



CSS DIVs, display, and other tricks



Review - IDs and Classes



IDs

The ID attribute is used to create a unique id for an element

As good practice an ID should be applied to only one element on a page

It is written as follow:

```
<p id="paragraphOne">This paragraph has a unique ID attribute</p>
```



Using an ID

In CSS to define an ID you use #

```
#paragraphOne {  
  
    background-color: black;  
  
    color: white;  
  
    text-align: left;  
  
}
```



Classes

The class attribute is used to create a shared “ID” across multiple elements

As good practice a class should be applied to more than one element on a page

It is written as follow:

```
<p class="paragraphClass">This paragraph has a unique class attribute</p>
```



Using a class

In CSS to define a class you use .

```
.paragraphClass {  
    background-color: blue;  
    color: black;  
    text-align: right;  
}
```

What's the difference?!?!?

- IDs are typically used when you want to write code for a specific element (IDs will become VERY important when we get into Javascript)
- Classes are used for when you want to apply code to multiple elements





A New Element - DIV

DIVs are empty containers

A DIV does not add or remove anything from a page

DIVs are a great to help style a page

DIVs will become more important in later lessons



Sizing an element such as a DIV or IMG

```
height:300px;
```

```
width:400px;
```



Sizing issues:

Add 50px padding to one of your divs.

```
padding: 50px;
```

Now make the divs the same size without changing the padding.

```
height: ???px;
```

```
width: ???px;
```



Why is this happening?

By default, the width and height of an element is calculated like this:

$\text{width} + \text{padding} + \text{border} = \text{actual width of an element}$

$\text{height} + \text{padding} + \text{border} = \text{actual height of an element}$

This means: When you set the width/height of an element, the element often appears bigger than you have set (because the element's border and padding are added to the element's specified width/height).

Source: https://www.w3schools.com/css/css3_box-sizing.asp



How do we solve this?

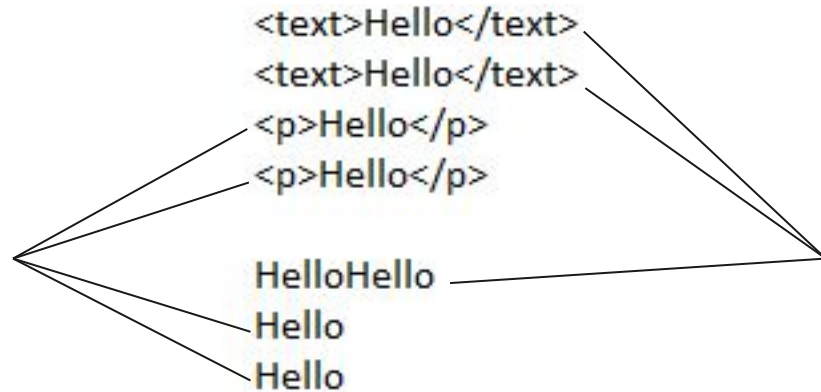
```
* {box-sizing: border-box;}
```

If you set `box-sizing: border-box;` on an element, padding and border are included in the width and height

Positioning on webpages

When you add elements to your page you are creating a series of **blocks** that can be manipulated around your page. Elements behave different based on how they are defined. An example of this is text vs paragraph.

Paragraph does will take up the full span of a page



Text does not take up the full span of a page



Using the CSS display property

`{display: none;}` - removes the element

`{display: inline;}` - displays similar to a span - inline with other elements

`{display: block;}` - displays similar to a `<p>`, starting a new line and claiming the entire block

`{display: flex;}` - block-level flex container (see next slide)



More about flex:

```
div {  
  
    display: flex;  
  
    flex-direction: row;  
  
}
```

Other Options: row | row-reverse | column | column-reverse | inherit



Another Cool Trick - overflow

The `overflow` property specifies whether to clip the content or to add scrollbars when the content of an element is too big to fit in the specified area.

The `overflow` property has the following values:

- `visible` - Default. The overflow is not clipped. The content renders outside the element's box
- `hidden` - The overflow is clipped, and the rest of the content will be invisible
- `scroll` - The overflow is clipped, and a scrollbar is added to see the rest of the content
- `auto` - Similar to `scroll`, but it adds scrollbars only when necessary

Source: https://www.w3schools.com/css/css_overflow.asp



Example of overflow: visible

By default, the overflow is `visible`, meaning that it is not clipped and it renders outside the element's box:

You can use the overflow property when you want to have better control of the layout. The overflow property specifies what happens if content overflows an element's box.



Let's finish our menu page

Launch Page / Menu Design

Mark MacCabe		
<div>Link</div> <div>Link</div> <div>Link</div> <div>Link</div>	Main Content Goes Here	Random Cool Info
Footer		



Homework

- Finish your menu page
- Read about the following topics:
 - CSS Sizing: https://www.w3schools.com/css/css_dimension.asp
 - Float: https://www.w3schools.com/cssref/pr_class_float.php
 - Position: https://www.w3schools.com/css/css_positioning.asp