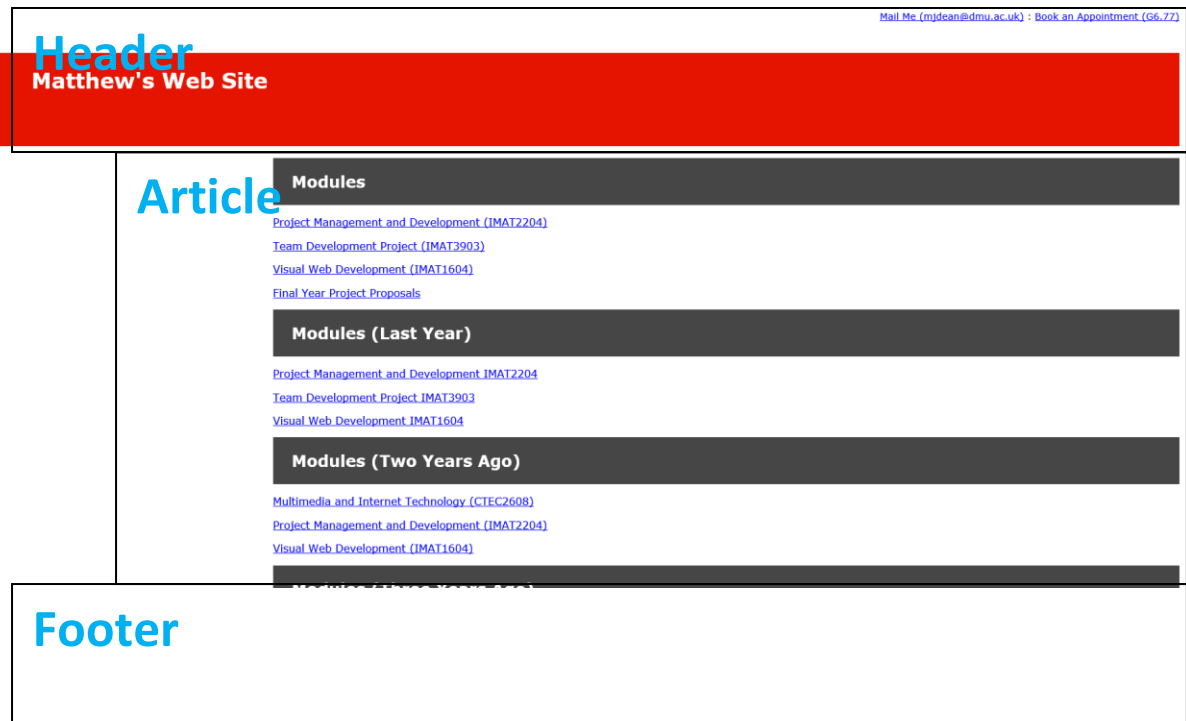## Styling your Interface

There have been various versions of HTML since its first inception. HTML 5 being the latest has benefited from being able to look back on these previous versions and make some very sensible changes.

One thing that was noted in the creation of a web site was that people tended to create the same sections to their sites.

Fred's DVD Swap Shop

Sign Up

EMail Address

Password

Login

Resend Password

Fred's Wish List

About

Search   Search   Clear   6 record(s) found

Blade Trinity
Pulp Fiction
Seven
Shirley Valentine
The Amazing Mr Blunden
X Men

For example most web sites have a top bit that appears on all pages of the site, i.e. the header. Many sites also include a footer at the bottom of every page. There is also a section for the main article of the text which is common to all pages. Lastly there is often a section containing all of the navigation buttons in the site.

Taking into account that many people were adding these sections anyway, the designers of HTML 5 decided that it made sense to build them into the mark-up language.

This lab session will give an outline of these tags, explain how to use them and explain why External Cascading Style Sheets are so important.

CSS Zen Garden

CSS Zen garden is a site that helps to illustrate why CSS is so important when designing pages with HTML 5.

Visit CSS Zen Garden by following this link…

http://www.csszengarden.com/

You should see a page like the following one.

CSS ZEN GARDEN
The Beauty of CSS Design

A demonstration of what can be accomplished through CSS-based design. Select any style sheet from the list to load it into this page.

Download the example ⬇ HTML FILE and ⬇ CSS FILE

THE ROAD TO ENLIGHTENMENT

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, broken CSS support, and abandoned browsers.

We must clear the mind of the past. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WASP, and the major browser creators.

The CSS Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with clarity. Learn to use the time-honored techniques in new and invigorating fashion. Become one with the web.

