HACK YALE

< INTRODUCTION />

WWW.HACKYALE.COM

WELCOME!

The Agenda

- HackYale
- The Internet
- Rapid fire HTML, CSS, and JavaScript
- Teaching lessons
- Q&A

HACK YALE

INTRODUCTIONS

ZACK RENEAU-WEDEEN

- HackYale director, four-time instructor.
- Senior, Trumbull, Computer Science / Econ major
- Worked at Google, Brewster
- Enjoys “PG-rated movies starring Martin Lawrence”

HACK YALE

HACKYALE

THROUGH THE AGES

HACK YALE

HACKYALE

"HackYale provides student-run lectures in web development, introductory programming and design."

HACK YALE

CHRONOLOGY

- Huge demand from the start
- Considered expansion
- Settled in as a provider of educational, cultural, and professional resources to Yale students.

HACK YALE

Current Work

WHAT WE'RE UP TO

CURRENT WORK

- Flagship lecture
- Workshops
- Collaboration
- Jobs

HACK YALE

LECTURE

- Once a week + office hours
- Homeworks

HACK YALE

SHOWCASE

- Yale Film Society
- Friends With Wheels
- Inta
- Yale Taxi Stand
- Screw Me Yale
- Yale Travelogue
- Etc.

HACK YALE

The Future

WHAT'S IN STORE

THE FUTURE

Incremental Change

- Meeting demand
- Jobs and opportunities
- Closer partnership with STC, CEID, etc.
- Cultural presence (e.g. blog)
- Moving into the physical realm

HACK YALE

THE FUTURE

Moon Shots

- HackYale Outreach
- Hack University

HACK YALE

THE INTERNET

THE LONGEST LINE EVER

HACK YALE

THE VIDEO WAS GREAT

THANKS, AARON TITUS

HOWEVER, WE'D LIKE TO GET A BIT MORE TECHNICAL AND ADD THE CONTEXT OF WEB DEVELOPMENT

HACK YALE

TERMINOLOGY

Client-server model

- Client == (you and your) browser
- Server == machine sending (or “serving”) you the data and files you request

HOST ~== “server”

- “to host” (code, files, applications) ~== “to serve”

LOCAL == hosted on the machine in question

REMOTE == hosted on a different machine

HACK YALE

HACK YALE

REQUEST-RESPONSE CYCLE

(1) Client (browser) makes a "request"

- REQUEST == textual message whose syntax and semantics are defined by HyperText Transfer Protocol (*HTTP*)
- Think of a protocol as a "language"

(2) Server issues a "response"

- RESPONSE == textual message defined by HTTP
- Contains status code. Ex: 404 ("*Not Found*"), 200 ("*Okay*"), 500 ("*Internal Server Error*")

(3) Cycle repeats itself

HACK YALE

KEY TECHNOLOGIES

WEB DEVELOPMENT 101

THE CLIENT-SERVER MODEL

REQUEST FOR

- HTML, CSS, JS
- IMAGE
- DOCUMENT
- ETC...

CLIENT

RESPONSE FROM

SERVER

- REQUESTED CONTENT... OR
- SOME OTHER REASONABLE RESPONSE BASED ON CONTEXT (EX. 404 NOT FOUND)

HACK YALE

ON THE CLIENT SIDE

REQUEST FOR

- HTML, CSS, JS
- IMAGE
- DOCUMENT
- ETC...

RESPONSE FROM

CLIENT

SERVER

HTML
CSS
JAVASCRIPT

- REQUESTED CONTENT... OR
- SOME OTHER REASONABLE RESPONSE BASED ON CONTEXT (EX. 404 NOT FOUND)

HACK YALE

ON THE SERVER SIDE

REQUEST FOR

- HTML, CSS, JS
- IMAGE
- DOCUMENT
- ETC...

RESPONSE FROM

- REQUESTED CONTENT... OR
- SOME OTHER REASONABLE RESPONSE BASED ON CONTEXT (EX. 404 NOT FOUND)

CLIENT

HTML
CSS
JAVASCRIPT

SERVER

PHP
Javascript (with Node.JS)
Ruby (perhaps on Rails)
Perl
Java
Python

HACK YALE

THE FRONT-END IS HOW TO DISPLAY CONTENT FROM THE SERVER

CLIENT

HTML
CSS
JAVASCRIPT

SERVER

PHP, Node.JS, RoR, etc.

