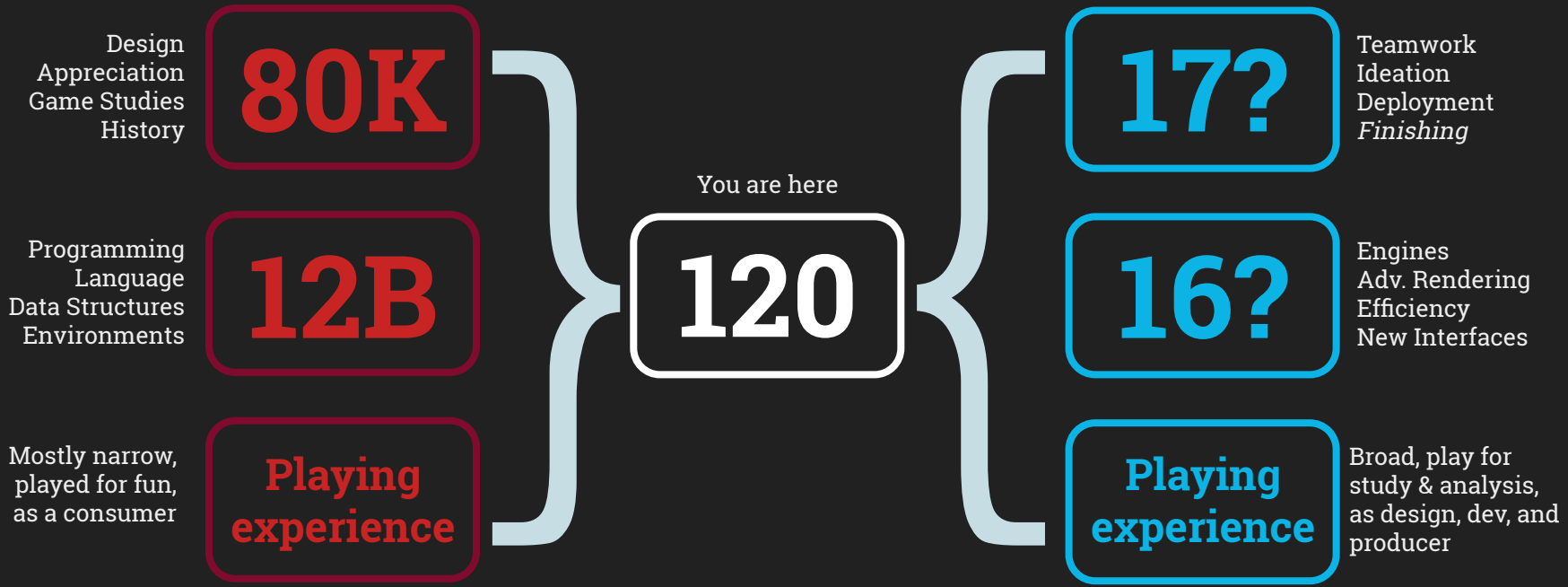
CMPM 120

Game Development Experience

Where you are



Why You Are Here

- Learn the basic principles of game programming and put them into practice
- Learn how to do the low-level implementation so we can turn ideas into working games
- Learn how technology *and* teamwork affect game design (PLO 7 & 8)

The Team

Isaac Karth

- he/him
- Teaching CMPM 120
- ikarth@ucsc.edu
- Computational Media PhD student (I do procedural generation stuff)

Tad Leckman

- Teaching AGPM 120

Who are you?

If you think of yourself as a programmer:
What is one artwork that you like?

If you think of yourself as an artist:
What is one programming element that you like?

Who are you?

If you think of yourself as a programmer:

Albrecht Dürer, *Young Hare*

If you think of yourself as an artist:

```
(map (reduce '(:plus-python [list-comprehension]))
```

Some Bad News

Programming Videogames is Difficult

Especially in 10 weeks,
in Summer,
and working in a team.

```
var Environment = {
    code: [ "HTML5", "CSS", "JavaScript"],
    framework: "Phaser",
    collaboration: [ "git", "GitHub" ],
    editor: [ "Sublime Text", "Atom", "Chrome", "etc" ],
    server: "Python"
}
```

NEWS LEARN DOWNLOAD EXAMPLES COMMUNITY PLUGINS

Desktop and Mobile HTML5 game framework

A fast, free and fun open source framework for Canvas and WebGL powered browser games.

DOWNLOAD & GET STARTED
Download or Fork via Github

3.18.1

PHASER FEATURES

- WEBGL & CANVAS
- PRELOADER
- PHYSICS
- SPRITES
- GROUPS
- ANIMATION
- PARTICLES
- CAMERA

- INPUT
- SOUND
- TILEMAPS
- DEVICE SCALING
- PLUGIN SYSTEM
- MOBILE BROWSER
- DEVELOPER SUPPORT
- WEB FIRST

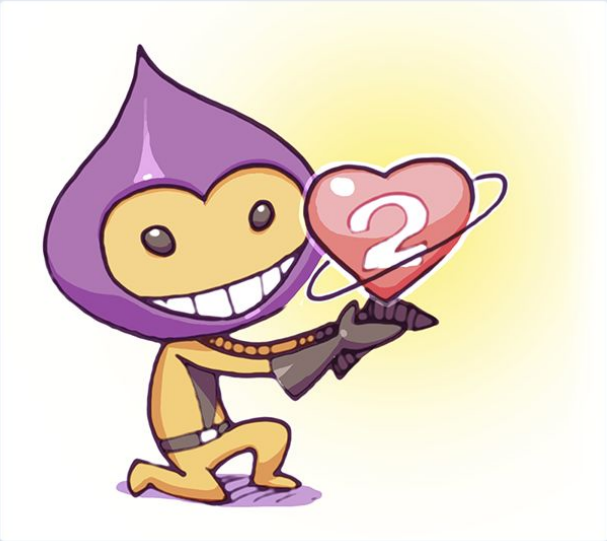
Why Phaser?

- Free!
- Fast
- Because I said so!
- Actively supported and documented
- Well-structured and (generally) genre agnostic
- Gives us lots of game-specific functions “for free” (e.g., game loop, state management, physics, input handling, etc.)
- Lots and lots and lots of community resources
- Used in high-quality, actual, real-life, professional games
- Will help you learn to love again despite the deep void in your heart

Ten years ago, we would have used Flash

Nathan told me this, so it must be true

Phaser CE - Community Edition



Phaser CE is the Community Edition of Phaser. In short, it's a version of Phaser that you, the community, have direct control over. We started it off with the 2.7.0 release and then handed it over to you.

Phaser 2 was a massive milestone for us, and we're still constantly amazed at all the cool things you've created, and continue to create with it. Thank you to everyone who has

Why not Phaser 3?

- Probably *will* be using 3 next Spring
  - Nathan is in the process of updating his slides
- A few major features are still not in place
- The tutorials are in the process of being converted
- Phaser CE is not "worse" than Phaser 3
- Phaser CE is still actively supported
  - Don't underestimate stability
- This is not the last time that you will encounter this situation in your career

There is no perfect...

...no perfect language

...no perfect framework

...no perfect engine

The sooner you learn this, the better.

You should be grateful
I'm not forcing you to
learn ClojureScript.

Your game's scope:
small

Your game's scope:

tiny

⇦ No


Thrak the Rat
So you're gonna murder me, are you? Shrakh, I say! Just try it!