SO WHAT IS THIS ABOUT?

There is a continuing need to show the power of CSS. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view some of the existing designs in the list. Clicking on any one will load the style sheet into

Before doing anything else take a look at the headings for the main article.

- The Road to Enlightenment
- So what is this about?

Note also the text for the paragraphs…

"Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, broken CSS support, and abandoned browsers.

We must clear the mind of the past. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP, and the major browser creators.

The CSS Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with clarity. Learn to use the time-honored techniques in new and invigorating fashion. Become one with the web."
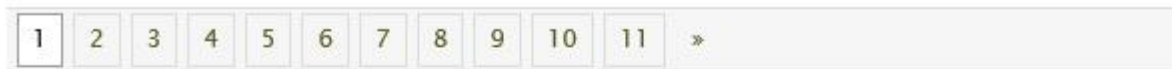
Ok – now on the right hand corner of the site press the button that says to view all designs.

You should be presented with a selection of different page designs.

This page is an archived list of Zen Garden designs added to date. You can also see the ones that didn't make the cut. Total designs listed: **218**

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | » |

221

Mid Century Modern
ANDREW LOHMAN, United States

220

Garments
DAN MALL, United States

219

Steel
STEFFEN KNOELLER, Germany

218

Apothecary
TRENT WALTON, United States

217

Handcrafted
ELLIOT JAY STOCKS, United Kingdom

216

Fountain Kiss
JEREMY CARLSON, United States

215

A Robot Named Jimmy
MELTMEDIA, United States

214

Verde Moderna
DAVE SHEA, Canada

213

Under the Sea!
ERIC STOLTZ, United States

Select a design you like the look of and you should see that design applied to the page.

CSS Zen Garden

The Beauty of CSS Design

A demonstration of what can be accomplished through CSS-based design. Select any style sheet from the list to load it into this page. Download the example html file and css file

The Road to Enlightenment

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, broken CSS support, and abandoned browsers.

We must clear the mind of the past. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP, and the major browser creators.

The CSS Zen Garden invites you to relax and meditate on the

So What is This About?

There is a continuing need to show the power of CSS. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view some of the existing designs in the list. Clicking on any one will load the style sheet into this very page. The HTML remains the same, the only thing that has changed is the external CSS file. Yes, really.

CSS allows complete and total control over the style of a hypertext document. The only way this can be illustrated in a way that gets people excited is by demonstrating what it can truly be, once the reins are placed in the hands of those able to create beauty from

As before, notice the headings and the test for the paragraphs, they are exactly the same.

So what is the point of this?

The point is that be splitting the presentation of a page by storing it in a CSS from the content of the page i.e. the text it is possible to change the layout of the entire site by switching to a different style sheet.