HACK YALE

WHOA WHOA, BACK UP

OKAY, LET'S WALK AND TALK THROUGH WHAT GOES ON BEHIND THE SCENES WHEN YOU VISIT A WEBSITE LIKE FACEBOOK

HACK YALE

OUR FIRST WEBSITE

AN EXERCISE

HACK YALE

ANATOMY OF A WEBPAGE

HEAD, SHOULDERS, KNEES AND FOOTER

FRONT-END DEVELOPMENT

An interaction between three “languages”

- HTML - HyperText Markup Language
- CSS - Cascading Style Sheets
- Javascript - A programming language, NOT Java

To be a successful developer, you’ll need to learn all three

HACK YALE

IMAGINE FOR A SEC...

THAT A WEBSITE IS A HUMAN BODY

HACK YALE

HTML

The “bones”

- The “content” of the Internet
- Builds the layout, structure and connections
- Easy to learn, easy to master

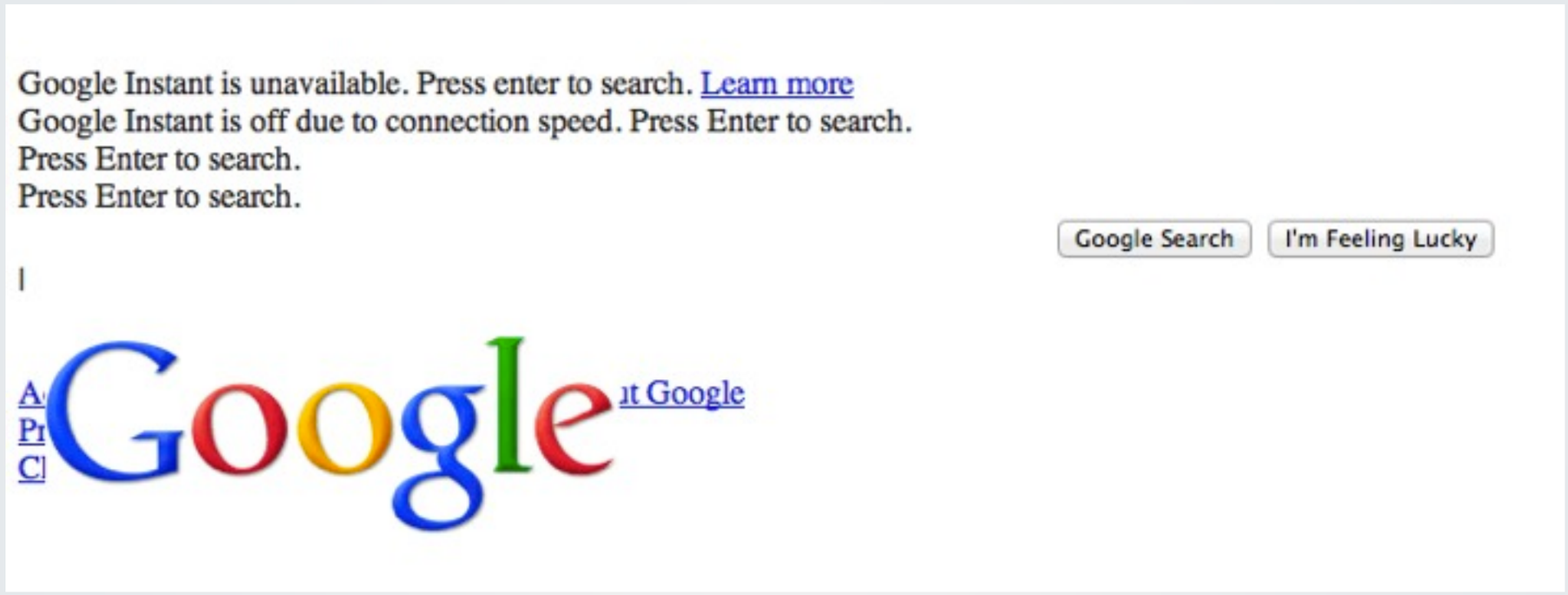
HACK YALE

THE HTML OF GOOGLE

IN A WEEK, YOU'LL BE ABLE TO MAKE THIS

HACK YALE

THE HTML OF GOOGLE



IN A WEEK, YOU'LL BE ABLE TO MAKE THIS

HACK YALE

CSS

The “skin” or “physical features”

- The “style” of the Internet
- Defines how HTML elements look
  - Width, height, color, position…
- Easy to learn, difficult to master
- The web designer’s best friend

HACK YALE

JAVASCRIPT

The “muscles”

- The “interaction” or “animation” of the Internet
- Makes HTML elements interact with one another
  - And with other pages!
- Tougher to learn, and unfortunately, tough to master

HACK YALE

HTML: IMPLEMENTATION

TYPE ALONG!

