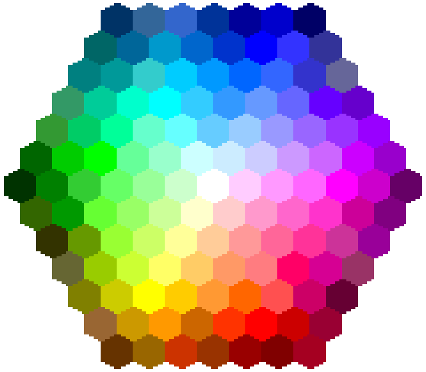
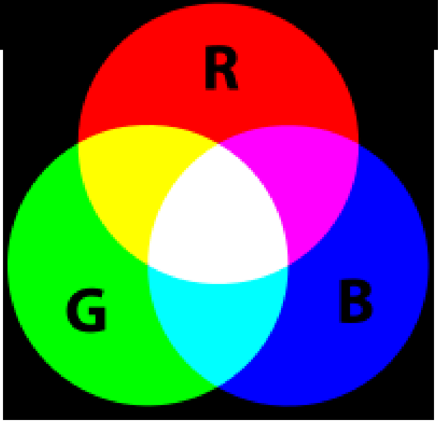
CM1102

Cascading Style Sheets (continued)



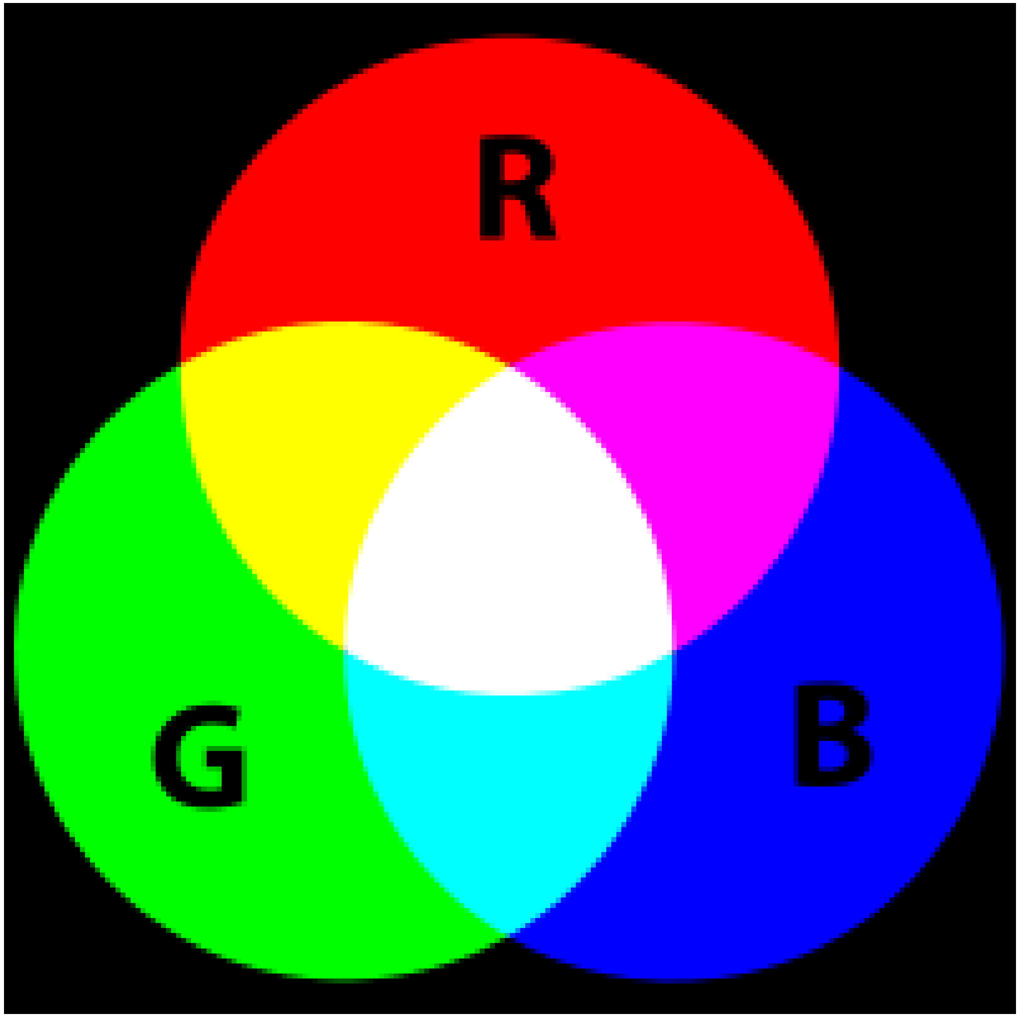
Colour



- can specify the colour of
  - text
  - background of the whole page
  - background of parts of the page
  (e.g. the cells of a table, or a block of text).
- Use hexadecimal or decimal numbers to represent the values of the red, green and blue (the primary colour components)
- Each colour component has a value between **`00`** and **`ff`** (hex) = 0 - 255 decimal, e.g. (with different colours)

```
<body style="background-color:#d2691e">
<h1 style="color:rgb(210,105,30)">
```

Colour – RGB Model



- #ff0000 = red
- #00ff00 = green
- #0000ff = blue
- #ffff00 = yellow
- #ff00ff = magenta
- #00ffff = cyan
- #000000 = black
- #ffffff = white
- #d2691e = rgb(210,105,30) = chocolate

