



Let's make a web game!

credit: Photon Storm

# What will we cover today?

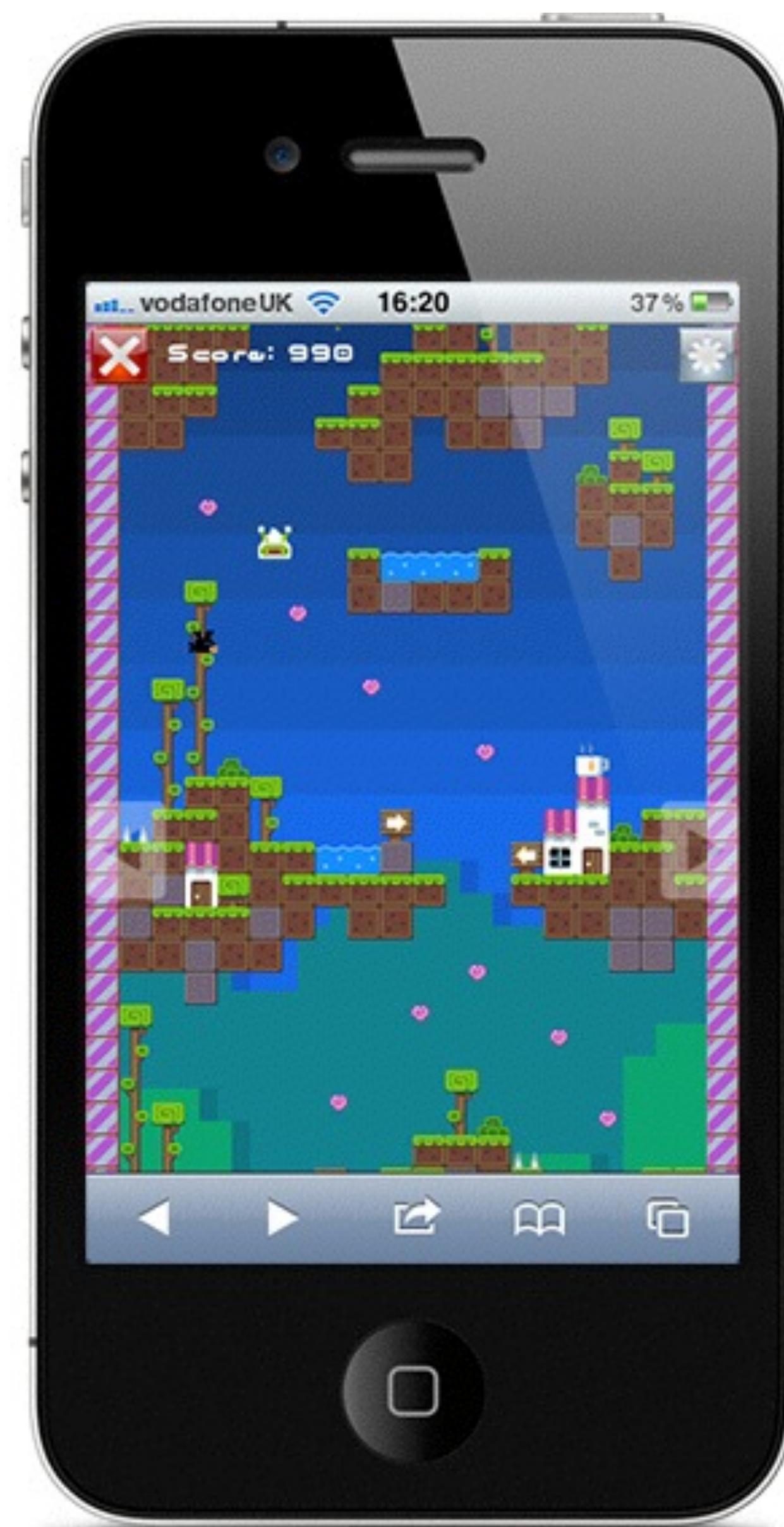
- The canvas element
- Phaser, a JavaScript framework
- Setting up a local server
- Making an HTML5 page
- Setting up Phaser
- Creating a simple game

# What is canvas?

A canvas is a rectangle in your HTML page where you can use JavaScript to draw anything you want.



credit: Nintendo



credit: Photon Storm

# What is Phaser?

Phaser is an open source JavaScript framework made for HTML5 game developers by HTML5 game developers.

## PHASER FEATURES



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

- INPUT
- SOUND
- TILEMAPS
- DEVICE SCALING
- PLUGIN SYSTEM
- MOBILE BROWSER
- DEVELOPER SUPPORT
- BATTLE TESTED

credit: Photon Storm

# What's a framework?

Frameworks help to reduce the amount of time spent reinventing the wheel.

They come with a large set of tools to help you accomplish tasks faster.



