



# GENERAL ASSEMBLY

---

# INTRO TO CODING

# IN THE NEXT 2 HOURS...

- ▶ What is the world wide web?
- ▶ Front-End vs. Back-End Development
- ▶ An intro to HTML
- ▶ An intro to CSS
- ▶ Q&A
- ▶ What's next?



# GOALS

- ▶ Understand what and how the web works
- ▶ Understand how web pages are created
- ▶ Gain an intro to HTML and CSS
- ▶ Know where you can learn more after this class



WHAT IS THE  
WEB?



# THE FIRST WEB PAGE EVER!

## World Wide Web

The WorldWideWeb (W3) is a wide-area [hypermedia](#) information retrieval initiative aiming to give universal access to a large universe of documents.

Everything there is online about W3 is linked directly or indirectly to this document, including an [executive summary](#) of the project, [Mailing lists](#) , [Policy](#)

### [What's out there?](#)

Pointers to the world's online information, [subjects](#) , [W3 servers](#), etc.

### [Help](#)

on the browser you are using

### [Software Products](#)

A list of W3 project components and their current state. (e.g. [Line Mode](#) ,[X11](#) [Viola](#) , [NeXTStep](#) , [Servers](#) , [Tools](#) , [Mail robot](#) , [Library](#) )

### [Technical](#)

Details of protocols, formats, program internals etc

### [Bibliography](#)

Paper documentation on W3 and references.

### [People](#)

A list of some people involved in the project.

### [History](#)

A summary of the history of the project.

### [How can I help ?](#)

If you would like to support the web..

### [Getting code](#)

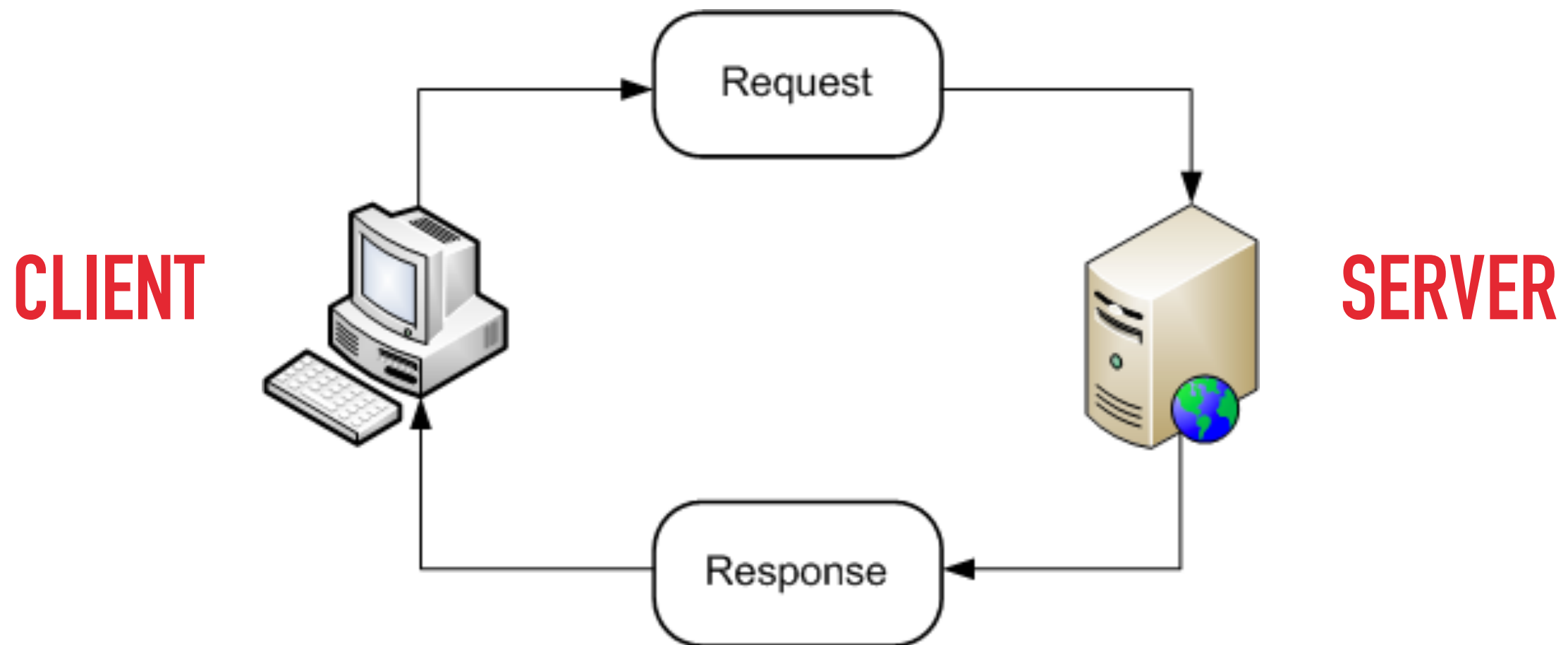
Getting the code by [anonymous FTP](#) , etc.



