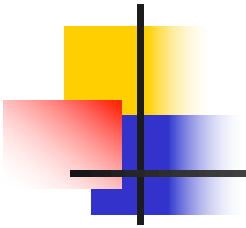


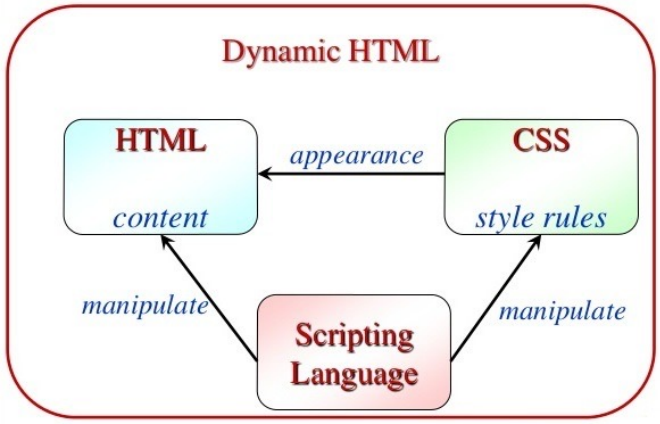
Lecture 3

Introduction to Dynamic HTML

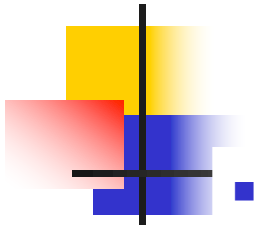
Dynamic HTML

Dynamic HTML, or **DHTML**, is an umbrella term for a collection of technologies used together to create interactive and animated websites by using a combination of a static markup language (such as HTML), a client-side scripting language (such as JavaScript), a presentation definition language (such as CSS).

Language	Type	Purpose
HTML	Markup Language	To create Structure of a webpage.
CSS	Stylesheet Language	To give style to the structure.
JavaScript	Scripting Language	To create a user interactive page and validate inputs.



HTML



- **HTML = Hypertext Markup Language**
- **It is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web.**
- **Markup Language = In the context of HTML it refers to text which is added to a document and which specifies how the document is to appear to software designed to view it (i.e. a browser).**
- **HTML has been through a number of versions. The latest is HTML5.**
- **The body that determines the HTML standard is the W3C or World Wide Web Consortium (https://www.w3.org/Consortium/)**

Hypertext

The Technical Definition:

- Hypertext is a type of database that has active cross-references and allows the reader to ``jump'' to other parts of the database as desired.
- This makes the reading process non-sequential.
- A hypertext database can be conceptualized as a network of **nodes** and **links** , where documents are the nodes and the links are are cross-references. We name these documents as **hyper-documents**, and these cross-references as **hyperlinks**.
- The underlying data model of this hypertext network can be viewed as a very complex graph.

The World Wide Web as a massive Hypertext Database

- The world wide web is a distributed database of documents stored on many computers networked together using the TCP/IP protocol.
- The retrieval of documents from the hypertext system that is the World Wide Web is done through a client server system using the HTTP protocol.
- This has had a number of consequences:-

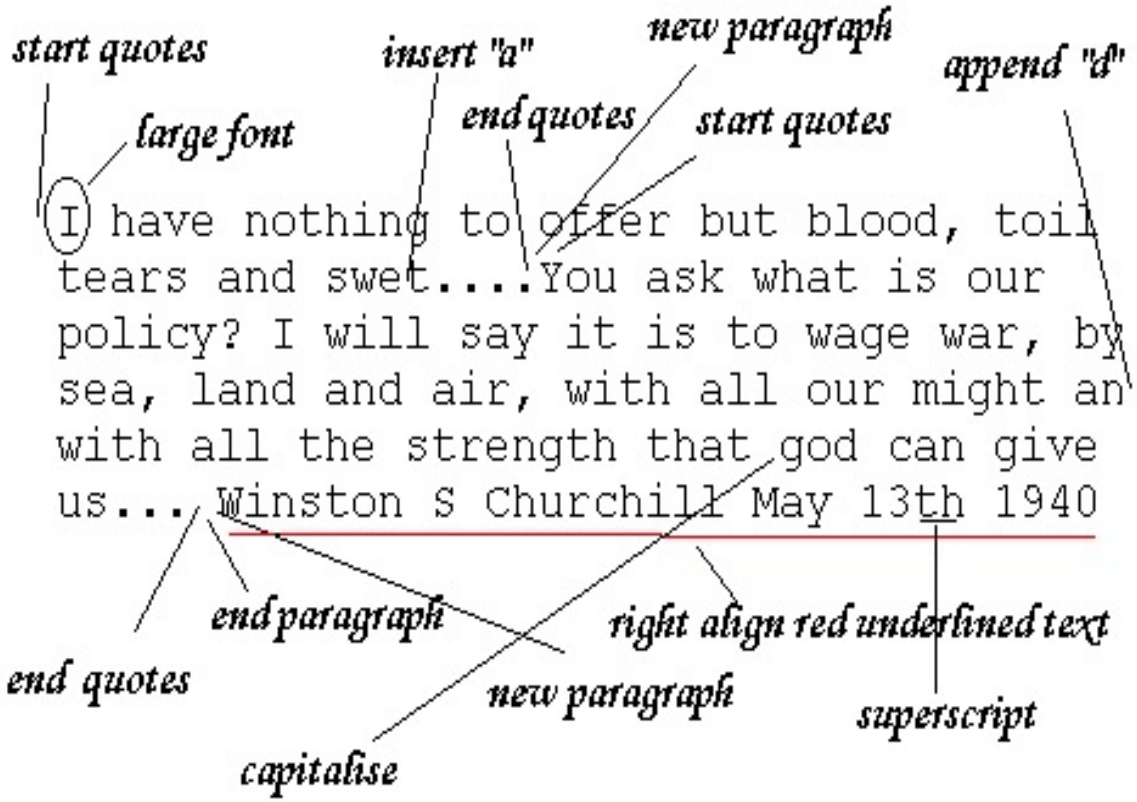
The number of users of the web is limited only by the size of the internet.

The number of documents available is only limited by the storage capacity of the web servers on the internet

The web (along with email) was a "killer app" for the Internet.

The web is the largest hypertext database in the world.

Text Markup



A hand coded piece of marked up text

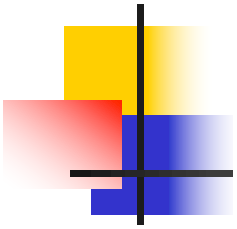
"I have nothing to offer but blood, toil tears and sweat...."

"You ask what is our policy? I will say it is to wage war, by sea, land and air, with all our might and with all the strength that God can give us..."