Creating your CSS

For this to work you will need to use the work you completed last week.
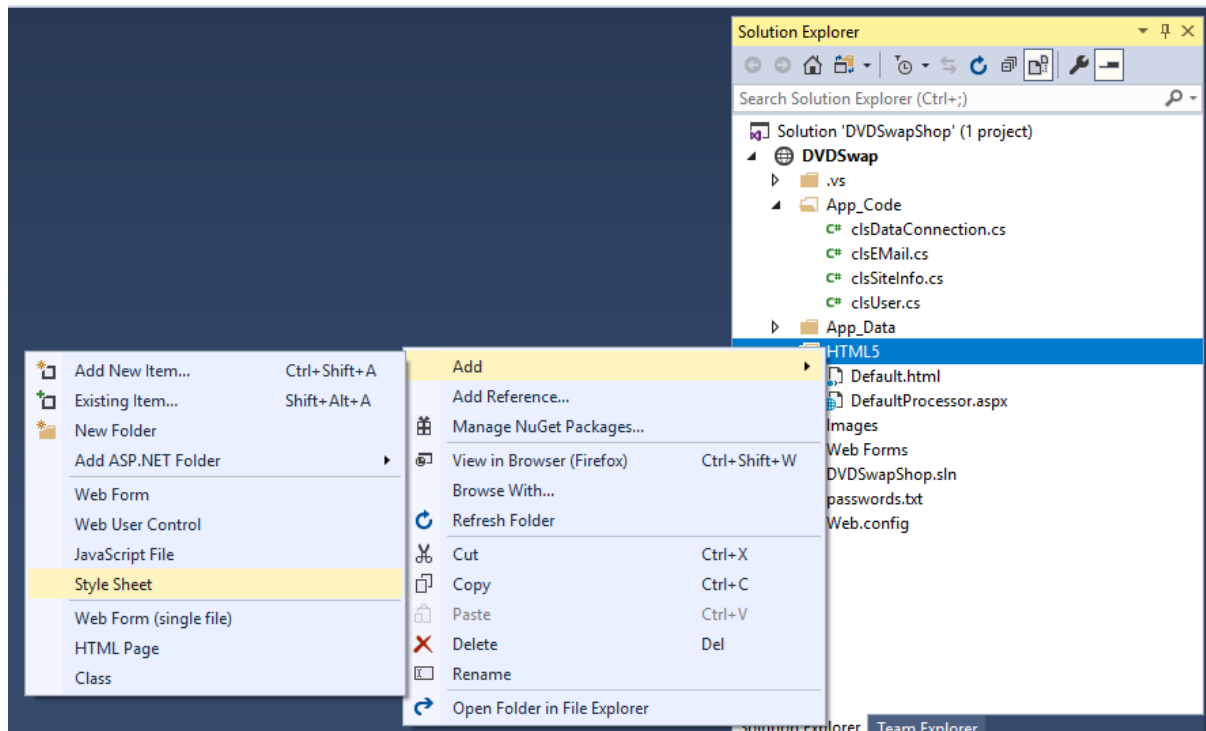
Once opened in Visual Studio you will see the following two elements already in place.
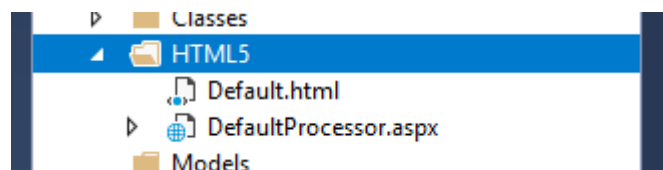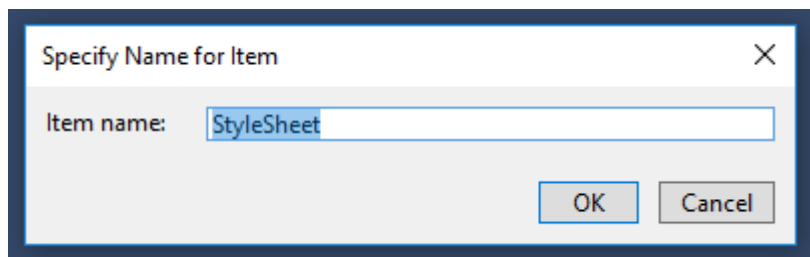
- Default.html   our HTML5 password page
- DefaultProcessor.aspx   the form processor for the above HTML page

We shall now add to this an external style sheet to allow us to store the styling we want to apply to the page.

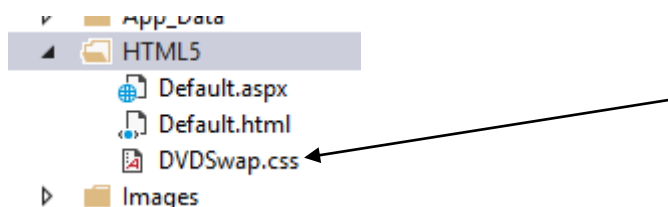If you right click on the web-site you may add the CSS like so…

You should be prompted to give the CSS a name…

Try to give it a sensible name that makes sense to you rather than letting the system do it for you.

In this example we shall call it DVDSwap.

Now that we have the CSS we need to add some mark-up to the HTML page to let it know that it exists.

Open the page Default.html and edit the HTML like so…

```
<!DOCTYPE html>
<html>
<head>
    <title>DVD Swap Shop</title>
    <meta charset="utf-8" />
    <link href="DVDSwap.css" rel="stylesheet" />
</head>
<body>
    <form method="post" action="DefaultProcessor.aspx">
        E-Mail Address
        <br />
        <input type="text" name="txtEMail" />
        <br />
        Password
        <br />
        <input type="password" name="txtPassword" />
        <br />
        <input type="submit" value="Submit" />
        <br />
    </form>
</body>
</html>
```
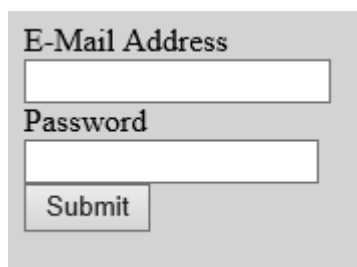
This extra line of mark-up tells the HTML page that its formatting should come from the CSS page DVDSwap.css.