⇦ No

No

No

No


WORLD WHERE
HOUSE-BUYING JOBS
BECAME RENT-PAYING
JOBS
BECAME LIVING WITH
FAMILY JOBS

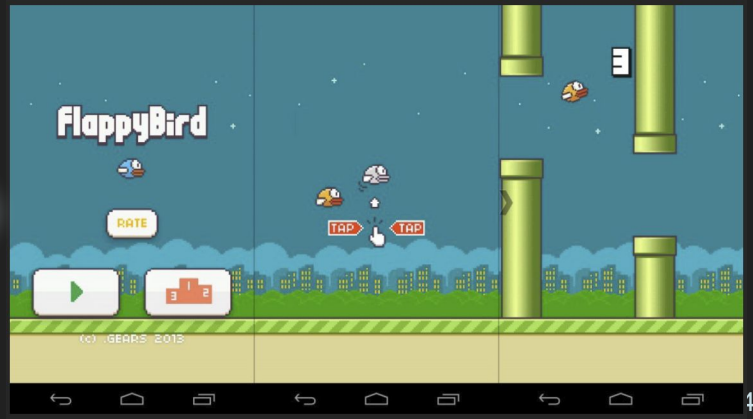
No

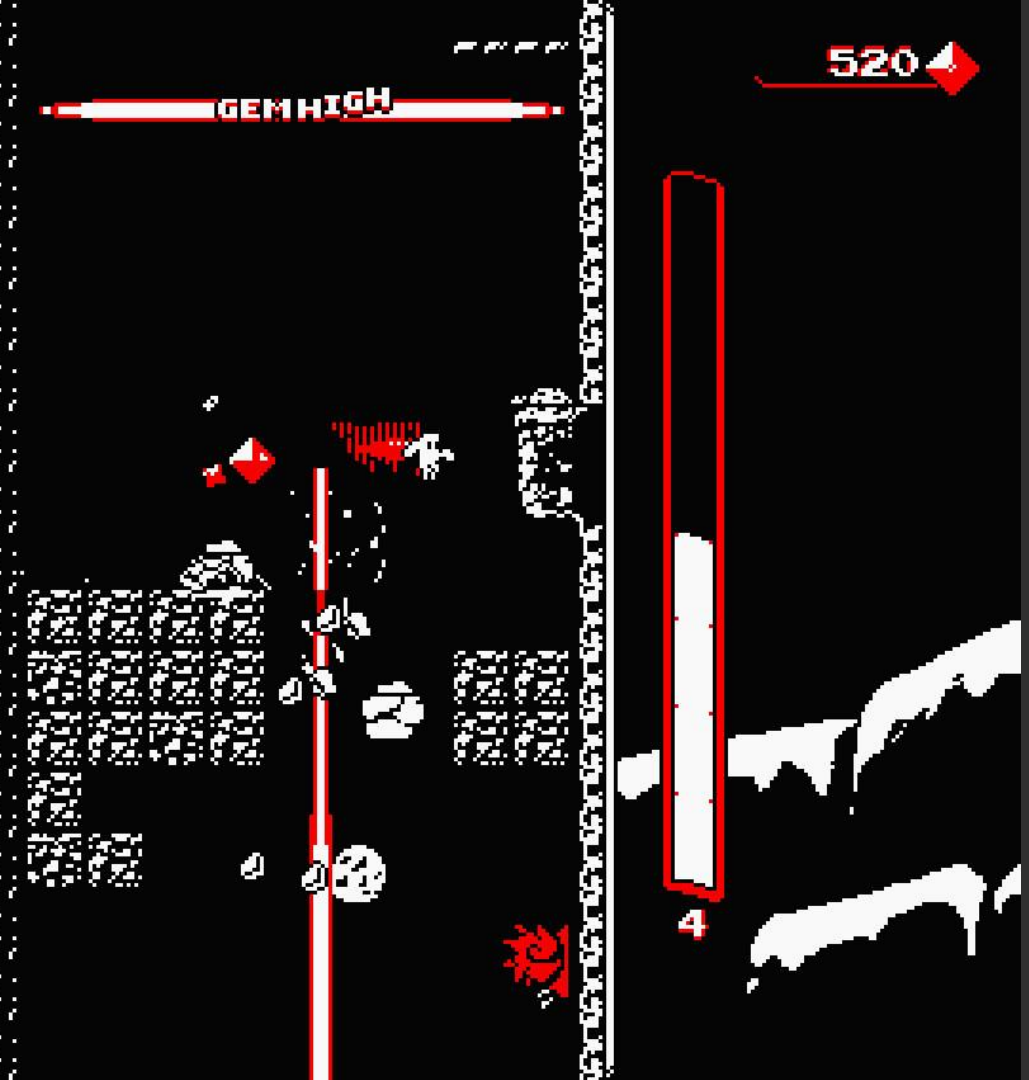
No

No

No

Yes





Ambitiously small

Mechanic-centric

Well-structured

Expressive

Idea-driven

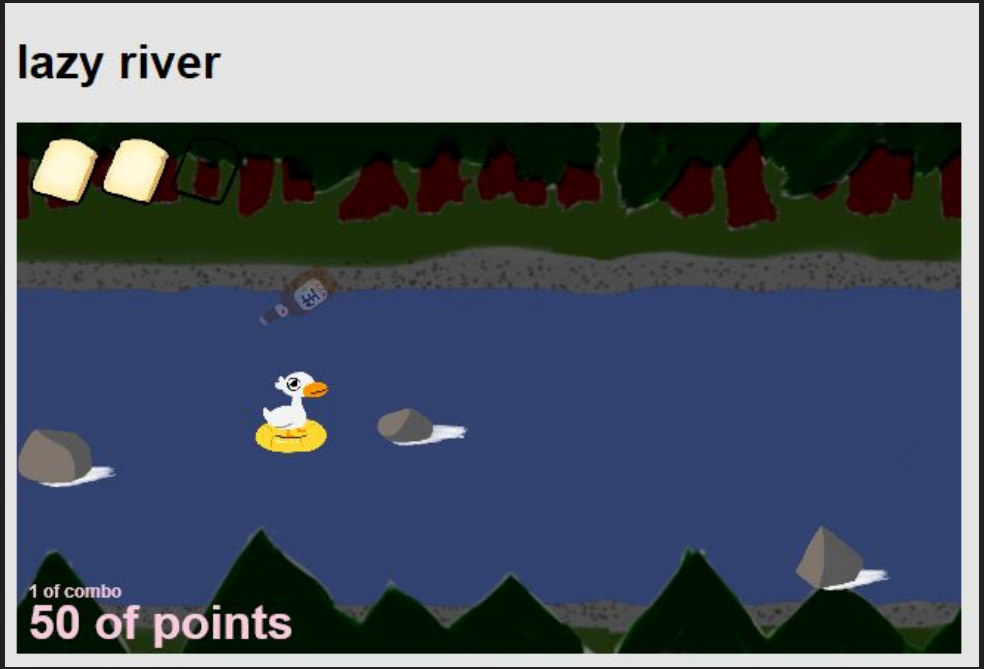
Achievable

2d :)

Some Good News

Every one of you can program a videogame

(And we have proof)



Lazy River: an endless drunker

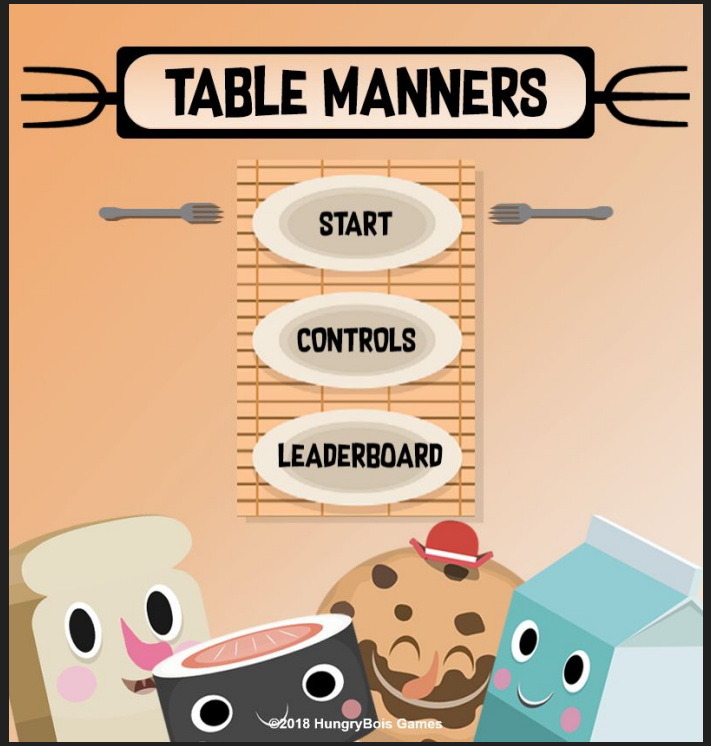
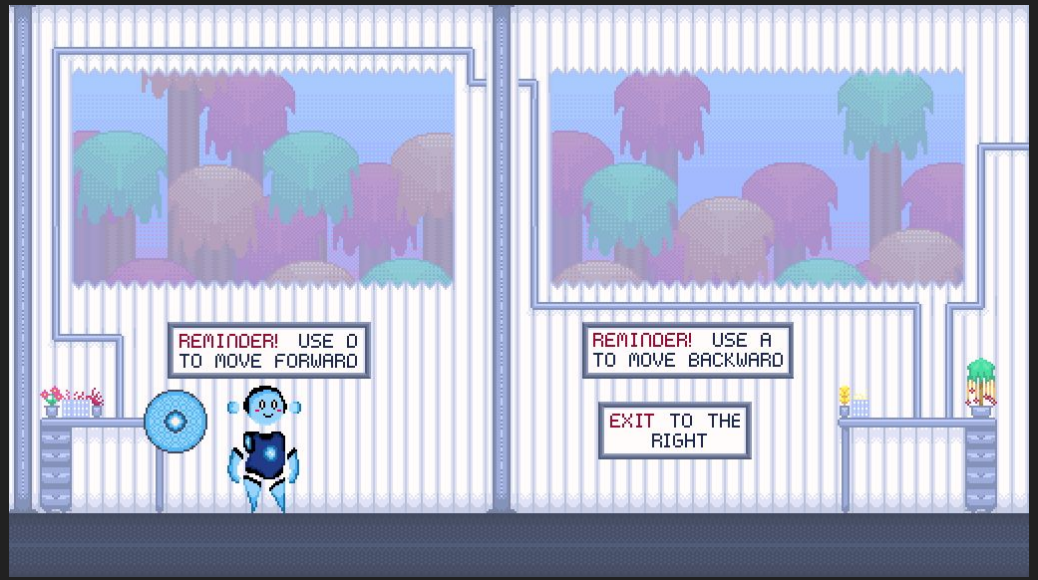
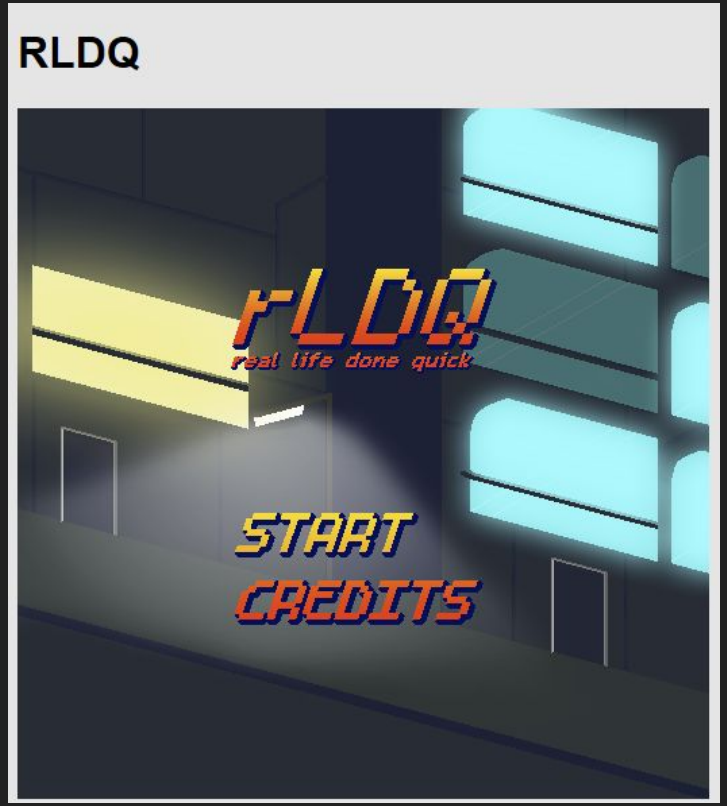


