

# JavaScript

# Recap

- HTML
- CSS
- Design Thinking
- JavaScript

# Where does JavaScript fit in?

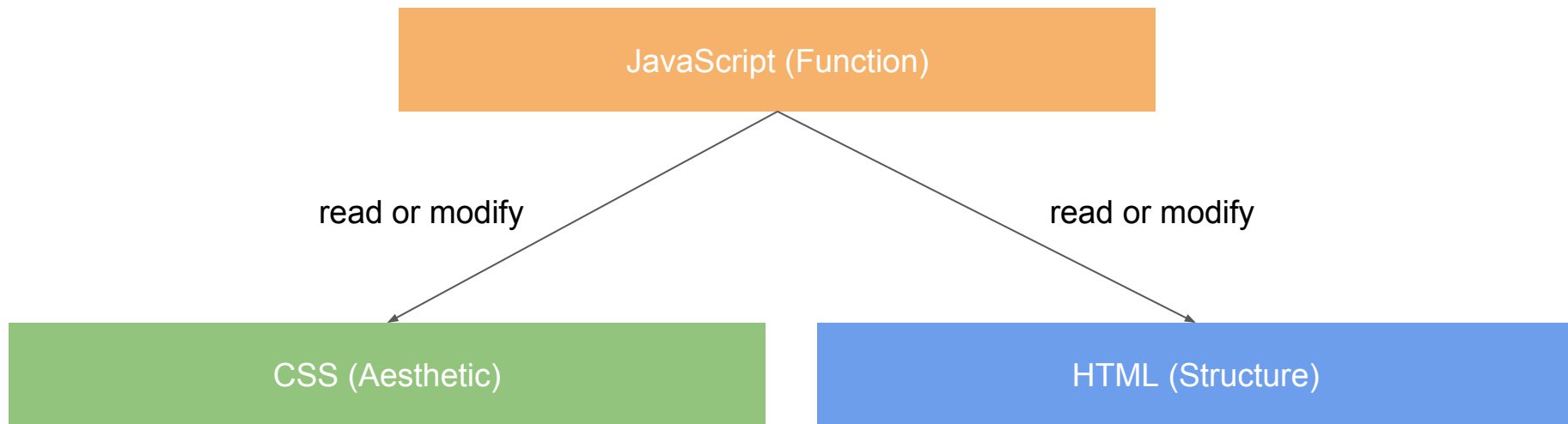


JavaScript (Function)

CSS (Aesthetic)

HTML (Structure)

# Where does JavaScript fit in?



# Examples of JavaScript

Type in search?

Click on Home?

Click on profile picture?

Click on Reply?

Click on Show this thread?

Scroll down?

The screenshot shows a Twitter interface with a tweet from Ray Tsang (@saturnism) celebrating Pi Day. The tweet includes a code block with Pi calculation statistics. Annotations with arrows point to various UI elements:

- Click on Home?**: Points to the Home tab in the top navigation bar.
- Click on profile picture?**: Points to the profile picture of Ray Tsang.
- Click on Reply?**: Points to the reply icon (speech bubble) below the tweet.
- Click on Show this thread?**: Points to the "Show this thread" link at the bottom of the tweet.
- Type in search?**: Points to the search bar at the top right.
- Scroll down?**: Points to the bottom of the page, indicating scrolling.