CODING IN HTML

To program the nested structure of HTML, we use *tags*

- Tags are just 1-4 letters that refer to something
  - Ex. “p” for “paragraph”; “h1” for “header 1”
- To differentiate tags from plain text, we enclose them in brackets
  - `<p>, <h1>`
  - This *opens* a tag
  - To close a tag, do `</h1>`

HACK YALE

MORE ABOUT TAGS

Tags delineate content blocks

- `<h1>` *I'm inside a tag!* `</h1>`
- Tags have "properties," and these properties are then passed on to the content within the tags
- Mr. `<tag>` says "abide by my laws until further notice"
- Then Ms. `</tag>` says "further notice"

TAGS CAN BE NESTED

```
<div>
    <h1>A sick header</h1>
    <div>
        <p>My sweet paragraph</p>
    </div>
</div>
```

This is how we can create boxes within boxes

HACK YALE

HTML DOCUMENT STRUCTURE

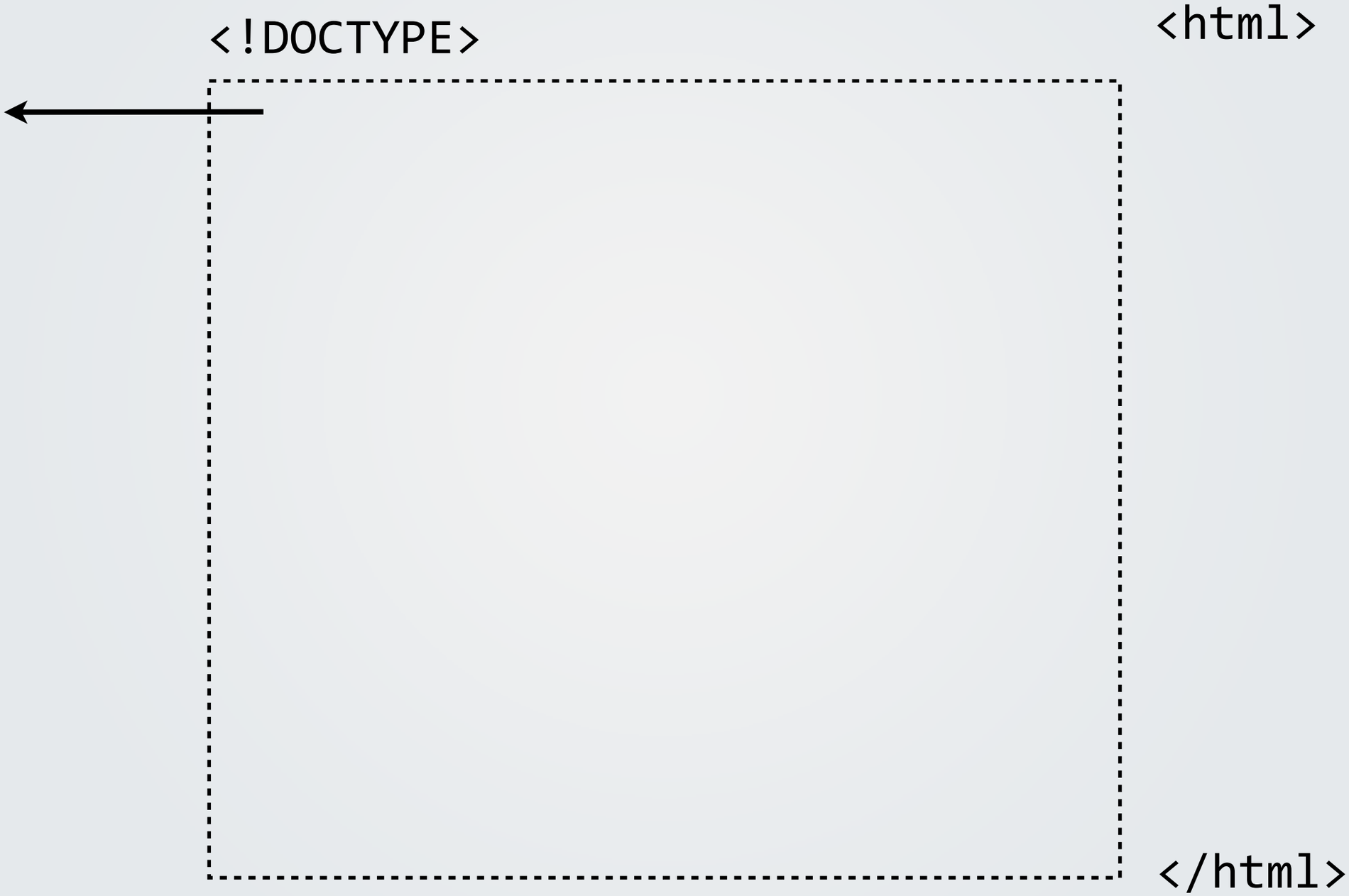
HACK YALE

HTML DOCUMENT STRUCTURE

<!DOCTYPE>

HACK YALE

HTML DOCUMENT STRUCTURE



HACK YALE

HTML DOCUMENT STRUCTURE

HACK YALE

HTML DOCUMENT STRUCTURE

HACK YALE

POPULAR TAGS

`<div>` (“division”) the content blocks of HTML (an empty shell)

```
<div>This content will be in an div block!</div>
```

`<a>` (“anchor”) links

```
<a href=”http://www.hackyale.com”>Home</a>
```

`<img>` images

```
<img src=”http://www.hackyale.com/logo.png” />
```

`<p>` paragraphs

```
<p>This text will be in a nice paragraph</p>
```

HACK YALE

CASCADING STYLE SHEETS (CSS)

HACK YALE

SO YOU'VE BUILT A SITE...

But it's ugly. Not to be mean, but by that I mean:

- Aside from images, not much color
- Not much control of where to put the "boxes"
- Spacing and alignment may not be ideal
- Not too responsive: hovers and clicks don't do anything

HACK YALE

THAT'S WHY WE HAVE CSS!

Style specifies how your HTML elements look

- We want to **select** some elements
- And apply a **style** to them