# HOW DOES THE WEB WORK?



# CLIENT-SERVER ARCHITECTURE



# WHAT MAKES UP A WEBPAGE?

URL: "HTTP://WWW.GOOGLE.COM"

HTML



index.html

CSS



styles.css



**SO WHAT'S WEB  
DEVELOPMENT?**



## WHAT'S WEB DEVELOPMENT?

- ▶ The creation and management of both websites and web based applications
- ▶ Made up of the Front End and Back End
- ▶ Most developers will try to convince you that it's magic, but it's mainly editing text files (coding/programming) and solving problems.



# FRONT END VS. BACK END

## FRONT END

- ▶ How things look to the user
- ▶ Involves: Images, content, structure
- ▶ HTML, CSS and JavaScript

## BACK END

- ▶ The logic and “how things work”
- ▶ Involves: “Business logic” and “data”
- ▶ Ruby, PHP, C++, Java etc



**WHAT TOOLS WILL  
WE USE?**



# TEXT EDITOR

- ▶ For this workshop, we'll use **Visual Studio Code**
- ▶ It's free
- ▶ Provides syntax highlighting, code hinting, auto completion and lots of great features!
- ▶ Word, Pages, Notepad type editors are NOT suitable for coding.



Visual Studio Code

# BROWSER



- ▶ We recommend using **Google Chrome**
- ▶ It's free and provides many developer-friendly tools!

# GOOGLE



- ▶ Googling is an important skill as an engineer!
- ▶ Google a question in the same wording you may ask someone in person
- ▶ You will find many engineers have been stuck with the same problem you are trying to solve

# GETTING STARTED

- ▶ In Chrome, go to: [www.light-html.com](http://www.light-html.com)
- ▶ Download the "HTML + CSS Boilerplate"
- ▶ Extract the folder that downloads:
  - ▶ On a Mac: double click on the zip file and it will extract
  - ▶ On Windows: open the zip folder and right click and hit "Extract All" in the menu bar
- ▶ Open Visual Studio Code
- ▶ Drag the entire folder you unzipped into the Visual Studio Code window





# HTML

# HYPertext MARKUP

# LANGUAGE



## THINK OF HTML AS...



- ▶ A markup language used to describe the **content** and **structure** of documents
- ▶ Describing what goes into a web document
- ▶ Not describing how a web document looks

# A TYPICAL HTML TAG

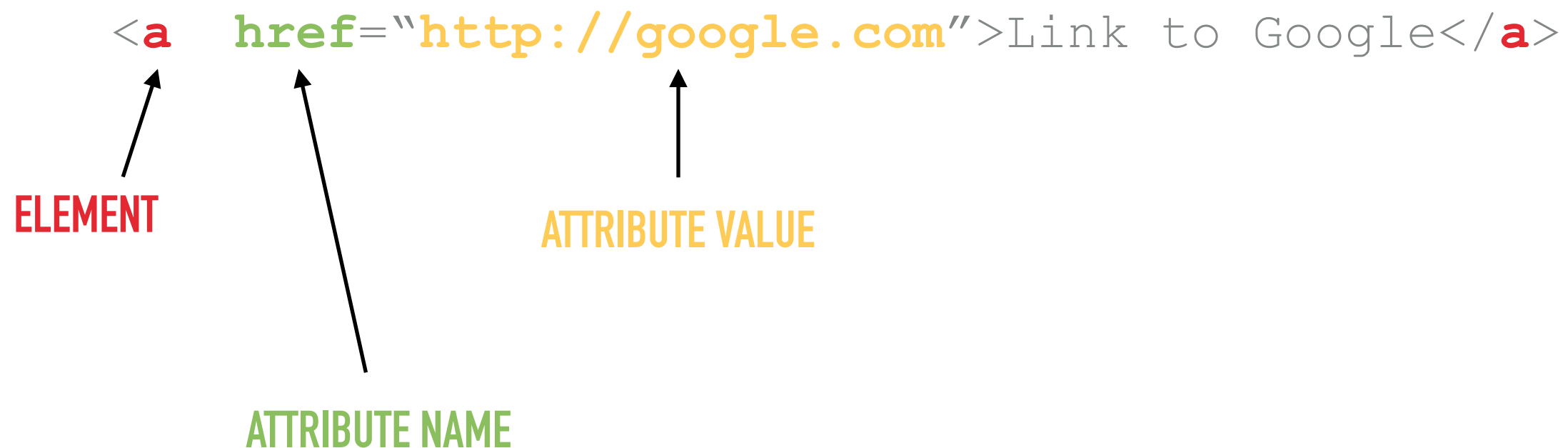
`<tag name attribute name="value">Contents</tag name>`