Winston S Churchill May 13th 1940

The same text after the markup has been applied

HTML Document Structure



```
<!DOCTYPE html>

<html>
    <head>
        <title>

        </title>
    </head>

    <body>

    </body>
</html>
```

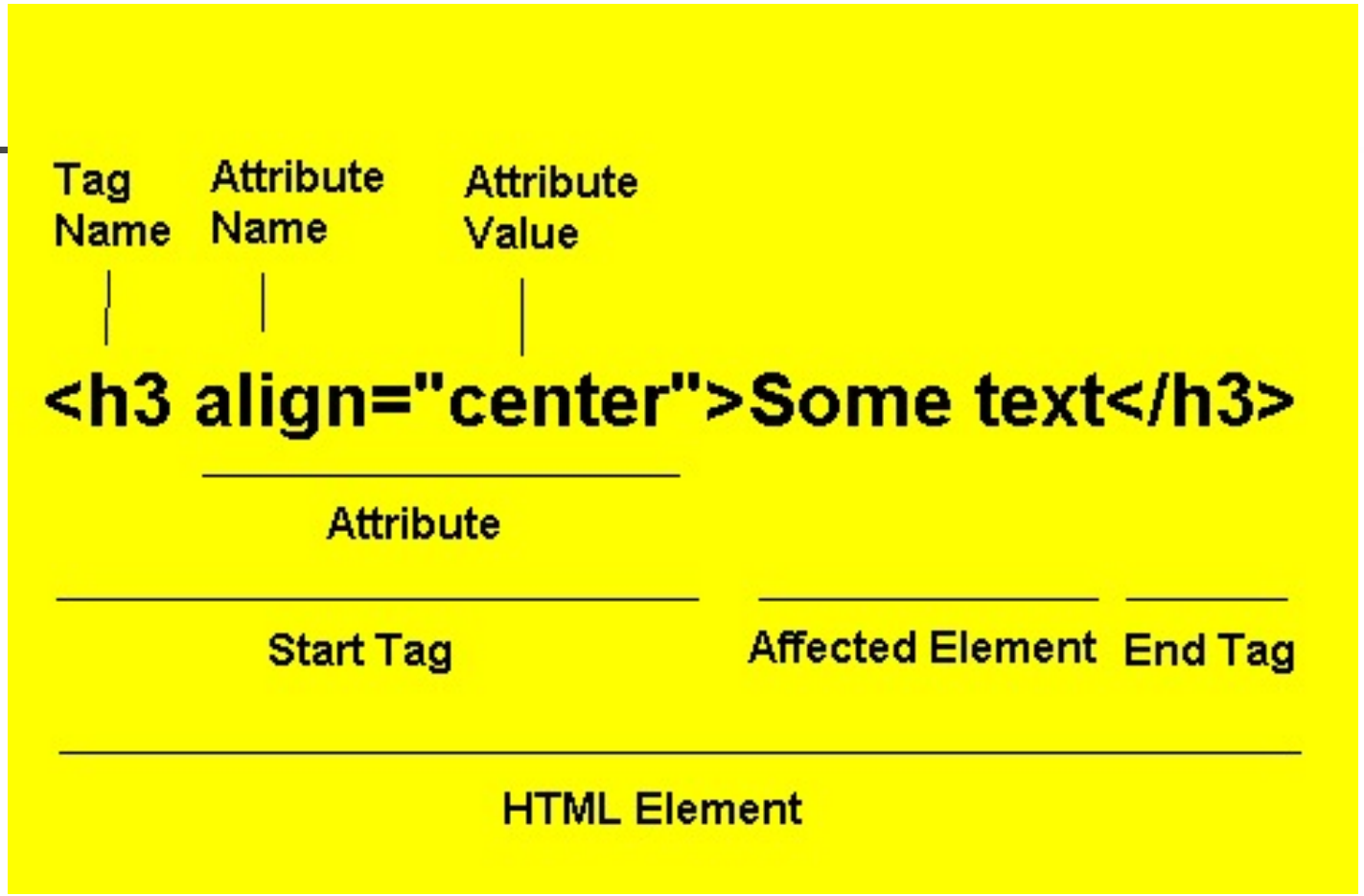
Head of the HTML Page

Body of the HTML Page

Whole HTML Document

The sequence of text above consists of the minimal set of html required for a valid html document. The basic text content of the page (This is my first web page) is surrounded by a set of tags.
A tag consists of some text contained in right and left angle brackets like this <body>. All the tags in the example are paired. This is not always the case however, some tags can be unpaired.

HTML Tag Structure



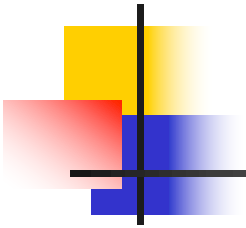
Paired tag: Opening tag (< >) and closing tag (</ >).
Singular tag: < />, such as <img />

https://www.w3schools.com/tags/default.asp

Links

- **The <a>Content html element marks off a portion of a document that is a hyperlink.**
- **Hyperlink=An element in an electronic document that links to another place in the same document or to an entirely different document.**
- **The <a> tag has a large number of possible attributes. Probably the most popular one is the href attribute.**

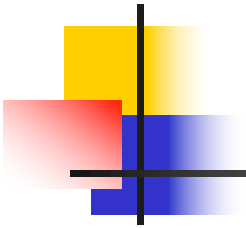
```
<body>
<a href="http://uts.edu.au">UTS</a> is a
public university in Australia.
</body>
```



Forms

- Forms are a way of entering data into a web page and processing it.
- Forms can specify an action to be taken on the data entered. The form tag specifies an action to be taken.
- The action can be a PHP, perl, asp or java server pages script.
- We will be looking at using PHP scripts to process the input from forms later in this course.

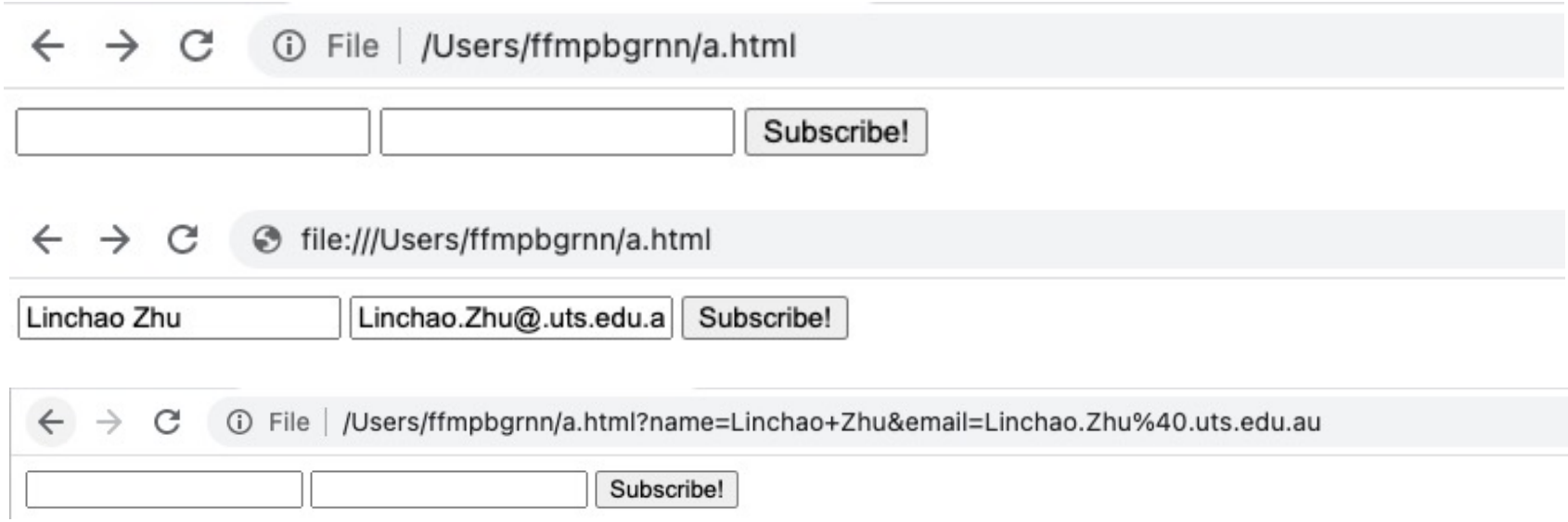
Forms (continued)