Colour names

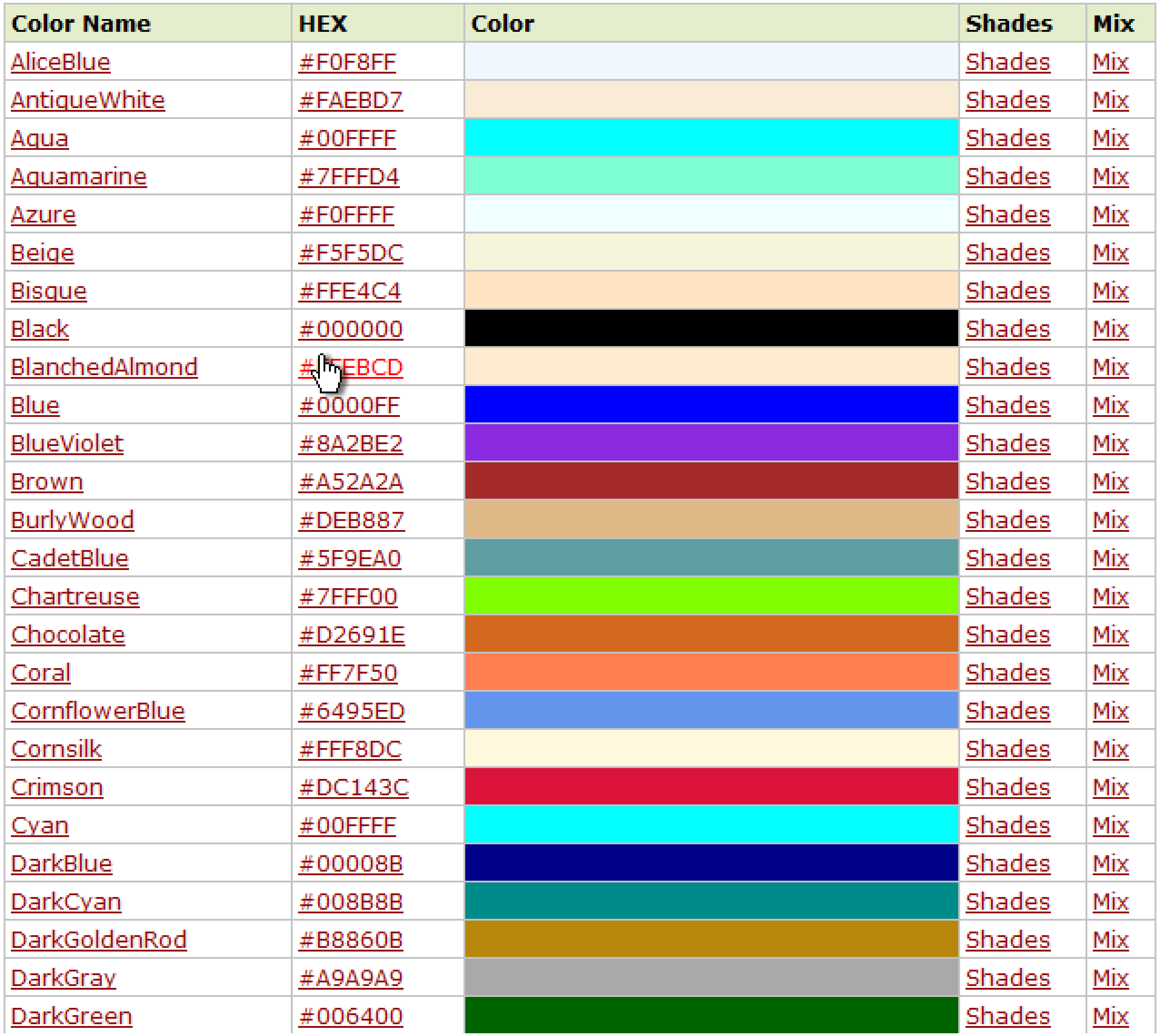
- Browsers allow textual names for colours (as well as hexadecimal and decimal), e.g.

```
<body style=“background-color:gray;
             color:black;” >
```

- For a list and illustration of the available colour names and their hexadecimal equivalent see appendix D of “Beginning HTML and CSS” or

