <div>, <p>          | division sections and paragraphs        |
| • <ul>, <li>          | bullet point list containing list items |
| • <table>, <tr>, <td> | rows and data (cells) within a table    |

**Step 5** First you must link the HTML file to your CSS file.

Inside the <head> tag, at the top of the HTML file, add the line of code shown below.

```
<head>
<title>Standard Algorithms</title>
<link rel="stylesheet" href="my own.css">
</head>
```

Remember to put your own .css file name into the link and save the changes.

**Step 6** Now you are ready to begin creating and testing your cascading style sheet.

Begin by typing in a selector and empty curly brackets for each of the tags you wish to style as shown.

From the list given on the previous page we know that a total of 10 tags could be styled.

```
body {
}
h1 {
}
h2 {
}
↓
```

**Step 7** Using Appendix A (along with CSS reference pages on the Internet) add properties and values to each of the selectors.

```
table {
  table-layout:auto;
  border-collapse:collapse;
  background-color:dark blue;
  margin-left:120px;
  margin-right:120px;
}
```

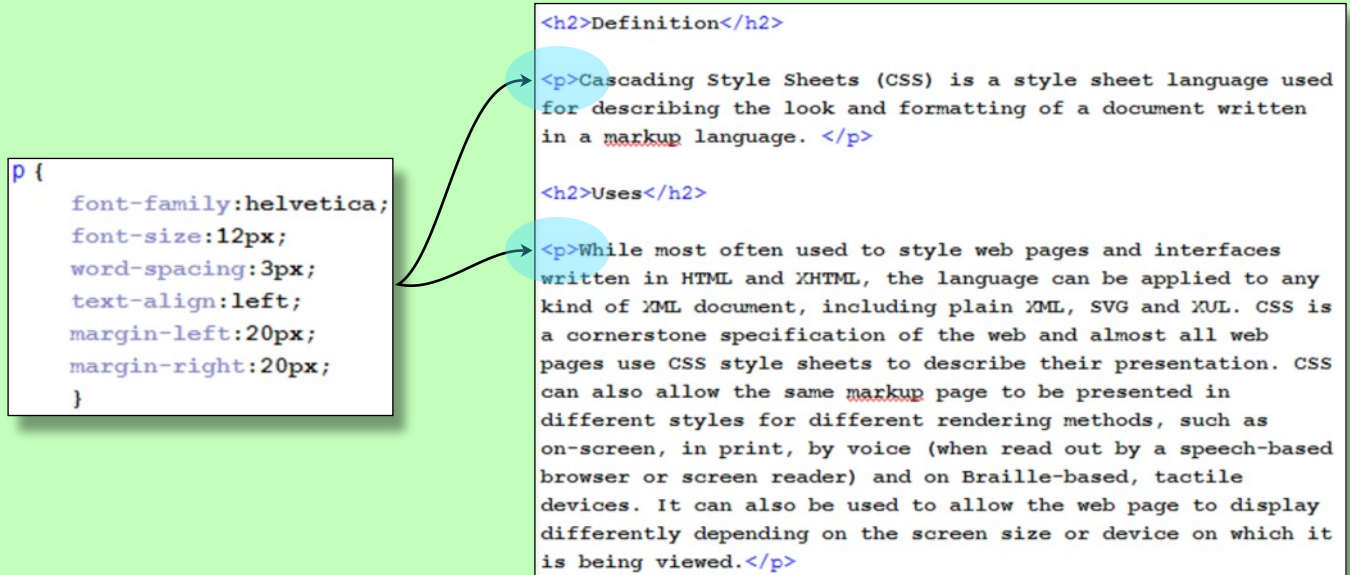
Unlike other tasks, where you may complete a large section of a project before testing it, you should continually save the CSS file and then refresh your browser to check your CSS code is formatting the HTML file correctly.

**Step 8** When you have finished, print the HTML file in the browser, write your name on it and submit it to your teacher.

## IDs and Classes

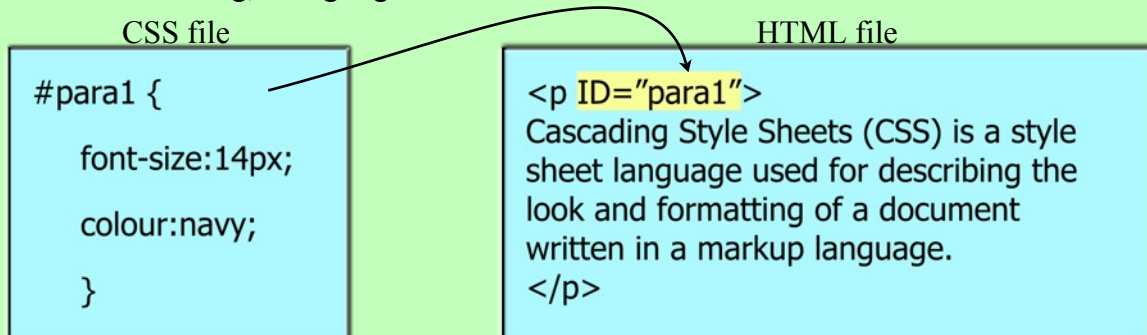
The selectors used in Practical Tasks 1 and 2 all made use of existing HTML tags.