- The basic tag used by forms is the <form></form> element
- The <form> tag has several important attributes.
- These attributes include action and method.
- The action attribute specifies the program or script to execute when the form is submitted.
- The method attribute specifies how the data in the form is sent to the Web Server. There are 2 methods "get" and "post".
- The "get" method appends the data to the URL in the request. The "post" method keeps it separate.

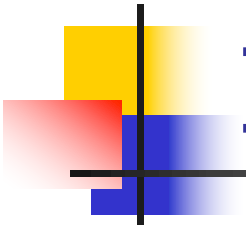
Forms in html

```
<form action="" method="get">
    <input type="text" name="name">
    <input type="text" name="email">
    <input type="submit" value="Subscribe!">
</form>
```



Forms in html

```
<form action="" method="get">
    <label>Enter your name: </label>
    <input type="text" name="name">
    <label for="email">Enter your email: </label>
    <input type="text" name="email">
    <input type="submit" value="Subscribe!">
</form>
```



Images

- The HTML <img> tag is used to embed an image in a web page.
- <img src="*url*" alt="*alternatetext*">
- <img alt="Google" src="/images/branding/googlelogo/2x/googlelogo_color_272x92dp.png">

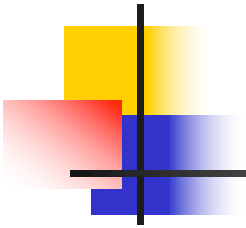


Image Maps

- Image Maps are a way of putting one or more links in an image
- One Image has one or more <area> tags each of which has a **href** attribute to a URL.
- Each area is known as a “hotspot”
- The areas can be circular, rectangular or polygonal

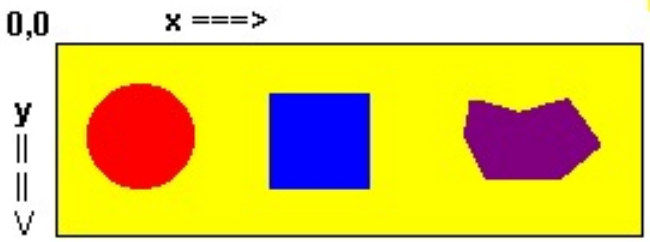
ImageMap Demonstration

```
<html>
<head>
<title>
Imagemap Demo</title>
</head>
<body>
<img id="pic" src="image1.gif"
     border="0" usemap="#shapes">
<map id="shapes" name="shapes">
   <area shape="circle" coords="47,35,22"
            href="circle.html">
   <area shape="rect" coords="100,18,143,56"
            href="rect.html">
   <area shape="poly"
            coords="186,20,205,25,224,17,238,39,
            220,53,191,53,180,34"
            href="poly.html">
</map>
</body>
</html>
```

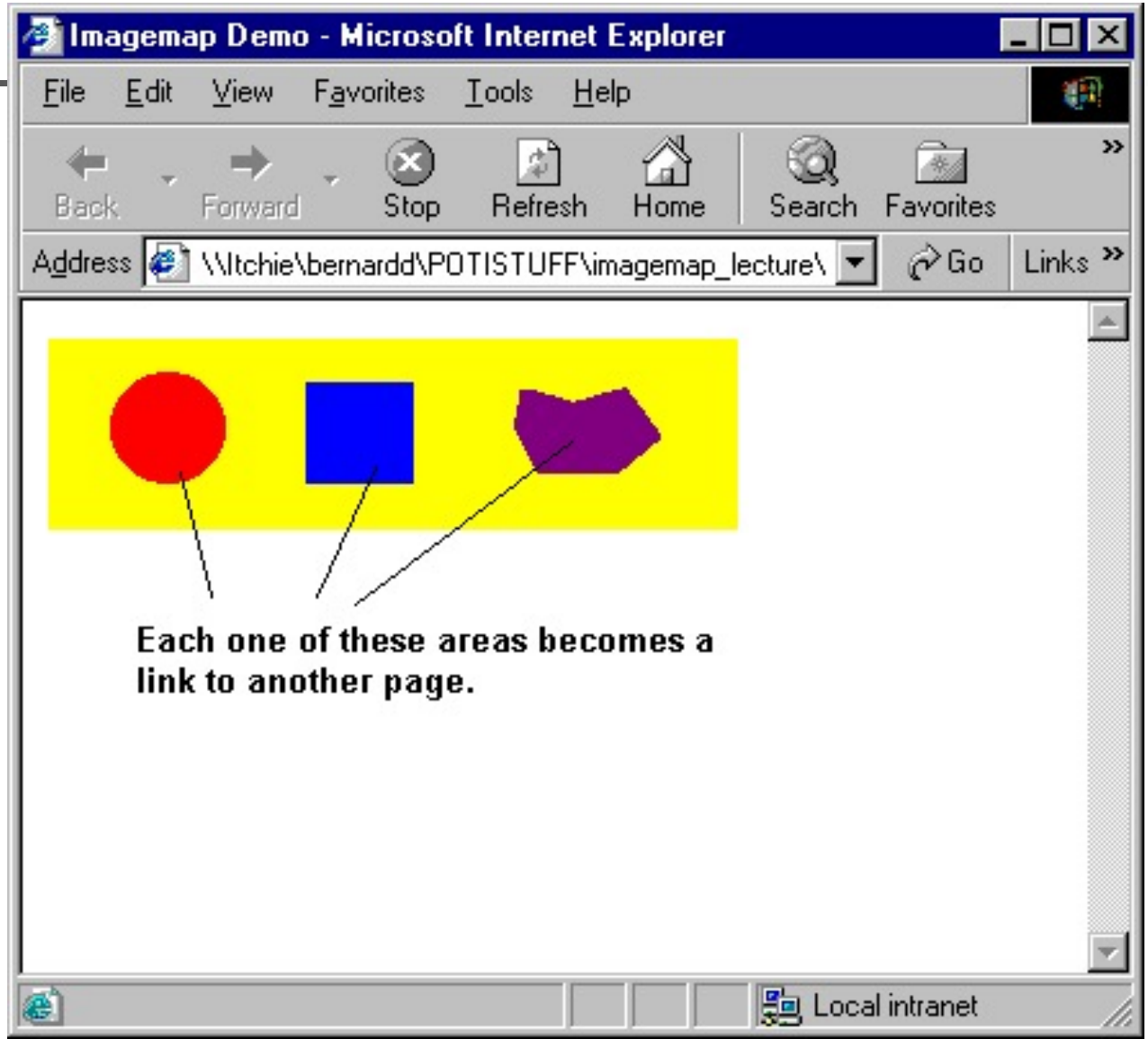
For the circle the coord are x,y and radius in that order.