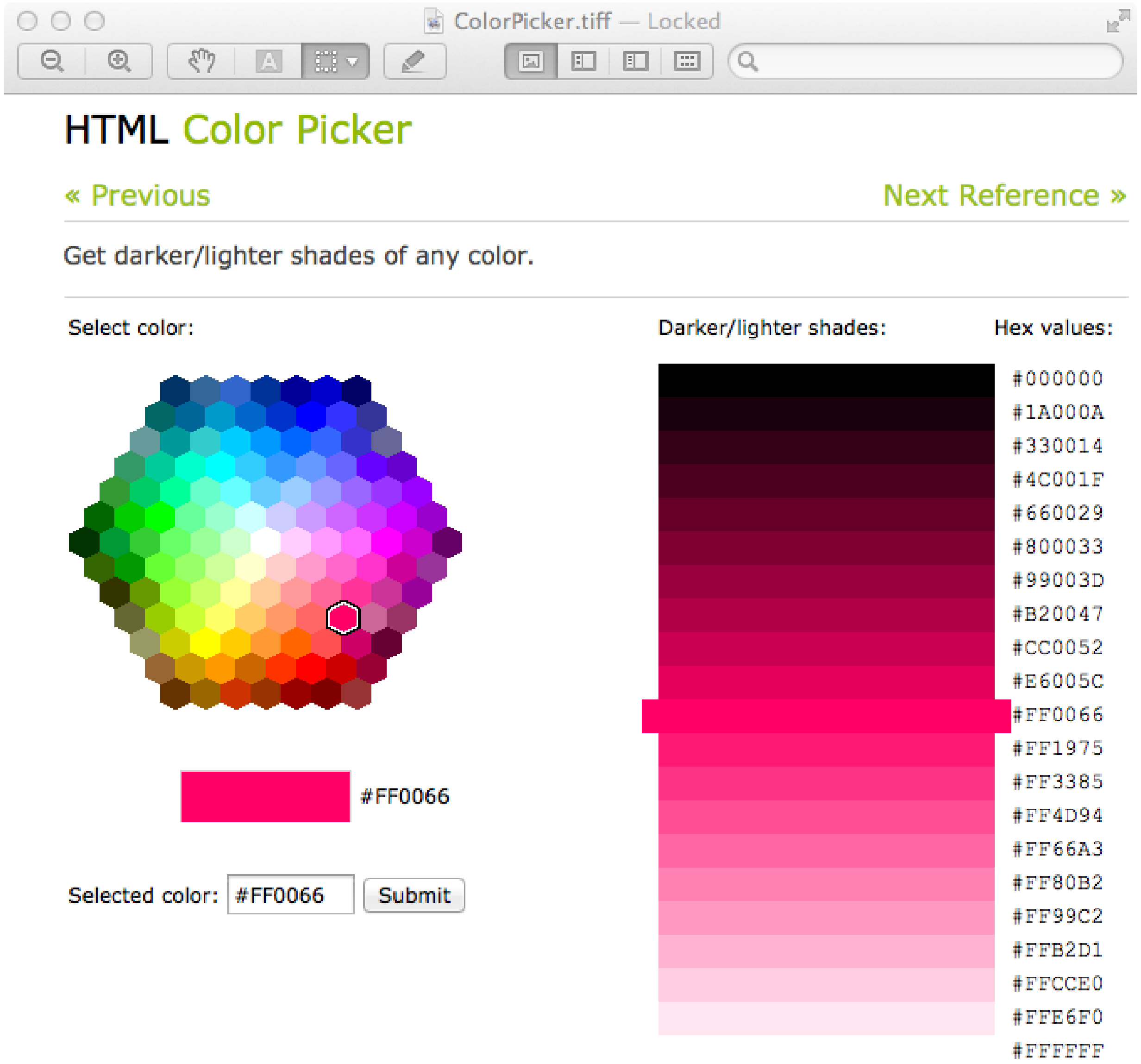
  http://www.w3schools.com
  /colors/colors_names.asp

Part of http://www.w3schools.com/colors/colors_names.asp



Color Name	HEX	Color	Shades	Mix
AliceBlue	#F0F8FF	⬜	Shades	Mix
AntiqueWhite	#FAEBD7	⬜	Shades	Mix
Aqua	#00FFFF	🟦	Shades	Mix
Aquamarine	#7FFFD4	🟩	Shades	Mix
Azure	#F0FFFF	⬜	Shades	Mix
Beige	#F5F5DC	⬜	Shades	Mix
Bisque	#FFE4C4	⬜	Shades	Mix
Black	#000000	⬛	Shades	Mix
BlanchedAlmond	#FFEBCD	⬜	Shades	Mix
Blue	#0000FF	🟦	Shades	Mix
BlueViolet	#8A2BE2	🟪	Shades	Mix
Brown	#A52A2A	🟫	Shades	Mix
BurlyWood	#DEB887	🟫	Shades	Mix
CadetBlue	#5F9EA0	🟦	Shades	Mix
Chartreuse	#7FFF00	🟩	Shades	Mix
Chocolate	#D2691E	🟧	Shades	Mix
Coral	#FF7F50	🟧	Shades	Mix
CornflowerBlue	#6495ED	🟦	Shades	Mix
Cornsilk	#FFF8DC	⬜	Shades	Mix
Crimson	#DC143C	🟥	Shades	Mix
Cyan	#00FFFF	🟦	Shades	Mix
DarkBlue	#00008B	🟦	Shades	Mix
DarkCyan	#008B8B	🟦	Shades	Mix
DarkGoldenRod	#B8860B	🟨	Shades	Mix
DarkGray	#A9A9A9	⬜	Shades	Mix
DarkGreen	#006400	🟩	Shades	Mix

http://www.w3schools.com/colors/colors_picker.asp



HTML Color Picker

« Previous

Next Reference »

Get darker/lighter shades of any color.

Select color:

#FF0066

Selected color: #FF0066 Submit

Darker/lighter shades:

Hex values:

#000000
#1A000A
#330014
#4C001F
#660029
#800033
#99003D
#B20047
#CC0052
#E6005C
#FF0066
#FF1975
#FF3385
#FF4D94
#FF66A3
#FF80B2
#FF99C2
#FFB2D1
#FFCCE0
#FFE6F0
#FFFFFF

Multiple stylesheets

- An external stylesheet is included using the `<link>` element
- It is possible to use multiple `<link>` elements
- Alternatively multiple stylesheets can be imported with `@import` rules

```
<head>
 <title>Stylesheets</title>
 <style>
    @import url(“http://www.abc.com/deptstyles.css”)
    @import url(“mystyles.css”)
 </style>
</head>
```

And can be followed in the same <style> element by explicit style rules

Cascading stylesheets

- Multiple stylesheets can be included in a document
- Styles defined in the first stylesheet are overridden by corresponding styles defined in the second stylesheet
  - the stylesheets are said to cascade
- Example of external stylesheets:
  - **`mainstyles.css`** – the company's stylesheet
  - **`deptstyles.css`** – the department's stylesheet
  - **`mystyles.css`** – the user's stylesheet
- In this order, in the event of conflicting declarations, the user's style definitions will override the department styles, which will override the company styles
- The document may then have a mix of all three

More on cascading priority

- Inline styles (using the **`style="…"`** attribute) have highest priority
- Internal style sheets (in the **`<style>`** element) take priority over external style sheets
- If multiple rules in an individual style sheet set the same property, the last one takes precedence