# Setting up a local server



credit: Silicon Angle

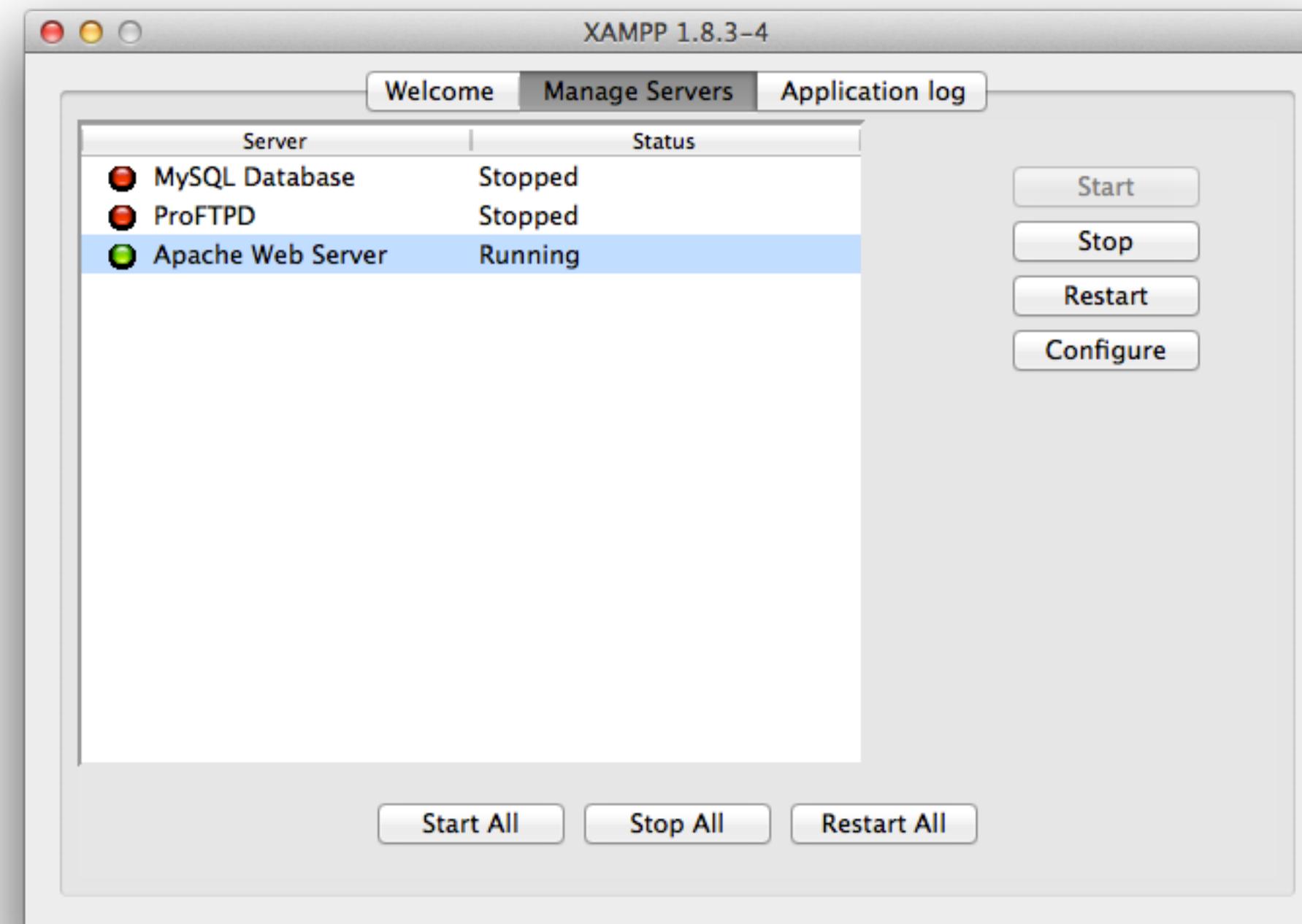
# Local servers

Local servers allow you to test websites you've programmed without an internet connection.

Phaser requires a server to run for security reasons.