**Tweet Content:**

Google Developers Retweeted

**Ray Tsang** @saturnism · 11h  
Happy #PiDay! We just calculated 750-billion digits of Pi calculation in 10 h (compared to 2-days it took in 2017) using something we'll announce soon @GCPcloud. Stayed tuned! #PiDay2018

```
π: 24571.086 seconds ( 6.825 hours )
Base Converting:
Time: 1683.936 seconds ( 28.066 minutes )
Writing Decimal Digits: 750,000,000,000 digits written
Verifying Base Conversion:
Time: 436.295 seconds ( 7.272 minutes )
Verifying Decimal Output:
Time: 4983.281 seconds ( 83.055 minutes )
Start Time: Mon Mar 12 21:26:29 2018
End Time: Tue Mar 13 07:22:16 2018
Total Computation Time: 26255.022 seconds ( 7.293 hours )
Start-to-End Wall Time: 35747.929 seconds ( 9.930 hours )
CPU Utilization: ██████████ + 1.96 % kernel overhead
Multi-core Efficiency: 93.17 % + 0.01 % kernel overhead
Last Digits: π
7802900108 9020660178 9996901755 7358537919 1208396296 : 749,999,9
9728959182 9959700411 0463747323 2418512577 4343912477 : 750,000,0
Spot Check: Good through 750,000,000,000
Version: 0.7.5.9481 (Linux/14-BDW ~ Kurumi)
Processor(s): Intel(R) Xeon(R) CPU @ 2.20GHz
Topology: 2 Cores/Socket 14 Sockets 28 Cores 56 Hyper-Threads : 4 200MB Cache
Usable Memory: 8.00GB RAM 500.00GB 100.00GB RAM
CPU Base Frequency: 2,200,098,864 Hz
Validation File: Pi - 20180313-072217.txt
```

12 replies 141 retweets 292 likes

[Show this thread](#)