Table Manners: a game about stealing sushi



Gravobot: a 2D puzzle platformer with a gravity orb mechanic

Tiny Steps: a storybook game wherein you play as the CUTEST mouse



rLDQ: a minigame collection of mundane daily tasks

The Light: a survival typing game (!?)

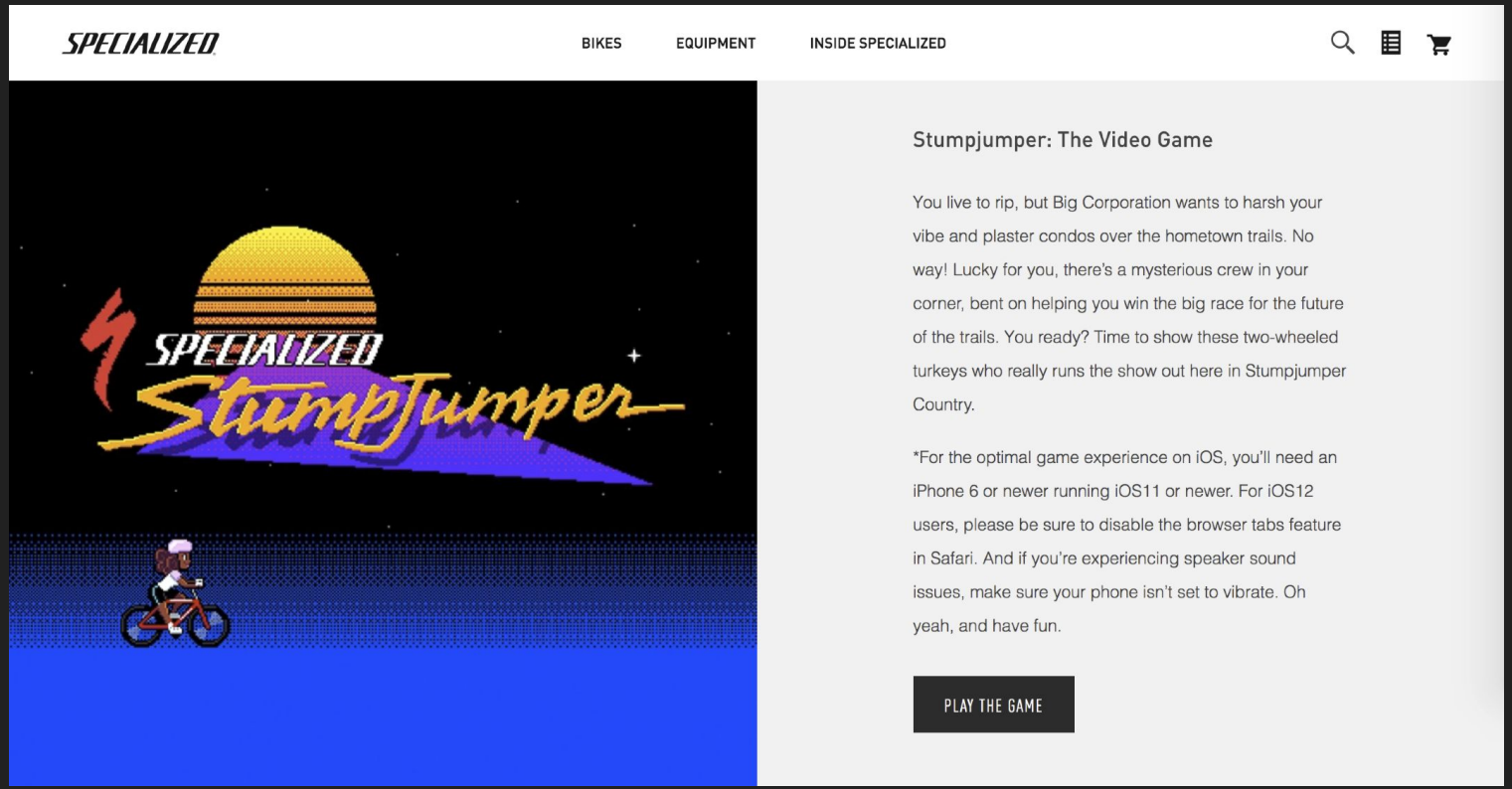
Some More Good News

Any style or genre you choose is fine with us*

*As long as you do so with creativity, thoughtfulness, and professionalism

The Runs: a very mature game

StumpJumper: a commercial UCSC game



Team of six, 10-minute game, 6 months, $25K Budget

Foreshadowing Your Future

Schedule Overview*

Week 1: Intro, Web Dev, Phaser Intro
Week 2: Loops, States, Assets, Pong
Week 3: JavaScript, Prefabs, Input, Collision
Week 4: Debugging, Camera, State Machines
Week 5: Tilemaps, P2 Physics, Runner Showcase
Week 6: Particles, git, Time
Week 7: Text, Fonts, CSS
Weeks 8–9: Audio, Guest Talks, Open Topics
Week 10: Final Presentations aka The TRIAL of WILL

*This will inevitably change a bit

There are no labs in the summer

However, I will have office hours:

Engineering 2, room 256

- ➔ Wednesday, 10am - 11am
- ➔ Thursday, 11am - 12pm

Canvas Resources

Syllabus: https://canvas.ucsc.edu/courses/26569

Policies & Expectations

Attendance

Respect your classmates' time

Attendance is mandatory for presentations

Lateness

This is a fast class and you need to stay on track.

Presentations need to happen in person

Devices

Respect the time and attention of those around you.

(And respect yourself too.)

Readings

Designed to help you learn the material

Slides & Source Code

These will be available in our class Drive folder a day or two after class

Collaboration & Help

You are encouraged to help each other,
but don't shortcut your own programming.

The Citation Model

Link to your sources!

I trust you to program ethically and responsibly

Grading

25% Readings & Small Assignments

25% Endless Runner Project

50% Final Game

Disability Resources

Please let me know how I can help

Communication

Talk to us early and often.

Don't let problems pile up.

Respect & Honesty

A reciprocal operation.

We're excited you're here!

Any questions?

Your first reading assignment

"Learning Web Design"

There's a quiz.

It is supposed to be easy if you know the subject,
and helpful if you don't.

You can retake it as many times as you want.