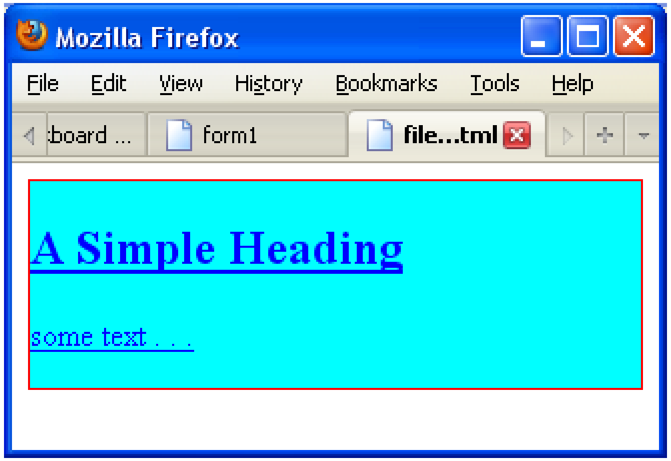
divisions and spans

- Rather than applying styles to an element itself, we wrap one or more elements in
  - a div (division) element (usually for block elements), or
  - a span element (usually for inline elements – part of a piece of text)
- Any required formatting can then be applied to the **`<div>`** or **`<span>`** element.
  - each can have **`class`** and **`id`** attributes
- The HTML5 elements of **`header, nav, article, section, aside`**, **`footer`** can be regarded as divs with particular semantics

Divisions

- Styles can be applied to blocks of HTML code using divisions

```
<head>
 <style>
  .myclass {
   color: blue;
   background: cyan;
   text-decoration: underline;
   border: thin groove red;
  }
</style>
</head>
<body>
 <div class="myclass">
  <h2>A Simple Heading</h2>
  <p>some text . . . </p>
 </div>
</body>
```



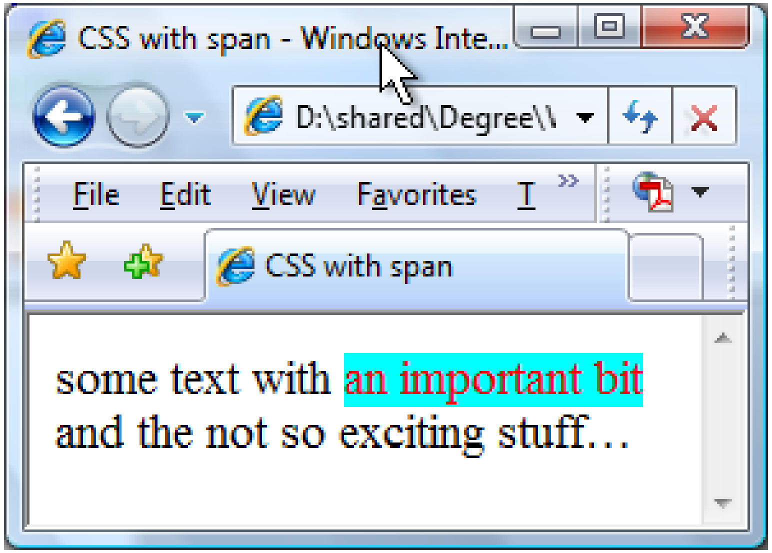
A Simple Heading

some text . . .

Spans

spans usually refer to a selected piece of text (inline).

```
<head>
 <style>
  .myclass {
   color: red;
   background: cyan;
   text-decoration: none;
  }
 </style>
</head>
<body>
  <p>some text with <span
class=“myclass”>an important
bit</span> and the not so
exciting stuff… </p>
</body>
```



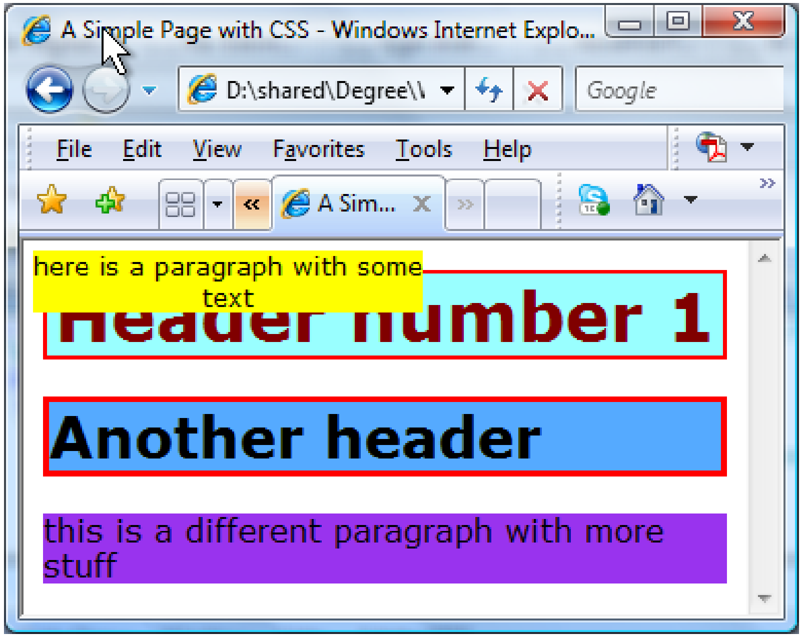
some text with an important bit and the not so exciting stuff…

Positioning - absolute

Absolute:

The values **`left`** and **`top`** are distances horizontally and vertically from ***the top left corner of the containing element.***

```
<style>
#mypara {position: absolute;
             top: 5px;
             left: 5px;}
</style>
.........
</head>
<body>
  <h1 class="headers">
  Header number 1</h1>
  <p id="mypara"> here is a
  paragraph with
  some text </p>
  ........
```



here is a paragraph with some text

Header number 1

Another header

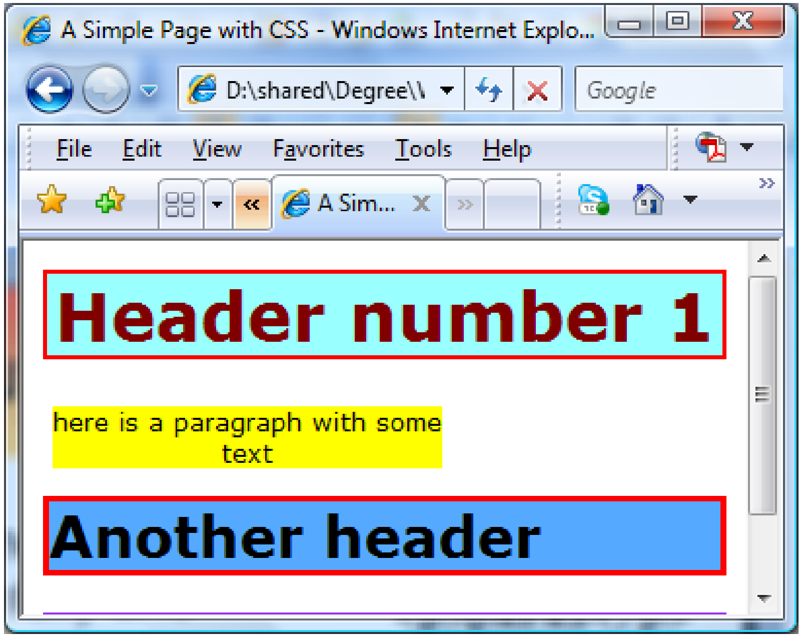
this is a different paragraph with more stuff

Positioning – relative

Relative

Values are relative to the top left of where the element would have been placed otherwise (with normal flow)

```
<style>
#mypara1 {position: relative;
            top: 5px;
            left: 5px;}
</style>
.........
</head>
<body>
  <h1 class="headers">
  Header number 1</h1>
  <p id="mypara1"> here is a
  paragraph with
  some text </p>
  ........
```



Header number 1

here is a paragraph with some text

Another header

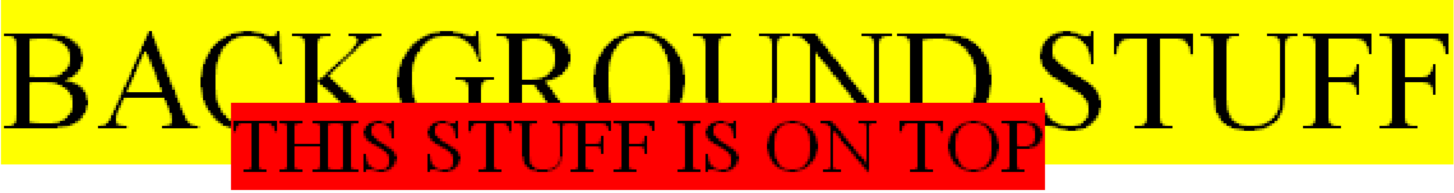
Note on absolute positioning

- The values of **`top`** and **`left`** are relative to the containing element *that has a position property set* (other than to static)
- By default the containing element is the `<body>` element
- To position relative to an element that is contained in the body or at some lower hierarchical level, *the parent element must be given a* **`position`** *property,* (irrespective of whether **`top`** or **`left`** are set) :

```
  {position: relative; }
```

Layers

- The browser maintains a stack of layers, each containing text, images etc. The browser displays layers on top of each other (in order).



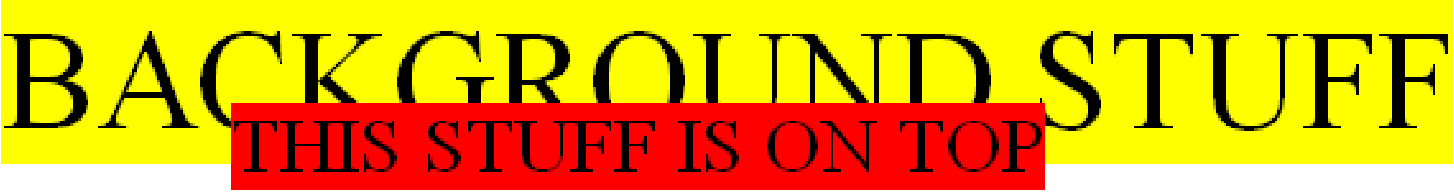
BACKGROUND STUFF

THIS STUFF IS ON TOP

  - **`z-index`**: specifies the order of an element in the stack
  - Higher numbered elements display on top of lower numbered elements
  - The z-index numbers can be negative

- The effect above is obtained by combining z-index with positioning properties (see next slide)

Layers



BACKGROUND STUFF

THIS STUFF IS ON TOP

```
<body>
<div style="z-index:2; left:100px; top:50px;
 position:absolute; background-color:red; font-size:30pt">
<p>THIS STUFF IS ON TOP</p>
</div>

<div style="z-index:1; left:10px; top:10px;
     position:absolute;  background-color:yellow;
     font-size:56pt">
<p>BACKGROUND STUFF</p>
</div>
</body>
```

`nth-child and nth-of-type` pseudo-classes

`:nth-child(n)` Selects the nth child of specified type to occur within any parent element

`p:nth-child(2)` selects a **`<p>`** element within any immediate parent, if it is the second child element (of any type) of that parent

`p:nth-of-type(2)` selects the second **`<p>`** element within any immediate parent

`nth-child and nth-of-type` pseudo classes *(cont)*

For example:

`tr:nth-of-type(2n)` selects every other **`<tr>`** child of a parent element, starting at the second

[**`tr:nth-of-type(even)`** does same thing**]**

`tr:nth-of-type(2n+1)` selects every other **`<tr>`** element of a parent element starting at the first