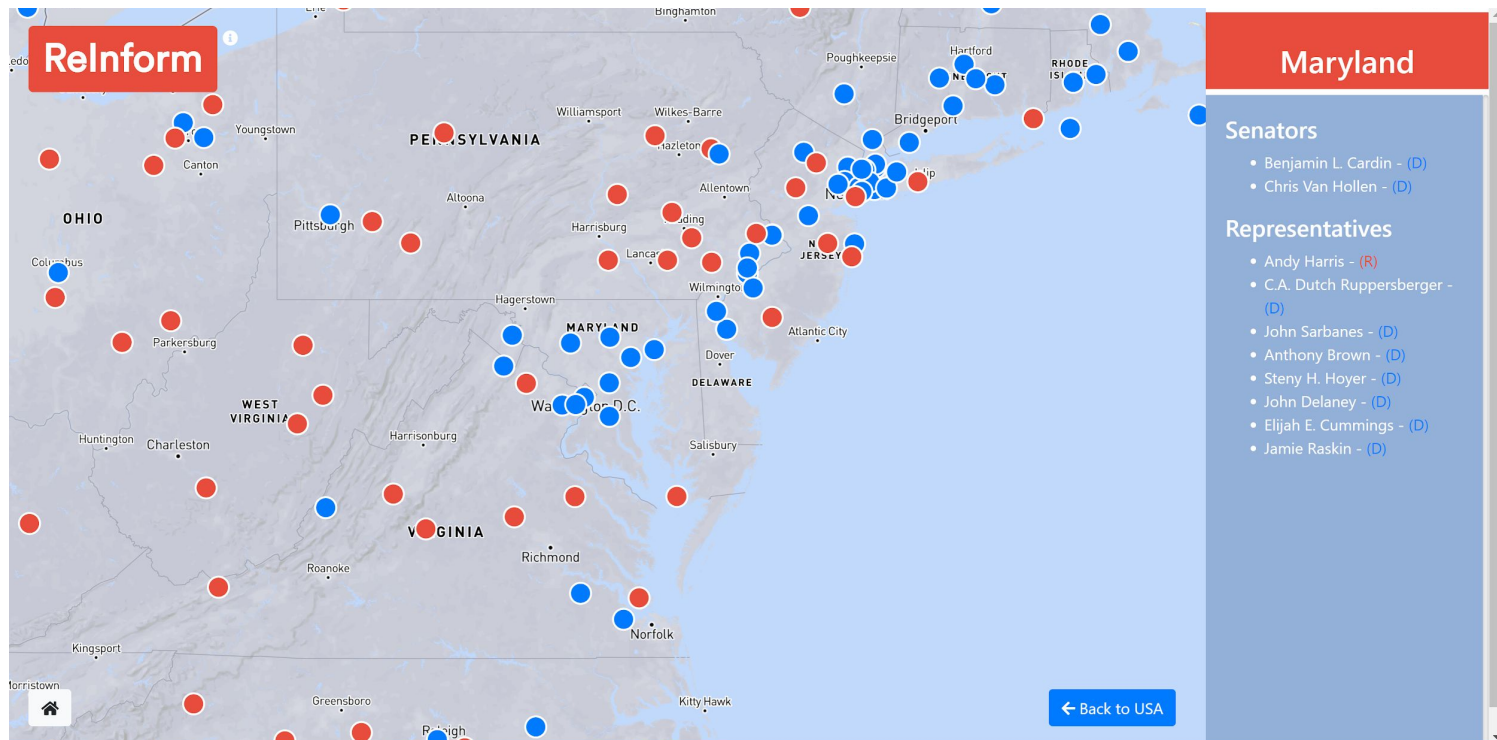
**TwitterU Recruiting** @TwitterU · 4h

**Search results for UVA:**

- #UVA
- UVA injury
- [uva baseball](#)
- uva hunter
- UVA** @UVA
- Virginia Men's Basketball** @UVAmen...
- Virginia Football** @UVAFootball
- Virginia Baseball** @UVABaseball
- Darden School at UVA** @DardenM...
- UVA Law School** @UVALaw