It is due by the end of Friday

Modern Web Development

HTML + CSS + JS

indeed

What: front end developer
Where: Santa Cruz, CA
Find jobs | Advanced Job Search

front end developer jobs in Santa Cruz, CA

My recent searches
front end developer - Soquel, CA
» clear searches

Sort by:
relevance - date

Distance:
within 50 miles

Salary Estimate
$100,200+ (861)
$115,000+ (673)
$120,000+ (582)
$130,000+ (356)
$140,000+ (172)

Job Type
Full-time (962)
Contract (87)
Internship (24)
Part-time (13)
Temporary (9)
Commission (3)

Location
Sunnyvale, CA (161)
San Jose, CA (153)
Santa Clara, CA (116)
Palo Alto, CA (98)
Mountain View, CA (96)
+ more »

Company
Walmart eCommerce (35)
Apple (34)
Walmart (20)

Software Engineer, Front End
Google ★★★★☆ 3,009 reviews
Mountain View, CA
Experience with front end technologies and/or front end frameworks. As a Front End Software Engineer at Google, you will specialize in building responsive and...
save job

Front End Engineer
Amazon.com Services, Inc. ★★★½☆ 34,428 reviews
Cupertino, CA
You will design, build, and operate web services and web front end at scale, interface with front-end and back-end teams, and deliver the plumbing in-between....
save job

Front-End JavaScript Developer (Contract)
LG Electronics ★★★★☆ 2,069 reviews
Santa Clara, CA
Front-End JavaScript Developer. Advanced Platform Lab of LG ARC (America R&D Center) in Santa Clara, CA, is looking for passionate and talented JavaScript &...
3 days ago save job

Front End Developer
BlockRules
Fremont, CA
As a Front End Developer, you will work on development and implementation of features such as smart contracts, integration of crypto-exchanges, and registration...
Easy apply
save job

Sr Frontend Engineer (Remote)
Noom Inc. ★★★★½ 8 reviews
United States
$100,000 - $150,000 a year
At Noom, we use scientifically proven methods to help our users create healthier

Software Engineer, Front End

Google ★★★★½ 3,009 reviews - Mountain View, CA

Apply On Company Site

In school or graduated within last 9 months? We encourage you to apply to openings on the Student Jobs site

Our engineering teams bring the best of Google to our users around the world with high-impact, innovative work. Learn about career opportunities in Bangalore and Tokyo today.

Note: By applying to this position your application is automatically submitted to the following locations: **Mountain View, CA, USA; San Francisco, CA, USA; San Bruno, CA, USA; Sunnyvale, CA, USA**
Minimum qualifications:

- BA/BS degree in Computer Science or related technical field or equivalent practical experience.
- 1 year of work experience.
- Experience in JavaScript, and one or more programming languages including but not limited to: Java, C/C++, Python or Go.
- Experience with front end technologies and/or front end frameworks.

Preferred qualifications:

- Experience developing user-facing software.
- Experience with the latest and greatest web standards, including HTML5 and CSS3.
- Knowledge of web libraries and frameworks such as AngularJS, Polymer, Closure or Backbone.
- Strong sense of web design and attuned to the fundamentals of user experience.
- Familiarity with the whole web stack, including protocols and web server optimization techniques.
- An understanding of the principles of accessibility and can build products that are accessible to users with disabilities.

About the job

Google's software engineers develop the next-generation technologies that change how billions of users connect, explore, and interact with information and one another. Our products need to handle information at massive scale, and extend well beyond web search. We're looking for engineers who bring fresh ideas from all areas, including information retrieval, distributed computing, large-scale system design, networking and data storage, security, artificial intelligence, natural language processing, UI design and mobile; the list goes on and is growing every day. As a software engineer, you will work on a specific project critical to Google's needs with opportunities to switch teams and projects as you and our fast-paced business grow and evolve. We need our engineers to be versatile, display leadership qualities and be enthusiastic to take on new problems across the full-stack as we continue to push technology forward.

Do you want to help Google build next-generation web applications like Inbox, Gmail, Google Search, Google Maps and

Front End Developer

Apple ★★★★☆ 5,034 reviews - Santa Clara Valley, CA

Apply On Company Site

Apple is seeking a Front-End Developer to drive user experience innovations for apple.com. This developer will not only be responsible for defining the architectural strategy for front-end technologies, including HTML5, CSS3 & JavaScript, but for evangelizing that technology across the team and Apple as a whole.

Key Qualifications

- Comfortable with source version control software and package managers (SVN, Git, NPM)
- Well-versed in fundamental visual and interactive design discipline
- Strive to use web standards to build solutions using semantic markup, templates (Handlebars) and SASS
- Understanding of all major browsers and the special considerations required for all various quirks
- Competent JavaScript programmer who doesn't need to rely on libraries to accomplish innovative interactions
- Aware of the interplay between JavaScript and HTML & CSS, and can dynamically create, modify, and style elements on a page with ease
- Experience with WebGL is a plus.

Description

Lead development efforts on large scale web-based projects, ensuring robust and lasting solutions are implemented Awareness of Apple's mobile platform with the ability to build solution that take advantage of the latest iOS features while remaining performant on the latest iOS devices Maintain existing JavaScript libraries: making sure they support the engineering and creative needs of apple.com Mentor team members: Educate on software development best practices and new technologies, especially HTML5 & CSS3 Innovate: Build things that people will blog and Twitter about

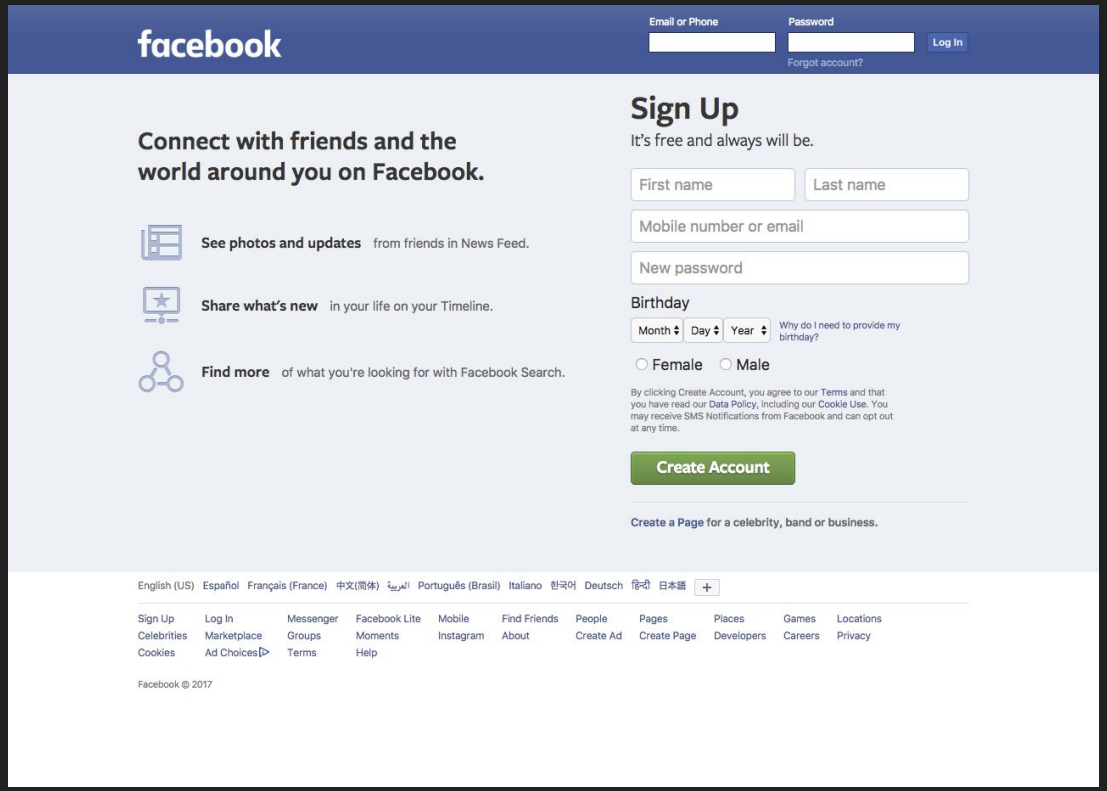
HTML: semantic layer
(how a page is described)

CSS: presentation layer
(how a page looks)

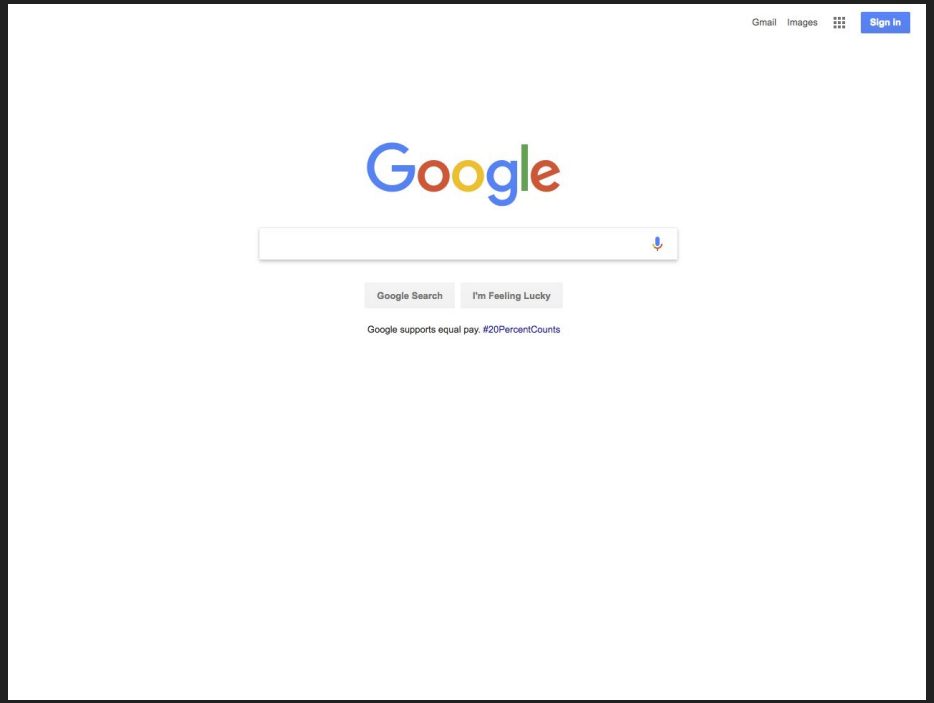
JavaScript: interaction layer
(how a page behaves)

Before we start...

