WEEK 5 - JAVASCRIPT

GENERAL ASSEMBLY

FEWD

Joe Bliss

Would be hooked-up to an IV drip of coffee if possible.

AGENDA

Mid-Course Feedback

Questions on Startup Matchmaker

Javascript

-

MID-COURSE FEEDBACK

Please take 10 minutes to fill-out the mid-course feedback.

START-UP MATCHMAKER

Questions? Queries? Concerns?

EXERCISE - WHAT CAN JS DO?

Go out on the internet and find some cool effects on your favorite sites. We want to try to explore the different things we can add to our sites by utilizing Javascript.

(SOME) USES OF JAVASCRIPT

Adding / Removing Elements

Changing CSS “on-the-fly”

Animating content

Detecting user interactions

Form validation

Loading dynamic content

Etc.

PSEUDO CODE RECAP

Computers don’t think, they act as if they do, by sequentially executing simple instructions.

The only things a computer knows are the things we tell it.

Pseudocode is the process of writing a program without using the syntax of a programming language.

EXERCISE – PAYCHECK CALCULATOR

Write pseudo code for how to calculate a paycheck for an hourly employee (example is below). What values do you need to keep track of? What operations repeat?

Hourly Rate: $12.50, Time-and-a-half for > 35 hours.

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
1200 - 2000	-	1000 - 1600	1000 - 1200	1400 - 2000	-	0900 - 1400

READING JAVASCRIPT CODE RECAP

Color Switcher

http://codepen.io/josephjbliss/pen/lIuyr

JS BASICS – SYNTAX

Syntax: Spelling and grammar rules of a programming language.

Like with any language, there are formal rules around how to write it. This is the syntax.

Every line must end in a semicolon ;

We will learn more as we go.

SYNTAX

EXERCISE - TRAFFIC LIGHT

Take the following Codepen and see if you can understand what it is doing to manipulate it to match the example I will show you.

http://codepen.io/josephjbliss/pen/vkDmB

INCLUDING JS IN YOUR PROJECT

Any file with the “.js” extension. Like CSS, we include it one of two ways:

External (most common)

```
<script type=”text/javascript” src=”js/project.js”></script>
```

Internal

```
<script type=”text/javascript”>
    //Do stuff here
</script>
```

CODEALONG - COLORFUL PAGE

Look at the scripts in the “js” folder.

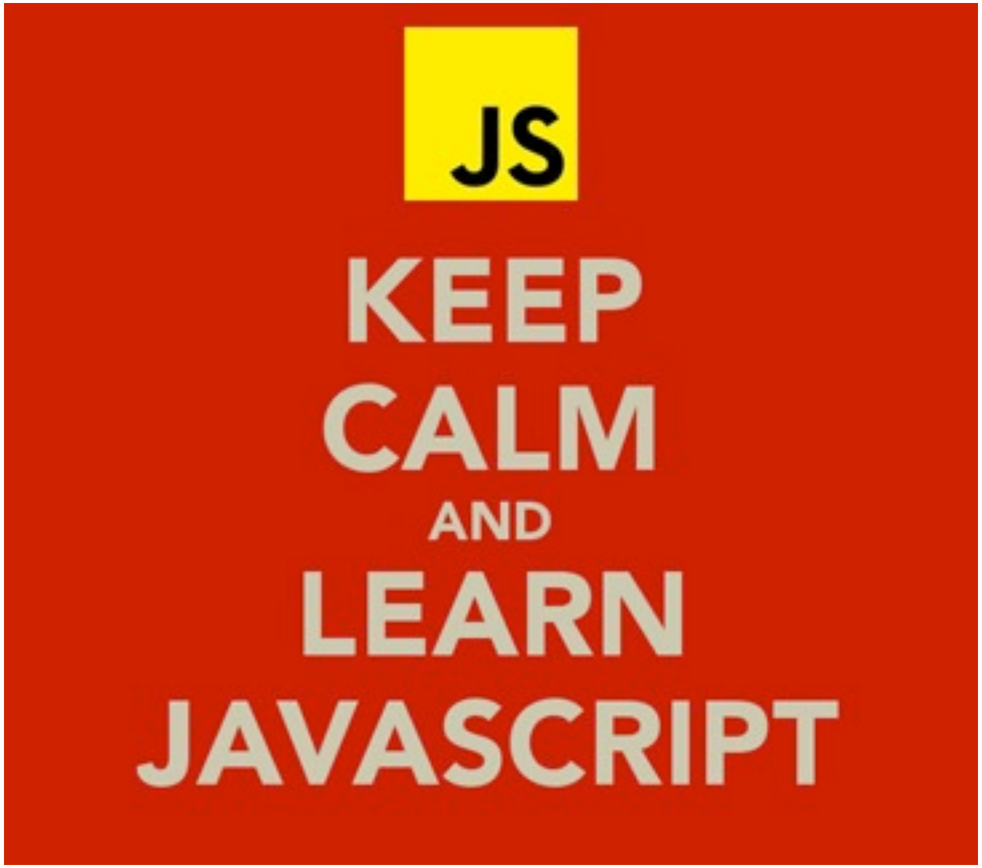
We are going to add them in the following order:

- jquery-1.9.1.js
- jquery.easing.min.js
- jquery.animate-colors-min.js
- project.js

CODEALONG - OUR FIRST JAVASCRIPT

We will write our first Javascript together.

alert("Message");
- Creates a pop-up in the browser that will display "Message"



TYPES OF DATA

Numbers

- Integers, i.e. 1, 2, 3, 4, 5
- Floats (numbers with decimal points), i.e. 3.14159, 2.718281828459045

Strings

- A sequence of characters enclosed in quotes, i.e. “I am a String”, “Hello!”, “Joe Bliss”

MORE ON NUMBERS

Represents numerical data

- Integers: i.e. 42
- Floating-point numbers: i.e. 3.14159265

Can be Signed or Unsigned (- or +)

- 6
- -8.2

We can perform arithmetic on number data types

MORE ON NUMBERS

Operator	Description	Example
+	Addition	1+1
-	Subtraction	3-2
*	Multiplication	5 * 3
/	Division	10 /2
%	Modulus	1%2

EXERCISE – OUR FIRST JAVASCRIPT

Add to our script the following alerts:

- The product of 5 and 23 (using *)
- The difference of 4 and 2 (using -)
- The quotient of 42 and 6 (using /)
- The sum of 7 and 8 (using +)
- Your first and last name (“Joe Bliss”)
- A warning message of your choosing (“These are not the droids you are looking for!”)

RETAINING INFORMATION

What does the following do?

```
alert(2+3);
```

What about this?

```
2+3;
```

VARIABLES

We can tell our program to remember values for us to use later on.

The entity we use to store the value is called a variable.

VAR OUT, MAN

Declaration - Creating a variable, reserves a space in memory and gives it a name.

- var age;

Assignment - gives that variable a value.

- age = 2;

Intialization (Declaration and Assignment):

- var age = 2;

VAR OUT, MAN

Re-assignment:

- var name = “Joe”;
- name = “Eddie”;
- name = “Constantin”;

alert(name);

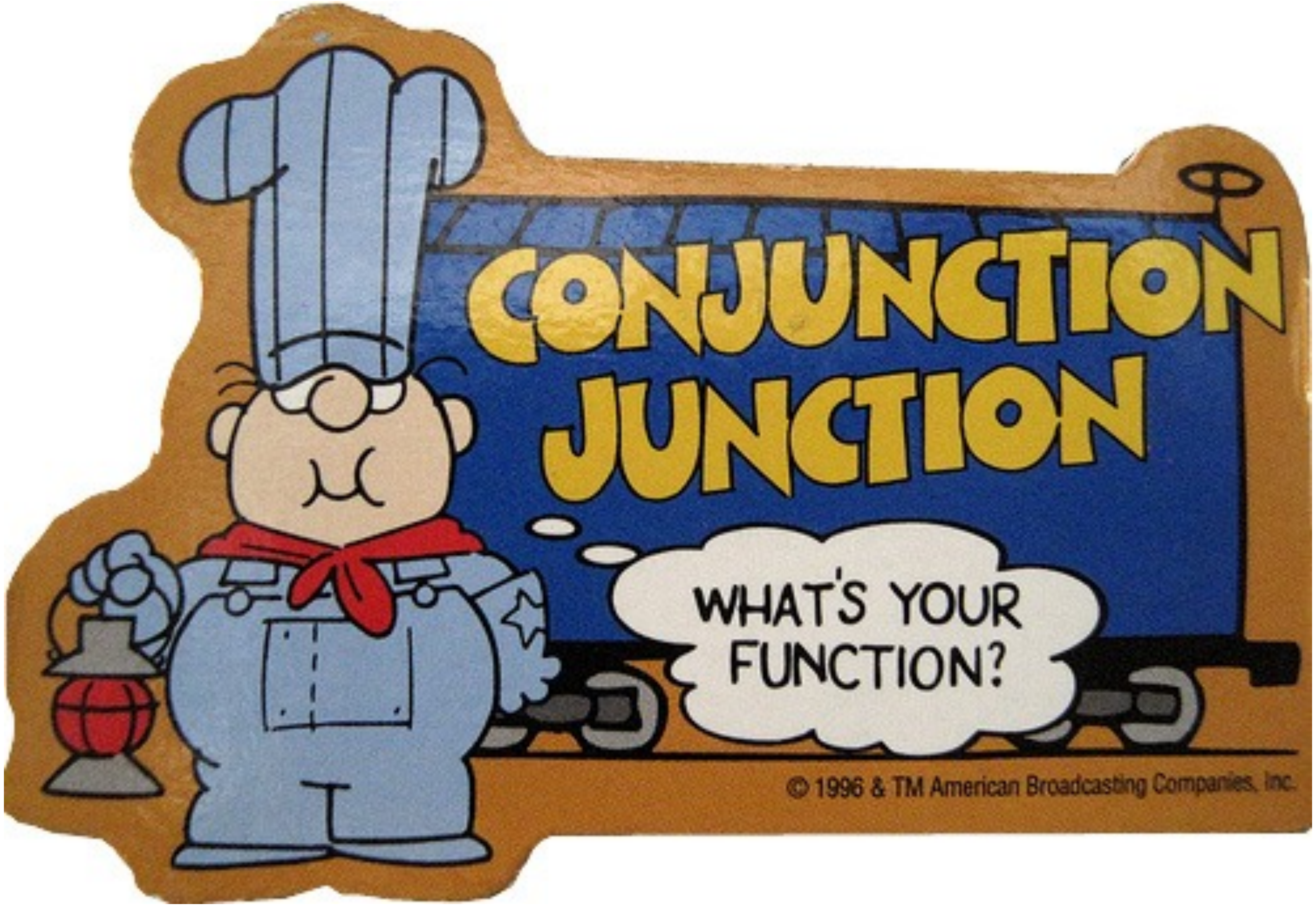
- Will display “Constantin”