So let's test this to make sure that it works.

Open the CSS and modify the styling for the body of the page like so.


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

If you view the web page in the browser you should see the background of the page is grey.

So what did we do there?

What we did was apply some styling to an HTML tag.

As discussed last week one of the required tags for an HTML page is the body tag.

```
<!DOCTYPE html>
<html>
<head>
    <title>DVD Swap Shop</title>
    <meta charset="utf-8" />
    <link href="DVDSwap.css" rel="stylesheet" />
</head>
<body>
    <form method="post" action="Default.aspx">
        E-Mail Address
        <br />
        <input type="text" name="txtEMail" />
        <br />
        Password
        <br />
        <input type="password" name="txtPassword"  />
        <br />
        <input type="submit" value="Submit" />
    </form>
</body>
</html>
```

By adding the style to the CSS like so…


```
body {
    background-color:lightgrey;
}
```
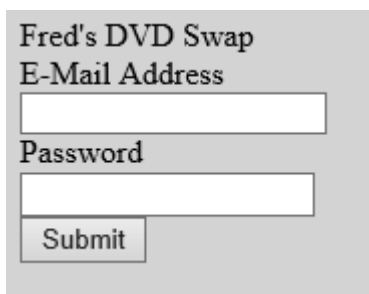
We are telling the browser whenever it has to apply the tag <body> apply the background colour light grey.

Let's have a play now with the built-in HTML 5 tags.

Let's start by creating a header for the page like so…

```
DVDSwap.css        Default.html    ₊ ✕
    <!DOCTYPE html>
  <html>
  <head>
      <title>DVD Swap Shop</title>
      <meta charset="utf-8" />
      <link href="DVDSwap.css" rel="stylesheet" />
  </head>
  <body>
      <header>
          Fred's DVD Swap
      </header>
      <form method="post" action="Default.aspx">
          E-Mail Address
          <br />
          <input type="text" name="txtEMail" />
          <br />
          Password
          <br />
          <input type="password" name="txtPassword"  />
          <br />
          <input type="submit" value="Submit" />
      </form>
  </body>
  </html>
```

When you view the page you should see the following.

Fred's DVD Swap
E-Mail Address
[   ]
Password
[   ]
[Submit]

It won't look right until we add some styling via the CSS.

Let's add a new style to the CSS and redefine the <header> tag, like we did for the body tag.

This time we will apply a style to the text within the header.

Modify the style sheet to include a new style called header like so…

```
DVDSwap.css    + X   Default.html
  body {
       background-color:lightgrey;
  }


  header {
       border:solid;
  }
```

When you view the page you should see the section of text marked header is now enclosed in a border.

```
Fred's DVD Swap
E-Mail Address

Password

 Submit
```
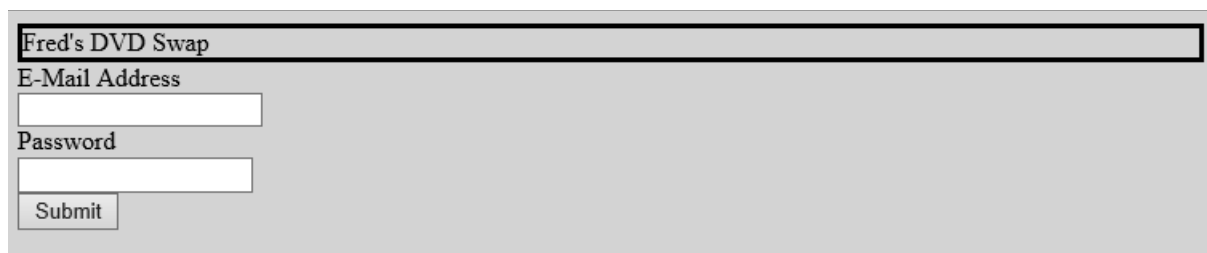
Let's mark-up the header a bit more to differentiate between the main header and the main text on the page.

Modify the HTML for the header like so…

```
<header>
    <h1>
        Fred's DVD Swap
    </h1>
    <h2>
        Swapping DVDs Since 2016
    </h2>
</header>
```

Now look at the page in the browser…

Fred's DVD Swap

Swapping DVDs Since 2016

E-Mail Address

Password

Submit

By marking up the text using the <h1> and <h2> tags we applied default formatting to the different sections of text. These heading tags make the text go smaller as you increase the number e.g. h3, h4 etc. If we decide that we want the sub heading to be smaller still we could mark it up as h3 or h4.