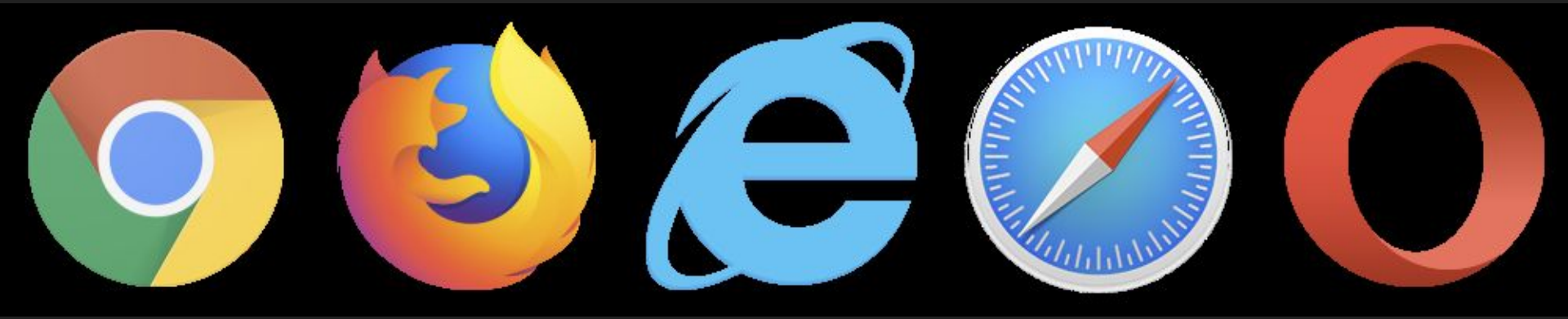
What the heck is the Internet?



On the Internet, but not actually *the* Internet...

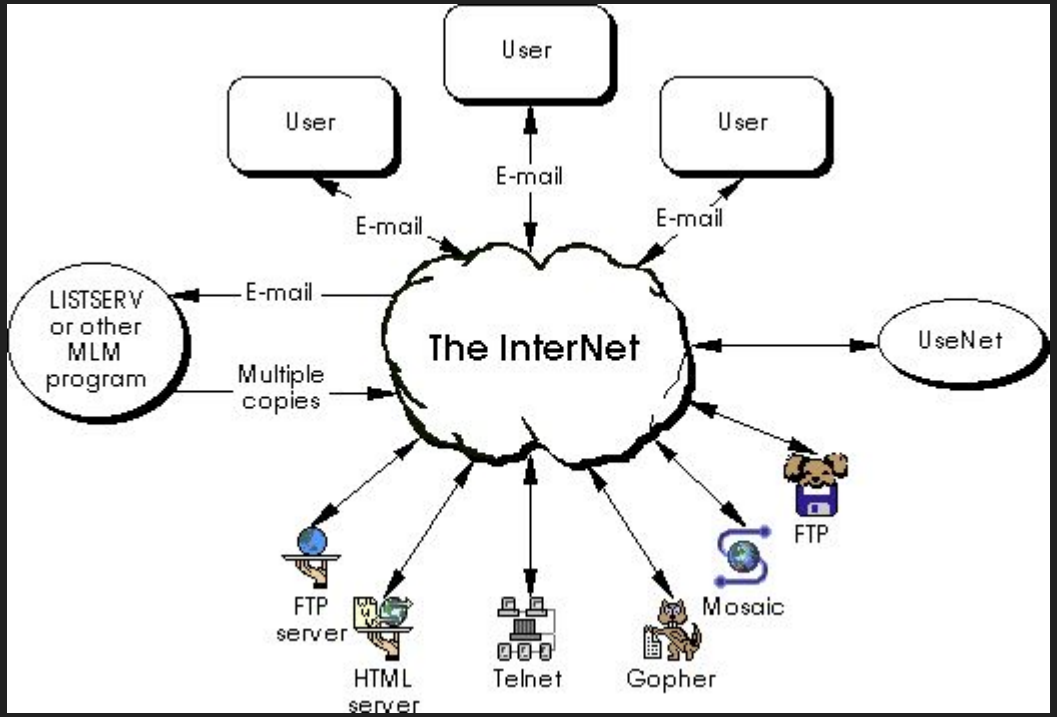


Also not the Internet...



Still not the Internet...

This *is* The Internet (but not the Internet we're talking about)

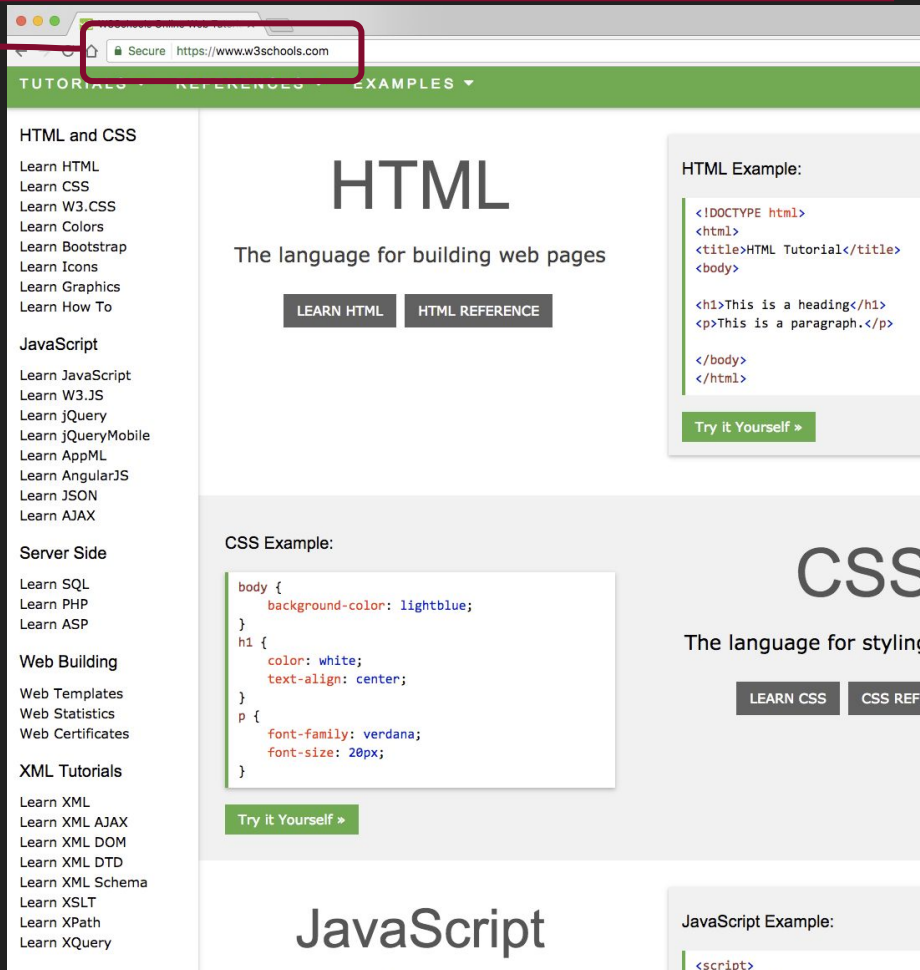


The Internet is a diverse network of connected computers that use a variety of standardized protocols to send and receive information

The World Wide Web is just one (of many) ways to send and receive information via the Internet.

It uses HTTP as its communication protocol, HTML as a language to describe and structure information, browsers to interpret HTML, and hypertext to link documents together

What's this stuff?