If cascading style sheets were limited to simply styling tags, each tag would have to have the same properties and values throughout every page that was linked to the style sheet.

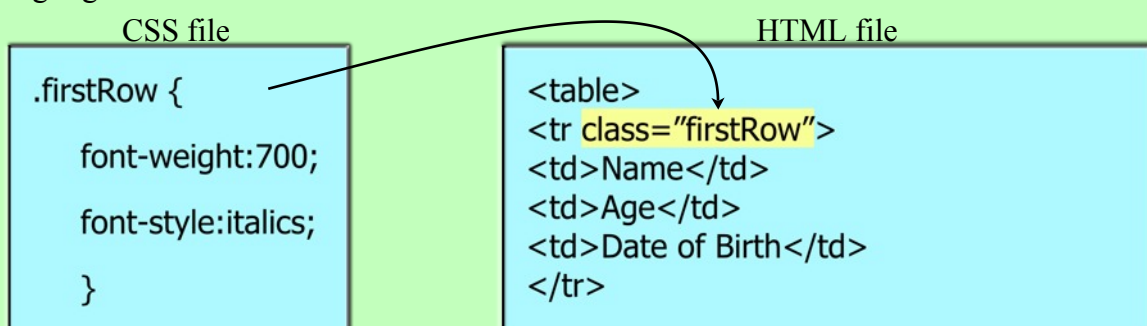


*IDs* and *Classes* are used to define additional style information that is not tied to a specific tag. These two additional selectors behave in a similar way and should be used as follows:

- **ID** selectors are created to style a single, unique element in your web page. For example a paragraph that is formatted differently for any other tag in the HTML. IDs are added to the CSS file with a preceding # symbol and then referenced in the HTML by adding the ID within a tag, as highlighted below.



- **Classes** are created when you wish to apply the same style to multiple tags. Classes in CSS files are preceded with a full stop and are added to the HTML tags as highlighted below.



### Practical Exercise 3 - Adding Classes to a Style Sheet

**Step 1** Open the HTML and CSS file you used for the previous exercise in a suitable editing application.

The html file contains 5 tables. An example of the table for the linear search algorithm is shown below. Note that there are no borders set in the table.

Each table is constructed using several rows (<tr> </tr>), one for each line of pseudocode and two columns (<td> </td>) for each row.


#### Linear Search

The pseudocode below details how to display all the values in an array called

Line 1 SEND "The values found are listed below:" TO DISPLAY

Line 2 FOR counter FROM 1 TO length(weights[])

Line 3 IF weights[counter] >= 10 AND weights[counter] <= 20 THEN

Line 4 SEND weights[counter] TO DISPLAY

Line 5 END IF

Line 6 END FOR

The first column is used to display the line number and the second is used to display the pseudocode.

**Step 2** Your task is to create classes in the CSS file and use them in multiple <td> tags to indent the loops and if statements in the pseudocode. You will have to work out how to increase the blank space before some of the text.

A finished example is shown below.

#### Linear Search

The pseudocode below details how to display all the values in an array called "weights"

Line 1 SEND "The values found are listed below:" TO DISPLAY

Line 2 FOR counter FROM 1 TO length(weights[])