`<a href="http://google.com">Link to Google</a>`

ELEMENT

ATTRIBUTE NAME

ATTRIBUTE VALUE



The diagram illustrates the structure of an HTML tag using the example `<a href="http://google.com">Link to Google</a>`. Three labels with arrows point to specific parts of the tag: 'ELEMENT' points to the opening tag `<a`, 'ATTRIBUTE NAME' points to `href`, and 'ATTRIBUTE VALUE' points to the value `"http://google.com"` within the attribute.



# SELF CLOSING TAGS

```

```

# SOME ELEMENTS CAN BE REPEATED – OTHERS CAN'T

```
<p>My name's Tasja</p>
```

```
<p>I love coding!</p>
```

```
<body></body>
```

THERE CAN ONLY BE ONE BODY ELEMENT



# TAGS CAN BE PLACED INSIDE OTHER TAGS

```
<ol>
```

```
  <li>First Item</li>
```

```
  <li>Second Item</li>
```

```
</ol>
```



1. First Item

2. Second Item



# BASIC LAYOUT OF A HTML DOCUMENT



# YOUR TURN!

- ▶ Add some tags anywhere inside the **<body>** tags
- ▶ Add a main heading to your page using the **<h1>** tag
- ▶ Add an unordered list using the **<ul>** and **<li>** tags
- ▶ Add some text with a **<p>** tag
- ▶ Add a link to another cool website using an **<a>** tag
- ▶ Add an image to your page
  - ▶ Save an image and give it a simple file name (make sure there are no spaces) e.g. **mypet.jpg**
  - ▶ Make a folder called **images** by right clicking in your root (main) folder and selecting "New Folder"
  - ▶ Now embed the image in the page using the **<img>** tag
  - ▶ Drop the image you saved into the folder called **images**
  - ▶ Now embed the image in the page using the **<img>** tag
  - ▶ Go to [https://www.w3schools.com/tags/tags\\_img.asp](https://www.w3schools.com/tags/tags_img.asp) for help if you need
  - ▶ If you finish, go to: [https://www.w3schools.com/html/html\\_exercises.asp](https://www.w3schools.com/html/html_exercises.asp)