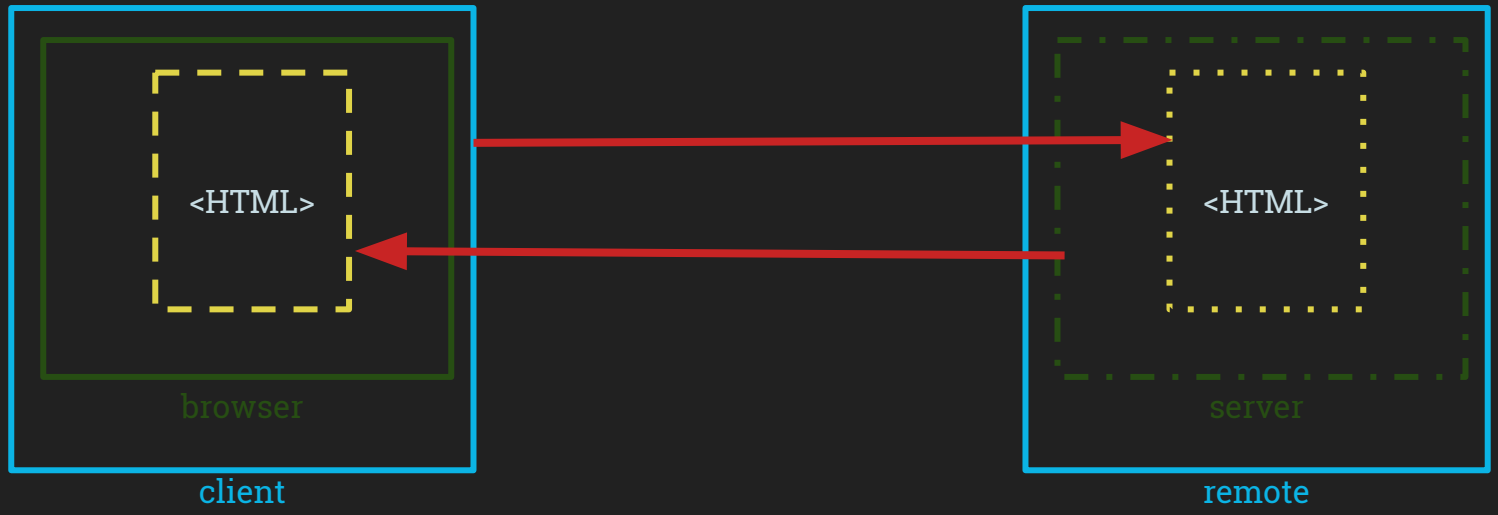


Uniform Resource Locator (URL)

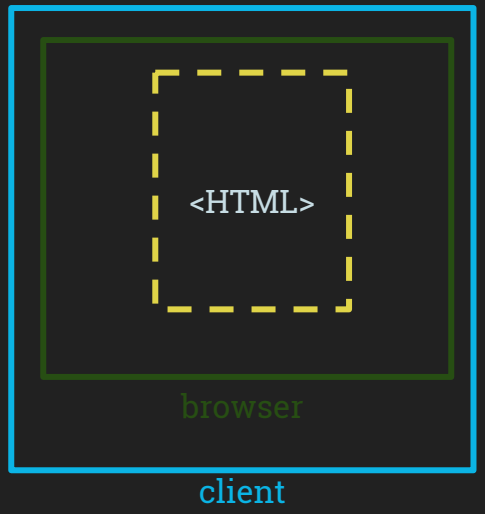
User Agent (browser)

protocol://subdomain.domain.tld:port-number/path?parameters

A basic model of how the web works



Client



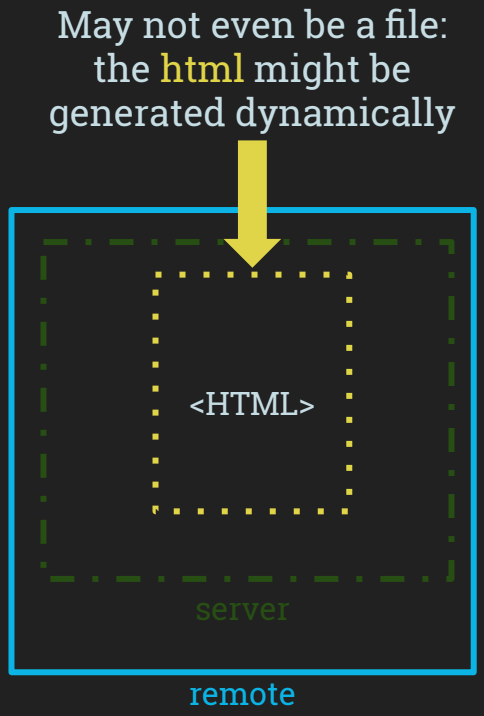
"Client-side" or "front end" applications run on our local machine

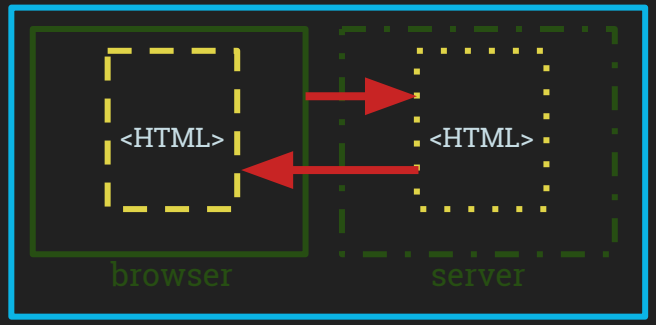
➔ Limited by the resources of the local machine

Server

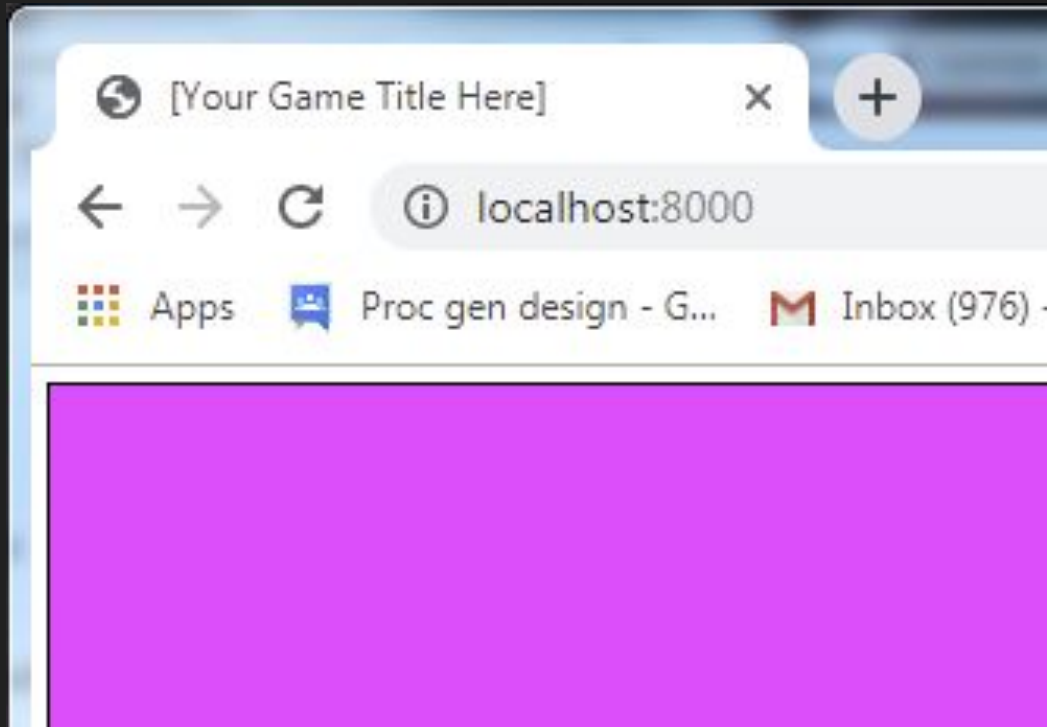
"server-side" or "back end" applications run on a remote machine

- ➔ Limited by the resources of the remote machine
- ➔ Are often virtual machines
  - Several can share a host machine
  - Or can be running on a cluster of host servers





You can serve websites locally



Python 2:

python -m SimpleHTTPServer

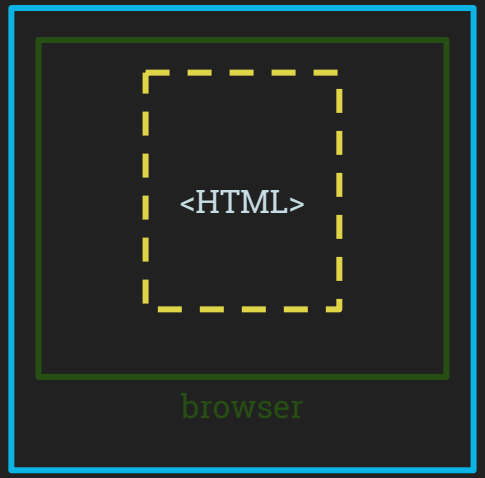
Python 3:

python -m http.server

If you don't have Python on your machine, install it.
https://www.python.org/downloads/
https://www.anaconda.com/distribution/

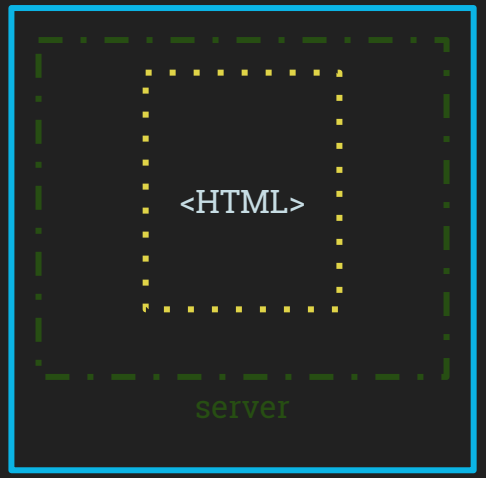
The easiest way to run a local server is with Python

Technologies



client

HTML
CSS
JavaScript



remote

PHP
Python
Ruby
Clojure
node.js

HTML = ???

