

---

---

# JavaScript

Ari Cohn

---

---

# Introduction: What is JavaScript?

- Main scripting language of websites
- Allows for dynamic content
- Contains object oriented features
- Weakly typed

# Setup

- Text Editor (Sublime Text, Atom, etc.)
- Browser (preferably Chrome or Firefox)
  - JavaScript Console



# Using JavaScript

- Developer Console (Chrome, Firefox)
- Attach script to HTML webpage

- by reference:

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

- inline:

```
<script>  
// code  
</script>
```

# Variables

```
var message = "Hello!";
```

Type	Examples
Number	5, 3.14, NaN
String	"John Smith"
Boolean	true, false
Object	{}, {x: 0, y: 1}, [1, 2, 3], null
Function	Math.abs
Undefined	undefined

# Arrays

- Dynamically sized
- Generic

```
var arr = [1, 2, "dog", null];  
arr[2];    // dog  
arr.length; // 4
```

```
var arr2d = [ [1, 2, 3], [4, 5, 6] ];  
arr2d[1][1]; // 5
```

# Truthiness

Type	Falsy	Truthy
Number	0, NaN	all other numbers
String	""	all other strings
Boolean	false	true
Object	null	all other objects
Function	n/a	all functions
Undefined	undefined	n/a

# Conditional Statements

```
if (x % 2 == 0) {  
    console.log("even");  
} else {  
    console.log("odd");  
}
```

```
if (val) {  
    console.log(val);  
}
```



# Equality

- Abstract Equality (==)
  - Tries to resolve types before comparing
- Strict Equality (===)
  - Returns false if the types are different

# Equality

5 == "5" true

5 === "5" false

0 == false true

0 === false false

"john" == "john" true

"john" === "john" true

# Loops

```
for (var i = 1; i <= 10; i++) {  
    console.log(i);  
}
```

```
var i = 1;  
while (i <= 10) {  
    console.log(i);  
    i++;  
}
```

# Functions

```
function add(x, y) {  
    return x + y;  
}
```

```
var multiply = function(x, y) {  
    return x * y;  
}
```

```
add(3, 4);  
add("John ", "Smith");
```

# Miscellaneous Syntax

- Comments
  - Single-line: `//`
  - Multi-line: `/* */`
- Semicolons
  - Optional, but good practice
- Adding string and another value casts to string
  - `"high " + 5` becomes `"high 5"`

# Practice

1. Write a function that when given a number  $n$  as input, prints the  $n$ th term of fibonacci. (ex:  $\text{fib}(0) = 1$ ,  $\text{fib}(1) = 1$ ,  $\text{fib}(2) = 2$ )
2. Write a function that calculates the average of an array of numbers.  
(Bonus: include values in the array that are not numbers and ignore them)
3. Print out the multiplication table for numbers up to 10.
4. Write a function that determines if a number is prime.

# Objects

- Mappings between keys and values
- Defining objects

```
var coord = {  
  x: 5,  
  y: -3  
};
```

- Accessing properties

```
coord.x  
coord["x"]
```

# Objects

```
var myPerson = {  
  name: "Joseph",  
  "age": 19,  
  eat: function (food) {  
    console.log(this.name + " ate " + food);  
  }  
};
```

```
console.log(myObj["bday"]);    // undefined  
myObj["bday"] = "3/8/1917";  
myObj.age = 100;
```



# Constructors

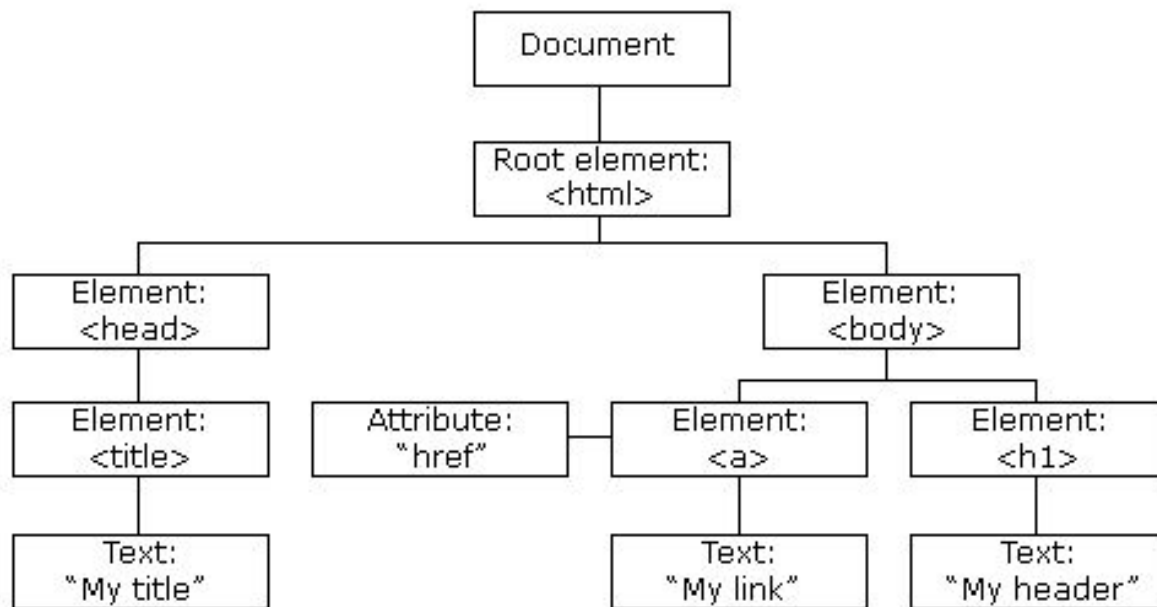
- Construct objects similarly to using a class

```
function Person(name, age) {  
  this.name = name;  
  this.age = age;  
  this.eat = function (food) {  
    console.log("yummy!");  
  };  
}
```

```
var myPerson = new Person("Tom", 18);
```

# Interacting with the DOM

- The HTML DOM is the Document Object Model for a web page



# Interacting with the DOM

- `document.getElementById(id)`
  - Makes a reference to the HTML element with an id of `id`
- `element.innerHTML`
  - Read or edit the contents of `element`'s inner HTML
- `element.onclick`
  - Assign a function to be called when `element` is clicked

```
var nameTag = document.getElementById("name");  
nameTag.innerHTML = "Nathan"
```

# Lab

<https://scottylabs.org/wdw/javascript/lab/>