Line 3 IF weights[counter] >= 10 AND weights[counter] <= 20 THEN

Line 4 SEND weights[counter] TO DISPLAY

Line 5 END IF

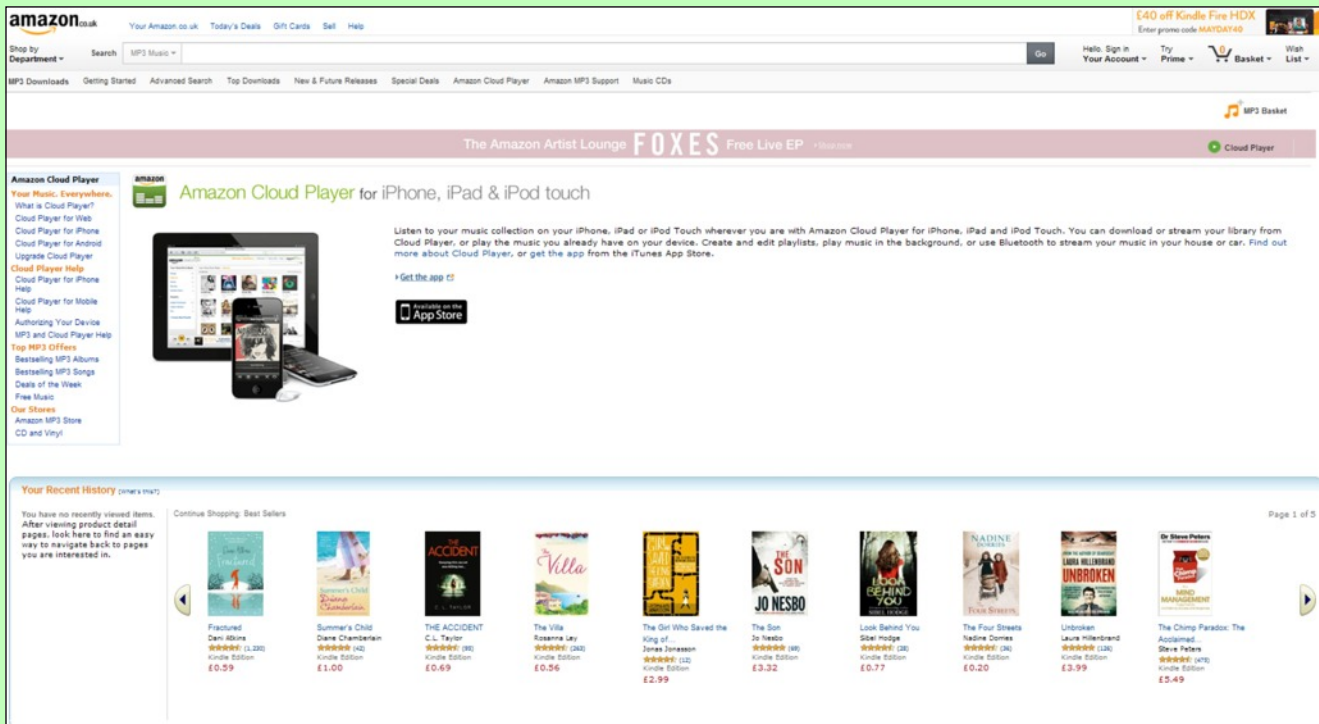
Line 6 END FOR

**Step 3** Show your teacher your completed work.

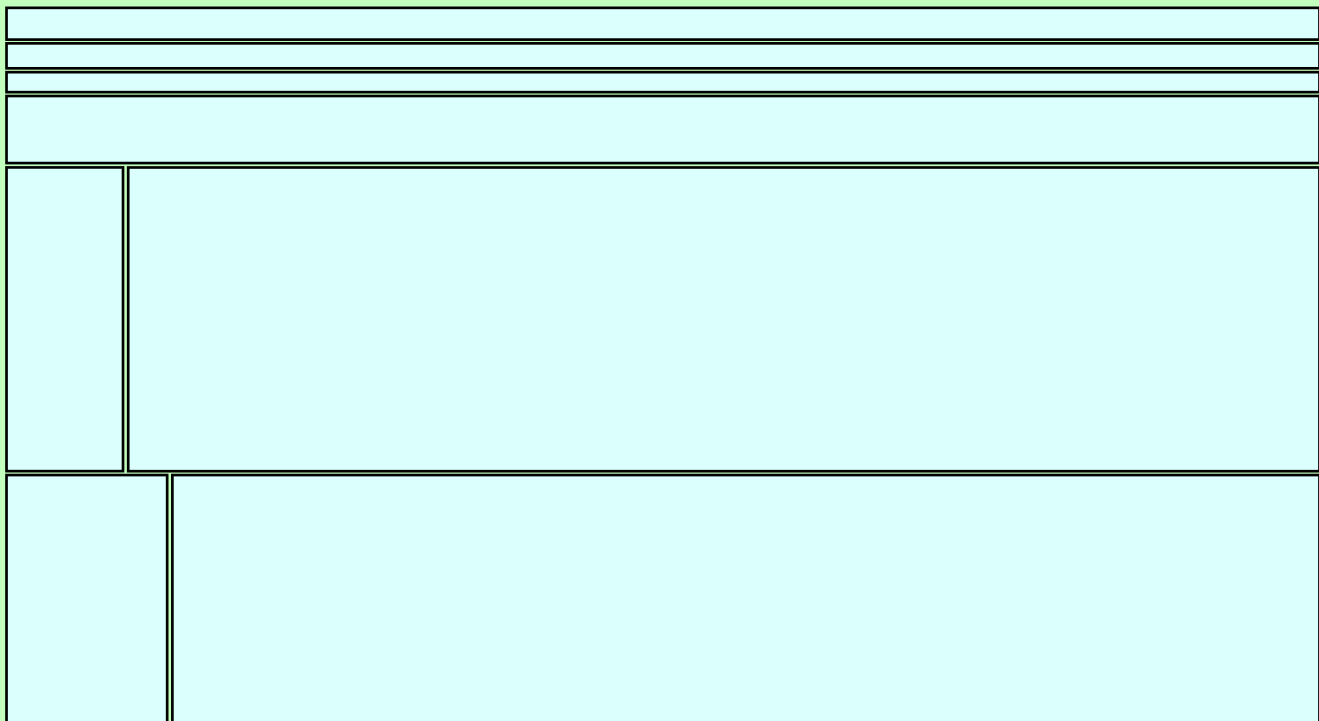


## Extension Work - Page Layout Using Positioning Properties

If you look closely at modern web pages you will see an underlying structure of rows and columns.