  - like color, size, alignment...

HACK YALE

WAYS TO SELECT ELEMENTS

By tag name:

- Very easy, just `p`, `img`, `div` or what have you
- Very rarely used, and often to define a global 'feel' for the site
  - For example, make all links red instead of the default blue
  - Make all images have a border
- But what happens when you don't want *all* of the p tags?

HACK YALE

WAYS TO SELECT ELEMENTS

By class name:

- A **class** is an attribute given to one or more HTML elements
  - `<p class="blue-text">Some text</p>`
- Many elements may have the same class
- An element may have many classes
  - Just separate with spaces!
  - `<p class="blue underline">More text</p>`

HACK YALE

WAYS TO SELECT ELEMENTS

By **ID**:

- An ID is an attribute given to **only one** element
  - `<p id="my-only-red-paragraph">ELMO</p>`
- Only one element may have a particular ID
- An element may only have one ID

Why IDs and not classes?

HACK YALE

PRACTICAL USE CASES

Tag name

- To override defaults or define global styles

Class

- The most widely used, allows for nice element selection

ID

- HackYale discourages using IDs in CSS selectors
  - Some other people do too

HACK YALE

CSS SYNTAX

```
p {
  font-size: 18px;
  color: blue;
}
```

HACK YALE

CSS SYNTAX

```
p {
  font-size: 18px;
  color: blue;
}
```

SELECTOR

CSS SYNTAX

```
p {
  font-size: 18px;
  color: blue;
}
```

SELECTOR

DEFINITION BLOCK

HACK YALE

CSS SYNTAX

```
p {
  font-size: 18px;
  color: blue;
}
```

SELECTOR

DEFINITION BLOCK

PROPERTY

HACK YALE

CSS SYNTAX

```
p {
  font-size: 18px;
  color: blue;
}
```

SELECTOR

DEFINITION BLOCK

PROPERTY

VALUE

HACK YALE

CSS SYNTAX

```
p {
  font-size: 18px;
  color: blue;
}
```

SELECTOR

DEFINITION BLOCK

PROPERTY

VALUE

END OF DEFINITION

HACK YALE

JAVASCRIPT

THE CONCEPT

HACK YALE

A PROGRAMMING LANGUAGE

But what does that mean?