# Start your Apache server

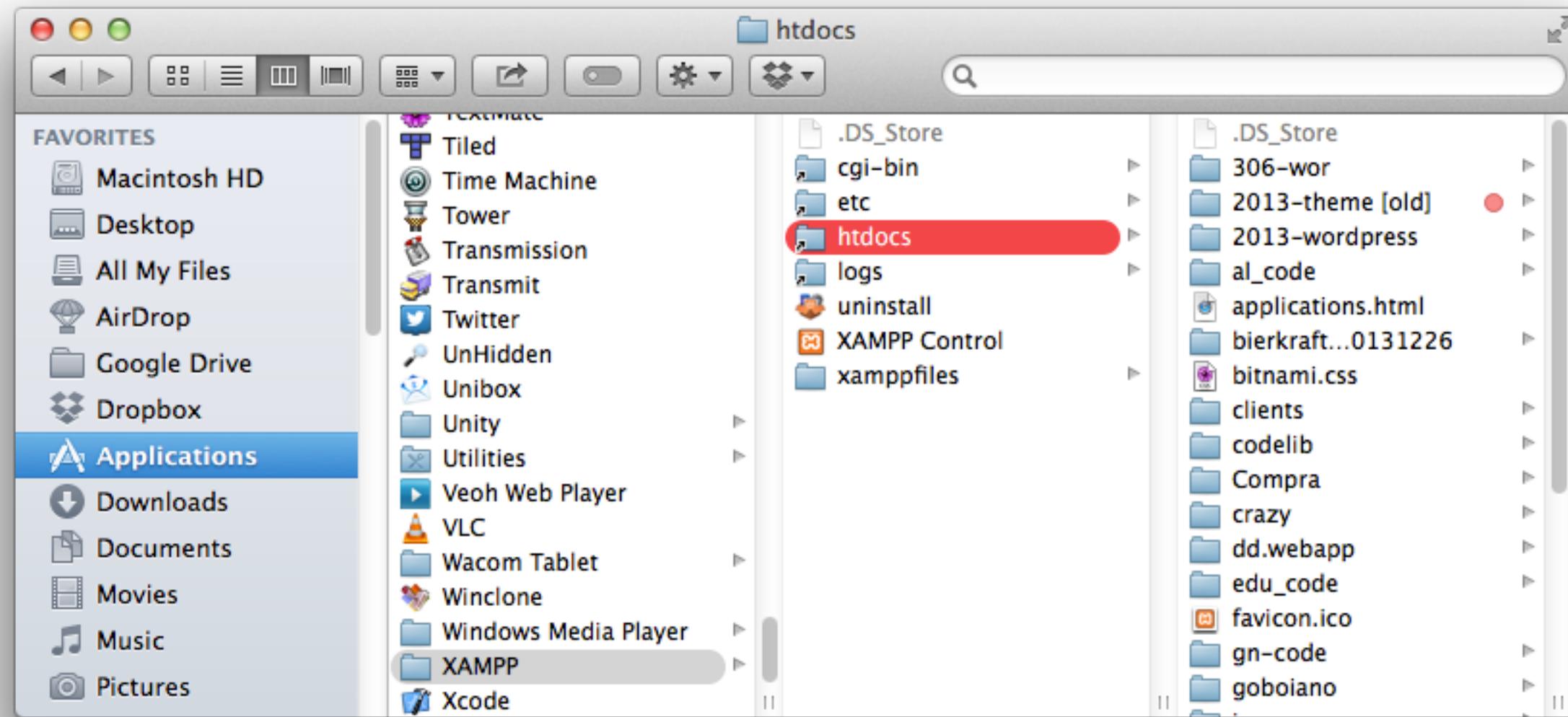


Open XAMPP and turn on Apache

credit: The Matrix

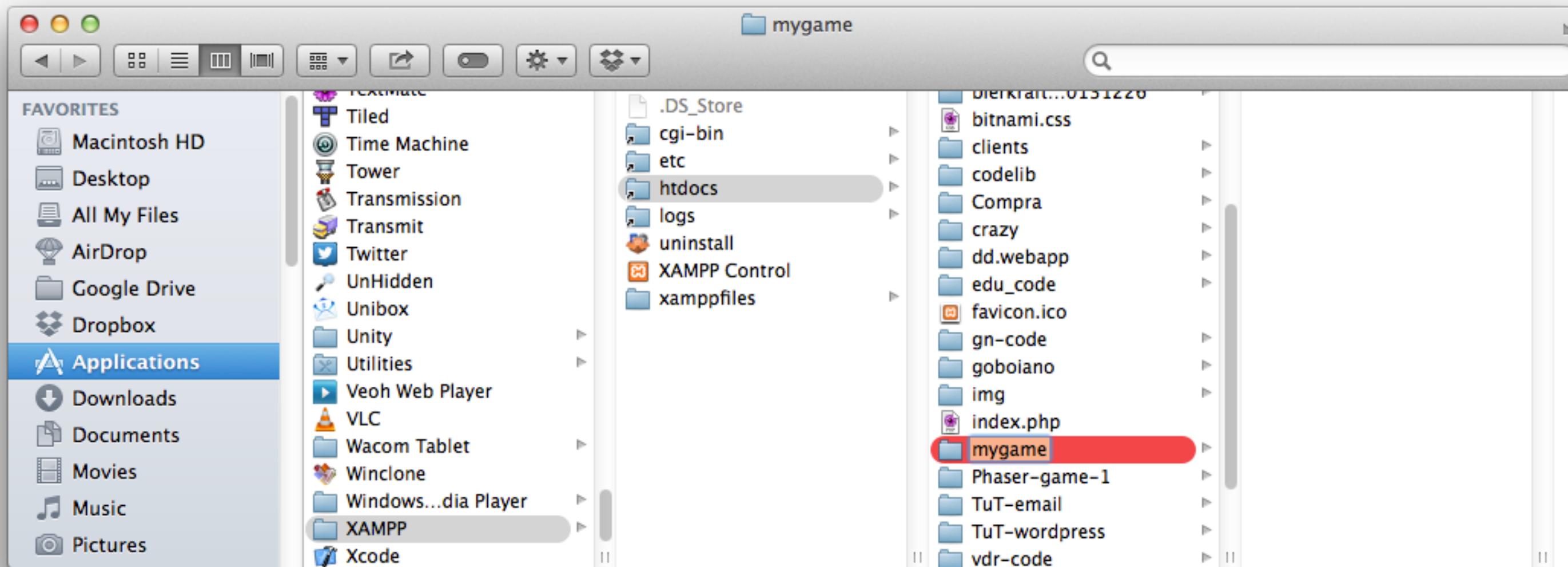
# Making an HTML5 page

# Find htdocs



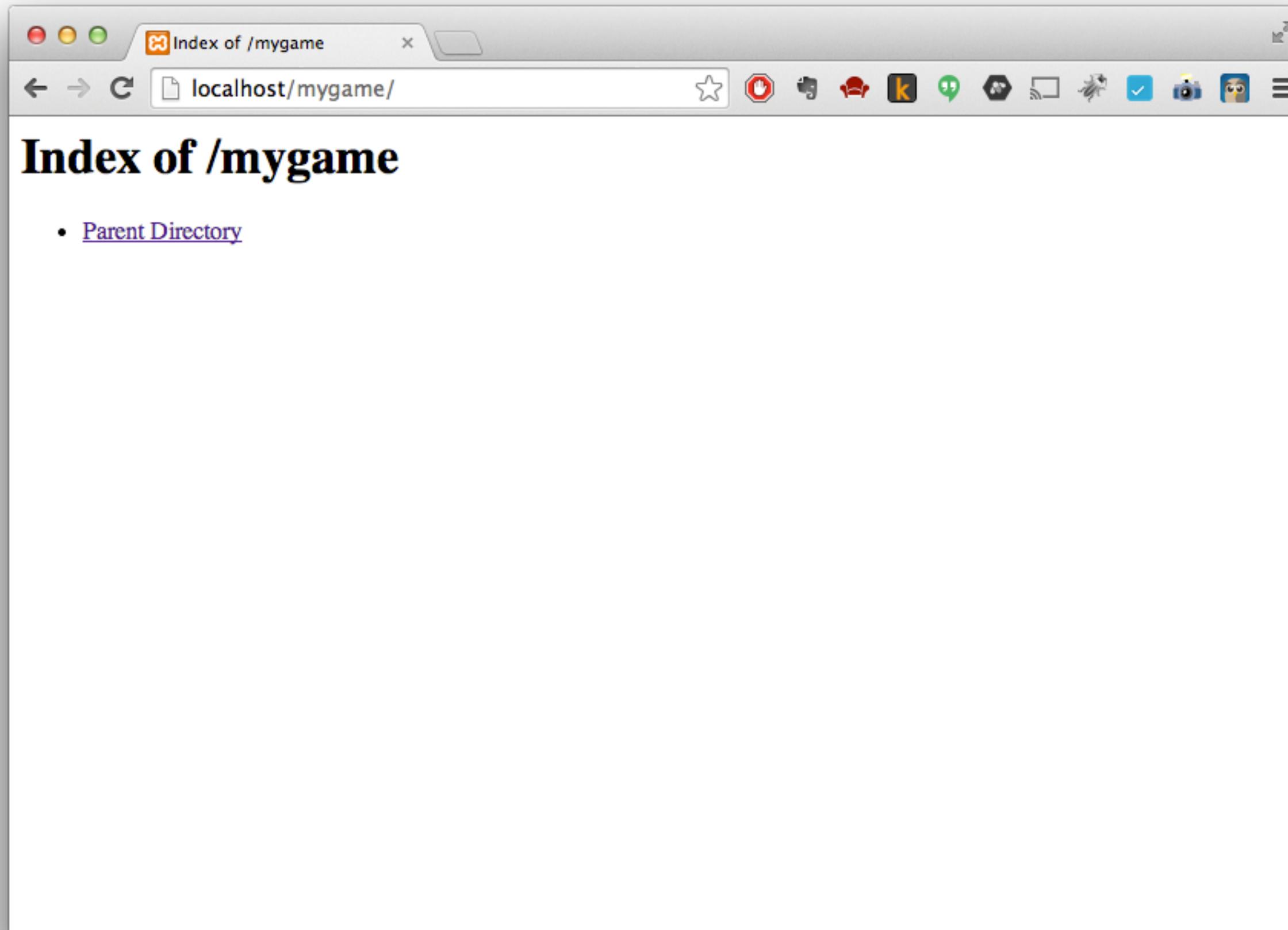
Open your XAMPP folder, then htdocs

# Create a folder to work in



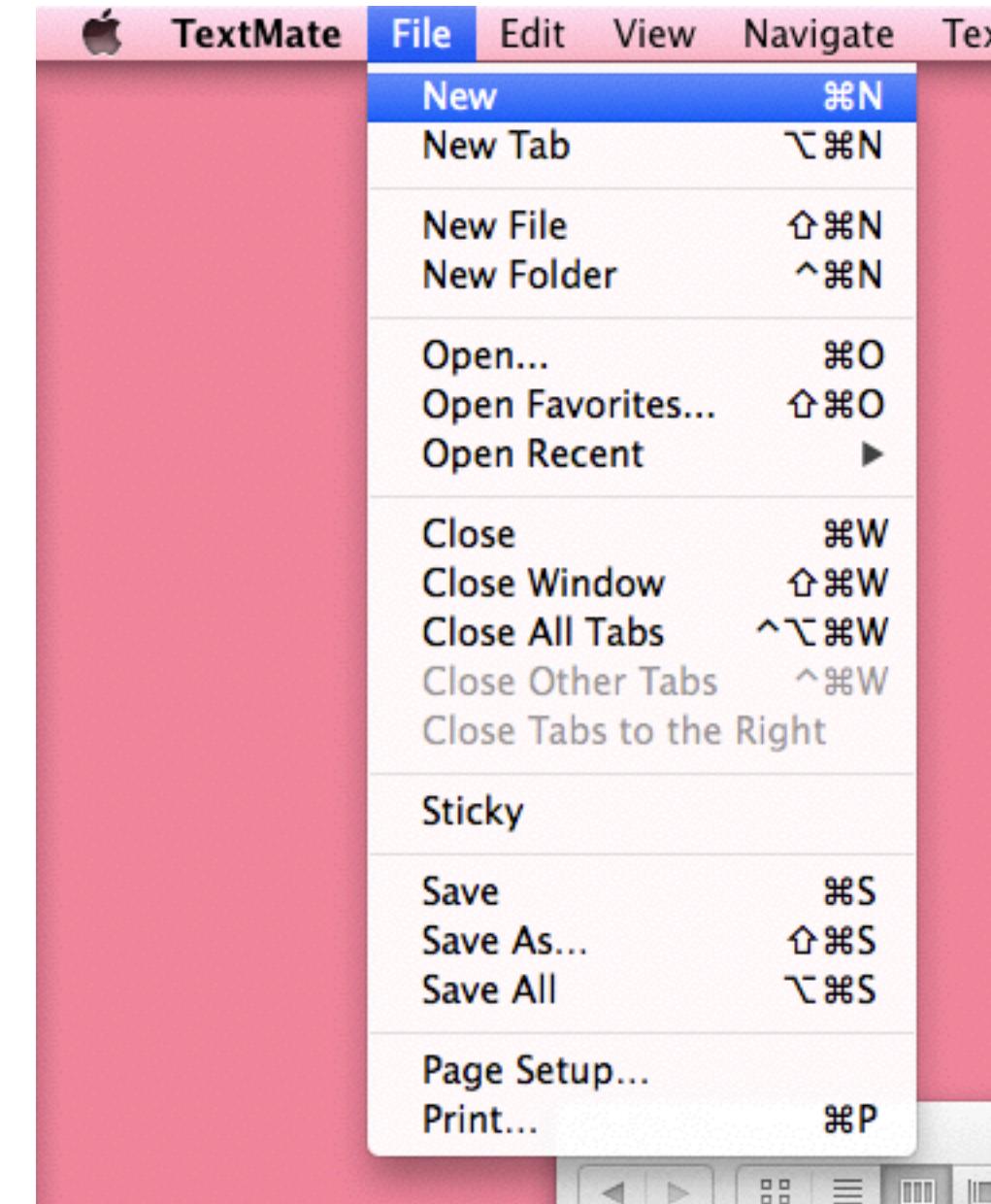
Name your folder **mygame**

# Check if it worked!



Go to [localhost/mygame](http://localhost/mygame)

# Create a new file in your text editor



Save your new file as `index.html` in your folder

# Components of an HTML5 page

- `<!DOCTYPE html>`: tells the browser it is looking at an HTML5 page
- `<html>`: begins the HTML code
  - `<head>`: the area where meta information will be located
    - `<title>`: the website title
  - `<body>`: the part of the page where the HTML shows up!