HTML = Hypertext Markup Language

Nathan Altice
Santa Cruz, CA

Dear Mom,

I had a fun time at summer camp today. I stole a kid's swimsuit and set it on fire. He didn't mind because we are friends.

When you have the time, please send me a birthday cake in the mail. It isn't my birthday, but I like cakes with my name on them. Please do not send cake in a poster tube like last time.

OK, that's all for now. Please don't touch the things in my room.

Love,
Nathan

```
<address>
<name>
Nathan Altice
</name>
<location>
Santa Cruz, CA
</location>
</address>

<salutation>
Dear Mom,
</salutation>

<body>
I had a fun time at summer camp today. I stole a kid's swimsuit and set it on
fire. He didn't mind because we are friends.

When you have the time, please send me a <important>birthday cake</important>
in the mail. It isn't my birthday, but I like cakes with my name on them.
Please do not send cake in a poster tube like last time.

OK, that's all for now. Please don't touch the things in my room.
</body>

<closing>
Love,
</closing>
<signature>
Nathan
</signature>
```

Marking up text gives it structure and meaning

language

semantic

HTML uses standardized tags to markup text

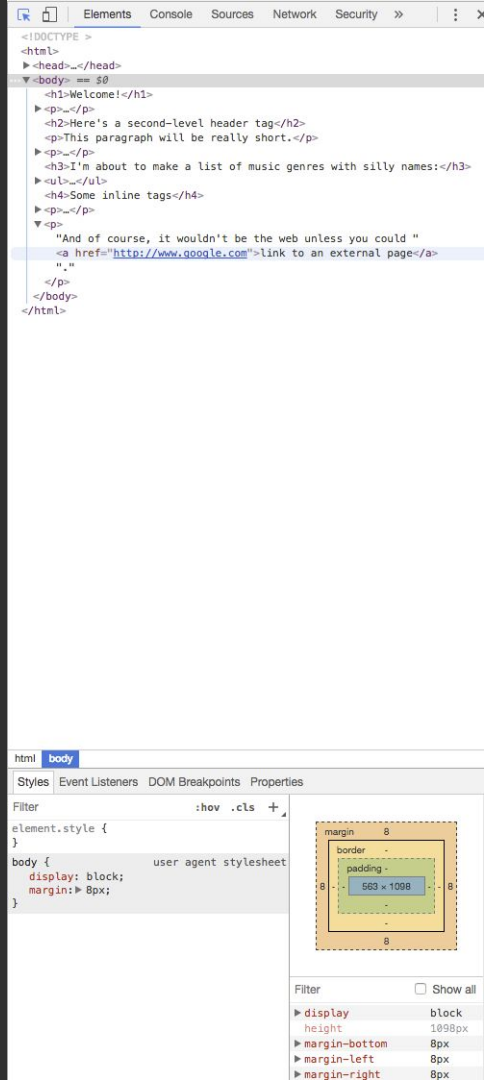
Tags provide semantic meaning to content

```
<!DOCTYPE>
<html>
</html>
```

All major browsers have developer tools

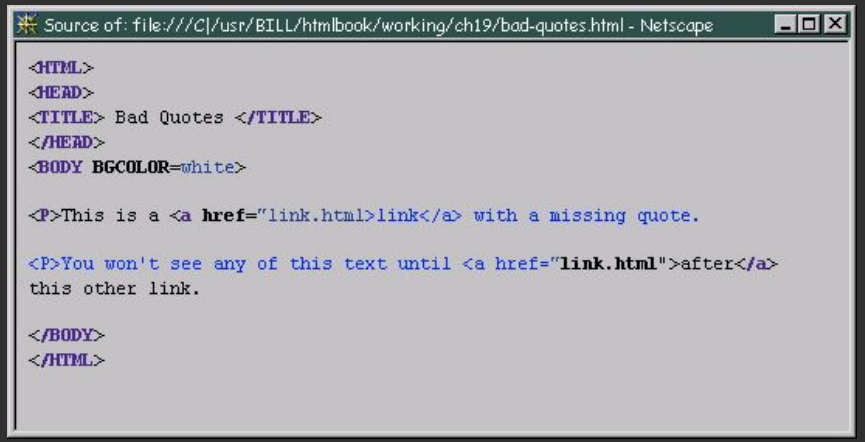
But every browser renders webpages slightly differently.

The BROWSER WARS! A period of civil war. Rebel open-source startups, striking from a hidden base, have won their first victory against the evil non-standards-compliant INTERNET EXPLORER...

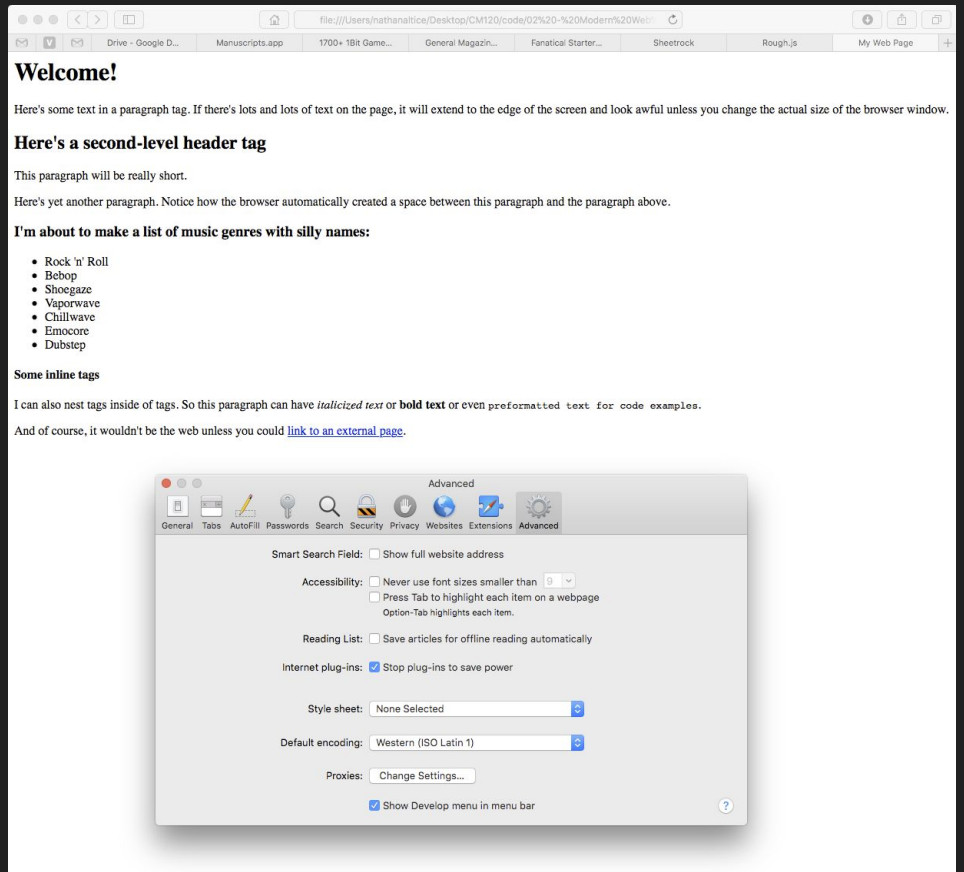


View Source

The 90s web designer's best friend



In Safari: Preferences > Advanced > Show Develop Menu...



Developer Tools Keyboard Shortcuts

```
Command+Option+I
OsX

F12 or Control+Shift+I
Windows
```

CSS = ???

CSS = Cascading Style Sheets

CSS: Let me sum up

- CSS separates presentation from structure
- CSS is a separate language with its own syntax
- CSS statements are called rules
- Rules contain a selector and a declaration
- Style rules "cascade" downward
- CSS definitions may live in a <style> tag (usually bad) or be linked externally (much better)

```css
body {
	font-family: "Arial";
	font-size: 14px;
	background-color: #facade;
}

h1 {
	border: 1px dotted red;
}

h2 {
	font-variant: small-caps;
}

.green {
	color: green;
}

/* h1 {
	font-family: serif;
	font-size: 5em;
} */
```

But what about...

What is **HTML5**?

MDN MOZILLA DEVELOPER NETWORK

Sign in

mozilla

WEB TECHNOLOGIES ▾ MOZILLA DOCS ▾ DEVELOPER TOOLS FEEDBACK ▾

MDN › Web technology for developers › Web developer guides › HTML developer guide › HTML5

LANGUAGES EDIT

HTML5

see all contributors

HTML5 is the latest evolution of the standard that defines HTML. The term represents two different concepts:

- It is a new version of the *language* HTML, with new elements, attributes, and behaviors,
- and a larger set of **technologies** that allows more diverse and powerful Web sites and applications. This set is sometimes called *HTML5 & friends* and often shortened to just *HTML5*.