Since we have marked up each section within the header with specific heading styles we can even mark them up individually.

Add the following styles to your style sheet.


```css
body {
    background-color:lightgrey;
}

header {
    border:solid;
}

h1{
    color:blue;
}

h2{
    font-style:italic;
}
```

Now look to see what we have…

Having styled the parent tag (header) along with two child tags (h1 & h2) we shall make a change to the parent tag again to centre the text like so…

```css
header {
    border:solid;
    text-align:center;
}
```

This will create the following effect on our page…

There are some points to note so far. Firstly by styling the parent tag <header> with centred text, the child tags <h1> and <h2> inherit this formatting.

Secondly we can apply new formatting to the child tags overriding the parent formatting, in this case a different font colour of green or italics.

Let's have a go at doing something different this time using the article tag.

Mark-up the HTML such that we have a section for the main body of the page styled using the article tag, like so…

```
<body>
    <header>
        <h1>
            Fred's DVD Swap
        </h1>
        <h2>
            Swapping DVDs Since 2016
        </h2>
    </header>
    <article>
        This is where the main text of the page will go.
    </article>
    <form method="post" action="Default.aspx">
        E-Mail Address
        <br />
        <input type="text" name="txtEMail" />
        <br />
        Password
        <br />
        <input type="password" name="txtPassword"  />
        <br />
        <input type="submit" value="Submit" />
    </form>
</body>
```

The first bit of styling may seem a little pointless at first but it will make sense shortly.

Add a new style for this tag with the following formatting…

```
DVDSwap.css*  -ㅁ ✕  Default.html*
body {
      background-color:lightgrey;
}

header {
      border:solid;
      text-align:center;
}

h1{
      color:blue;
}

h2{
      font-style:italic;
}

article {
      border:solid;
}
```
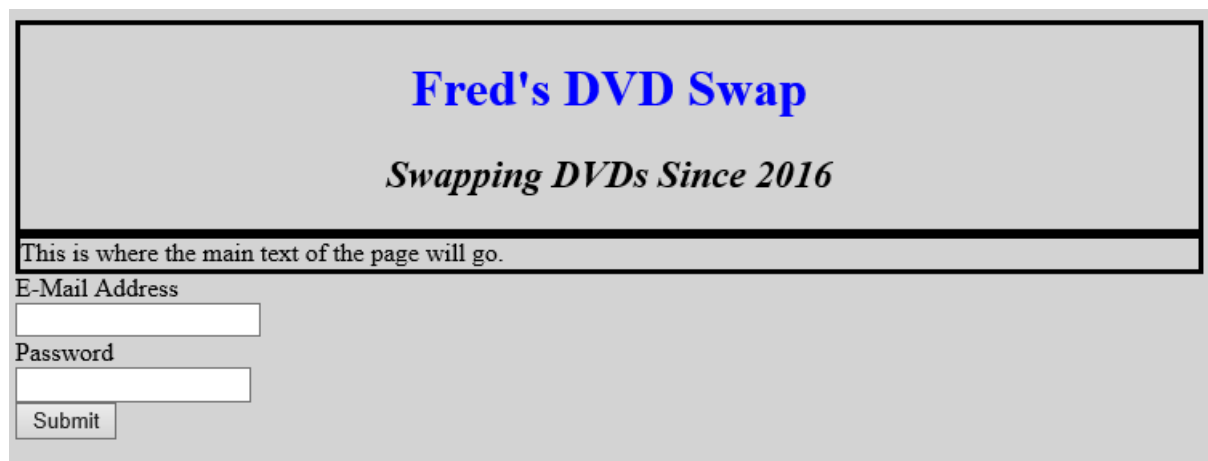
View the page and you should see something like this…

Fred's DVD Swap

Swapping DVDs Since 2016

This is where the main text of the page will go.

