GENERAL ASSEMBLY 6.2

Front-End Web Development

Albert Penticoss
Web Designer, University of Melbourne

Start your engines!

- Connect to wifi
- Open your text editor
- Open GitHub desktop and GitHub website
- Open schoology

Submit your assignment

- Please submit assignment 3
- REMINDER: No “auto” extensions - if you need an extension please ask me personally

Reminder – Graduation requirements

- To graduate from the course, students must:
  - 1. complete at least 80% of assigned homework
    - 8 assignments so you can only miss one!
  - 2. attend 80% of classes - so you can miss up to a maximum of four
  - 3. satisfactorily complete the final project (as determined by the course instructor and stated on the class syllabus).

What are we going to cover today?

- Lesson 8 JQuery Selection Exercise - 15min
- Lesson 8 Review - 10min
- GitHub pages setup - 5min
- CSS Position property - 10min
- JS Variables - 20min
- Guest talk - Laura Summers - 20min
- Break - 15min
- JS Variables exercise (Scorekeeper) - 30min
- JS Conditionals - including exercises (Compare that and Light Switch) - 55min

Exercise – jQuery Basics Review

KEY OBJECTIVES

Practice jQuery selection

DELIVERABLE

dom selectors

AGENDA

15 mins

1. Download lesson exercise 1 starter code from schoology
2. Follow instructions in the exercise.js file

RESOURCES

Starter code
Code editor

LESSON 8 REVIEW

FEWD – LESSON 8

INTRODUCTION TO JQUERY

What is jQuery?

- jQuery is JavaScript
- jQuery is a cross-browser JavaScript library designed to simplify the client-side scripting of HTML.
- jQuery is a JavaScript function some smart people have written and support
- jQuery is the most widely used JavaScript library - its the VHS in VHS vs Betamax
- There are other library/frameworks such as https://angularjs.org/ and https://nodejs.org/en/ which are used more by app developers
- jQuery can be used for apps too!

What is so good about jQuery?

- jQuery makes DOM manipulation much simpler than writing your own code
- jQuery is “Cross browser” - it works the same in all* browsers (yes JavaScript also has same cross browser issue like HTML and CSS)
- jQuery allows you to use CSS3 selectors
- jQuery has some helpful utilities that aren’t in pure JavaScript e.g. string trimming
- jQuery has been adopted widely so its well backed and supported and there’s a wealth of information on it and plugins to use.

What can we do with jQuery?

- Document traversal
- CSS Manipulation
- Event Handling
- Animation
- and more!

Note: jQuery is just JavaScript so it can't do anything JavaScript can't do but it makes it a lot easier to do it!

How do we use jQuery?

- Since jQuery is just JavaScript we include it the same way we include any other JavaScript script, with the <script> HTML tag
- We can download the jQuery .js file from http://jQuery.com and save it in our website (usually in a js folder) and include it in our HTML like so:

```
<script src="js/jquery-1.12.0.min.js"></script>
```

How do we use jQuery?

- Or we can just included it without downloading it by making use of a Content Delivery Network (CDN).
- Using jQuery's CDN:

```
<script src="//code.jquery.com/jquery-1.12.0.min.js"></script>
```

- Using googles CDN:

```
<script src="//ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>
```

Note: above examples are using protocol relative URL's, these won't work if you are not viewing your site using a local development web server (like Brackets live preview) or actual web server. I.e. if you double click your local html file to view it, these won't work, you'll need to add https: to start of URL.

How do we use jQuery?

- HTML5 boilerplate includes jQuery by default using jQuery's CDN and providing a local version as backup for developing when not connected to the internet:

```
<script src="https://code.jquery.com/jquery-1.12.0.min.js"></script>

<script>window.jQuery || document.write('<script src="js/vendor/jquery-1.12.0.min.js"><\/
script>')</script>

<script src="js/plugins.js"></script>

<script src="js/main.js"></script>
```

- HTML5 boilerplate also provides two empty js files, plugins.js and main.js for us to paste plugin code into and write our jQuery/JavaScript, respectively

How do we use jQuery?

- HTML5 boilerplate includes jQuery at the bottom of our HTML file, just before the closing </body> tag
  - This is best practice for including JavaScripts for performance reasons
  - There are exceptions, for example modernizer.js, which needs to included in the <head>
- You may have also noticed the ‘min’ in the file name jquery-1.12.0.min.js
  - This indicates that we are using the minified version of the juery.js file, which is also good for performance.
  - The 1.12.0. bit is the version we are using.

Exercise - Colour switcher

KEY OBJECTIVES

Let's find out if jQuery is easier than pure JavaScript

AGENDA

10 mins

1. Download lesson exercise starter code from schoology
2. Code along with instructor

DELIVERABLE

jQuery version of colour switcher

RESOURCES

Starter code
Code editor

FEWD – LESSON 8

JQUERY BASICS

Syntax

- Syntax: is the spelling and grammar rules of a programming language.
- Like any language, there are formal rules around how to write it. This is the syntax.

