

# HTML

## Part 2

# More HTML tags

- We've covered quite a lot of different HTML tags since we've started
- There are a few more basic ones I'd like to cover
  - id's
  - anchor tags
  - div
  - span
  - em, i
  - strong, b
  - u

# ID attribute

- So we've seen you can add classes to any HTML tag. The “class” is called an attribute, as it tells the HTML tag to do something a little bit extra

```
<h1 class="yourclass">Heading 1</h1>
```

- We can also add another attribute called an ID

```
<h1 id="myid" class="yourclass">Heading 1</h1>
```

- An ID is unique, so it should only be used once on a page, which means if we have used 'myid' already, we shouldn't put a second, or third 'myid' on the page, as the HTML will get confused



# Anchor Links

- The reason we want it to be unique is if we want to add a link to a section on the page.
- This is called an Anchor Link
- If there are more than two ids the same, the link won't know which to go for
- For this example, if I want to link to this heading:

```
<h1 id="myid" class="yourclass">Heading 1</h1>
```

- I need to create a new link

```
<a href="#myid" title="Heading 1">Heading 1</a>
```

- The important part is the # (hash symbol), in CSS, this means an ID. As we've seen already, .myclass is a class, so #myid is an ID.

# Span Tag

- Span tags are great, they can be used anywhere

`<span id="myid" class="myclass">some text</span>`

- They can have id's and classes added to them

# Div Tag

- Div tags are almost like span tags
- They are flexible
- They can be used anywhere, for anything
- They can have ID's and classes

`<div id="myid" class="myclass">some text</div>`

- The main difference between span's and div's is how the browser displays them
- A span is very handy for putting styling on some words in the middle of a sentence.
-



# Div Tag (Cont...)

- This is called “inline”

```
class="shadow imgright"  
src="images/dog2.gif"/>  
Harry McKillen Rhoncus  
pulvinar nec magna  
facilisis tempor <span  
class="redtext">tincidunt  
parturient integer</span>  
in hac vel nec facilisis?
```

## Heading 1

Harry McKillen Rhoncus pulvinar  
nec magna facilisis tempor  
tincidunt parturient integer in hac  
vel nec facilisis? Vel, cum,  
porttitor augue phasellus integer

# Div Tag (Cont...)

- But what happens if I switch the span in this example to a div

```
pulvinar nec magna  
facilisis tempor <div  
class="redtext">tincidunt  
parturient integer</div>  
in hac vel nec facilisis?
```

nec magna facilisis tempor

tincidunt parturient integer

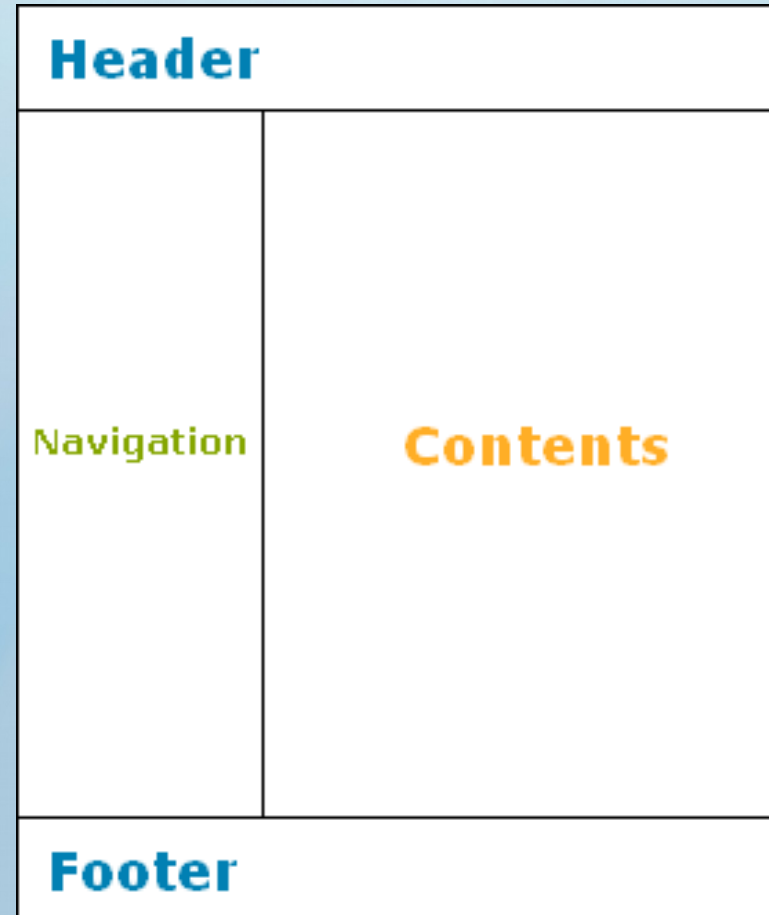
in hac vel nec facilisis? Vel, cum,  
porttitor augue phasellus integer

- It breaks the inline and puts a space above and below the text.
- This is called block (think of lego)
- We'll learn more about how divs can be used in the next slide



# Page Layouts

- Div's are really good for creating page layouts and designs
- A typical website usually looks like this



# Page Layouts (Cont...)

- To do this, we're going to have to put a div around our entire page
- So at the very top of your HTML file put

```
<div class="page">
```

...

(Everything on your page)

...

```
</div>
```

- Remember to close the div tag at the bottom

# Page Layouts (Cont...)

- When you refresh your page, you should see no difference
- Next we're going to create the CSS

```
.page{  
  width:900px;  
  margin: 0 auto;  
}
```

- This creates a 900 pixel wide column, which is centered in the middle of the browser



# Page Layouts (Cont...)

- Now we need to create the navigation column in your HTML

```
<div class="navigation">  
    (put some of your HTML here)  
</div>
```

- This creates a 200 pixel wide column and floats it to the left.

# Page Layouts (Cont...)

- Now we need to create the CSS for the navigation

```
.navigation{  
  width:200px;  
  float:left;  
}
```

- This creates a 200 pixel wide column and float it to the left.

# Page Layouts (Cont...)

- Next we'll create a div block for where all our content will go

```
<div class="content">  
  (put some of your HTML here)  
</div>
```



# Page Layouts (Cont...)

- And now we'll create the CSS for the where all the content will go

```
.content{  
  width:700px;  
  float:left;  
}
```

- This creates the 700 pixel wide column for the content and floats it to the left
- $200 + 700 = 900$ , so you can see we've create the columns just the right size to fit on our “page” div

# CoderDojo Notes

- I'll try to keep these updated with the latest notes, all the current slides are up there now:
- <http://www.harrymckillen.com/coderdojo/>

# Online Resources

- If you want to learn HTML & CSS on your own you can try the following good websites:
- <http://learn.shayhowe.com/html-css/>
- <http://www.codecademy.com/en/tracks/web>
- <https://www.khanacademy.org/computing/computer-programming/html-css>