E-Mail Address

Password

Submit

What we are going to do next is position this new text explicitly on the page.
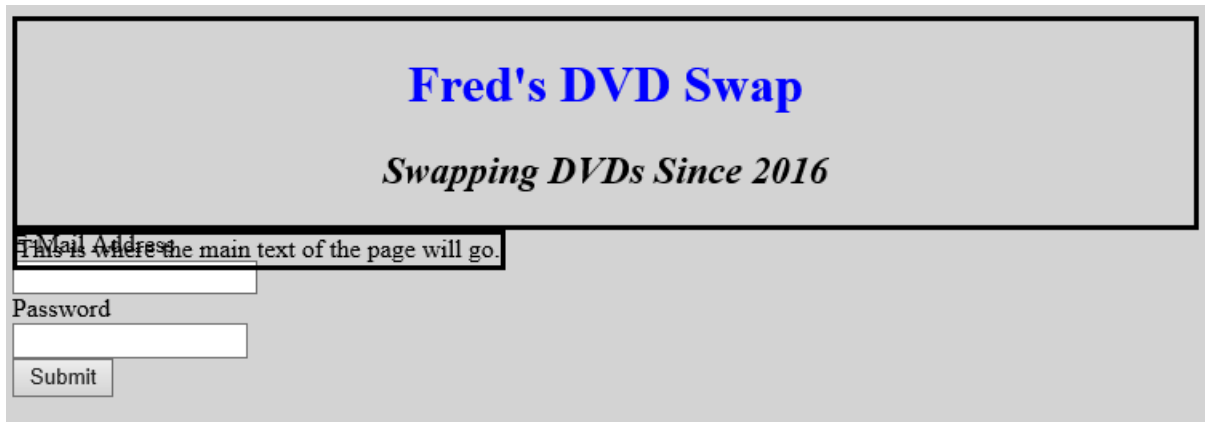
We shall change the positioning of the section to "fixed" so that we can tell the browser exactly where we want to place section on the page.

```
article {
    border:solid;
    position:fixed;
}
```
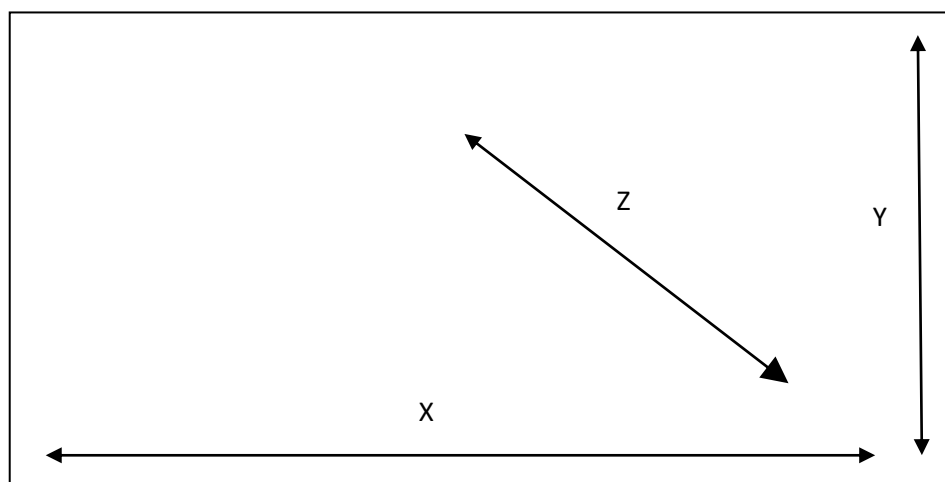
This should produce the following effect…

We will now specify a specific location on the page for this text.

Positioning Considerations

If we imagine the page displayed in the browser there are a number of ways that we can describe the dimensions of a page.

The page is typically a rectangle…

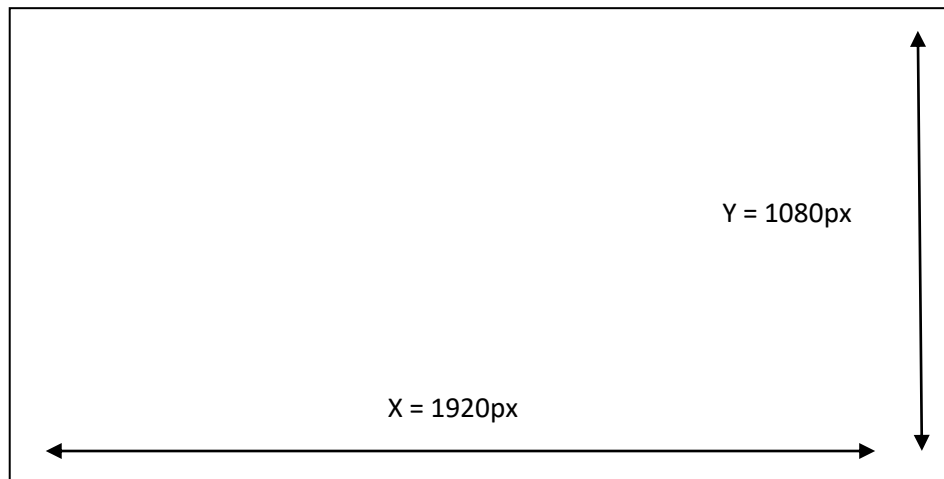The page has both height and width x and y.