For rect its x, y of left top and bottom right.

For poly it's each set of points in the polygon.

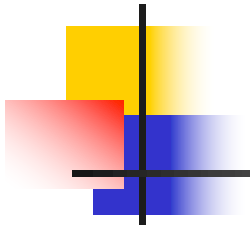


How the previous Page appears



Dynamic HTML Technique
Inline Style Sheets

Inline Style Sheets

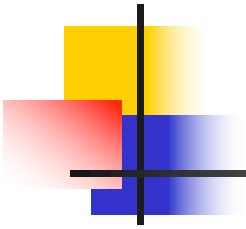


- Inline style sheets use the style attribute of a **HTML tag** to set the style of a particular tag. We are mentioning them now because they are used for the positioning of images in dynamic html

Inline Style sheet - Example

```
<h1 style="color:red; font-size:30px;">
    This is a heading
</h1>
<p style="color:green; font-size:18px;">
    This is a paragraph.
</p>
```

Graphics (the <img> tag), visibility and layering

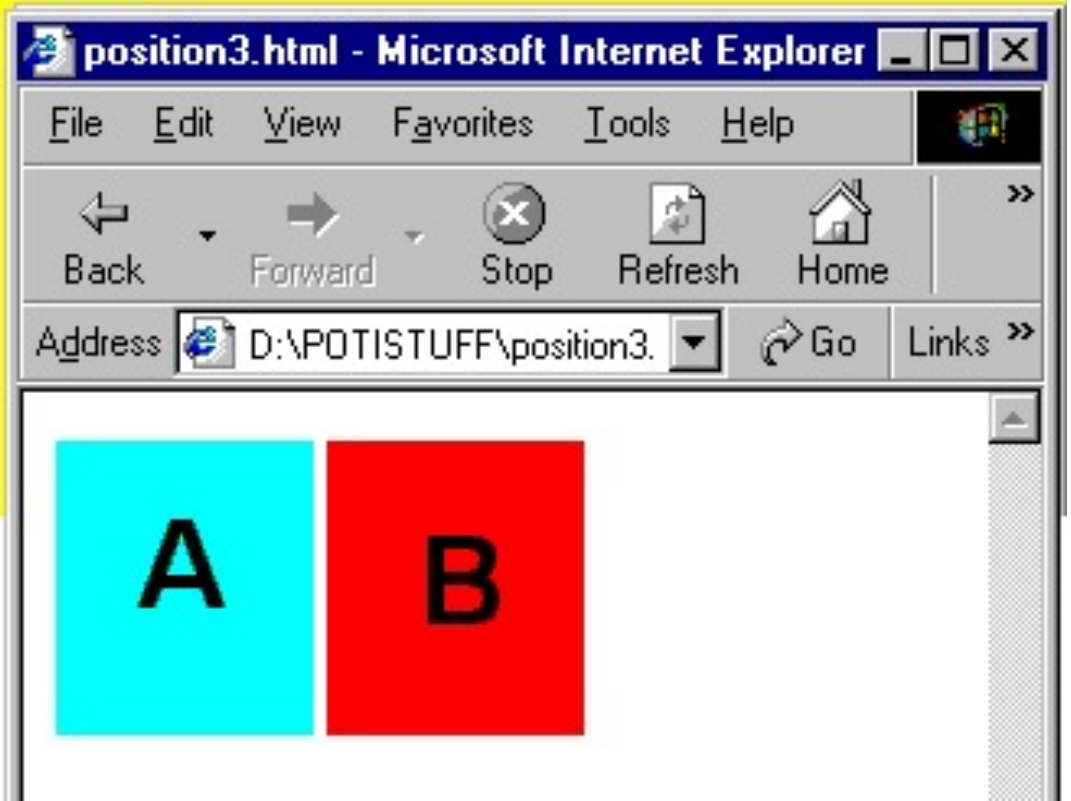


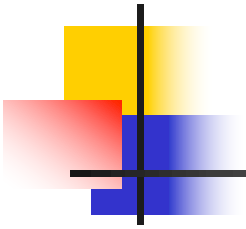
- The <img> tag in HTML can have a lot of its attributes changed by using an inline style sheet.

- Amongst the attributes that can be changed are an image's visibility (hidden|visible) and its position (absolute|relative|static).

Positioning Images (1)

```
<html>
<head><title> position3.html</title>
</head>
<body>

<img src="a.gif">
<img src="b.gif">

</body>
</html>
```





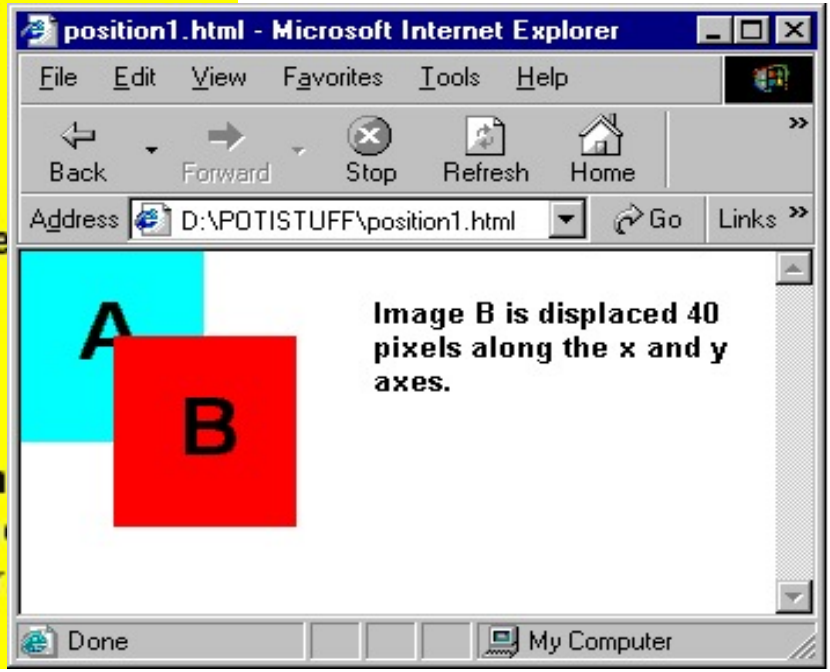
Render Flow

Positioning Images (2)

```
<html>
<head><title> position1.html</title>
</head>
<body>

<img src="a.gif" style="position: absolute;
        top: 0px;
        left: 0px">
<img src="b.gif"
        style="position: absolute;
        top: 40px;
        left: 40px">

</body>
</html>
```

