

Welcome to  
**Learn to Code**

Updated: March 17, 2017

powered by  galvanize

# About Galvanize

Dynamic learning community  
for technology

- web development
- workspace
- data science
- networking

To learn more,  
visit [galvanize.com](http://galvanize.com)



#SeaLTC

powered by galvanize

# Workshops Available

## Web Development Foundations in JavaScript

Email [enrollment@galvanize.com](mailto:enrollment@galvanize.com) for more information.

Visit:  
[galvanize.com/courses/web-development-foundations-with-javascript/](http://galvanize.com/courses/web-development-foundations-with-javascript/)  
[bit.ly/galvanize-wdfjs](http://bit.ly/galvanize-wdfjs)



powered by **galvanize**



# About Web Development Immersive

- 24 Week Full-Time Program
- 91% Job Placement Rate within six months
- Average starting salary: \$77,000 per annum
- Scholarships available for those who qualify

More information:

[galvanize.com/seattle/web-development](http://galvanize.com/seattle/web-development)

Email [lauren.lark@galvanize.com!](mailto:lauren.lark@galvanize.com)

powered by **galvanize**

# For more information

Email Lee Ngo at  
[lee.ngo@galvanize.com](mailto:lee.ngo@galvanize.com)

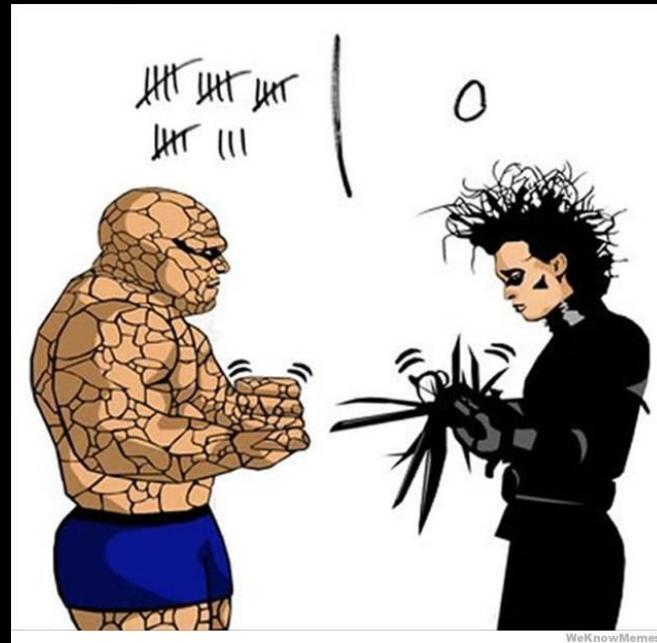
or  
Visit our website at  
[galvanize.com](http://galvanize.com)



#SeaLTC

powered by  galvanize

# But first...



#SeaLTC

powered by galvanize

# EPIC ROCK PAPER SCISSORS

- Best 2 out of 3 contests to advance
- Players say “Rock, Paper, Scissors, SHOOT” simultaneously and present their choice on “SHOOT” - otherwise, re-do that round
- Winner advance and finds someone else to battle
- Non-winner becomes acolyte of the winner and provides moral support
- When we are down to the last 2 - FINAL SHOWDOWN



powered by galvanize

# Workshop Intro to JavaScript

#SeaLTC

powered by galvanize

# About this Workshop's Architect

Graham McBain

@grahammcbain

Graduate of the Web

Dev Immersive

Program (g3)

CEO of Soapbox

Developer Evangelism



# About this Workshop's Architects



Lee Ngo

[github.com/lee-ngo](https://github.com/lee-ngo)

Galvanize Evangelist  
based in Seattle

Made a Game of  
Thrones text  
adventure game

# Source Code for this Workshop

The screenshot shows a GitHub repository page for 'Learn-to-code-week-2'. At the top, there are navigation links for 'Explore', 'Features', 'Enterprise', and 'Pricing', along with 'Sign up' and 'Sign in' buttons. The repository name 'GalvanizeOpenSource / Learn-to-code-week-2' is displayed, along with 'Watch 2', 'Star 4', 'Fork 74', and a 'Code' button. The repository has 24 commits, 1 branch, 0 releases, and 5 contributors. A commit by 'GalvanizeEvangelists' is shown: 'added question outlines' at 'fb441ca' 6 days ago. Other files listed include 'css', 'README.md', and 'index.html'. A 'Learn-to-code-week-2' section contains a 'Basic JavaScript and jQuery' section with a note about following a tutorial and a 'Please email if you are doing this at home and have any questions!' message.