[**`tr:nth-of-type(odd)`** does same thing**]**

`:nth-child` works in similar way

`:before` and `:after` pseudo elements

`p:before` inserts specified content at the beginning of specified element (**`<p>`**), e.g

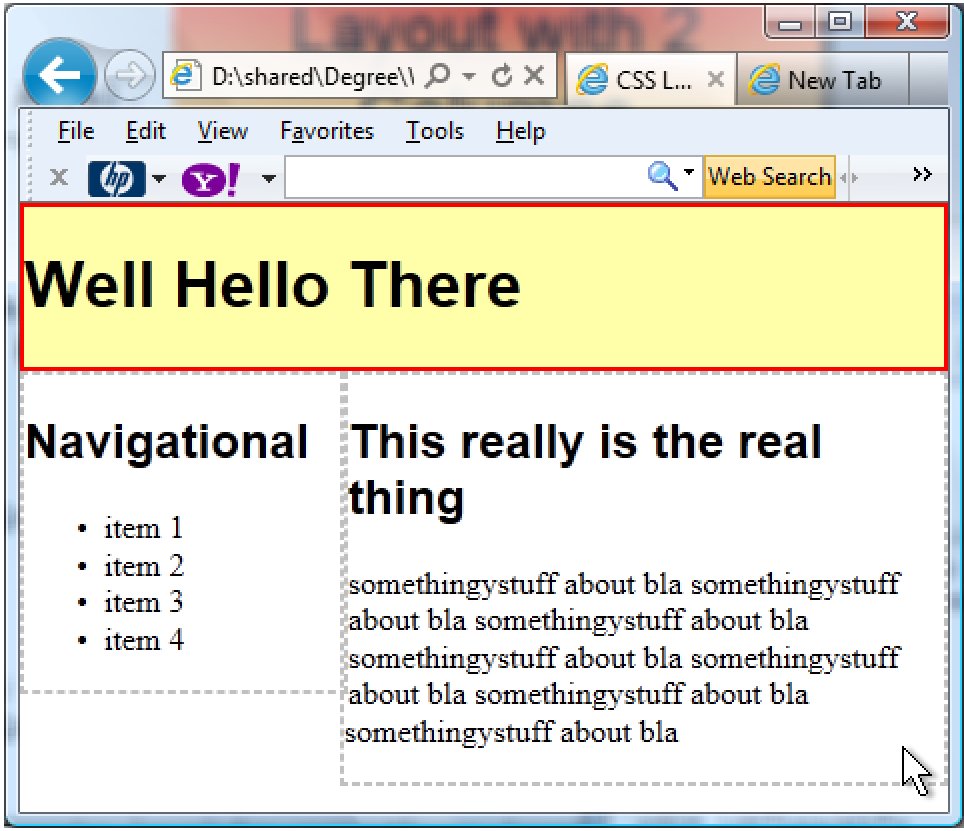
```
p:before {content: "My thoughts are as follows "; color:red;}
```

Inserts the text “My thoughts are as follows:” at the beginning of all **`<p>`** elements

`:after` inserts content at the end of specified elements

Layout with 2 Columns

```
<header>
<h1>Well Hello There</h1>
</header>
<div id="wrapper">
<nav id="leftNavigation">
  <h2>Navigational</h2>
  <ul>
    <li>item 1</li>
    <li>item 2</li>
    <li>item 3</li>
    <li>item 4</li>
  </ul>
</nav>
<div id="contentcol">
 <h2> This really is the real
     thing</h2>
 <p> somethingystuff about
    bla somethingystuff about
    bla……. .
 </p>
</div>
</div>
```



Well Hello There

Navigational

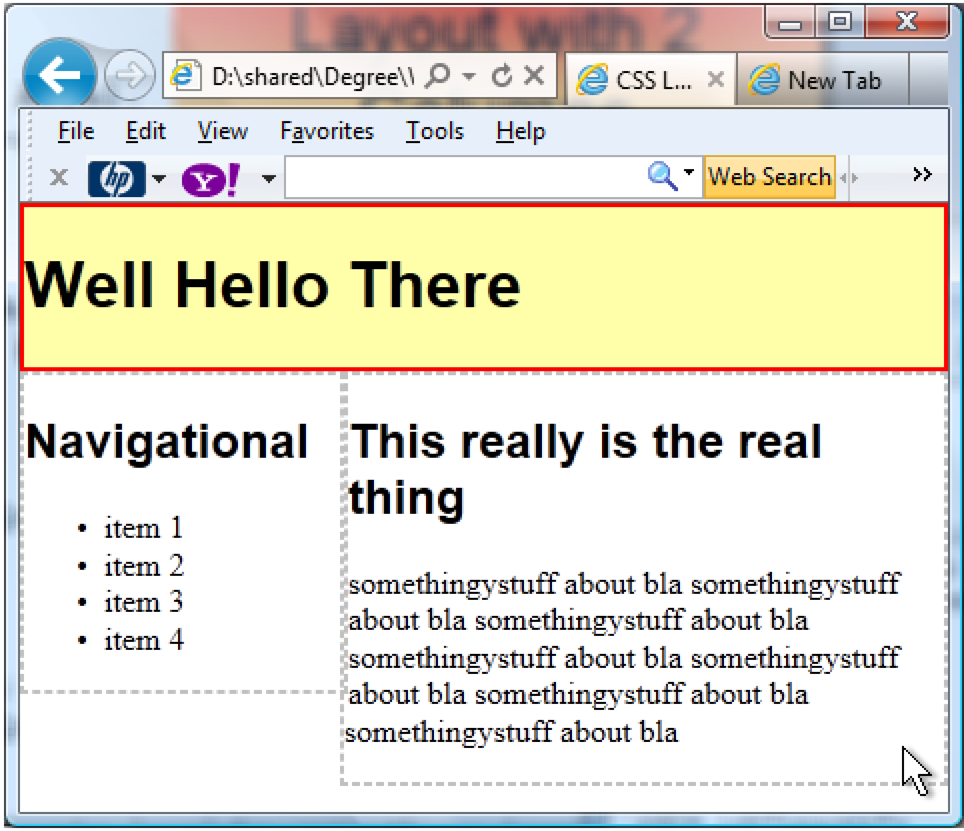
- item 1
- item 2
- item 3
- item 4

This really is the real thing

somethingystuff about bla somethingystuff about bla somethingystuff about bla somethingystuff about bla somethingystuff about bla somethingystuff about bla somethingystuff about bla