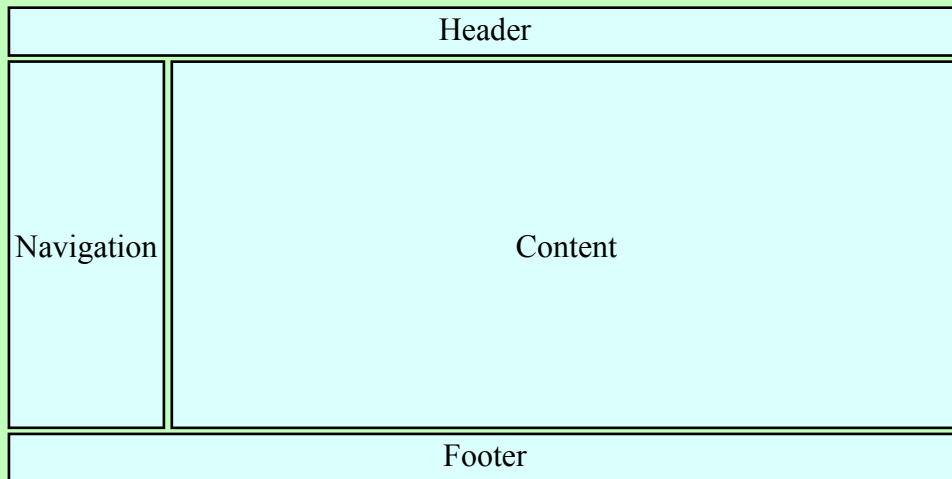
The boxes below can be mapped to the different areas of the Amazon web page shown above.



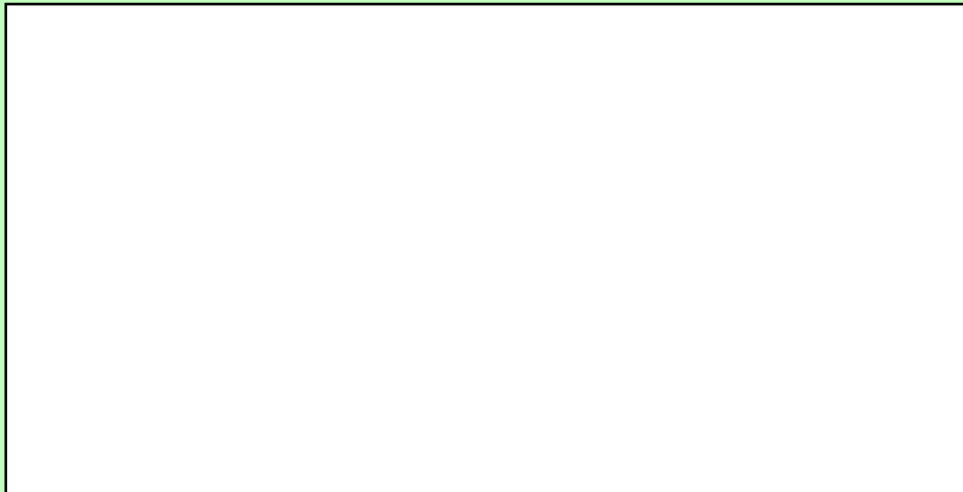
To create this page, each of the above boxes would be created using division `<div>` `</div>` tags. Style sheet rules are then created to resize and position each of the boxes.



The diagram below shows a common structure for a simple web page.