# Getting started with Phaser



credit: Photon Storm

# Download the latest version

This repository

Search or type a command

Explore Gist Blog Help

cattsmall

Watch 382 Star 4,412 Fork 1,202

photonstorm / phaser

Phaser is a fun, free and fast 2D game framework for making HTML5 games for desktop and mobile web browsers, supporting Canvas and WebGL rendering. <http://phaser.io>

1,649 commits 2 branches 26 releases 62 contributors

branch: master phaser / +

Phaser 2.0.4 zero hour update.

photonstorm authored 6 days ago latest commit e0083dcb26

File	Description	Time Ago
build	Phaser 2.0.4 zero hour update.	6 days ago
docs	Docs update.	6 days ago
examples	Removed the examples build script from the Gruntfile (fix #592)	2 months ago
filters	Fixed the vectors used in the BlurX and BlurY filters (thanks @nickryall)	25 days ago
plugins	Color.HSVColorWheel will return an array with 360 color objects for e...	8 days ago
resources	New Phaser Project Template specifically for requireJS in the 'resour...	7 days ago
src	P2 fixed creation of RevoluteConstraint by passing maxForce in the op...	6 days ago
tasks	Phaser.ArrayList is a new iterative object, similar in principal to a...	10 days ago
.gitignore	Spanish translation of the tutorials	5 months ago
.jshintrc	Unsuppress warnings for things that don't appear in the codebase.	a month ago
.travis.yml	Fixed Travis file naming (thanks Windows)	a month ago

Code Issues 32 Pull Requests 12 Wiki Pulse Graphs Network

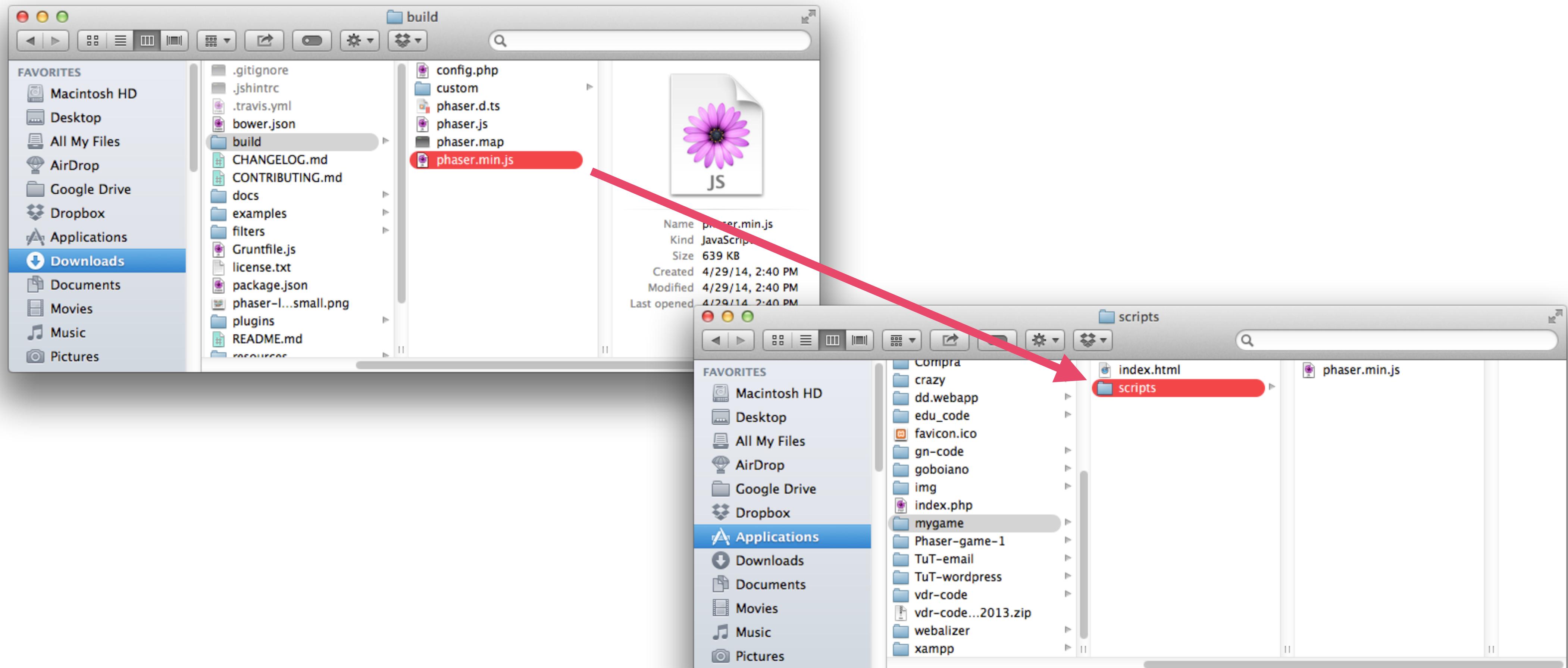
HTTPS clone URL <https://github.com/photonstorm/phaser>

You can clone with **HTTPS**, **SSH**, or **Subversion**.