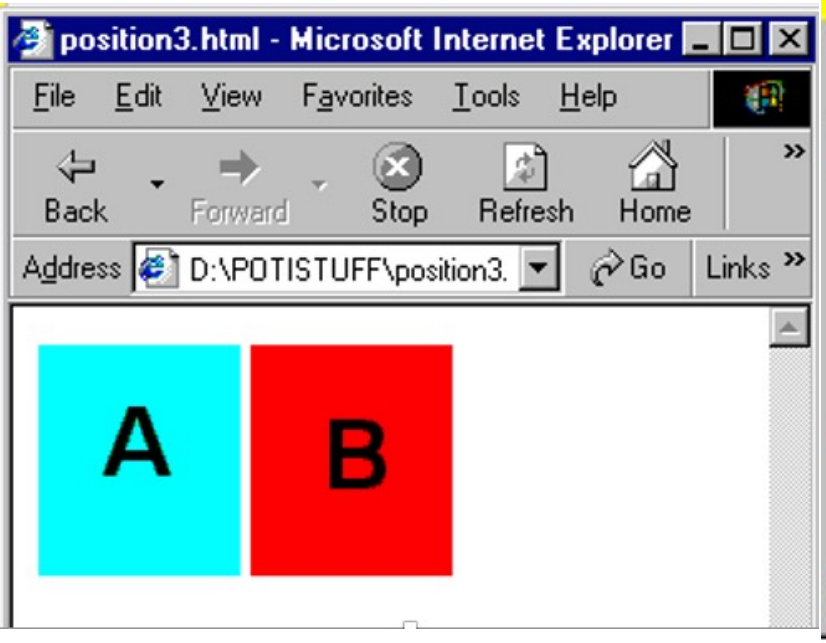
By using an inline style we can place images precisely.

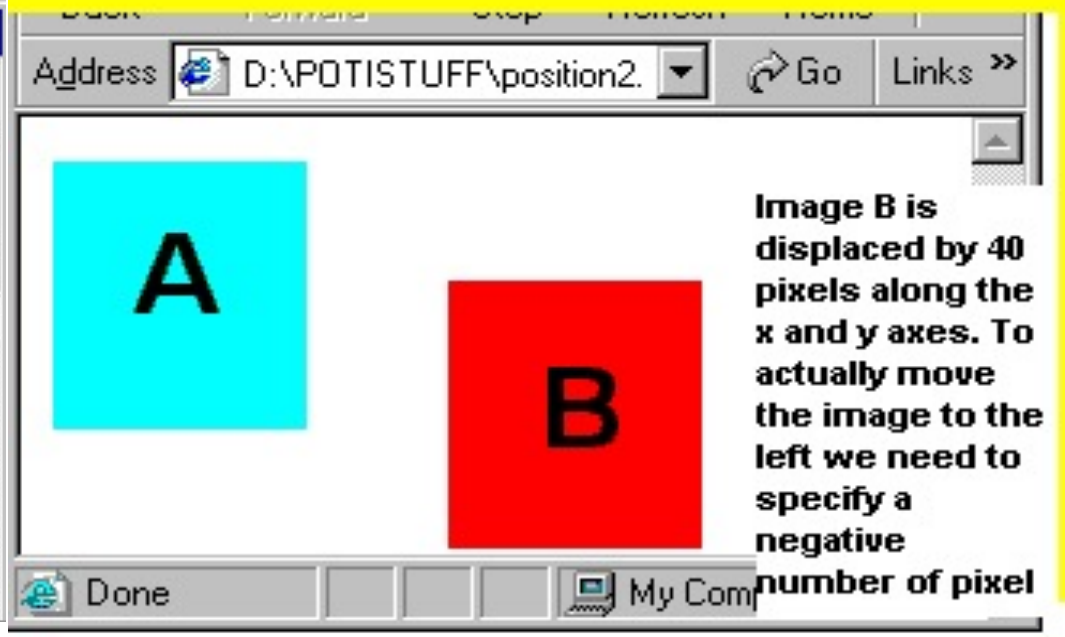
The position: absolute feature of the in stylesheet means that the image is pla relationship to the top left hand corner

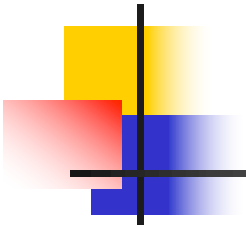


Positioning Images (3)

```
<html>
<head><title> position2.html</title>
</head>
<body>

<img src="a.gif">
<img src="b.gif"
      style="position: relative;
      top: 40px;
      left: 40px">

</body>
</html>
```







Dynamic HTML: Rollovers

- A Rollover is when the mouse pointer is over a particular part of the page and the appearance of the page changes.

- A common example is the items in menus in web pages where the colour of the item text and/or its background changes when the mouse moves over the item.

Graphics Rollover Example

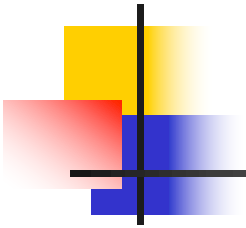
```
<html>
<head>
</head>
<body>

<img id=pic src="shade2.gif"
     onmouseover=pic.src="shade1.gif";
     onmouseout=pic.src="shade2.gif">

</body>
</html>
```

The code for a graphics rollover is similar to that for a text rollover. Inline code has been used to change the value of the src attribute when the mouseover and mouseout events occur.

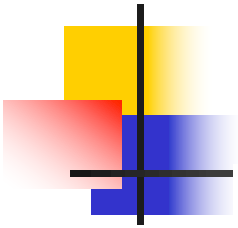
Graphics Rollover Example

```
<html>
<head>
</head>
<body>

<img id=pic src="shade2.gif"
     onmouseover=pic.src="shade1.gif";
     onmouseout=pic.src="shade2.gif">

</body>
</html>
```

The code for a graphics rollover is similar to that for a text rollover. Inline code has been used to change the value of the src attribute when the mouseover and mouseout events occur.



Cascading Style Sheets



The Problem with HTML

HTML tags can define both the structure and the presentation of web pages.

Examples of Structural Tags :

<head>, <p>, , <a>

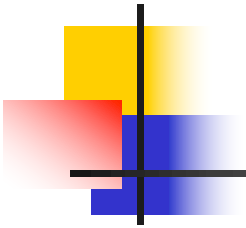
Examples of Presentation Tags :

<h3>, ,

Several years after the web was first created, it was realised that these two functions of HTML should be separated. Cascading Style Sheets were designed to separate out the presentation of web pages from HTML Tags.