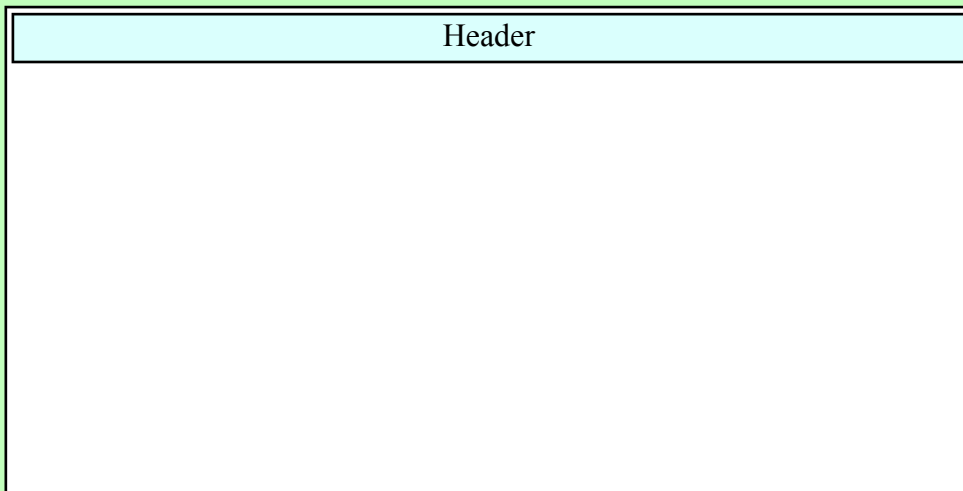
To create the structure for the above web page start by creating a division for the whole page.



```
<div>
```

```
</div>
```

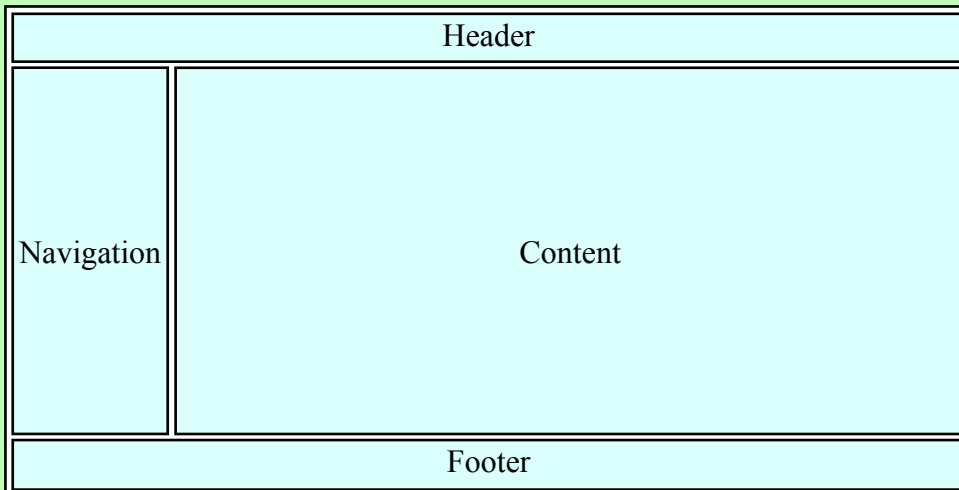
Next add a division for the header.



```
<div>  
<div>  
</div>
```

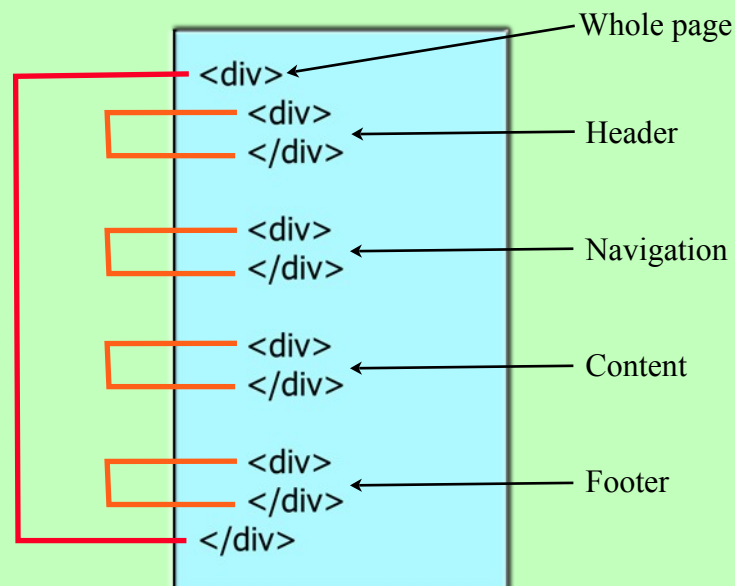
```
</div>
```

Complete the structure by adding a division for each of the four boxes.



```
<div>
<div>
</div>
<div>
</div>
<div>
</div>
</div>
```

You should now have 5 divisions with one of them enclosing the others.