Clone in Desktop Download ZIP

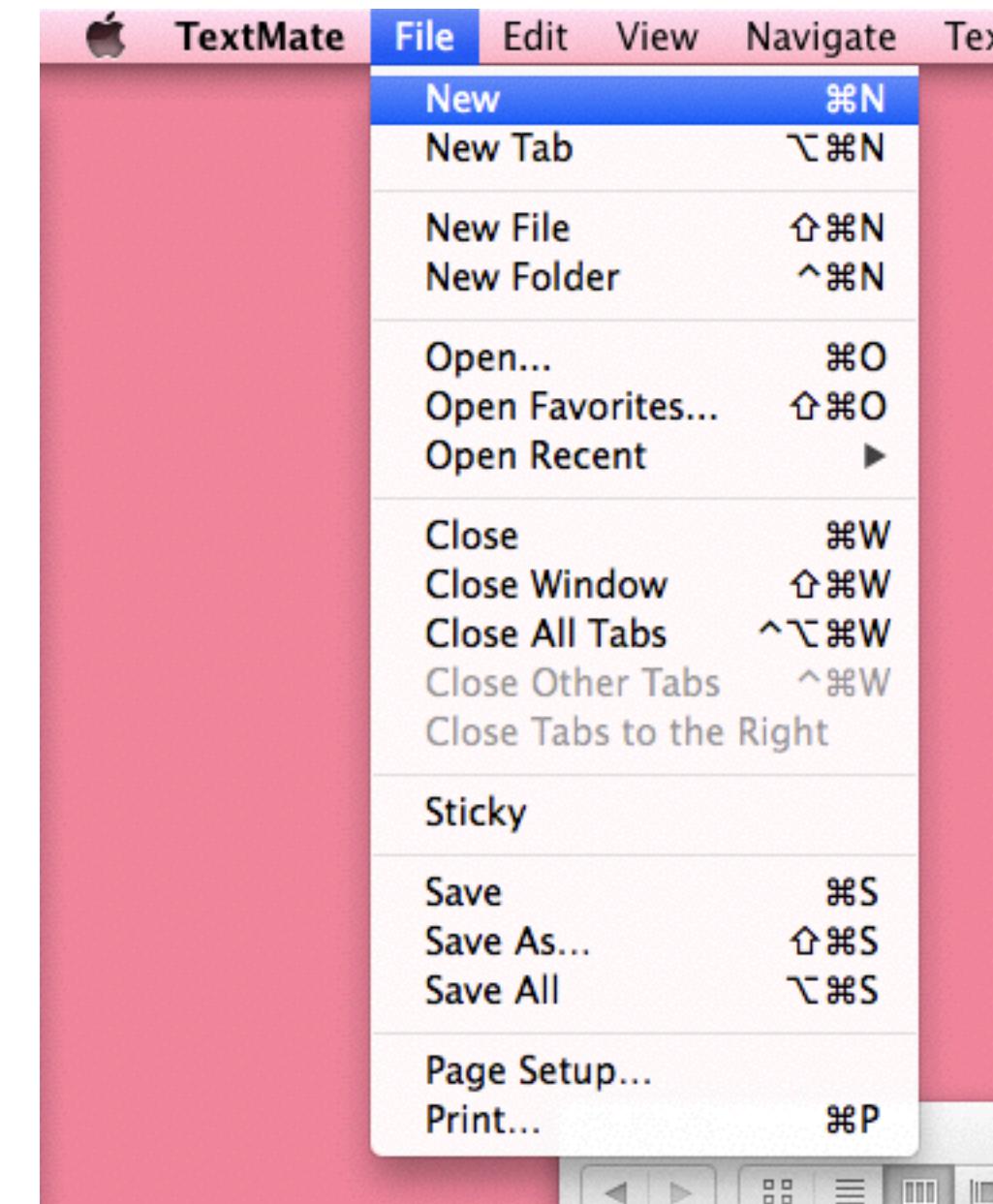
[github.com/photonstorm/phaser](https://github.com/photonstorm/phaser)

# Move phaser.min.js



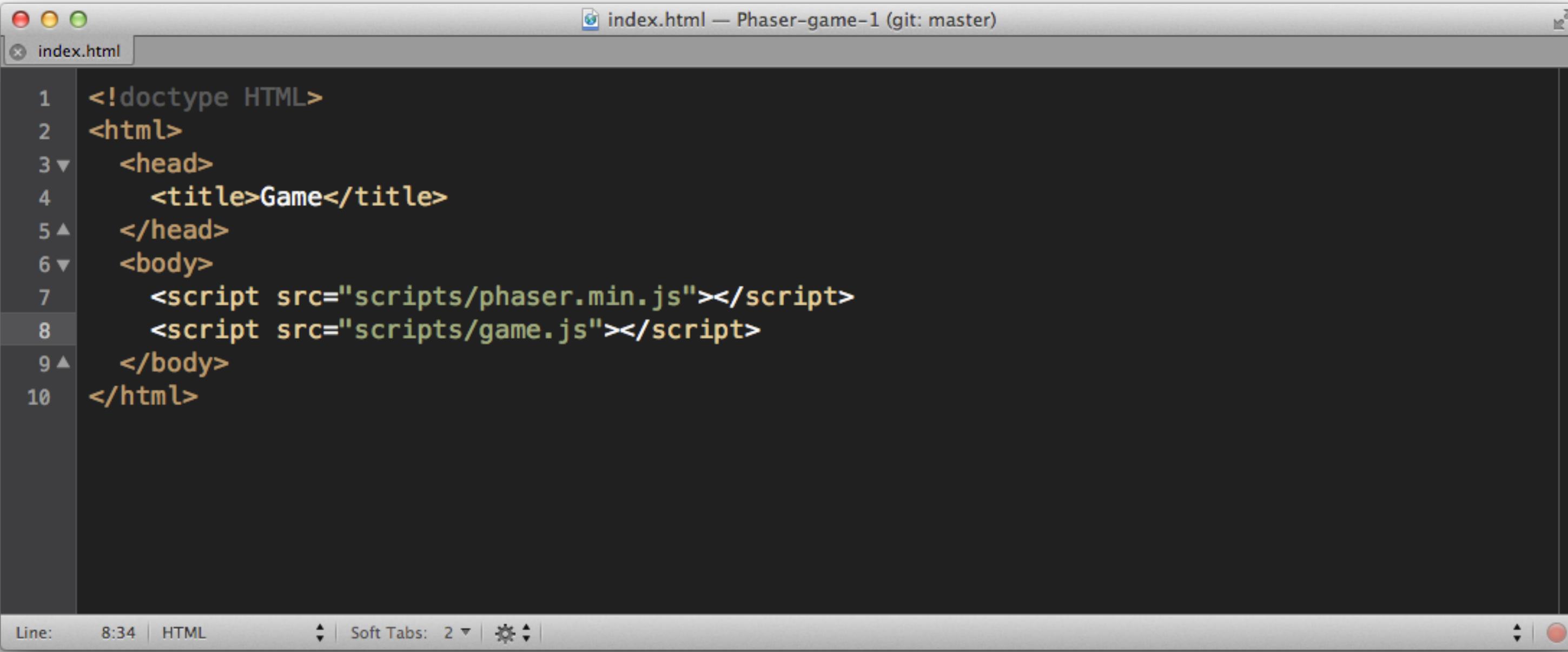
create a scripts folder & move phaser.min.js there