Search all people for UVA

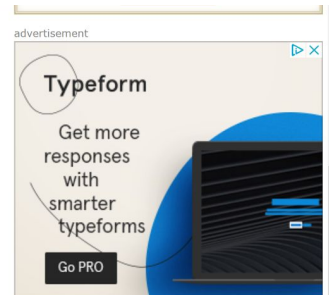
# Examples of JavaScript



Reinform.me

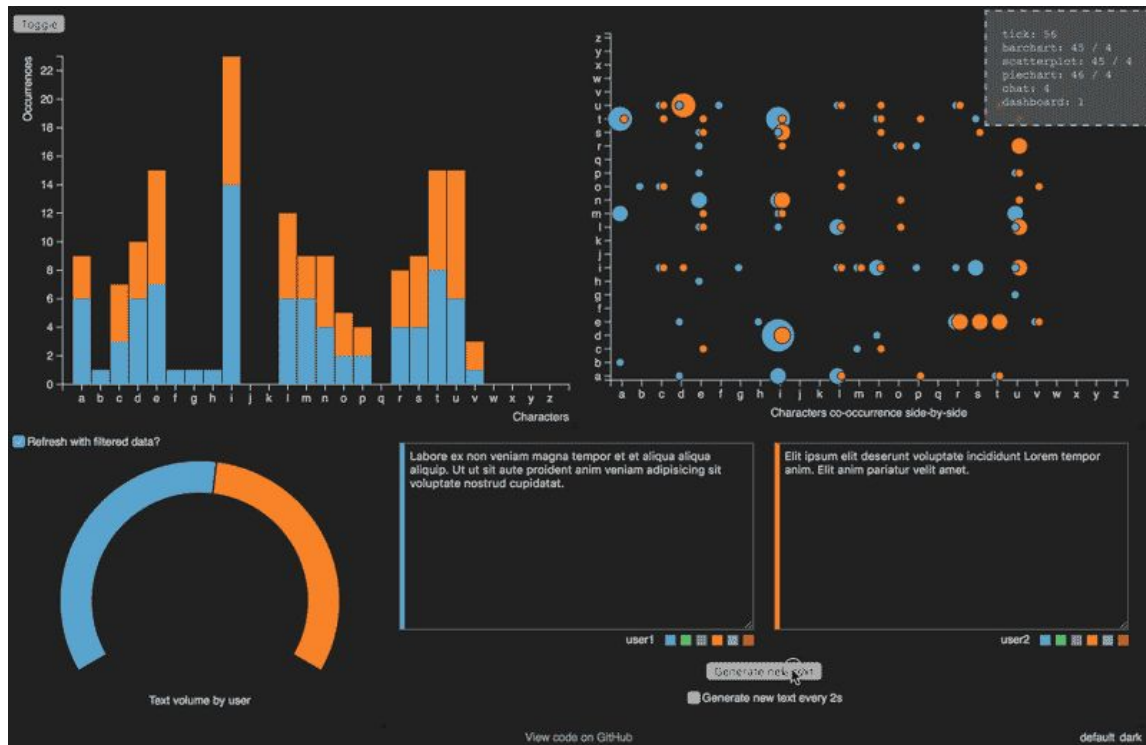
# Examples of JavaScript

- 5 44.8k  [Astronaut's DNA No Longer Matches His Identical Twin's After Year Spent in Space, NASA Finds](#) (ktla.com)  
submitted 6 hours ago by IAmClaytonBigsby to r/news  
1328 comments share save hide report
- 6 19.2k  [New glider concept : no glider at all, you just hit the ground full speed and die](#) HUMOR (i.redd.it)  
submitted 5 hours ago by DaLinkster to r/FortNiteBR  
337 comments share save hide report
- 7 15.0k  [I sing to her in the morning before I leave to work. She has a favorite spot where she lays to listen right next to me.](#) (i.redd.it)  
submitted 5 hours ago by mark503 to r/dogpictures  
294 comments share save hide report
- 8 37.1k  [Fluff enjoying a peaceful snowfall](#) (i.imgur.com)  
submitted 6 hours ago by mike\_pants to r/aww  
251 comments share save hide report



Clicking on the “expand”  
button makes a large  
image/video appear

# Examples of JavaScript



<https://rd3.now.sh/>



# Where is JavaScript written?

```
<!DOCTYPE html>
<html>
  <head>...</head>
  <body>
    <p>...</p>
    <script>
      // javascript goes here
    </script>
  </body>
</html>
```

# Where is JavaScript written?

- Can also be written in a separate .js file and imported in

```
<!DOCTYPE html>
<html>
  <head>
    <script src="myscripts.js"></script>
  </head>
  <body>...</body>
</html>
```

# Variables

- Store information
  - numbers, strings, arrays, objects, functions
- Variable names
  - must be unique
  - can contain letters, digits, underscores and dollar symbols
  - cannot start with a digit
  - are case sensitive
  - cannot be a keyword

# Variables

- Declaration

```
var x; // undefined
```

- Initialization

```
x = 3;
```

- Both

```
var y = "hi";
```

# Variable Types

```
var anumber = 3;
```

```
var anothernumber = 3.0;
```

```
var astring = "hello";
```

```
var anotherstring = 'world';
```

```
var anarray = [];
```

```
var anotherarray = [3, "hello", anotherstring];
```

```
var anobject = {};
```

```
var anotherobject = {"key": "value"};
```

# Operators

```
var x = 5; // 5
```

```
x = x + 1; // 6
```

```
x += 1; // 7
```

```
x -= 1; // 6
```

```
x *= 4; // 24
```

```
x /= 3; // 8
```