CSS

CASCADING STYLE SHEETS



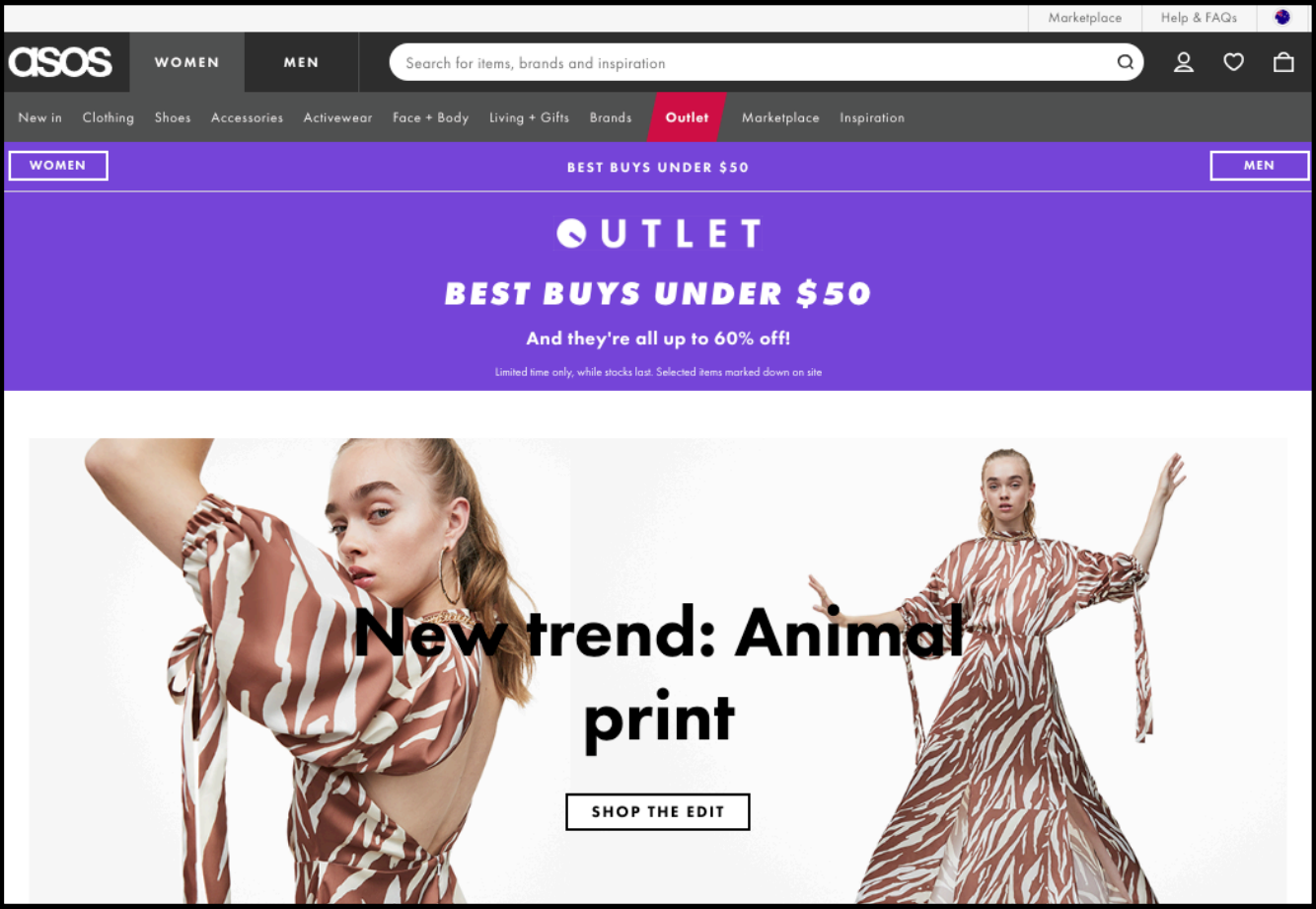


# THINK OF CSS AS...

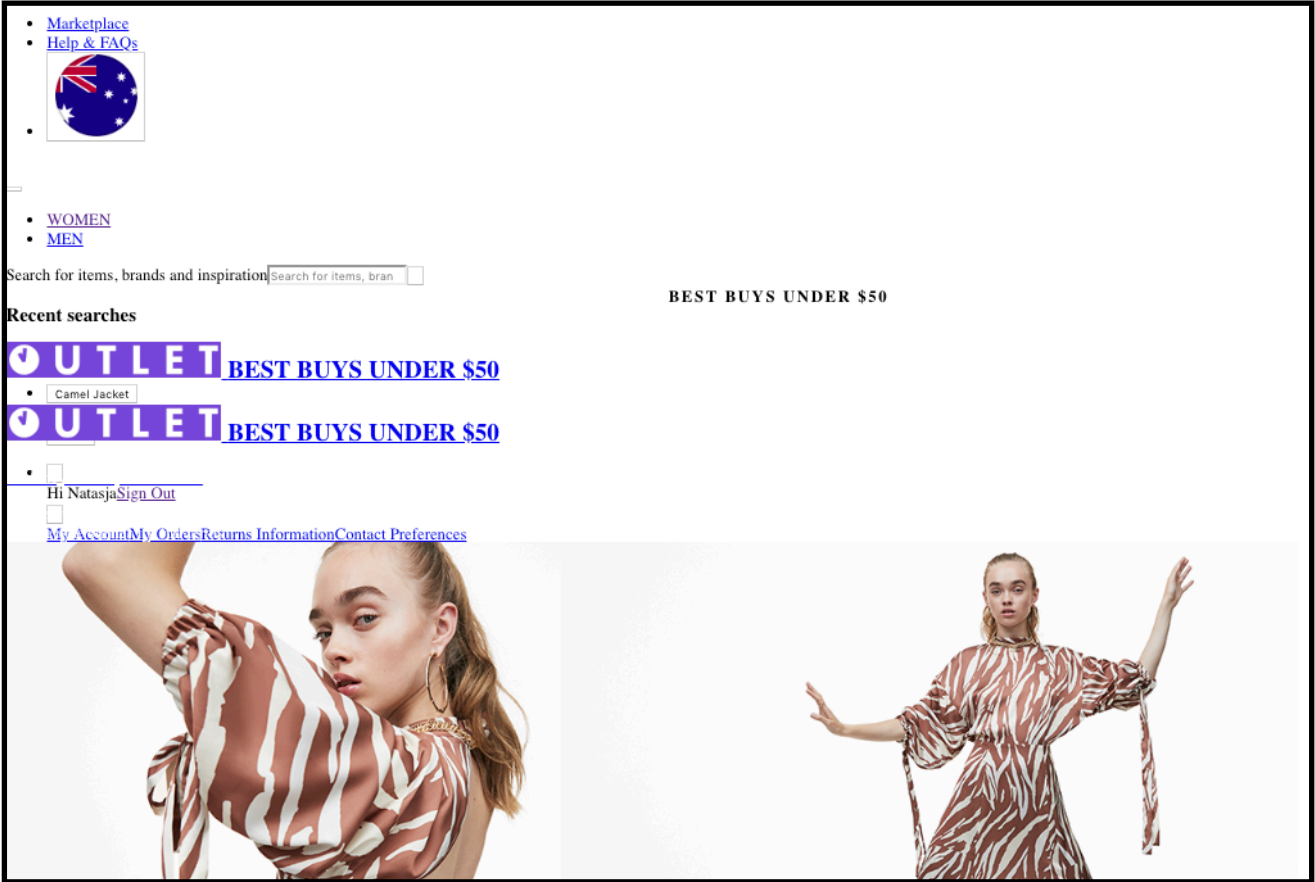


- ▶ A separate document telling the browser how your HTML looks
- ▶ A "Cascading" document that means styles can fall from one style sheet to another. So multiple style sheets can be used on one HTML document.

# GENERAL ASSEMBLY - INTRO TO CODING



WITH CSS



WITHOUT CSS



## CSS SYNTAX

```
selector { property: value; }
```

```
a { color: pink; font-size: 12px; }
```

Diagram illustrating the components of a CSS rule:

- SELECTOR** (red text) points to the `a` selector.
- PROPERTY** (blue text) points to the `font-size` property.
- VALUE** (orange text) points to the `12px` value.

# SELECTING BY A HTML ELEMENT

## IN HTML FILE:

```
<p>I'm learning how to code tonight!</p>
```

## IN CSS FILE:

```
p {  
    color: blue;  
}
```



# CSS CLASSES

## IN HTML FILE:

```
<p class="green">I'm a green paragraph!</p>
```

## IN CSS FILE:

```
.green {  
    color: green;  
}
```



## LINKING OUR CSS

```
<link rel="stylesheet" type="text/css" href="css/styles.css">
```



# YOUR TURN!

- ▶ Make sure your CSS is linked by adding the link to your CSS Stylesheet between the two **<head>** tags: **<link rel="stylesheet" href="css/styles.css">**