# Create a JavaScript file



Save your new file as `game.js` in your scripts folder

# Link your files to the HTML



A screenshot of a code editor window titled "index.html — Phaser-game-1 (git: master)". The editor displays the following HTML code:

```
1 <!doctype HTML>
2 <html>
3 <head>
4   <title>Game</title>
5 </head>
6 <body>
7   <script src="scripts/phaser.min.js"></script>
8   <script src="scripts/game.js"></script>
9 </body>
10 </html>
```

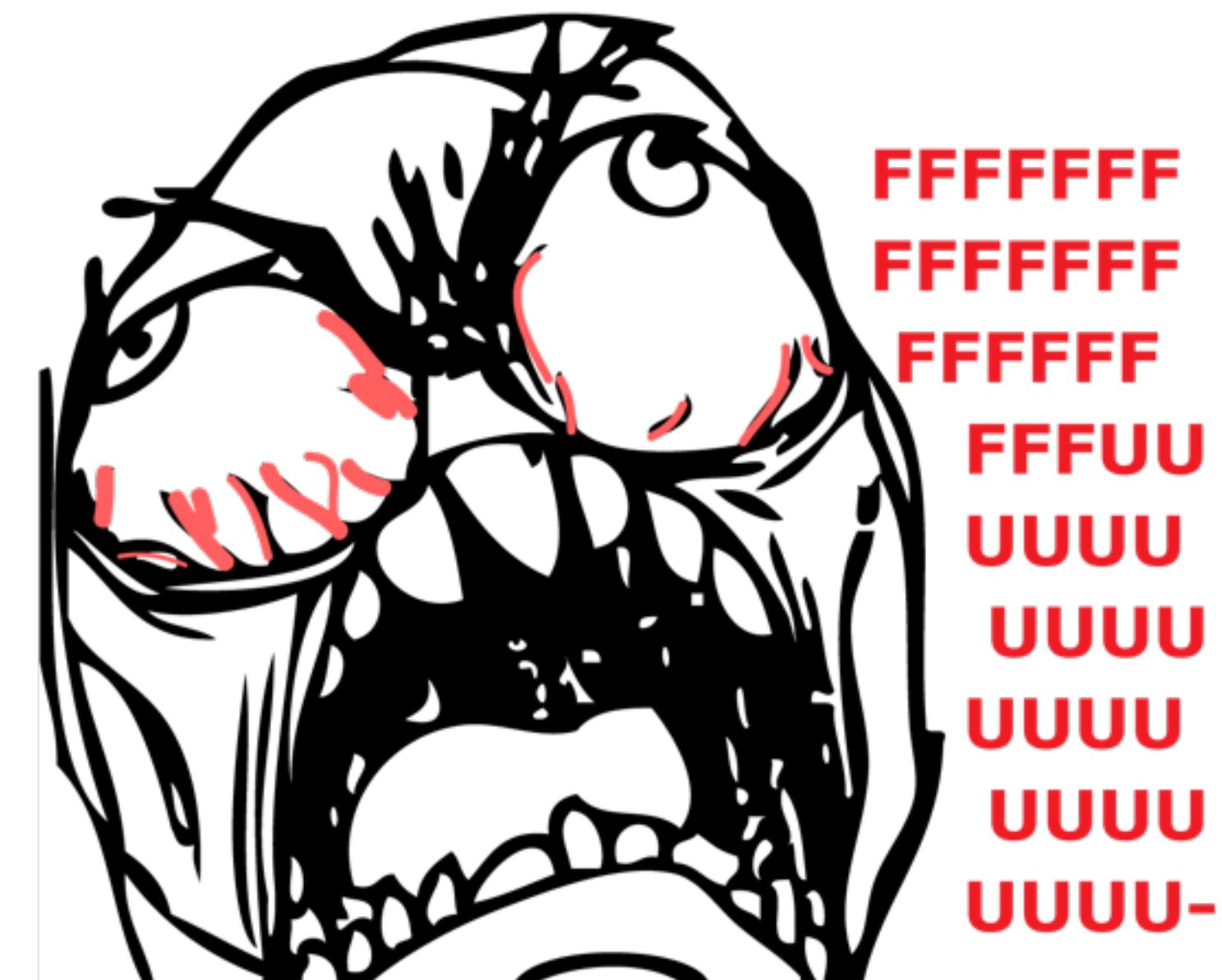
The code editor interface includes a toolbar at the top with icons for file operations, and a status bar at the bottom showing "Line: 8:34 | HTML".