Why do we program or write code?

To what end?

HACK YALE

WHAT IS PROGRAMMING?

PROGRAMMING IS THE PROCESS OF
WRITING *functions* THAT TELL COMPUTERS
TO DO WHAT WE TELL THEM TO DO

HACK YALE

FUNCTIONS

A FUNCTION IS A SET OF INSTRUCTIONS THAT TAKE SOME INPUTS AND PRODUCE A PREDICTABLE OUTPUT

EXACTLY LIKE IN ALGEBRA!

HACK YALE

CONCEPT

By defining and using functions, we can tell computers things we want to do. For example...

- Double a number
- Make all text on a web page double its current size
- Make an image change when a user clicks on it
- ...everything a computer does is implemented by a function

HACK YALE

VARIABLES

Sometimes, we want functions to have different outputs when we have different inputs.

- In algebra, you *solve* for a variable.
- But in programming, you get to assign variables values so that we can use those values later.

HACK YALE

VARIABLES AND FUNCTIONS

IF YOU UNDERSTAND THESE TWO CONCEPTS,
YOU'VE CLEARED A MAJOR HURDLE IN
UNDERSTANDING PROGRAMMING

HACK YALE

JAVASCRIPT SYNTAX

HACK YALE

FUNCTIONS

In algebra, we write $f(x) = \ldots$

- f is the name of the function
- x is the input
- … denotes the actual instruction of the function (ex. $x+1$)
- $f(x)$ is shorthand for the output (sometimes called y)

- I hope this sounds familiar.

HACK YALE

FUNCTIONS

In Javascript we write *f = function(x) { ... };*

- *f* is the name of the function
- *x* is the input
- ... denotes the instructions of the function (ex. **return *x+1;***)
- *f(x);* is still the output

- I hope this is still familiar!

HACK YALE

VARIABLES

Only kind of like algebra

- But instead of finding the values of variables, we use variables to **store values**.

- `a = 7;`
- **`a = a + 7;`**
- In algebra, the second statement is impossible. In programming, it *happens all the time*.

HACK YALE

JAVASCRIPT

A BRIEF HISTORY

HACK YALE

Are JavaScript and Java the same thing?

HACK UNIVERSITY

“Are JavaScript and Java the same thing?”

NO

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?”

HACK UNIVERSITY

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

Sort of...

HACK UNIVERSITY

“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.

HACK UNIVERSITY

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

HACK YALE

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' ) {...}`

HACK YALE

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)

HACK YALE

BUT IT AIN'T BAD NEITHER!

Appreciation for JavaScript has grown in recent years thanks to several key figures

- Douglas Crockford - YUI, Javascript: The Good Parts
- John Resig - jQuery
- Jeremy Ashkenas - Backbone.js
- Ryan Dahl / Joyent - Node.js