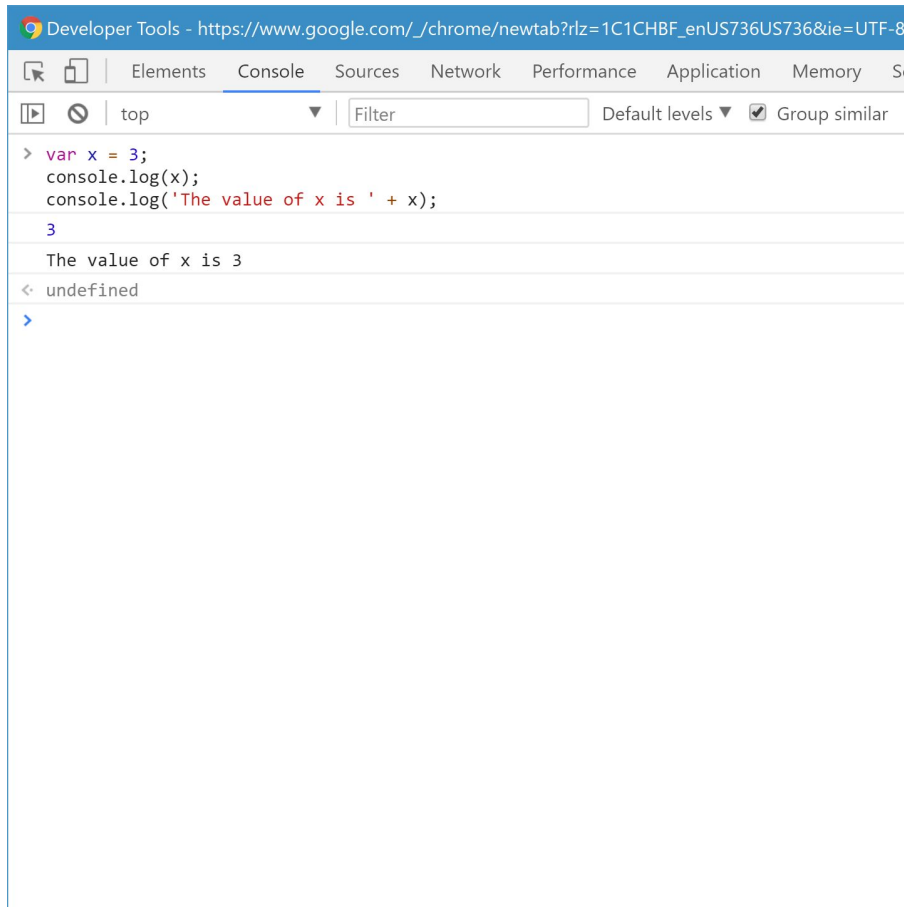
```
x %= 5; // 3
```

# Printing variables

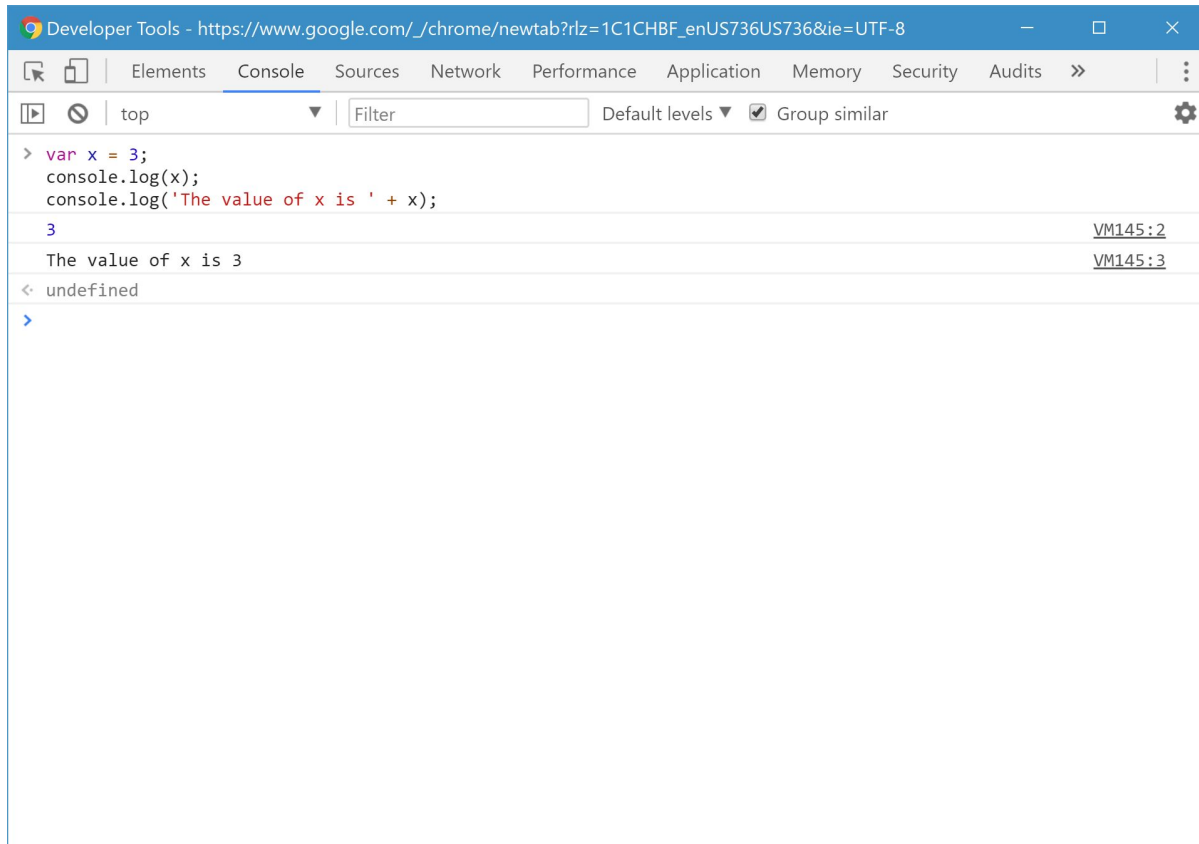
```
var x = 3;
```

```
console.log(x);
```

```
console.log('The value of x is ' + x);
```