Structure and Presentation

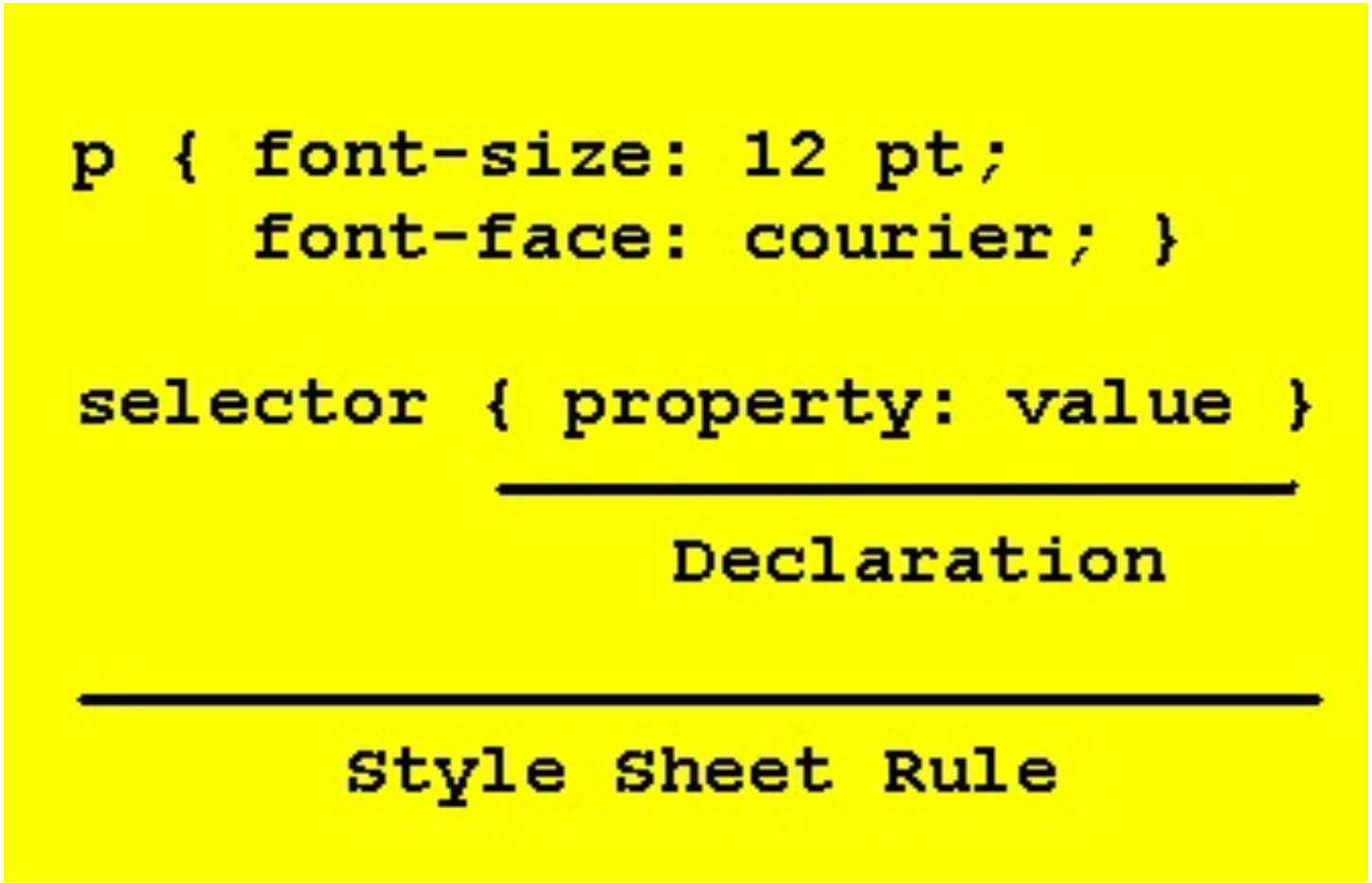
- **Structure** defines the components and areas on the page. This includes the head, body, title, links, tables, forms, and divs
- **Presentation** is the way the information is presented, such as its colour, size, background colour, font, etc.



Style Sheets and HTML Presentation

- Style Sheets move the presentation aspects (colour, font, etc.) out of HTML and into style sheets.

Style Sheet Structure



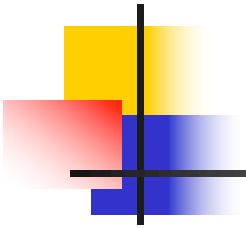
Cascading

- Cascading determines the precedence of a set of different types of style sheets when they all apply to the one html element.

```
h1 {
    color: red;
}
h1 {
    color: blue;
}
```

```
<h1>This is my heading.</h1>
```

This is my heading.



External Style Sheet

- This type has a Style Sheet in an external text file
- File must be named with the extension .css e.g. common.css
- An external style sheet just consists of a set of Style Sheet Rules
- **The document isn't tied to any particular page.**

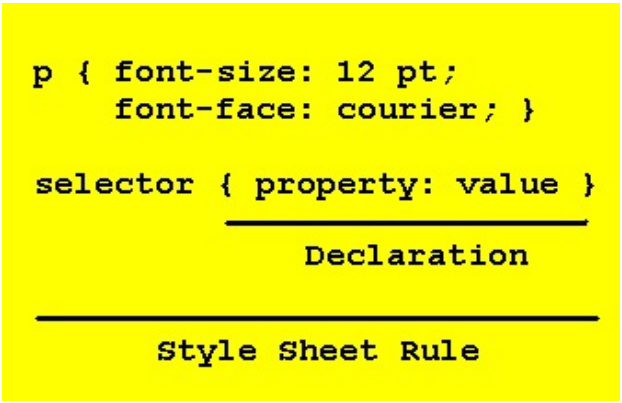
The <link> tag and External Style Sheets

- The <link> tag is an unpaired tag
- The <link> tag specifies relationships between the current document and other documents.
- Mainly used for linking in an external style sheet, but it can be used for other purposes.
- We use 3 attributes to read in an external style sheet rel, href and type.

```
<head>

<link rel = "stylesheet"

      href = "cssdir/thestyles.css"

      type = "text/css" >

<title> External CSS Example </title>

</head>
```

Style Sheet Selector

A selector tells the browser where a CSS rule is to be applied. Selectors can be user defined (class and id selectors) or based on a particular tag.



Style Sheet Selectors

- Tag : The style defines how the affected element of all instances of a particular tag display.
- ID : A style is given an identifier and tags are given the same identifier by using the id attribute.
- Class : A style is given a class selector. Tags can use this style by using the class attribute. Styles using the class selector are extremely versatile
- Inline : a tag has a style defined specifically for itself using the style attribute.