[github.com/  
GalvanizeOpenSource/  
Learn-To-Code-JavaScript](https://github.com/GalvanizeOpenSource/Learn-To-Code-JavaScript)  
[bit.ly/galvanize-ltc-js-1](http://bit.ly/galvanize-ltc-js-1)

We'll explain how you  
will use this link

# Do you have a **text editor**?



We recommend that you use Atom, which is build and maintained by GitHub at: [atom.io](http://atom.io)

Otherwise,  
use **CodePen**



You can do this entire lesson within your  
web browser:

<http://codepen.io/hienpd/pen/GqZNxj>

# Download the code!

1. Go to: [github.com/GalvanizeOpenSource/  
Learn-To-Code-JavaScript](https://github.com/GalvanizeOpenSource/Learn-To-Code-JavaScript)  
Or try: [bit.ly/galvanize-ltc-js-1](http://bit.ly/galvanize-ltc-js-1)
2. Download the zip file of our code and unzip the folder
3. Open the files in your text editor
  - a. index.html
  - b. CSS/style.css
4. Open the index.html file in your browser

# The Download ZIP is right there...

A screenshot of a GitHub repository page for 'GalvanizeOpenSource / Learn-To-Code-JavaScript'. The page shows basic repository statistics: 32 commits, 1 branch, 0 releases, and 6 contributors. A large green arrow points from the top left towards the bottom right, specifically highlighting the 'Clone or download' button. The repository description reads: 'Learn some basic JavaScript by building a "Rock, Paper, Scissors" application! — Ed.' Below the stats, there are buttons for 'Create new file', 'Upload files', 'Find file', and the highlighted 'Clone or download' button. The commit history lists several changes:

- lee-ngo committed on GitHub JavaScript is NOT Java
- css changes to CSS 8 months ago
- README.md JavaScript is NOT Java 5 minutes ago
- index.html removing the JS code so that people must type it 6 months ago

At the bottom, there is a partial view of another README.md file.

# Pictures of Kittens

Setting up everything can be stressful!



# Recap from Workshop 1

- Set up your computer for web development
- Overview of basic **HTML** concepts
- Overview of **CSS** concepts
- Working in the **sandbox**

# In this course you will learn

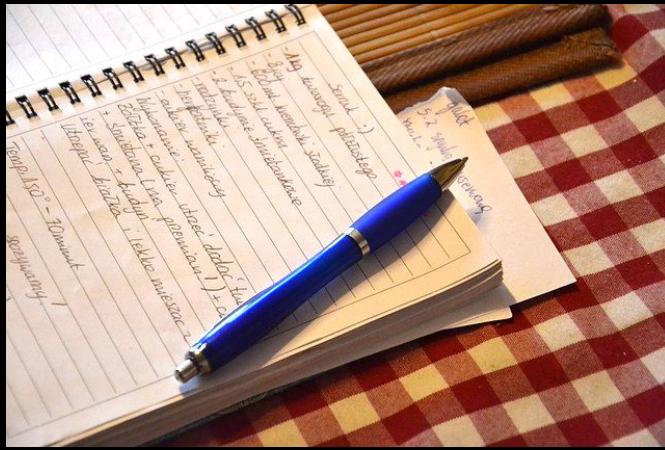
- **Basic syntax** of JavaScript
- **Variables and Functions**
- **Conditional statements** (if, else if, else)
- Build a “**Rock, Paper, Scissors**” application

# Gut check, Galvanize style!



- This course is for beginners
- Feel free to move ahead
- Help others when you can
- Be patient and nice
- We'll all get through it!