JavaScript Syntax

- Semicolon ;
- Brackets { }
- Parentheses ( )
- Quotation Marks ‘ ‘ “ “

JavaScript Syntax

- Comments:

  //Single Line Comments

  /* Multi line comments */

jQuery Syntax

selector	action	parameters
jQuery(‘p’)	.css	(‘color’, ‘blue’);
$(‘p’)	.css	(‘color’, ‘blue’);

We are calling the jQuery function (we use the alias $ to avoid typing jQuery all the time), we tell jQuery function what we want to select using CSS selector syntax and we choose an action to apply to the selected element(s)

jQuery Syntax

We will certainly be discussing this in more detail, but in general jQuery let's us grab some element from the page ($('selector')), and do something with it ($('selector').click(doSomething);).

$(document).ready()

We can’t manipulate the DOM until it has loaded.

So jQuery provides us with a function to test for document readiness.

Code placed inside the `$( document ).ready()` will only run once the DOM is ready

$(document).ready()

```
$( document ).ready(function() {
    // your code goes in here!
});
```

```
$( window ).load(function() {
    // use this for code you want to run after entire page including images and iframes are ready
});
```

Exercise - Traffic light

25

KEY OBJECTIVES

Learn about the click( ) event

AGENDA

15 mins

1. Download lesson exercise starter code from schoology
2. Code along with instructor

DELIVERABLE

jQuery version of traffic light

RESOURCES

Starter code
Code editor

Syntax Drill – 20 mins

http://codepen.io/anon/pen/zrMVxP

Fork it, then write some jQuery

- add a click handler to the button that adds a class to the paragraph called ‘highlight’
- write some css for .highlight to add background colour, padding and border
- if you finish quick I’ll give you something harder :)

Lab – lets build a Lesson review app!

We are going to design it and then code it!

10 mins design

20 mins groups discuss HTML

40 mins build it together!

What fun!

FEWD – LESSON 9

VARIABLES AND CONDITIONALS

FEWD – LESSON 9

GITHUB PAGES SETUP

GitHub Pages

- GitHub pages are a way to host a website on GitHub
- We are going to use this for our Final Project
- We are also going to use this today for our exercises! HOORAY

http://ipadisthin.github.io/fewd/

FEWD – LESSON 9

CSS POSTION PROPERTY

Position property

- Information in the position slides is from: http://www.w3schools.com/css/css_positioning.asp

Position property

- The position property specifies the type of positioning method used for an element.
- There are four different position values:
  - static
  - relative
  - fixed
  - absolute
- Elements are then positioned using the top, bottom, left, and right properties. However, these properties will not work unless the position property is set first. They also work differently depending on the position value.

Position: static;

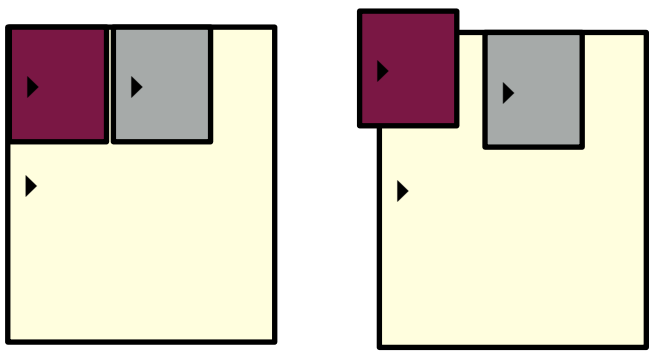
position: static;

- HTML elements are positioned static by default.
- Static positioned elements are NOT affected by the top, bottom, left, and right properties.
- An element with position: static; is not positioned in any special way; it is always positioned according to the normal flow of the page.

Position: relative;

position: relative;

- An element with position: relative; is positioned relative to its normal position.
- Setting the top, right, bottom, and left properties of a relatively-positioned element will cause it to be adjusted away from its normal position. Other content will not be adjusted to fit into any gap left by the element.



position: relative;

top: -10px;

right: -10px;

Position: fixed;

position: fixed;

- An element with position: fixed; is positioned relative to the viewport, which means it always stays in the same place even if the page is scrolled. The top, right, bottom, and left properties are used to position the element.
- A fixed element does not leave a gap in the page where it would normally have been located.

Position: fixed;

position: absolute;

- An element with position: absolute; is positioned relative to the nearest positioned ancestor (instead of positioned relative to the viewport, like fixed).
- However; if an absolute positioned element has no positioned ancestors, it uses the document body, and moves along with page scrolling.
- Note: A "positioned" element is one whose position is anything except static.

z-index

- If two elements overlap the one that comes second in the HTML will usually be on top.
- We can use z-index to change the stacking
- z-index is a numerical value and can be negative
- eg:

```
z-index: 100;
```

```
z-index: -1;
```

VARIABLES

Variables

What are variables?

- We can tell our program to remember values for us to use later on.
- The action of saving a value to memory is called assignment
- The entity we use to store the value is called a variable

Variables

- The action of getting the value from a variable is called accessing the variable
- We will use all the above techniques to store values into variables, and generate new values using existing variables

Variables Declaration

- Declaration: `var age;`
- Assignment: `age = 21;`
- Both at the same time: `var age = 21;`
- Note in JavaScript using var to declare the variable is optional, but if you don't declare the variable with var when inside a function, that variable will have global scope! Sometimes you don't want that. Declare your variables to avoid confusion.