# Recap: Chrome Developer Tools





# More on Variables

**let** declares a  
variable in the  
local scope

**var** declares a  
variable in the  
global scope

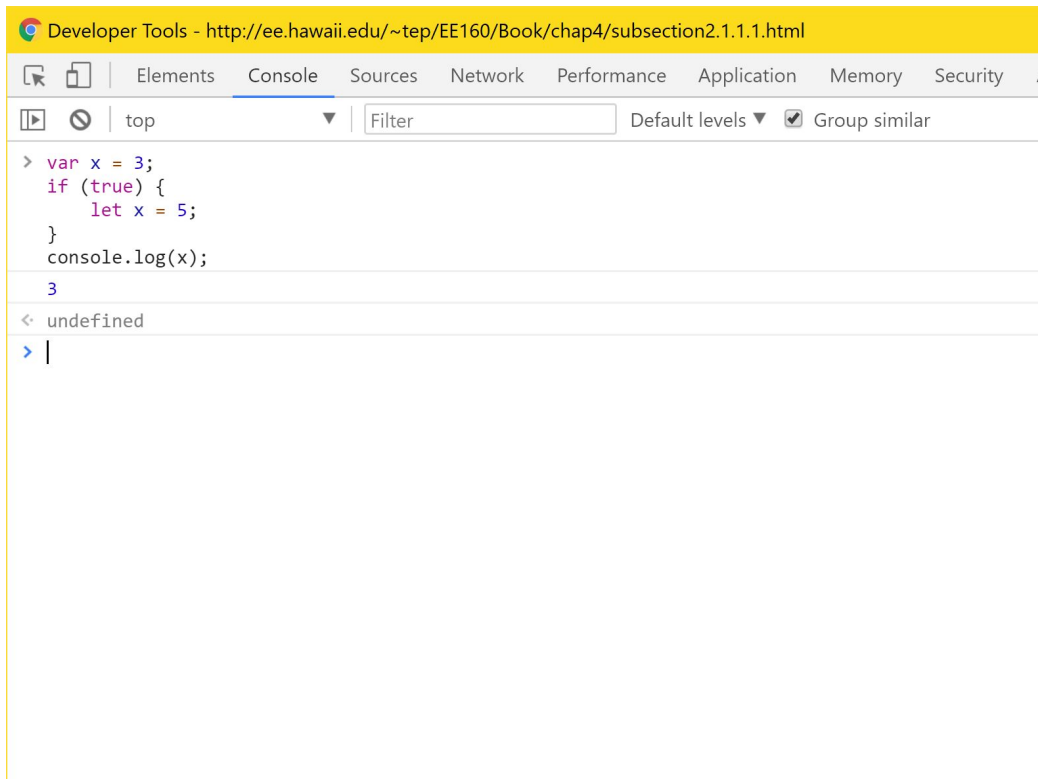
```
var x = 3;
```

```
if (true) {
```

```
    let x = 5;
```

```
}
```

```
console.log(x);
```