# What web coding is (really!)



&t



Recipes to give to your computer to “cook” up some awesome things for you online

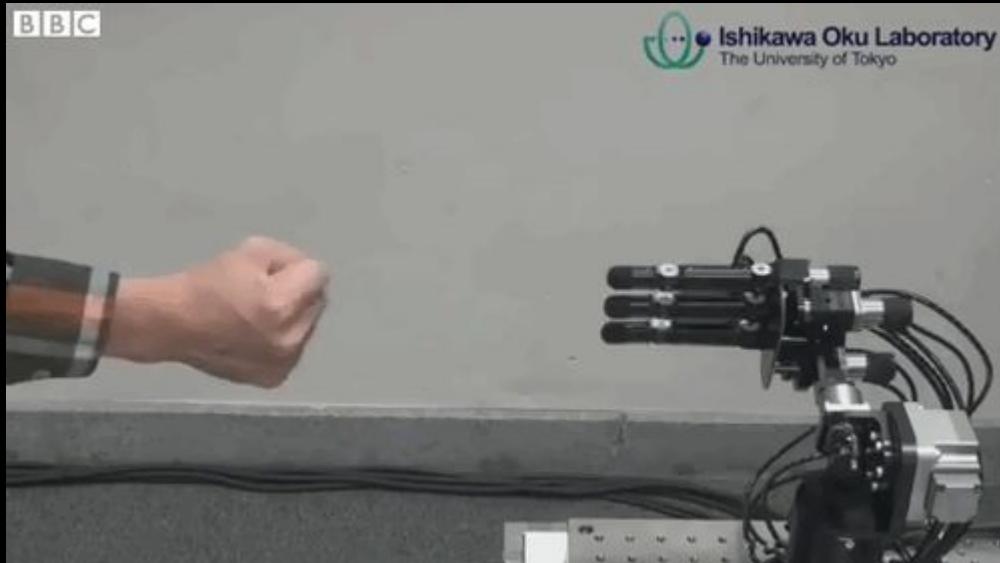
# Remember this?



#SeaLTC

powered by galvanize

# We're going to make our own app!



You are going to  
play with (against)  
the computer!

# 4 Steps to Building This App

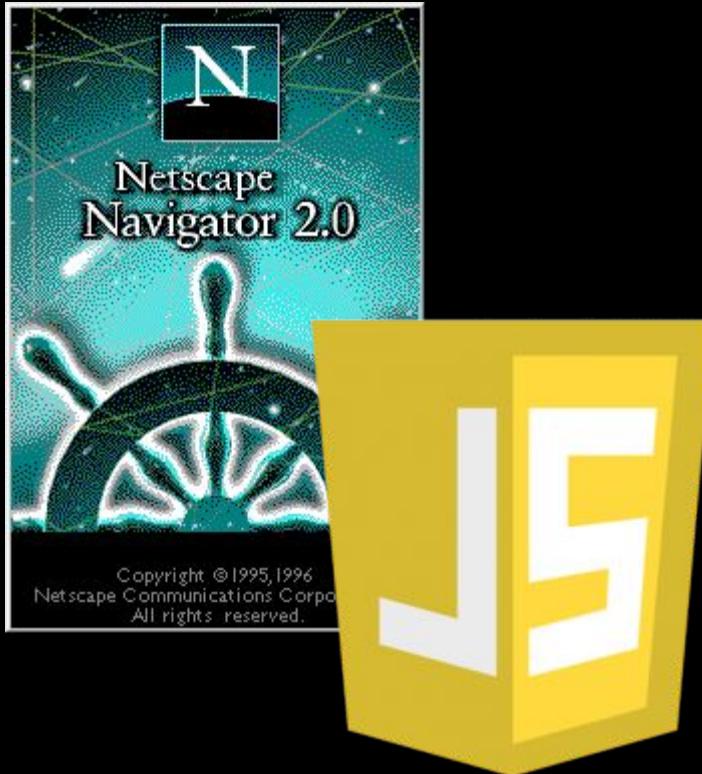
1. Get the user's choice
2. Get the computer's choice
3. Teach the computer how to guess rock, paper, or scissors
4. Compare their choices and tell the user the result

But first...