To resize and position the boxes we will create ID selectors. (Remember IDs are unique to one object.)

## Whole Page Style

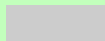
This allows us to set:

- the page width to 900 pixels
- the margin to 0 (no margin)
- the background colour of the page to white.

```
#container {
margin: 0 auto;
width: 900px;
background: #ffffff;
}
```

### Header Style

This allows us to set:

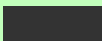
- a background colour of 
- a padding of 20 pixels inside the header

No positioning is required as the header <div> appears first out of the four boxes in the HTML. This means it will automatically appear at the top of the page.

```
#header {  
    background: #cccccc;  
    padding: 20px;  
}
```

### Navigation Style

This allows us to:

- ensure that the navigation box is positioned at the left of the web page
- set the width of the navigation
- set a padding of 20 pixels inside the box, pushing the text from the edges.
- set the background colour to 

```
#navigation {  
    float: left;  
    width: 200px;  
    background: #333333;  
    padding: 20px;  
}
```

### Content Style


This allows us to:

- ensure that the content box is positioned at the right of the web page
- set the width of the content box
- set a padding to push the text in from the top and left of the content box edges.

```
#content {  
    float: right;  
    width: 700px;  
    padding-left: 10px;  
    padding-top: 20px;  
}
```

### Footer Style

This allows us to:

- clear the left and right positioning of the previous two boxes (ignore them)
- set a background colour of 
- centre all the text
- set a padding of 20 pixels inside the footer

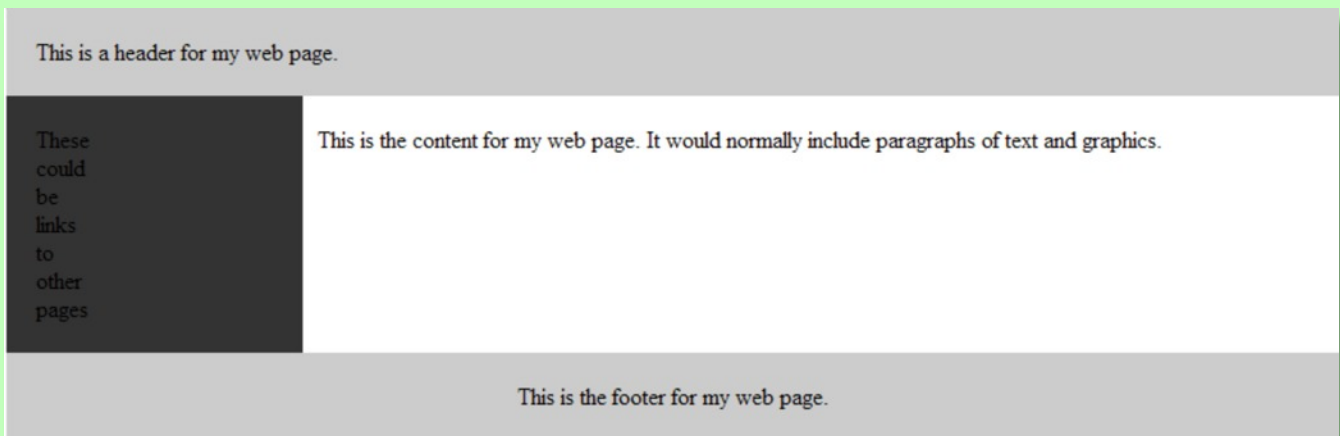
Note that a height of 1% will ensure the footer is never larger than its contents require.

```
#footer {  
    clear: both;  
    background: #cccccc;  
    text-align: center;  
    padding: 20px;  
    height: 1%;  
}
```

Finally the ID selectors must be added to the HTML to implement the styles and create the layout.