The page also has a third dimension that is depth z.

z controls how objects overlap each other.

The values of X and Y on the screen from the point of view of the display may be thought of as dots or pixels.

For example 1920 x 1080 pixels (px) is typical screen resolution.

Y = 1080px

X = 1920px

The issue faced is that not all screens are the same size when it comes to pixels.

A mobile phone screen may have different values for x and y.

For example the iPhone has had a range of screen resolutions over time...

| 1st Gen, 3G & 3GS | 4 & 4S | 5, 5C & 5S | 6 | 6 Plus |
| 320×480 | 640×960 | 640×1136 | 750×1334 | 1242×2208 |

downsampled to

1080×1920

More about the crazy downsampling business here and here.

http://iphoneresolution.com/

This may mean that if we design our pages to fit to a certain screen resolution we may not get the desired effect when viewed on a device with lower resolution.

One of the problems with the design of the DVDSwap is that we are using web forms (which are fine) but they are using absolute positioning (which is not)!

Absolute positioning is what allows you to drag and drop your controls around the page whilst not having to worry about style sheets and mark-up. Fine if you are a beginner but not so fine if you want to create web applications that may be used on a range of devices.

Imagine you want to look at the DVDSwap shop interface on an iPhone. The problem is that it would end up looking something like this…
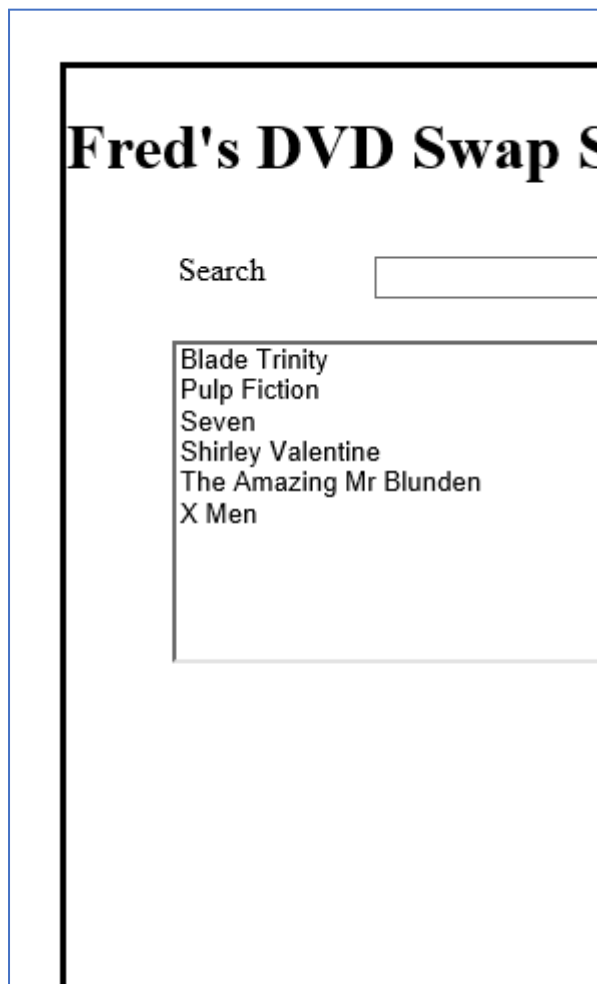
Why? Because absolute positioning is not useful for creating pages that scale gracefully over different devices. Because the size and position of each control is absolute in the grid it was originally created on the page has to be scrolled left and right, up and down to access the whole grid.

There is a better way of doing this

The other way of looking at the size of objects on the screen is to think of them as percentage values of the parent.