# What IS JavaScript?

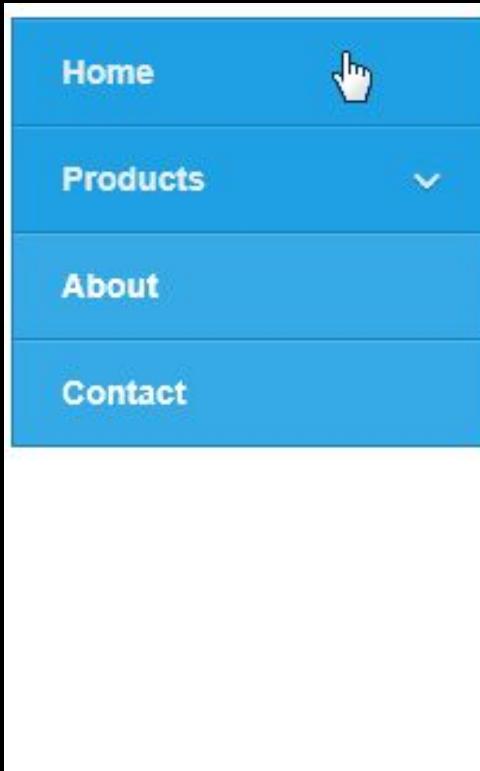
(and why is it called that?)

# Remember Netscape?



- First appeared in 1995
- Originally called “Mocha”, then “LiveScript”, then “JavaScript”
- With HTML + CSS, JS is essential in all web development

# From Static to Dynamic



- JS allows web pages to do more than just “sit there”
- You can animate, calculate, etc. - you can do it all!
- It is a bridge between “design” and “development”

# Java is NOT JavaScript!

Java

```
class HelloWorldApp {  
    public static void main(String [] args) {  
        System.out.println("Hello World!");  
    }  
}
```

JavaScript

```
var HelloWorldApp =  
function() {  
    console.log(  
        "Hello World!");  
}
```

# JavaScript's Basic Syntax

**var** - defines a variable

**;** - terminator

**“word”** - string creator

**function()** - does something

**{ }** - block notation

**.** - dot notation



# LET'S CODE!

(First exercise....)

# Let's run through a simple tutorial!

Go to [www.javascript.com/try](https://www.javascript.com/try) and do the quick 10-minute tutorial!

- Finished with the tutorial?

You're ready to move on to the next step!

# In this course you will learn

- ~~Basic syntax of JavaScript~~
- Variables and Functions
- Conditional statements (if, else, else if)
- Build a “Rock, Paper, Scissors” application

# Variables!

(A little bit of old-school algebra)

# What are variables?

Syntax:

```
var price1 = 5;
```

```
var price2 = 6;
```

```
var total = price1 + price2;
```

*What is the value of total?*

Variables are containers for storing data.  
In JS, you must declare them, then define.

# Variables can store...

**Strings** - “Hello, my name is Lee.”

**Numbers** - 40, 0.15

**Boolean** - true or false

**Null** - literally nothing

**“Nothing”** - undefined values

**Functions** - here we go...!

# What's the difference between...?

= - assignment operator

```
var foo = 1
```

== - abstract equality comparison (value)

```
“1” == 1 => true
```

=== - strict equality comparison (value & type)

```
“1” === 1 => false
```

# LET'S CODE!

(Make your first variable)

# Get the user's choice!

Assign a prompt method to the variable  
`userChoice`:

```
>> var userChoice = prompt("Do you  
choose rock, paper or scissors?");
```

*Why is this a terrible way to get user input?*

# Functions

(Make it do something...anything!)

# What are functions?

Syntax:

```
var multiply = function(a,b){  
    return a * b  
};
```

`multiply(2,4);` ← *What is the value?*

Block of code that performs a task  
In JS, you declare, define, CALL (invoke)

# Syntax of a Function

**Parameters** - (a,b,c) - hypothetically what passes through the function

**Arguments** - real values of the parameters the function affects

**Block** - {...} - the function's operational code

**Return** - the output of the function

# LET'S CODE!

(Make your first function)

# Get the computer's choice!

Assign a Math random method to the variable  
**computerChoice**:

```
>> var computerChoice = Math.random();
```

What is Math in JavaScript?