Layout with 2 Columns

```
body { margin: 0;
       padding: 0;}
h1, h2 {font-family: Arial,
                sans-serif;}
#wrapper{
   width: 100%;}
header {width 100%;
   border: red 2px solid;
  background-color: #ffffaa;}
#leftNavigation {
   width: 10em;
   float: left;
   border: dashed silver 2px;
   padding:0;}
#contentcol {
   padding: 0;
   margin-left: 10em;
   border: dashed silver 2px;
   min-width: 10em;}
```



Well Hello There

Navigational

- item 1
- item 2
- item 3
- item 4

This really is the real thing

somethingystuff about bla somethingystuff about bla somethingystuff about bla somethingystuff about bla somethingystuff about bla somethingystuff about bla somethingystuff about bla

Layout with 2 Columns

- Left Column (nav):
  fixed width **W** and floats left

- Content Column div:
  liquid, i.e. no width specified (but has min width)
  - has left margin >= **W**

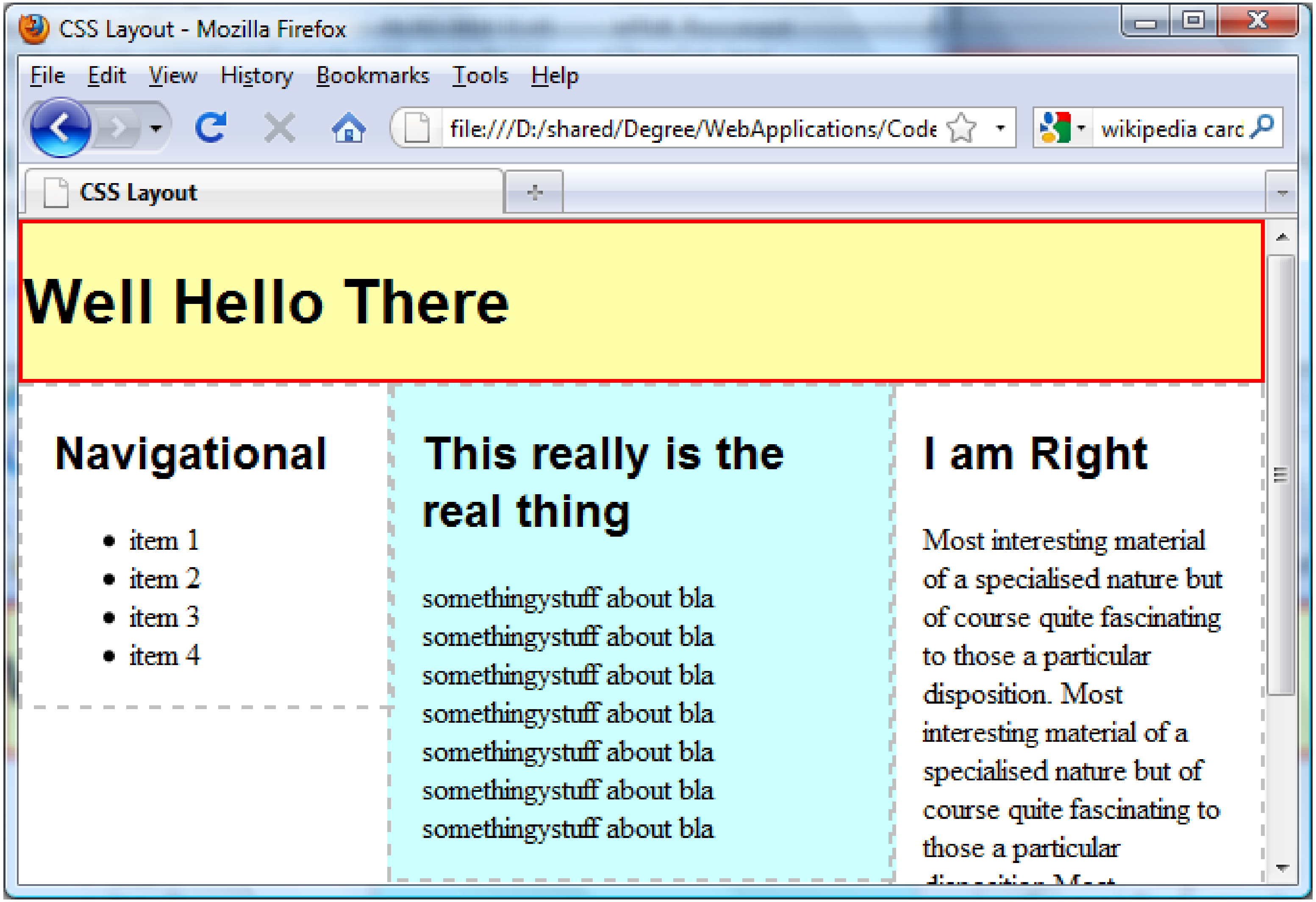
Both contained in Wrapper div

<body> set to have no margin or padding

Float property

- The **`float`** property causes an element to move to the left or right of its containing element.
- Text that follows the floating element will wrap around it
  - a block element that follows a floating element will not force a new line (unless the **`clear`** property is set)