HACK YALE

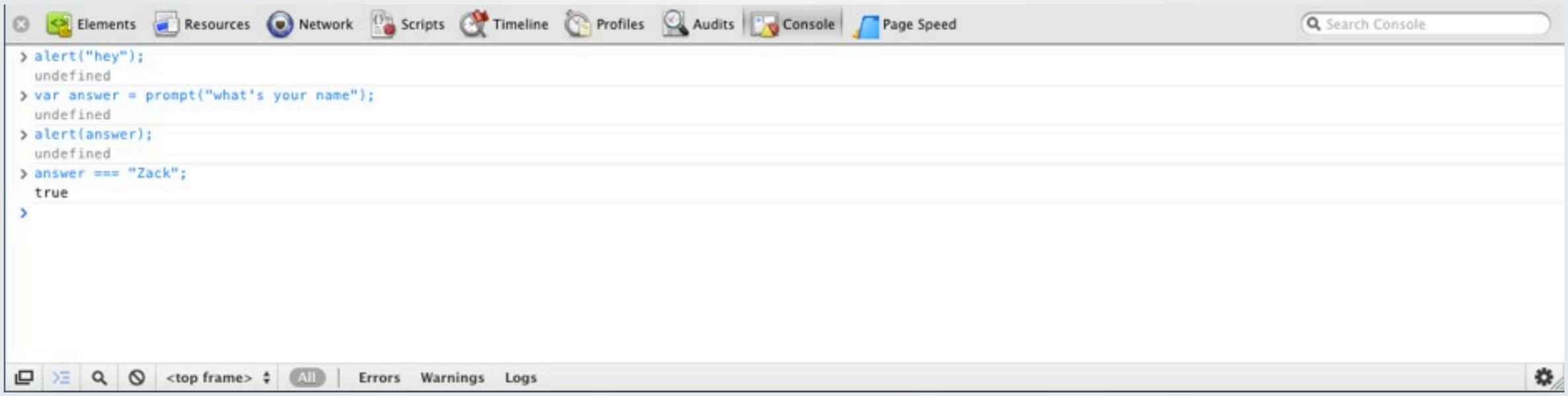
DEMO

FIRE UP THE CONSOLE

HACK YALE

CHROME CONSOLE

command + option + j
or
view -> developer -> javascript console



HACK YALE

VARIABLE SYNTAX

```
var a = 7;
```

- var tells the computer that a is variable
- = means to **store**. In this case, the number 7
- All Javascript statements end in a semicolon.

HACK YALE

VARIABLE SYNTAX

```
a = a + 7;
```

- Don't need to resay that a is a variable.
- = means to **store**. In this case, the number 7
- All Javascript statements end in a semicolon.

HACK YALE

OTHER EXAMPLES

We covered a lot of examples in class, that I'll list here.

- `var f = function(x) { return 2 * x };`
- `var numCatz = 7;` vs just `numCatz = 7;`
- `var numDogz = "7";` and `numDogz + numCatz;`
- `typeof numCatz;` and `typeof f;`
- `var g = function(func) { return func(2); };`
- `var ps = document.getElementsByTagName("p");`

HACK YALE

THE CONCEPTS

The concepts covered in the examples (respectively)

- Defining functions
- The use of the keyword ‘var’
- The difference in addition between string and numbers
- Types in Javascript (‘var’ is not a type. A variable only gets a type when it is assigned a value). Functions are a type!
- Functions can be inputs to other functions
- Getting elements on your HTML page (in this case, all paragraphs)

HACK YALE

UNDERSTANDING THE CONCEPTS

It is *incredibly* important that you understand all of the previous concepts, even if you have programmed before. Javascript does things a bit differently than C or Java, and when we start talking about more advanced things (like callbacks and the Event Loop), we're going to take these things as a given. Use Google or come to office hours...it's just 6 concepts!

HACK YALE

INCLUDING JAVASCRIPT

HACK YALE

SIMILAR TO CSS

```
<!DOCTYPE html>
<html>
<head>
   <!-- some HTML -->
</head>
<body>
    <!-- some more HTML -->
    <script src="main.js"></script>
</body>
</html>
```

HACK YALE

WHAT ARE THE DIFFERENCES?

```
<!DOCTYPE html>
<html>
<head>
   <!-- some HTML -->
</head>
<body>
    <!-- some more HTML -->
    <script src="main.js"></script>
</body>
</html>
```

HACK YALE

CSS

```
<html>
<head>
    <!-- some HTML -->
    <link rel="stylesheet" href="/style.css">
    <!-- some more HTML -->
</head>
<body>
    <!-- even more HTML -->
</body>
</html>
```

JAVASCRIPT

```
<!DOCTYPE html>
<html>
<head>
   <!-- some HTML -->
</head>
<body>
    <!-- some more HTML -->
    <script src="main.js"></script>
</body>
</html>
```

HACK YALE

LEARNINGS

FIRE UP THE CONSOLE

HACK YALE

STRAIGHTFORWARD

One mistake and it was clear

- Have workshops for esoteric topics, not full classes.
- Focus on the products.
- Be lively.

HACK YALE

LESS OBVIOUS

Some non-obvious, subtler lessons

- Have class when people *aren't* working.
- Avoid insecurity.
- Prepare to improvise.

HACK YALE

STILL WORKING ON

Some things we haven't quite figured out yet

- Ensuring long-term commitment.
- Biting off the right amount.

HACK YALE

HOMEWORK

YOU THOUGHT WE WERE DONE

HACK YALE

HOMEWORK

Begin your education

- Complete project 1 at https://dash.generalassemb.ly

HACK YALE

THANKS!

HACK YALE

HACK YALE

QUESTIONS? ZACK@HACKYALE.COM

WWW.HACKYALE.COM