Style with a Tag as a selector

```
h1 {color: #0000FF;
    background-color: FFFFFF; }
```

Note the use of the style tag here to bring in the embedded style

```
<html>
<head>
<style type="text/css">

h1{color: #0000FF;
   background-color: FFFFFF;
   }

</style>
</head>
<body>
<h1>
This heading gets its
properties from an embedded
stylesheet</h1>
</body>
</html>
```

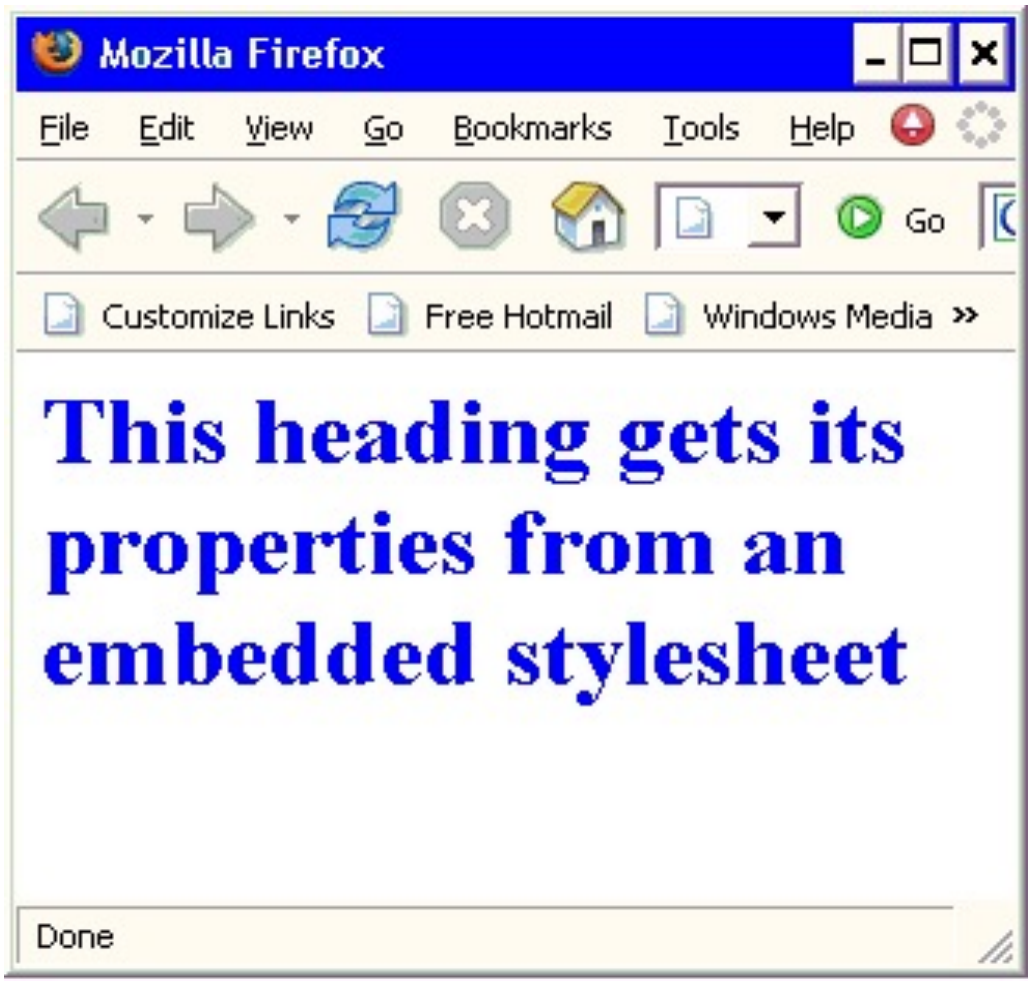
The style on the right is an example of a style where the selector is a tag.
In this case, the selector is h1. This means that any h1 tags will be displayed with the colours in the style.

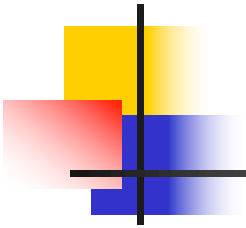
The HTML page at the left shows a full implementation of the style for the style at the top left.

We have embedded the style in the head of the HTML page.

Note that instead of embedding the style we could have put it in an external file and imported or linked it into the page.

How the previous page appears





Styles with ID as a selector

- You can only create one type of embedded style for any one type of tag.
- Creating a style using an id is a way of getting round this limitation.
- Can be used in a similar fashion to Class Stylesheets (which we will deal with next)

Example : Style with id selector

```
<html>
<head>

<style type="text/css">

p { color:red; }
#second { color: blue; }

</style>
</head>

<body>

<p>Here is a paragraph</p>

<p id="second">Here is another one</p>

</body>
</html>
```

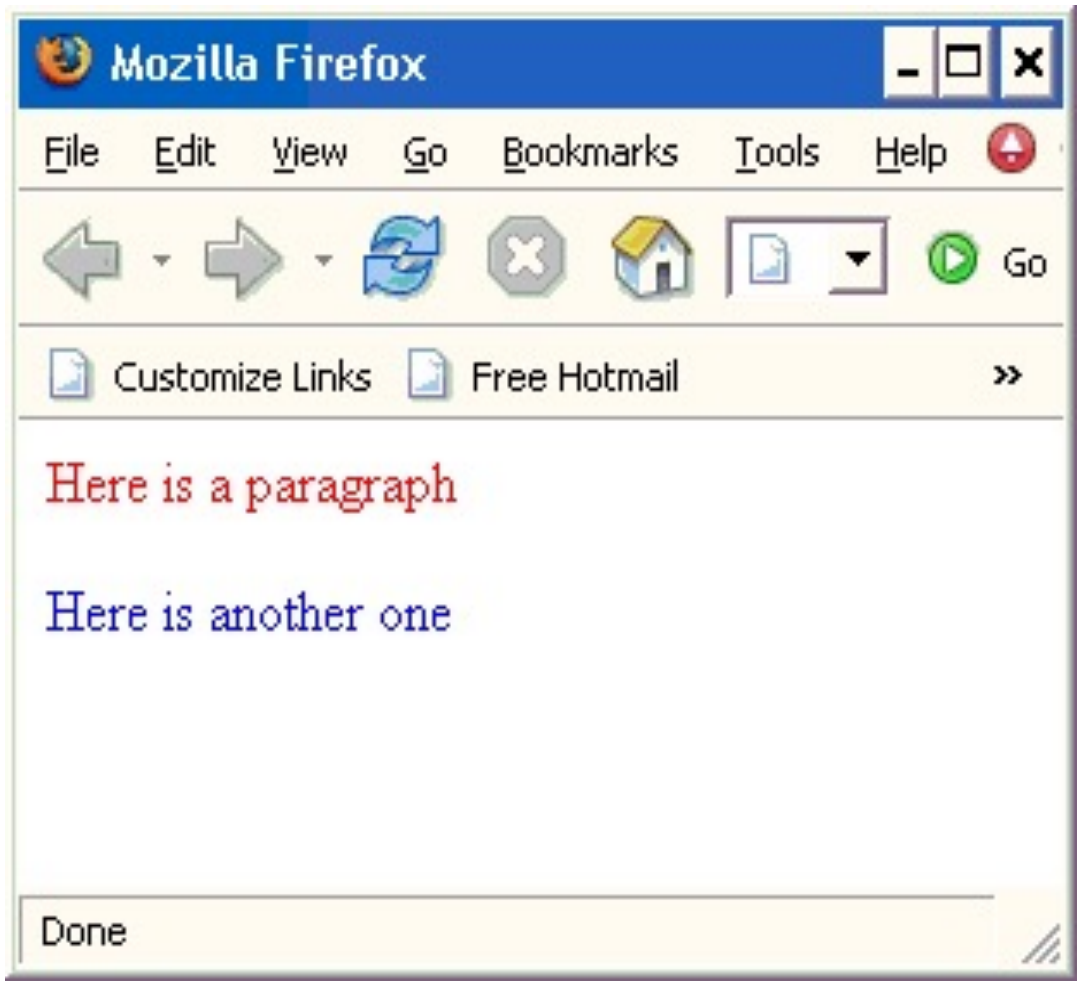
Paragraph tag <p> has a style associated with it here.