So for example the screen may be viewed like so…

```
                                                    │
                                                    ▲
                                                    │
                                                    │
                                                    │
                            Y = 100%                │
                                                    │
                                                    │
                                                    │
         X = 100%                                   │
                                                    │
   ◄────────────────────────────────────────────►  ▼
```
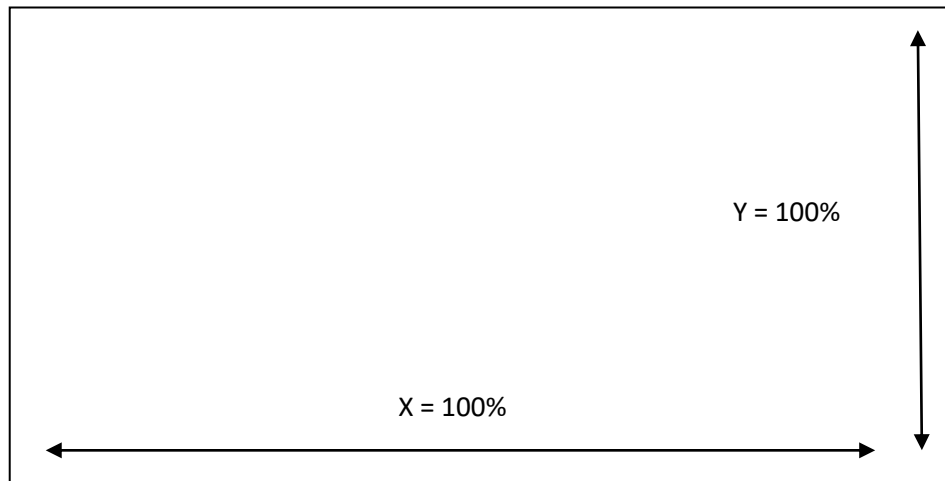
100% will always be 100% no matter what the size of the screen we are using.

So let's say we want to position something roughly in the middle of the screen?

If we make the object 40% of the screens height and width positioning it at 30% from the top and the left should land it in the middle…
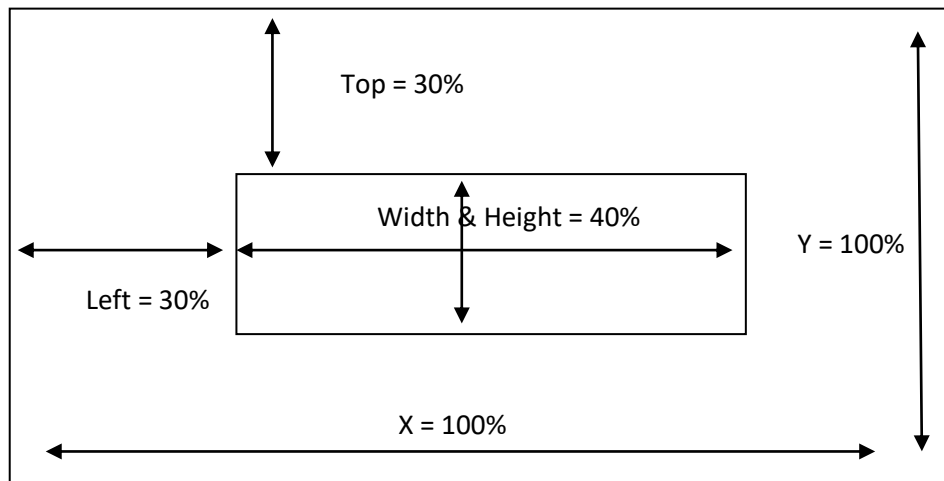
Let's try this with our CSS.

Modify the style for article like so…

```
article {
    border:solid;
    position:fixed;
    width: 40%;
    height:40%;
    top: 30%;
    left: 30%;
}
```

Since we have created a border to the styling it is easier to see where the boundaries of the formatting are.

Fred's DVD Swap

Swapping DVDs Since 2016

E-Mail Address

Password

[Submit]

This is where the main text of the page will go.

By specifying percentage values it doesn't matter what size the screen displaying the page is as it is always a percentage of the parent size. Try re-sizing the window and see what happens to the area you have just marked up. No matter what size the window is it will always keep the percentage values defined in the style sheet

This type of layout is called liquid layout as it flows with whatever screen size it is displayed in.

To complete this work you need to do the following.

Using the original DVDSwap as a guide set up the layout of your page with the header, article and nav to create a page that looks something like this…

Fred's DVD Swap Shop

Search [   ] [Search] [Clear] 6 record(s) found

Blade Trinity
Pulp Fiction
Seven
Shirley Valentine
The Amazing Mr Blunden
X Men

Sign up
EMail Address
[   ]
Password
[   ]
Login
Resend Password

Fred's Wish List
About

Don't worry about creating the additional buttons as yet just get the layout for the three main areas working.

Things to do…

Once you have done that settle on a colour scheme for your pages try to make it look like a professional web site.

Lastly create a second style sheet with a completely different colour scheme and layout to the first CSS.

Demonstrate that you can switch between the two style sheets by editing the HTML in your default page. In so doing you will have created a mini version of the CSS Zen Garden.