

HACK UNIVERSITY

< WEB DEVELOPMENT />

WWW.HACKUNIVERSITY.COM

HACK

UNIVERSITY

< WEB DEVELOPMENT />

WEEK 3
WAIT...JAVASCRIPT?



Plan

- Homework review and support groups!
- Javascript
 - The interesting history
 - Beginning programming techniques
 - Live follow-along demo after each section
- Homework Overview



CEID



JOIN



HOMEWORK REVIEW



**MOST
AWESOME**

ALL OF YOU!



**HERE TO ACCEPT THE
AWARD ON BEHALF OF
THE CLASS...**

DREW MCMAHON



HOMEWORK REVIEW

POSTPONED

OVERALL, MORE GREAT STUFF!

REVIEW OF WHAT ELSE TO KNOW

HTML

We will probably cover them, but no guarantees, and not today in class.

- › meta tags
- › forms
- › iframes
- › explore and google them!

CSS

We will probably cover them, but no guarantees, and not today in class.

- selector precedence
- float
- display
- position
- explore and google them!

PROGRAMMING

A QUICK INTRO



WHAT IS A PROGRAMMING LANGUAGE?

Multiple definitions we might consider:

- A language designed specially to communicate instructions to a machine
- A notation for writing programs, which are specifications of a computation or algorithm
- A vocabulary and set of grammatical rules for instructing a computer to perform specific tasks

WHAT IS A PROGRAMMING LANGUAGE?

Defined by two core components: syntax and semantics

- **Syntax:** describes the possible combinations of symbols (letters, numbers, shapes, etc.) that can be used to generate code
- **Semantics:** the meaning of a programming language; i.e. how the syntax is converted into something useful

KEY TERMINOLOGY

- ▶ **Code**
 - ▶ The program to execute, written in a high-level language (e.g. Javascript)
- ▶ **Parser**
 - ▶ A program that “reads” the code and turns it into machine language (i.e. something the computer can process)
- ▶ **Console**
 - ▶ An environment that sends input to and receives output from the parser and execution of the program

JAVASCRIPT

A BRIEF HISTORY



“

Are JavaScript and Java the same thing?

No

*Are JavaScript and JSON the same
in any way?*

THE NAME

In 1995, Java was hot stuff, and the Netscape guys wanted their language to be also.

- So they named it JavaScript
- Supposedly this was in exchange for bundling Sun's Java runtime with their market-leading browser: Netscape Navigator.
- Java Applets (e.g. slimezone.com) later fell out of favor, but JavaScript kept kicking.

“Isn’t Javascript that crappy little toy language that was invented in like twenty minutes by some guy at Netscape?”

“But isn’t Javascript that crappy little toy language that was invented in like twenty minutes by some guy at Netscape?...

“But isn’t Javascript that crappy little toy language that was invented in like twenty minutes by some guy at Netscape?”

Sort of...

but not really anymore.

BRIEF HISTORY

- Javascript was hurriedly developed by Brendan Eich at Netscape in 1995
 - “Mocha” > “LiveScript” > “JavaScript”
 - Submitted to Ecma International for standardization in 1996 > “ECMAScript”
 - Iterated, updated, and enhanced since then
 - For more on Javascript’s colorful history, give this a read: <http://ask.metafilter.com/195482/Lets-assume-that-I-am-the-stupidest-person-that-ever-lived-Explain-to-me-what-JavaScript-is-what-it-does-and-how-a-moron-would-go-about-learning-it#2813956>

JAVASCRIPT AIN'T PERFECT...

Given the rushed conditions of its development, JavaScript has some downsides

- Too permissive. Doesn't complain
- Type coercion (i.e. == vs. ===)
- Awkward typing
 - `if (typeof x == 'undefined') {...}`

BUT IT AIN'T BAD NEITHER!

JavaScript has its advantages

- Functions as first class objects
- Closures
- Event loop
- Prototype chain
- Tail recursion (in theory)
- Fast (thanks to Google's V8 engine)



SYNTAX AND SEMANTICS

SOME CORE RULES

VARIABLES

What is a variable? A named object to which we can assign a value (of any data type desired)

- e.g.

```
var name = "Rafi";
```

- This says: Assign the string “Rafi” to the variable “name”

VARIABLE NAMING

When naming variables in Javascript, you must:

- Always begin with a lowercase letter
- Never use spaces in the name
- Use underscores or camelCase to have a multi-word name
 - UNDERScores: `my_multi_word_variable`
 - CAMELCase: `myMultiWordVariable`

VARIABLE NAMING

Most people use camelCase in JavaScript

- Always begin with a lowercase letter
- Never use spaces in the name
- CAMELCASE: myMultiWordVariable

ASSIGNMENT VS. EQUALITY

To **assign** a value to a variable, put the variable name on the left, the value on the right, and a single = in between

```
var name = "Rafi";
```

ASSIGNMENT VS. EQUALITY

To test for equality between values or variables, use a triple `==`

```
name === "Rafi";
```

- The parser would evaluate this expression to the boolean values `true` or `false` depending on the value stored in `name`

ASSIGNMENT VS. EQUALITY

Never use the double ==

- Like ===, == is evaluative
- It is to be avoided, because it attempts to coerce the types of its operands (more on this later)

KEY SYNTAX POINTS

- Should end every line of declarative code with a semicolon
 - e.g. var name = “Rafi”;
- Encapsulate “code blocks” with curly braces
- Pass arguments or conditions inside parentheses
- White space doesn’t matter

WHITE SPACE BEST PRACTICES

White space doesn't matter, but...