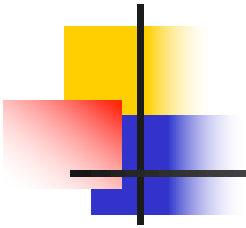
A second style with the id "second" is defined here.

This paragraph gets written with the style associated with the <p> tag

This paragraph is written with the style with an id of "second"

How the previous page appears





Class Selectors

- Style rules with class selectors can be applied to any appropriate html element by using the class attribute. They are similar to, but more versatile than id selectors.
- Can have any name the html writer decides on.
- Have no connection with classes in Object Oriented Programming languages.

Class Stylesheet Example

```
<head>

<style type="text/css">

.whatever
{ color: orange;
 background-color: rgb(0%,0%,0%)
}

.another { color: #888 }
</style>

</head>

<body>
<h1 class="whatever">Here is whatever class</h1>

<h1 class="another">Here is another class</h1>

</body>
</html>
```

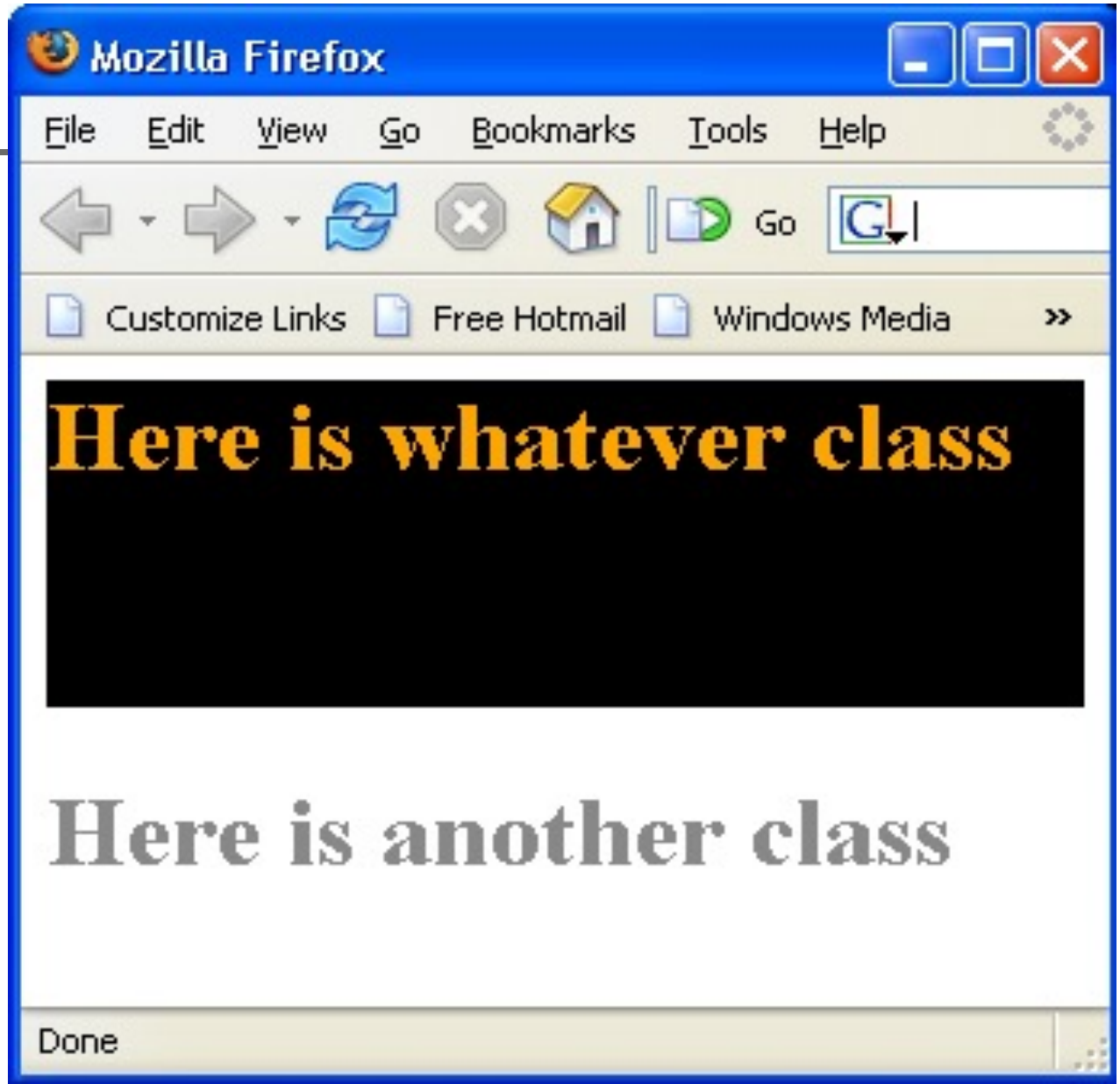
Two class based stylesheets are defined here .whatever and .another

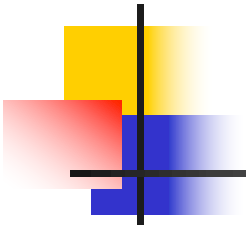
Notice that we have yet more ways of defining colors.

#888 is equivalent to 888888 in hex. Each number is repeated.

Here the two class styles are applied.

How the previous page looks



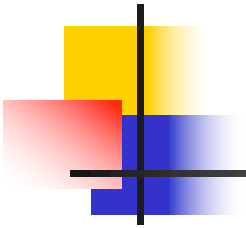


Style sheet Precedence

Browser style sheet have lowest precedence

External style sheets have next precedence

Inline styles have top precedence



- HTML tags
  - https://www.w3schools.com/tags/default.asp
  - Forms
  - Attributes
- CSS
  - Position
  - Selectors