Variable Re-Assignment

- var name = "Jo";
- name = Amir;
- Note: name is now Amir.

Variable Conventions

- Variables start with a lower case letter
- If they contain multiple words, subsequent words start with an upper case letter.

```
var numberOfStudents = 10;
```

- This is called camel case (it has humps!)

Variables & Data Types

- What can you store in a variables?

Variables & Data Types

- What can you store in a variables?
- The types of different values we support include:
  - String text
  - Number: int (whole numbers) or float numbers (numbers with decimal point)
  - Boolean: true or false

Strings

- Stores textual information
- String literal is surrounded by quotes

"How is the weather today?"

Arithmetic In JavaScript

- You can perform arithmetic on number data types

Operator	Meaning	Example
+	Addition	8 + 10
-	Subtraction	10 – 8
*	Multiplication	12 * 2
/	Division	10 / 5
%	Modulus	10 % 6

Escaping

- If you use a special character in your string you need to escape it with the \ character

"They \"purchased\" it"

'It\'s a beautiful day'

Numbers

- Represent numerical data
  - int: 42
  - float: 3.14159265
- Signed
  - int: +6
  - float: -8.2

Conversion: String To Number

- You can convert a string into a number:

```
parseInt(“4”);

parseFloat("3.14159");
```

Conversion: Number To String

- You can convert a number into a string:

```
var number = 4;
```

`number.toString();` or `number + "";`

number now equals “4”

GUEST SPEAKER

Exercise - Score Keeper

KEY OBJECTIVES

Use variables

AGENDA

25 mins

1. Download lesson exercise starter code from schoology
2. Code along with instructor

DELIVERABLE

Score card

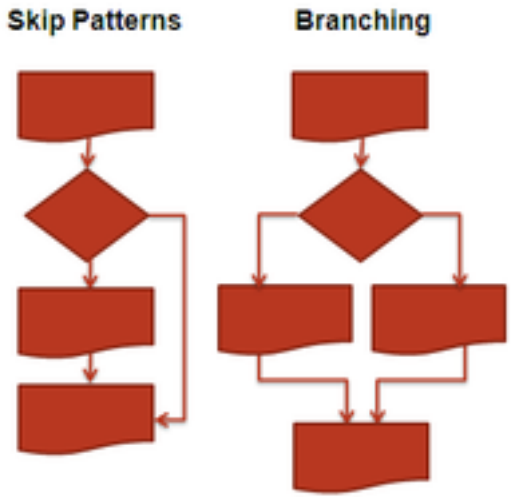
RESOURCES

Starter code
Code editor

FEWD – LESSON 9

CONDITIONALS

Conditionals



Making Decisions

- It's either TRUE or FALSE (like booleans)
- If you are greater than 18, you are an adult

```
if (age > 18){
    document.write("You are an adult");
}
```

Comparisons - Equality

- Are two things equal?
- == equal to value
- === equal to value and type

```
10 == 10 //true

10 == 5 //false

"hi" == "hi" //true


10 == “10” //true

10 === “10” //false
```

Logical Operators

x = 3

Logical Operators			
Operator	Description	Comparing	Returns
==	equal to	x == 8	FALSE
===	exactly equal to(value and type)	x=== “3”	FALSE
		x===3	TRUE
!=	is not equal	x!=8	TRUE
!==	is not equal(neither value nor type)	x!== “3”	TRUE
		x!==3	FALSE
>	greater than	x>8	FALSE
<	less than	x<8	TRUE
>=	greater than or equal to	x>=8	FALSE
<=	less than or equal to	x<=8	TRUE

Conditional Syntax

```
if(condition is true) {
    //Do something
}
```

Conditional Syntax

```
if(condition is true) {
    //Do something
}else{
    //Do something else
}
```

Conditional Syntax

```
var topic = "JS";
if (topic == "JS") {
    console.log("You're learning JavaScript");
} else if(topic == "JavaScript") {
    console.log("You're still learning JavaScript");
} else {
console.log("You're learning something else");
```

Multiple Conditions

```
if (name == "GA" && password == "YellowPencil"){
   //Allow access to internet
}
```

The Truth Table

```
if (name == "GA" && password
== "YellowPencil"){
   //Allow access to internet
}
```

AND - &&	TRUE	FALSE
TRUE	true	false
FALSE	false	false

The Truth Table

```
if (day == "Monday" || day ==
"Wednesday"){
   //We have class today
    }
```

OR - ||	TRUE	FALSE
TRUE	true	true
FALSE	true	false

Exercise - Compare That

KEY OBJECTIVES

Use comparison operators

DELIVERABLE

Value compare

AGENDA

20 mins

1. Download lesson exercise starter code from schoology
2. Code along with instructor

RESOURCES

Starter code
Code editor

Exercise - Blackout

KEY OBJECTIVES

Use variables and comparison operators

DELIVERABLE

Lights on/off function

AGENDA

15 mins

1. Download lesson exercise starter code from schoology
2. Code along with instructor

RESOURCES

Starter code
Code editor