How else can we get a random choice?

# Recap of Variables and Functions

- ❑ Variables are JS containers for data
- ❑ Functions perform tasks in JS

You're ready to move on to the next step!

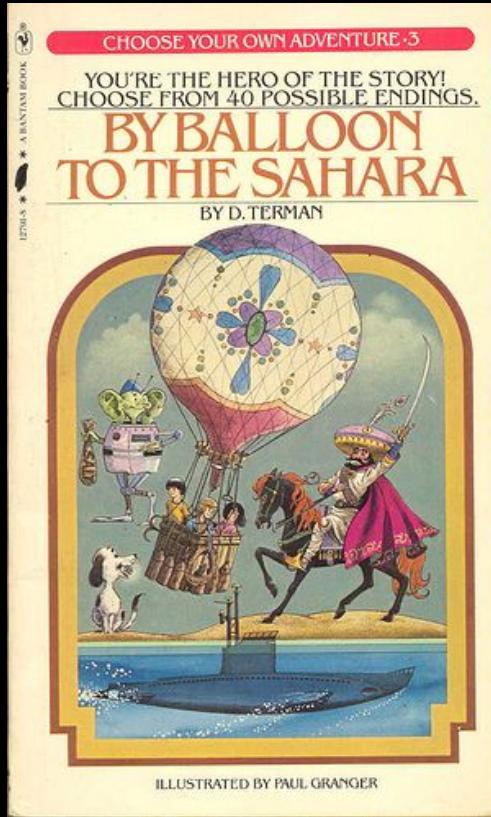
# In this course you will learn

- ~~Basic syntax of JavaScript~~
- ~~Variables and Functions~~
- Conditional statements (if, elseif, else)
- Build a “Rock, Paper, Scissors” application

# Conditional Statements

(If you want to move on, great! Or else...)

# If, else, else if - conditionals!



Coding is all about logic!  
You have to clearly define the rules of engagement.  
Think of creating conditionals as building your own “Choose Your Own Adventure”  
\*This is a book

# Conditional Syntax - if

**if** - if what's in the parameters is **true**, then a block of code will run.

If it's **false**, the code will not run.

```
if (hour < 18) {  
    greeting = "Good day";  
}
```

# Conditional Syntax - else

**else** - what if you wanted the code to do something else if it's **false**?

```
if (hour < 18) {  
    greeting = "Good day";  
} else  
{ greeting = "Go away.";  
}
```

# Conditional Syntax - else if

What if another scenario comes up?

```
if (hour < 18)
    {greeting = "Good day";}
else if (hour < 9)
    {greeting = "OK day";}
else {greeting = "Go away.";}
# THIS CODE IS NOT WELL WRITTEN - WHY? #
```

# LET'S CODE!

(Build a conditional for your app)

# Back to if, else, else if

```
if (computerChoice <= 0.33) {  
    computerChoice = "rock";  
} else if (computerChoice <= 0.66) {  
    computerChoice = "paper";  
} else {  
    computerChoice = "scissors";  
}
```

# Recap of Conditionals

- ❑ **if** statements perform an action if the statement is **true**
- ❑ **else** statements perform an action if the statement is **false**
- ❑ **else if** statements perform an action if the first is **false** but the second is **true**

You're ready to move on to the next step!

# In this course you will learn

- ~~Basic syntax of JavaScript~~
- ~~Variables and Functions~~
- ~~Conditional statements (if, elseif, else)~~
- Build a “Rock, Paper, Scissors” application

# LET'S CODE!

(Now, the final “bit” of code...)