3 columns



CSS Layout - Mozilla Firefox
File Edit View History Bookmarks Tools Help
file:///D:/shared/Degree/WebApplications/Code
wikipedia carc
CSS Layout

Well Hello There

Navigational

- item 1
- item 2
- item 3
- item 4

This really is the real thing

somethingystuff about bla somethingystuff about bla somethingystuff about bla somethingystuff about bla somethingystuff about bla somethingystuff about bla somethingystuff about bla

I am Right

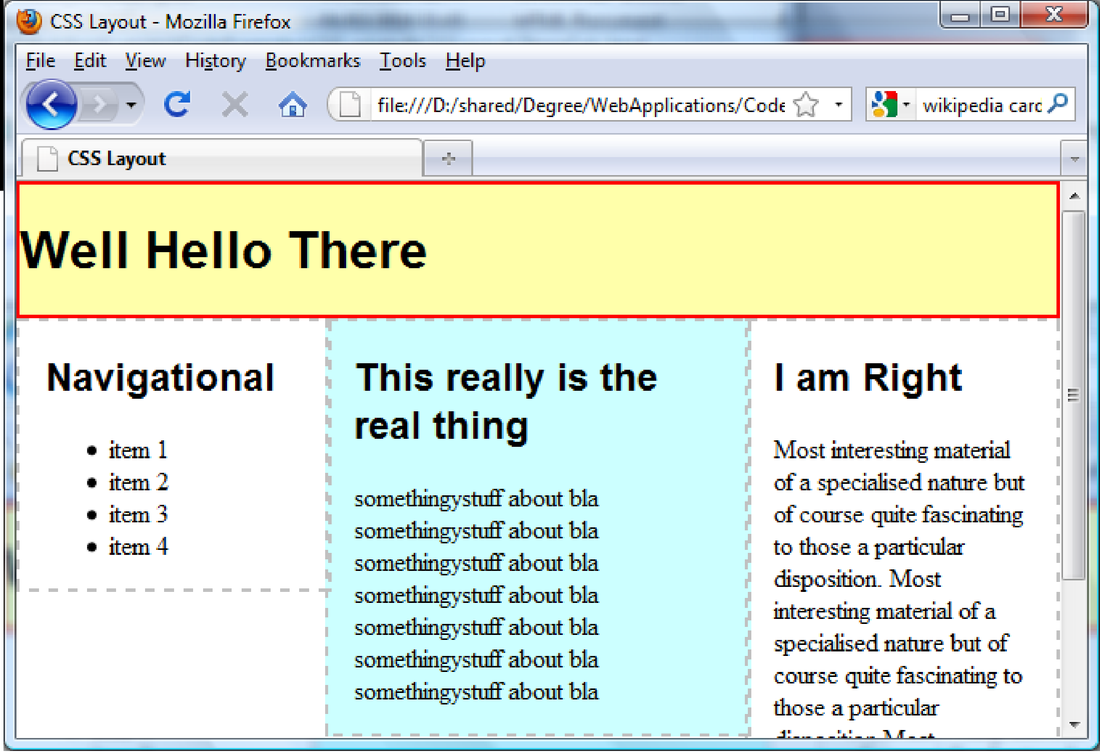
Most interesting material of a specialised nature but of course quite fascinating to those a particular disposition. Most interesting material of a specialised nature but of course quite fascinating to those a particular

3 columns

```
<header>
<h1>Well Hello There</h1>
</header>
<div id="wrapper">
<nav id="leftNavigation">
<h2>Navigational</h2>
<ul>
<li>item 1</li>
<li>item 2</li>
<li>item 3</li>
<li>item 4</li>
</ul>
</nav>
```

```
<div id="rightCol">
<h2> I am Right </h2>
Most interesting material of
……..
</div>
<div id="contentcol">
<h2> This really is the real
thing</h2>
<p> somethingystuff about
bla
somethingystuff about bla
…….
</p>
</div>
</div>
```

3 columns


CSS Layout - Mozilla Firefox
File
Edit
View
History
Bookmarks
Tools
Help
file:///D:/shared/Degree/WebApplications/Code
wikipedia carc
CSS Layout

Well Hello There

Navigational

- item 1
- item 2
- item 3
- item 4

This really is the real thing

somethingystuff about bla
somethingystuff about bla
somethingystuff about bla
somethingystuff about bla
somethingystuff about bla
somethingystuff about bla
somethingystuff about bla

I am Right

Most interesting material of a specialised nature but of course quite fascinating to those a particular disposition. Most interesting material of a specialised nature but of course quite fascinating to those a particular disposition. Most

```
body { margin: 0;
       padding: 0;}
h1, h2 {font-family:
       Arial, sans-serif;}
header {width 100%;
 border: red 2px solid;
 background-color: #ffffaa;}

#wrapper{ float: left;
   min-width: 30em;}
#leftNavigation {
   width: 10em;
   float: left;
   border: dashed silver 2px;
   padding:0 1em;}
#rightCol {
   width: 10em;
   float: right;
   border: dashed silver 2px;
   padding:0 1em;}
```

```
#contentcol {
   background-color: #ccffff;
   padding: 0 1em;
   margin: 0 12em 0 12em;
   border: dashed silver 2px;
   min-width: 5em;}
```

3 Columns

- Here left and right colums are fixed width

  left floats to left; right floats to right
- Content column liquid with min-width (could also be fixed)

  its left and right margins extend across full width of the left and right columns
- IMPORTANT : the floating div elements for the left and right columns must appear in the HTML doc **before** the content div – so that they can claim their space.