# If statements

```
if(condition) {  
    // then do this  
}
```

# But first, conditions...

<, <=, >, >=

```
"" == 0 // true
```

== vs. ===

```
0 == "0" // true
```

```
"0" == "" // false
```

!= and !==

```
"" === 0 // false
```

&&, ||

```
0 === "0" // false
```

```
"0" === "" // false
```

# If statements

```
if(hungerLevel >= 5) {  
    orderPizza(); // function call  
}
```

# If statements

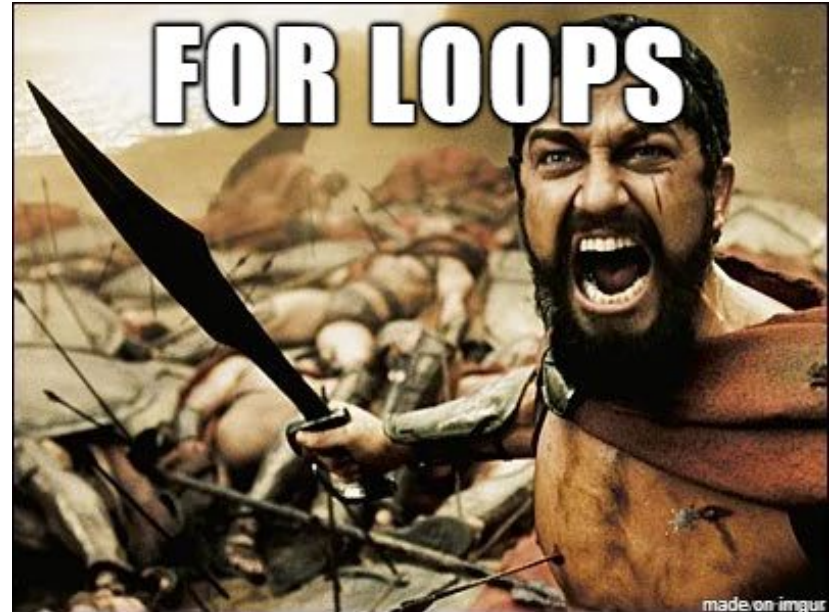
```
if(hungerLevel >= 5) {  
    orderPizza(); // function call  
} else if(hungerLevel >= 3) {  
    goToNewcomb();  
}
```

# If statements

```
if(hungerLevel >= 5) {  
    orderPizza(); // function call  
} else if(hungerLevel >= 3) {  
    goToNewcomb();  
} else {  
    subsist();  
}
```

# Loops

- **for**
- for-in
- **while**
- do-while



# For-loops

```
for (statement 1; statement 2; statement 3) {  
    // code you want looped  
}
```

statement 1: executes before the loop starts

statement 2: defines the condition for looping

statement 3: executes for every iteration of the loop



# For-loops

```
for(let i = 0; i < 6; i++) {  
    console.log(i); // do something using our index value  
}  
  
for(let r = 0; r < 6; r++) {  
    for(let c = 0; c < 6; c++) {  
        console.log(r + c); // for-loops can be nested  
    }  
}
```

# While loops

```
var check = true;  
while(check) {  
    check = checkIfSomethingIsTrue(check);  
}
```

# While loops

- Can also be implemented using a “base case”

```
var i = 0;
```

```
var check = true;
```

```
while(check) {
```

```
    if(i > 6)
```

```
        break;
```

```
    check = checkIfSomethingIsTrue(check);
```

```
}
```

# Functions

```
var i = myFunction(3,4);
```

```
console.log(i); // This will print out the number 12
```

```
function myFunction(p1, p2) {
```

```
    return p1 * p2; // The function returns the product of p1 and p2
```

```
}
```

# Functions

Normal function definition.

```
function foo(a, b) {  
    return a + b;  
}  
console.log(foo(3, 5)); // will print 8
```

Functions can be stored in variables.

```
let bar = function(a, b) {  
    return a + b;  
}  
console.log(bar(3, 5)); // will print 8
```

# Functions

Just a heads up, functions can also be defined in some weird ways.

```
let baz = (a, b) => {  
  return a + b;  
}
```

```
console.log(baz(3, 5)); // will print 8
```

# Linking Javascript to HTML

We discussed HTML's tree structure earlier

When an HTML document is loaded into a web browser, it becomes a document object

- Document object is the root node of the HTML document
- "Owner" of all other nodes
  - element nodes, text nodes, attribute nodes, and comment nodes
- Provides properties and methods to access all node objects from within JavaScript
- Very useful for dynamic programming

[https://www.w3schools.com/js/tryit.asp?filename=tryjs\\_intro\\_inner\\_html](https://www.w3schools.com/js/tryit.asp?filename=tryjs_intro_inner_html)