Designed to be usable by all Open Web developers, this reference page links to numerous resources about HTML5 technologies, classified into several groups based on their function.

- *Semantics*: allowing you to describe more precisely what your content is.
- *Connectivity*: allowing you to communicate with the server in new and innovative ways.
- *Offline and storage*: allowing webpages to store data on the client-side locally and operate offline more efficiently.
- *Multimedia*: making video and audio first-class citizens in the Open Web.
- *2D/3D graphics and effects*: allowing a much more diverse range of presentation options.
- *Performance and integration*: providing greater speed optimization and better usage of computer hardware.
- *Device access*: allowing for the usage of various input and output devices.
- *Styling*: letting authors write more sophisticated themes.

<canvas>The part we care about</canvas>

An HTML element that allows us to draw graphics using scripting (i.e. JavaScript).

plugin

native

HTML5 provides a

target container

for our games

02-03-important.html

02-02-firstpage.html | 02-03-important.html

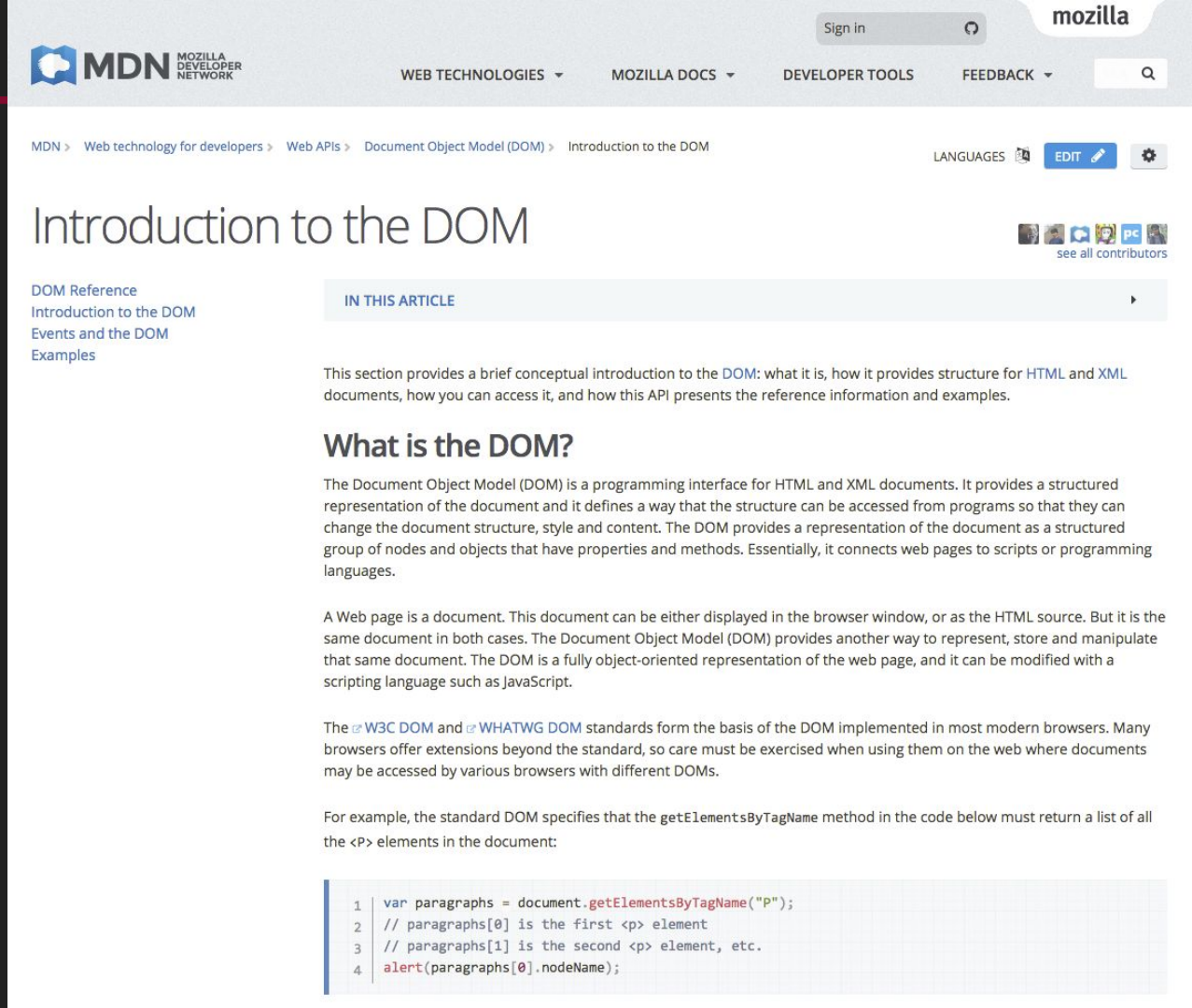
```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Page Name</title>
    <meta charset="utf-8">
    <style type="text/css">
        /* CSS goes here */
    </style>
    <link rel="stylesheet" type="text/css" href="css/styles.css">
    <script type="text/javascript" src="js/game.js"></script>
</head>
<body>

    <!-- content goes here -->

    <script type="text/javascript">
        // code goes here
    </script>
</body>
</html>
```

Line 9, Column 66 | Tab Size: 4 | HTML

HTML, CSS, and JS combined

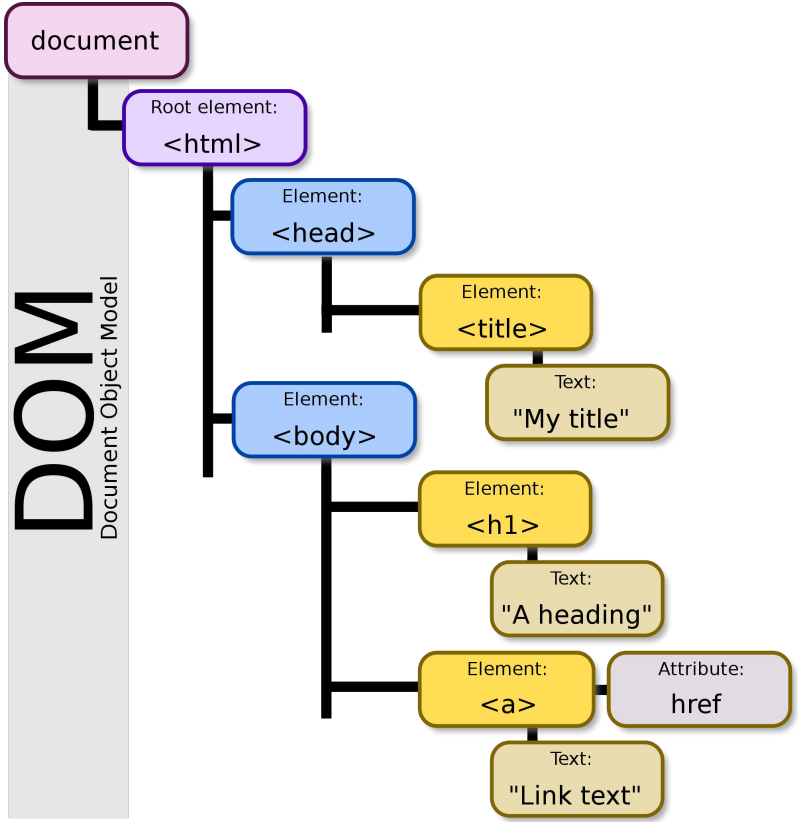
CMPM 120

The Document Object Model



"A Web page is a document. This document can be either displayed in the browser window, or as the HTML source. But it is the same document in both cases. The Document Object Model (DOM) provides another way to represent, store and manipulate that same document. The DOM is a fully object-oriented representation of the web page, and it can be modified with a scripting language such as JavaScript."

MDN



https://commons.wikimedia.org/wiki/File:DOM-model.svg

JS = JavaScript

JavaScript Overview

- First developed in 1995 at Netscape (for Navigator 2.0)
- Not really related to Java
- Actually a scripting language (domain-specific for web environment)
- Relies on host for input/output (e.g., browser)
- Multi-paradigm (e.g., procedural, functional, OOP, etc.)
- Dynamic (i.e., executes at runtime)
- Loosely typed
- Standardized as ECMAScript
- Historically maligned/praised for its flexibility

JavaScript Types

number

string

Boolean

Object
- Function
- Array
- Date
- RegExp

null

undefined

JavaScript Types

Some examples

```
// number
var year = 2019;
var course_number = 120;

// string
var name = "Isaac Karth";

// Boolean
var ownsCar = false;

//Object (Function)
var addNumbers = function(a, b) {
        return a + b;
}
//Object (Array)
var = favGames = ['Thief: The Dark Project', 'SimCity 2000',
'Heaven's Vault', 'Crusader Kings 2', 'Pathologic 2' ]
//Object (Date)
var today = new Date(2019, 6, 25);
//Object (RegExp)
var re= new RegExp('\\w');

var the_abyss = null;

undefined // it's complicated
```

Next Class:

Bring your laptop!
(if you have one)