WHAT'S IN A NAME?

Variable Naming Conventions:

- Start with a lowercase letter. If they contain multiple words, subsequent words will start with an uppercase letter.
- var number = 10;
- var numberOfClasses = 8;

FUNCTIONS


CONJUNCTION JUNCTION
WHAT'S YOUR FUNCTION?
© 1996 & TM American Broadcasting Companies, Inc.

FUNCTIONS

Functions are simply a collection of lines of code that you group together so that you can:

- Execute them at a given time
- Reuse them
- Respond to user input

We will do much more with these later ...

CODEALONG - COLORFUL PROMPT

When use clicks on button, change the text and the color of the center paragraph to the color they input.

MORE ON STRINGS

Stores textual information

- “Double Quotes”
- ‘Single Quotes’

These are fairly interchangeable.

- alert('Hello');
- alert("Hello");

MORE ON STRINGS

Double vs single quoted strings:

- 'They "purchased" it'
- "It's a beautiful day"

Escaping

- "They \"purchased\" it"
- 'It\'s a beautiful day'

MORE ON STRINGS

Get the number of characters in a String:

- variable.length;

For example:

- var name = “Matt”;
- alert(name.length);

CODEALONG - LIFETIME SUPPLY

Store your current age into a variable.

Store a maximum age into a variable.

Store a favorite drink (from a drop-down) into a variable.

Store an amount per day into a variable.

Calculate how many you would eat total for the rest of your life.

Output the result to the screen.

EXERCISE - FORTUNE TELLER

Using the last exercise as a guide, create a “fortune teller”:
Store the following into variables:

- number of children
- your partner’s name
- your geographic location
- your job title.

Output your “fortune” to the screen like so: “You will be a ___ in ___, and married to ____ with ___ kids.”

COMPARISONS

We often times have things that we want to compare.

Think back to our Thermostat pseudocode.

We will want to check if two values are equal.

We will also want to check if one is greater than, less than, or equal to another.

COMPARISONS (ASSUMING VAR 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

IF A PICTURE PAINTS A THOUSAND WORDS, THEN ... EXECUTE SOME CODE

```
if (this condition is true) {
   //Execute this code
}
//Otherwise continue, skipping the code above

if (true) {
    alert(“The condition is true.”);
}
```

CODEALONG - CLICK COUNT

Change the color of the dot when the count hits a certain threshold.

IF / ELSE

```
if (condition is true) {
    alert(“The condition is true”);
}
else {
    alert(“The condition is false”);
}
```

IF / ELSE IF / ELSE

```
if (condition is true) {
    alert(“The condition is true”);
}
else if (some other condition is true){
    alert(“The first condition was false, but this one is true”)’;
}
else {
    alert(“Neither was true”);
}
```

EXERCISE - CLICK COUNT

Add some more functionality to the click counter by adding an else if and an else to the existing code.

EXERCISE - TEMPERATURE CONVERTER

This assignment is open-ended. The HTML/CSS is up to you. There is no starter code.

Build an application using HTML/CSS and JS that converts a temperature from Fahrenheit to Celsius AND from Celsius to Fahrenheit, based on user input.

HOMEWORK

Finish the Temperature converter.

Add the Developer / Designer pages to the Startup Matchmaker. (both are based on the same template)