- ▶ Start each declarative statement on a new line
- ▶ Use logical indentation to help code readability
- ▶ Include spaces inside parentheses, curly braces, or square brackets e.g. `if(condition){ var x = 5; }`
- ▶ Use spaces to separate operators (e.g. `+`, `*`, `=`) from their operands (e.g. numbers, strings, variables)

CONTROL FLOW

YOUR APP'S "LOGIC"

CONTROL FLOW

Javascript control flow is similar to many other programming languages

- › if / else if / else
- › ternary operator
- › switch
- › for loops
- › while loops

CONDITIONAL OPERATORS

- `==`
 - Equality, allows for type conversion
- `=====`
 - Strict equality, no type conversion
- `!=`
 - Not equal to
- `> or <`
 - “Greater than” or “less than”
- `>= or <=`
 - “Greater than or equal to” or “less than or equal to”
- `&&`
 - Requires that both conditions are true
- `||`
 - One or both conditions are true

A FEW EXAMPLES

```
var jaredsAge = 22;
var zacksAge = 20;
if ( jaredsAge == "22" ) { /* true because of type conversion */ }
if ( jaredsAge === "22" ) { /* false */ }
if ( jaredsAge >= zacksAge ) { /* true */ }
if ( jaredsAge === 22 && zacksAge === 20 ) { /* true */ }
if ( jaredsAge === 20 || zacksAge === 20 ) { /* true */ }
```

IF / ELSE

Controls flow based on the truth or non-truth of a condition

- ▶ Syntax: `if(condition) { /* execute this code */ }`
- ▶ `else` specifies what to do if the `if` condition is false

```
var name = "Will";
if ( name === "Will" ) {
  console.log( "Wow, you're awesome." );
} else {
  console.log( "Seek a better name..." );
}
```

ELSE IF

If you want to specify more than 1 condition, each with different actions, you can use else if

```
var name = "Zack";
if ( name === "Zack" ) {
    console.log( "Wow, you're awesome." );
} else if( name === "Jared" ) {
    console.log( "Yeah, you're awesome too." );
} else {
    console.log( "Seek a better name..." );
}
```

FOR

Iterates a block of code **for** a “determinate” period

```
'var name = "Jared";
for ( var i = 0; i < name.length; i++ ){
    console.log( "Letter #" + i + " is " + name[i] );
}

/* OUTPUT:
   Letter #0 is J
   Letter #1 is a
   Letter #2 is r
   Letter #3 is e
   Letter #4 is d
*/
```

WHILE

Iterates a block of code **while** a condition is true

```
var name = "Jared";
var i = 0;
while ( i < name.length ){
    console.log( "Letter #" + i + " is " + name[i] );
    i++;
}

/* OUTPUT:
   Letter #0 is J
   Letter #1 is a
   Letter #2 is r
   Letter #3 is e
   Letter #4 is d
*/
```

HOMEWORK REVIEW

LINE BY LINE

LET'S DIG THROUGH THE CODE



HOMEWORK

YOUR EMPLOYER IS NOT
SATISFIED

“

You successfully mimicked the functionality of the leprechaun list, but somehow facebook is still in business. In order to really capture market share, he/she needs you to implement the features that he/she listed as optional on the first go-round.

YOU MUST CREATE FUNCTIONALITY TO...

- add todos to by hitting the enter button
- delete todos
- cross off todos as completed

SUPPORT GROUPS

SUPPORT GROUPS

A sub-network of diversely skilled people who you can work with and use as resources. More information to come.



ADMIN

ADMIN

➤ Homework

- Implement the upgrades to your todo list
- Upload it to CloudChill.in as hw3_your_name
- Send me the url, so I know it's final
- Eloquent Javascript
- More details in the wrap up email

➤ *There will be Oscars*

ADMIN

➤ Office Hours

- Paul Fletcher-Hill (TBA)
- Brandon Jackson (TBA)
- Rafi Khan (Thursday 8pm - 10pm)
- Zack Reneau-Wedeen (Monday 3pm - 5pm)
- Location TBA in the wrap-up email

ADMIN

➤ Support Groups

- Small groups you can direct questions to
- Will be announced next week!

ADMIN

➤ Workshops

- Two big ones this week
 - Twitter Bootstrap with Rafi Khan
 - Adobe Photoshop and Illustrator with Chika Ota
- Get on the mailing list at [hackyale.com!](http://hackyale.com)

THANKS!



HACK UNIVERSITY

QUESTIONS EVEN GOOGLE CAN'T ANSWER?

TEAM@HACKYALE.COM

WWW.HACKUNIVERSITY.COM