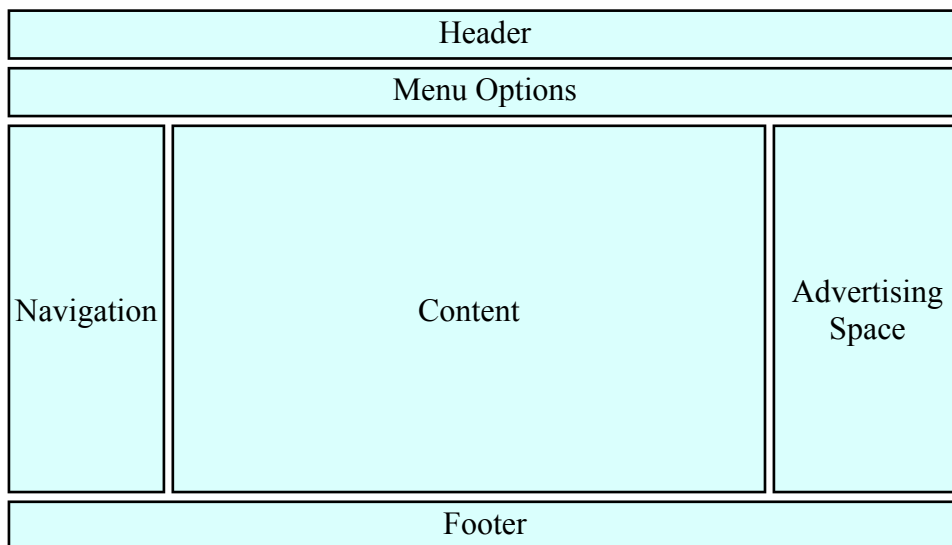
```
<div ID="container">  
  <div ID="header">  
  </div>  
  
  <div ID="footer">  
  </div>  
  
  <div ID="content">  
  </div>  
  
  <div ID="footer">  
  </div>  
</div>
```

If you add some content inside the completed `<div>` `</div>` tags you can now see what the finished page will look like.



## Practical Extension Exercise - Creating a Page Layout Using Positioning

Your task is create the layout shown below using `<div>` `</div>` tags, IDs, properties and values to position objects on the page.



## Appendix A - CSS Properties &amp; Values

## Text

Property	Example Values	Explanation
font-family	arial helvetica "times new roman"	Sets the selectors font. Note that if the name of the font is longer than one word it should be enclosed in inverted commas.
font-size	medium 14px 18pt 75%	Text may be set to a small, medium, large size. Set text to a fixed size in pixels (px) or points (pt). The percentage value indicates the size of text relative to a parent element.
font-weight	bold normal bolder lighter 400	The thickness of each character can be adjusted using either text values (bold etc) or more accurately using numerical values (500). For numerical values, 400 = normal and 700 = bold.
font-style	normal underline italics	Sets styles of text as seen in Word Processing applications.
text-decoration	underline overline line-through	This property works in a similar way to font-style and is commonly used to style hyperlinks.
text-transform	capitalize uppercase lowercase	Changes the first letter of every word into uppercase. Turns all the text to uppercase. Turns all the text to lowercase.
letter-spacing	0.5em	Sets the size of the gap between each letter.
word-spacing	2em	Sets the size of the gap between each word.
Word-wrap	Normal Break-word	Sets text to word wrap words at the end of a line (normal) or split words if they are too long (break-word)
text-align	left right center justify	Text justification as found in Word Processing applications. Note the spelling of 'center' uses American English.
text-indent	20px	This indents the first word of a paragraph by a distance given in px.
line-height	2	This specifies the distance between the lines in relation to the font-size. So a value of 2 would be double spacing.

## Colours

Property	Example Values	Explanation
color	blue rgb(255,67,0) #ff349f	Colors (note the American spelling in HTML and CSS) can be set on the majority of selectors using: <ul style="list-style-type: none"> <li>• a word - red, navy, yellow, black</li> <li>• an r/g/b value - expressed as numbers between 0 and 255 (237,129,255) or a percentage (60%,34%,0%)</li> <li>• a hexadecimal rgb code - using either one digit (0-f) for each color (f6b), or two digits (ff6c00)</li> </ul>
background-color		

## Margins, Padding & Borders

Property	Example Values	Explanation
margin	20px	Sets a space in pixels around the outside of the entire selector.
margin-left	30px	As above but allows for more control over which side of the selector will have a margin.
margin-right		
margin-top		
margin-bottom		
padding	40px	This defines a blank area of space inside the boundaries of a selector.
padding-left	16px	As above but allows for more control over which side of the selector will have a padded area.
padding-right		
padding-top		
padding-bottom		
border-width	3px	Sets a border width in px around the entire selector.
border-left-width	10px	As above but allows for more control over the border thickness applied to each side of the selector.
border-right-width		
border-top-width		
border-bottom-width		
border-color	red	Sets the colour of the border.
border-style	solid dotted dashed double	Sets the appearance of the border as described by the values shown.