- ▶ Let's centre our page with CSS
  - ▶ Add a **<main>** tag around ALL the elements we've added inside our document **<body>** tags.
  - ▶ Go to your CSS file and style the main by adding a max-width. Wrap all your CSS rules inside curly braces like this: **main { max-width: 600px; }**
  - ▶ Let's center it by using **margin: auto;**
  - ▶ All CSS properties that apply to the same selector, you include in the same set of curly braces. So the finishes CSS above would look like: **main { max-width: 600px; margin: auto; }**
- ▶ Add some colors
  - ▶ Change the color of the **<h1>** tag
  - ▶ Try changing the background color of the body and see what happens
- ▶ Try adding a custom font by yourself by going to <https://fonts.google.com>
- ▶ If you finish, go to: <https://flukeout.github.io>

# WHAT'S NEXT?





# GOOD REFERENCES

- ▶ [W3Schools.com](https://www.w3schools.com)
- ▶ [developer.mozilla.org](https://developer.mozilla.org)
- ▶ [sidebar.io](https://sidebar.io)
- ▶ [devdocs.io](https://devdocs.io)
- ▶ [css-tricks.com](https://css-tricks.com)
- ▶ [tympanum.net/codrops](https://tympanum.net/codrops)



# STAYING CURRENT

- ▶ Smashing Magazine
- ▶ CSS Tricks
- ▶ Medium
- ▶ Web Designer Weekly
- ▶ [codepen.io](https://codepen.io)





---

**THANK YOU!**