# But...who won the game?

Let's create a function called **compare**

(okay if you need to use your GitHub here... this code is a bit of a nightmare)

```
var compare = function(userChoice, computerChoice) {  
    if (userChoice === computerChoice) {  
        window.alert("The result is a tie!");  
    } else if(userChoice === "rock") {  
        if (computerChoice === "scissors") {  
            window.alert("Rock wins!");  
        } else {  
            window.alert("Paper wins");  
        }  
    } else if(userChoice === "paper") {  
        if(computerChoice === "rock") {  
            window.alert("paper wins!");  
        } else {  
            window.alert("scissors wins!");  
        }  
    } else if(userChoice === "scissors") {  
        if (computerChoice === "paper") {  
            window.alert("scissors wins!");  
        } else {  
            window.alert("Rock wins");  
        }  
    }  
};
```

# Some new functionality here

`====` - “is exactly equal to”

Not to be confused with `=` (the assignment operator) or `==` (abstract equality comparison)

`window.alert()` - pop-up notification (enable them for now)

# Now let's call it in our app in HTML!

Add this into your HTML file:

```
<button class="button"  
onclick="compare(userChoice,  
computerChoice);">LET'S PLAY!</button>
```

# Pictures of Kittens

Did it work? Great! No? Let's figure it out!

Let's look into the **Console** with **Inspect**



# Play around in the sandbox!

- "I want to play again!"
- "I want the game to congratulate me by name!"
- "Make it so I always win."
- "I don't want to have to click a button to play."



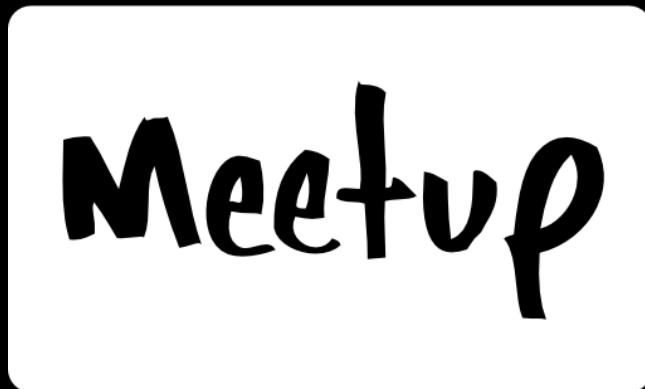
# In this course you will learn

- ~~Basic syntax of JavaScript~~
- ~~Variables and Functions~~
- ~~Conditional statements (if, elseif, else)~~
- Build a “~~Rock, Paper, Scissors~~” application

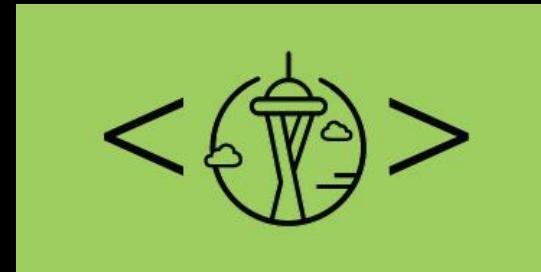
# You did it!

You are now a JavaScript coder.  
ACHIEVEMENT UNLOCKED!

# Keep the party going!



Sign up via the  
Learn to Code Seattle  
Meetup Group



# Workshops Available

## Web Development

### Foundations in JavaScript

Email [enrollment@galvanize.com](mailto:enrollment@galvanize.com) for more information.

Visit:  
[galvanize.com/courses/web-development-foundations-with-javascript/](http://galvanize.com/courses/web-development-foundations-with-javascript/)  
[bit.ly/galvanize-wdfjs](http://bit.ly/galvanize-wdfjs)



powered by **galvanize**



# About Web Development Immersive

- 24 Week Full-Time Program
- 91% Job Placement Rate within six months
- Average starting salary: \$77,000 per annum
- Scholarships available for those who qualify

More information:

[galvanize.com/seattle/web-development](http://galvanize.com/seattle/web-development)

Email [lauren.lark@galvanize.com!](mailto:lauren.lark@galvanize.com)

powered by **galvanize**

# Thank you for coming to galvanize

Email Lee Ngo at  
[lee.ngo@galvanize.com](mailto:lee.ngo@galvanize.com)

or  
Visit our website at  
[galvanize.com](http://galvanize.com)



This course has been brought to you by the evangelists of Galvanize.