## Positioning, Floating & Clear

Property	Example Values	Explanation
position	static	Ensures that the selector appears in the order given in the HTML.
	relative	Takes the selectors normal (static) position and moves it relative to this point.
	absolute	This sets the selector to ignore its order in the HTML code and move to a set point on the web page.
	fixed	Similar to absolute but selector will stay in their position on the web page, even when the page is scrolled in the browser.
top	0px 20px	Used in conjunction with the position property above to place a selector at an exact position in the web page. Left:20px would move the selector 20 pixels in from the left.
bottom		
left		
right		
float	left	Floating a selector will move it to the left or right of the page with surrounding content flowing round it.
	right	
clear	left	Clear is used to cancel the effect of floating left or right selectors.
	right	
	both	

## Appendix B - HTML Tag Reference

### Document

Tag	Explanation of Use
<html> </html>	These tags appear at the top and bottom of every web page. They tell the browser that this is an HTML document.
<head> </head>	The head tag acts as a container for head elements. The head tag must include a <title> and may include <style>, <base>, <link>, <meta> and <script> tags.
<title> </title>	The title tag is used to enclose a title of your choice that will appear at the top of the browser when the page is viewed. <title>My Hockey Home Page</title>
<body> </body>	The body tag encloses the all the content of the web page that will be displayed by the browser.
<b>Example</b>	<pre>&lt;html&gt;   &lt;head&gt;     &lt;title&gt;My Hockey Home Page&lt;/title&gt;   &lt;/head&gt;   &lt;body&gt;     I have been playing hockey since the age of 6.   &lt;/body&gt; &lt;/html&gt;</pre>

### Layout

Tag	Explanation of Use
<div> </div>	The division tag can be used several times throughout a document to spilt the HTML code into sections. They are often used with CSS files to layout a web page.
<p> </p>	The paragraph tag is used to define and enclose paragraphs of text.
 	The break tag tells the browser to move the next object onto the next line.

### Links

Tag	Explanation of Use
<a> </a>	The a tag encloses an object to turn it into a hyperlink. It is used like this: <a href="http://www.rslearning.co.uk">Visit RS Learning Ltd</a>
<link>	The link tag is used to load in external sources like style sheets and has no end tag. <link rel="stylesheet" href="Home Page Styles.css">

### Graphics

Tag	Explanation of Use
<img>	The image tag includes a link to a graphic file and tells the browser how to display the file. 

## Tables

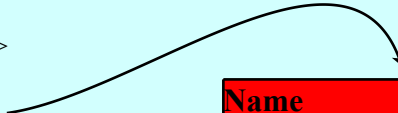
Tag	Explanation of Use
<code>&lt;table&gt;</code> <code>&lt;/table&gt;</code>	Used to enclose the content of a table.
<code>&lt;tr&gt;</code> <code>&lt;/tr&gt;</code>	The table row tags enclose the data for each row of the table. If the table has three columns each table row tag will enclose three sets of data tags <code>&lt;td&gt;</code> <code>&lt;/td&gt;</code> .
<code>&lt;td&gt;</code> <code>&lt;/td&gt;</code>	The table data tags each represent a cell in the table. All the information in a table, displayed by the browser, is enclosed by <code>&lt;td&gt;</code> <code>&lt;/td&gt;</code> tags.
<code>&lt;thead&gt;</code> <code>&lt;/thead&gt;</code>	The table head tag usually encloses the top row of the table. It can be used to style the headings or top row of the table differently from the rest.
<code>&lt;tbody&gt;</code> <code>&lt;/tbody&gt;</code>	The table body tag encloses all the rows that make up the body of the table. It can be used to style the body content of the table differently from the headings.
<code>&lt;colgroup&gt;</code> <code>&lt;/colgroup&gt;</code>	The column group tag allows groups of columns to be declared and then styled. Additional <code>&lt;col&gt;</code> tags contain the information about how the columns are grouped.

### Example

```

<table>
  <colgroup>
    <col style="background-color:red">
    <col span="2" style="background-color:yellow">
  </colgroup>
  <thead style="background-color:red">
    <tr>
      <td>Name</td>
      <td>Wins</td>
      <td>Draws</td>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Pars United</td>
      <td>29</td>
      <td>5</td>
    </tr>
    <tr>
      <td>Saints Academicals</td>
      <td>18</td>
      <td>16</td>
    </tr>
  </tbody>
</table>

```



Name	Wins	Draw
Pars United	29	5
Saints Academicals	18	16

## Lists


Tag	Explanation of Use
<code>&lt;ul&gt;</code> <code>&lt;/ul&gt;</code>	The unordered list tag creates a bullet point list. Each bulleted item should be contained between list item tags.
<code>&lt;ol&gt;</code> <code>&lt;/ol&gt;</code>	The ordered list tag creates a numbered list. Each numbered item should be contained between list item tags.
<code>&lt;li&gt;</code> <code>&lt;/li&gt;</code>	The list item tags are used to enclose each item in a list.

### Example

```

<ul>
  <li>Bread</li>
  <li>Butter</li>
  <li>Jam</li>
</ul>

```

- 
- Bread
  - Butter
  - Jam