



# BEGINNING HTML AND CSS

## CLASS 4

# WELCOME!

Girl Develop It is here to provide affordable and accessible programs to learn software through mentorship and hands-on instruction.

## Some "rules"

- We are here for you!
- Every question is important
- Help each other
- Have fun

# REVIEW

## Inline vs Block level elements

## Div tag to group content

```
<div id="header">  
  <h1>Main Heading</h1>  
</div>
```

# CSS POSITION - FLOAT

## BLOCK:



## INLINE:



- "Floating" an element takes it in the normal flow, as far to the left or right of it's containing element as possible.
- Any other elements, such as paragraphs or lists, will wrap around the floated element.
- Always specify a width when floating an element, otherwise the element is likely to take up the whole page and not appear floated.
- You can specify a) whether an element is floated or not, and b) which side it floats on.

# FLOAT: EXAMPLE

Below a `<blockquote>` is floated to the left, allowing text to wrap around it on the right

That's when they discovered Are You a Human. After switching to PlayThru, HiredMyWay 40% decrease in the amount of time it takes an average user to sign up for the site, from 4 minutes, 24 seconds to 2 minutes, 39 seconds. That's almost **two minutes saved** per user.

“The technology behind Are You a Human seemed like a breath of fresh air.”

 —Wes Weiler, CMO

Even better, after implementing PlayThru, the percentage of HiredMyWay users who complete profile after signing up Wes Weiler

CMO

HiredMyWay That's no surprise—CAPTCHA's studies indicating that one in four people will

And PlayThru didn't just improve HiredMyWay's numbers—it also improved their overall customer experience. “Coming onto my site, and that's a bad experience,” says Wes. “Everything's an emotional experience on our site.”

# FLOAT

```
.float{  
  float:left;  
  width:200px;  
  background:yellow;  
}
```

Hi, I'm a yellow box  
with black text.  
I like to hang out on  
the left side.

Not floating element

Not floating element

Not floating element with

wrapping Lorem ipsum dolor sit amet,  
consectetur adipisicing elit, sed do  
eiusmod tempor incididunt ut labore et  
dolore magna aliqua.

# USING FLOATS TO PLACE ELEMENTS SIDE BY SIDE

If you want two block level elements to be side by side, you need to float both elements. One left, and one right.



**WIDTH: 300PX;  
FLOAT: LEFT;**

**WIDTH: 400PX;  
FLOAT: RIGHT;**



# CLEAR

- Clearing tells the element on which side (right, left, both) other elements **cannot** appear.
- If you had an image floated left, and you did not want the paragraph to appear next to it, you would add **clear: left;** to the paragraph.
- Clearing both sides makes sure floats don't flow past the clear element.

```
clear: right;  
clear: left;  
clear: both;
```

# CLEAR

```
.float{  
  float:left;  
  width:50px;  
  background:yellow;  
}  
.clear-left{  
  clear:left  
}
```

hi Not floating element

hi Not floating element

hi

Non-floating element with a class of **.clear-left**

# LET'S DEVELOP IT

Let's float our side bar and content areas.

# STATIC POSITIONING

HTML elements are positioned static by default.

Static elements are positioned in the normal flow of the page

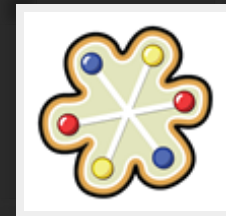
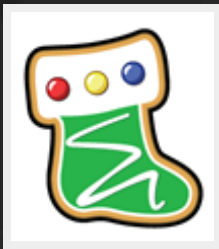
Static elements ignore top, bottom, right or left property specifications.

# STATIC POSITIONING

In normal flow, inline boxes flow from left to right, wrapping to next line when needed.

```
  
  
  
...  
  

```



# STATIC POSITIONING

In normal flow, block boxes flow from top to bottom, making a new line after every box.

```
<p>Greetings</p>  
<p>Hello</p>  
<p>Hi there!</p>
```

Greetings

Hello

Hi there!

# RELATIVE POSITIONING

- Takes the element out of the normal flow, allowing it to be moved to the top, left, right or bottom.
- Does not affect the elements surrounding it.
- Makes an element a "positioning context" in which to position other elements relative to it.
- Relative positioning and absolute positioning are used together.

# RELATIVE POSITIONING

The "relative" value will still put the element in the normal flow, but then offset it according to top/left/right/bottom properties.

```
.relative{  
  position: relative;  
  left: 80px;  
  top: 20px;  
  height: 100px;  
  background-color: yellow;  
}
```

Hello, hi!



# ABSOLUTE POSITIONING

- Positions element outside of the normal flow.
- An absolutely positioned element is offset from its container block, positioned relative.
- Its container block is the first element that has a position other than static.
- If no such element is found, the container block is `<html>`.
- Other elements act as if it's not there.
- Determined by its offset values in the properties top, bottom, right and left.

# ABSOLUTE POSITIONING

The "absolute" value will take the element out of the normal flow and position it in relation to the window (or the closest non-static element).

```
.top{  
  position: absolute;  
  top: -40px;  
  right: 10px;  
  background-color: yellow  
}
```

```
.bottom{  
  position: absolute;  
  bottom: -40px;  
  left: 60px;  
  background-color: green  
}
```

# EXAMPLE: ABSOLUTE POSITIONING

Here's an example of an image with a caption absolutely positioned over top of it.



The containing div has a position of relative, and the caption has a position of absolute.

# Z-INDEX

When you use positioning to move elements out of the normal flow of content, elements can overlap. You can change the order of overlapping with z-index.

The element with highest z-index goes on top.

```
.bottom{  
  position: absolute;  
  bottom: 10px;  
  left:60px;  
  background-color: yellow;  
}
```

```
.top{  
  position: absolute;  
  bottom: 15px;  
  left:60px;  
  background-color: green;  
  z-index: 2;  
}
```

# LET'S DEVELOP IT!

Let's add some positioning.

Let's create a div that contains an image and a caption, and position the caption absolutely overtop the image.

# QUESTIONS?



Top  
Down here

HTML/CSS ~ Girl Develop It ~