# Comment your code

5 minutes after you write code  
without comments

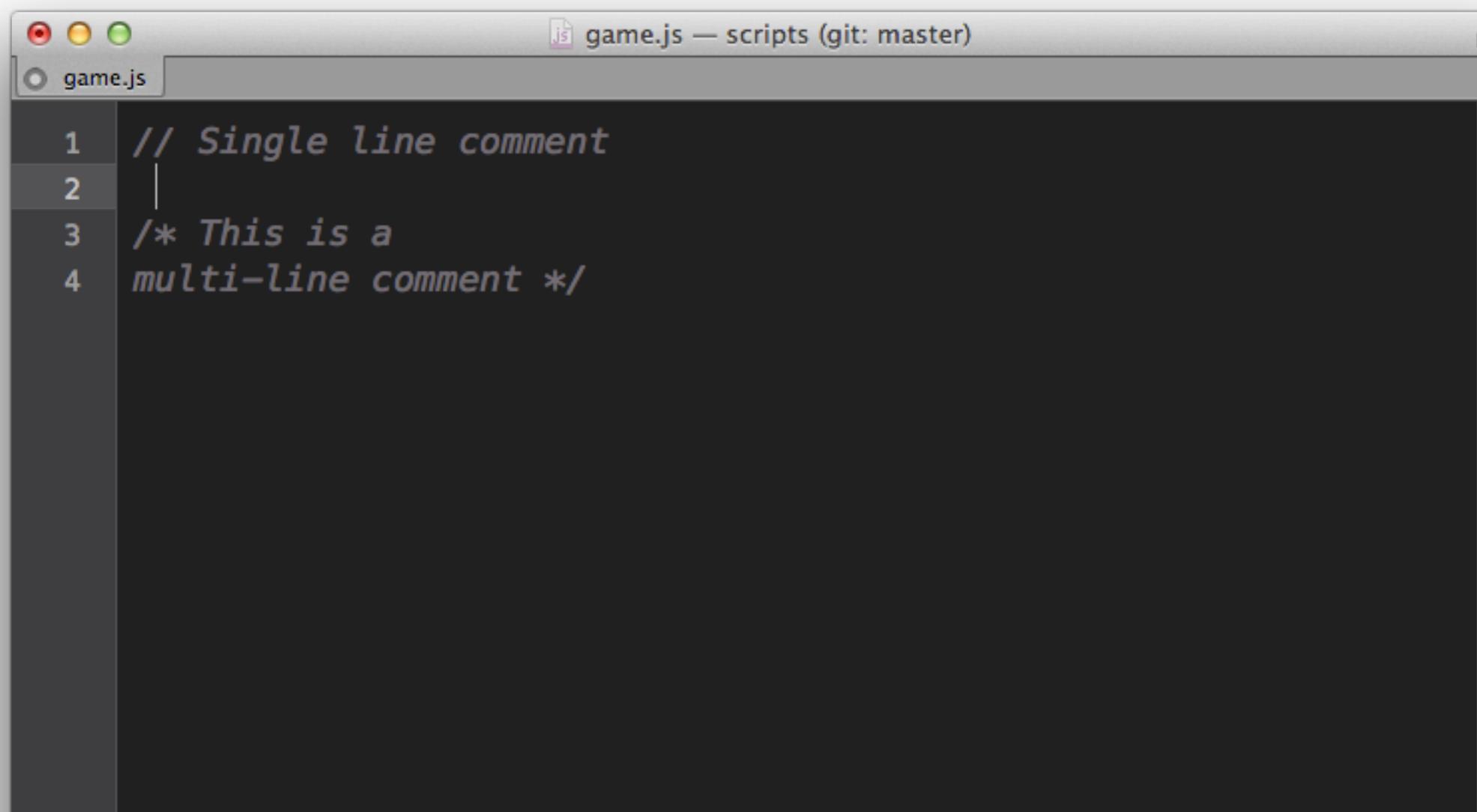


When you come back to it  
in 3 weeks



# Commenting code is easy

- Preface your one-line comment with two slashes `(//)`
- Two line comments require an asterisk and slash on each side `(/* */)`
- Most text editors have shortcuts (like  + /)



A screenshot of a text editor window titled "game.js — scripts (git: master)". The file contains the following code:

```
// Single line comment
/*
 * This is a
 * multi-line comment */

```



# Creating games with JavaScript



credit: Jandi Small

# Our game

- Collect the pigcats!
- Enemies switch between dangerous & safe
- If you collect the pigcats, you win!
- If your health goes down to 0, you lose!
- Polish it with start, win, and lose screens

# Variables

- Variables are useful for storing data that **may change** throughout the course of your app (e.g. your player's health)
- To create a variable, you have to tell JavaScript:
  - The name you're going to refer to it by
  - The value (information) that the variable contains

# Variables

- Variables let you refer to the same information many times
- If you need to change that information, you only have to do it once

For example, best friends may change but the label stays the same:

```
var myBestFriend = "Isaiah";
var myBestFriend = "Rebecca";
var myBestFriend = "Aileen";
```

# Functions

- **Function:** a named section of a program that does a specific task
- Wraps up code in an easy-to-reference way
- **Parameter:** additional information you can give the function to change the output

# Function structure

```
function fetch(dog) {  
    run to the ball;  
    pick up the ball;  
    bring the ball back;  
};
```

- Name of the function
- Parentheses: Hold any modifiers (also known as arguments)
- Brackets: What to do in the function
- Semicolon: end of line, move onto the next thing

# Calculations

**+** (add)

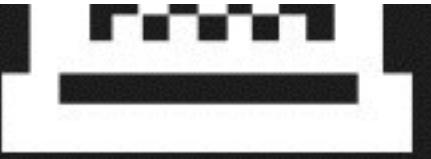
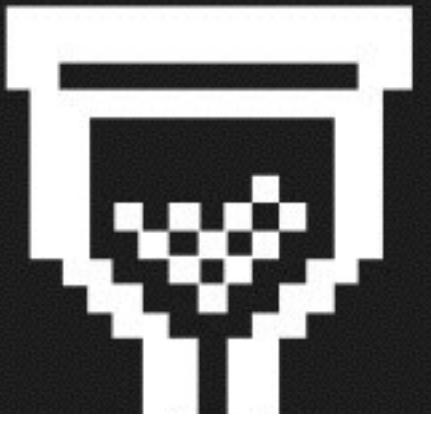
**-** (subtract)

**\*** (multiply)

**/** (divide)

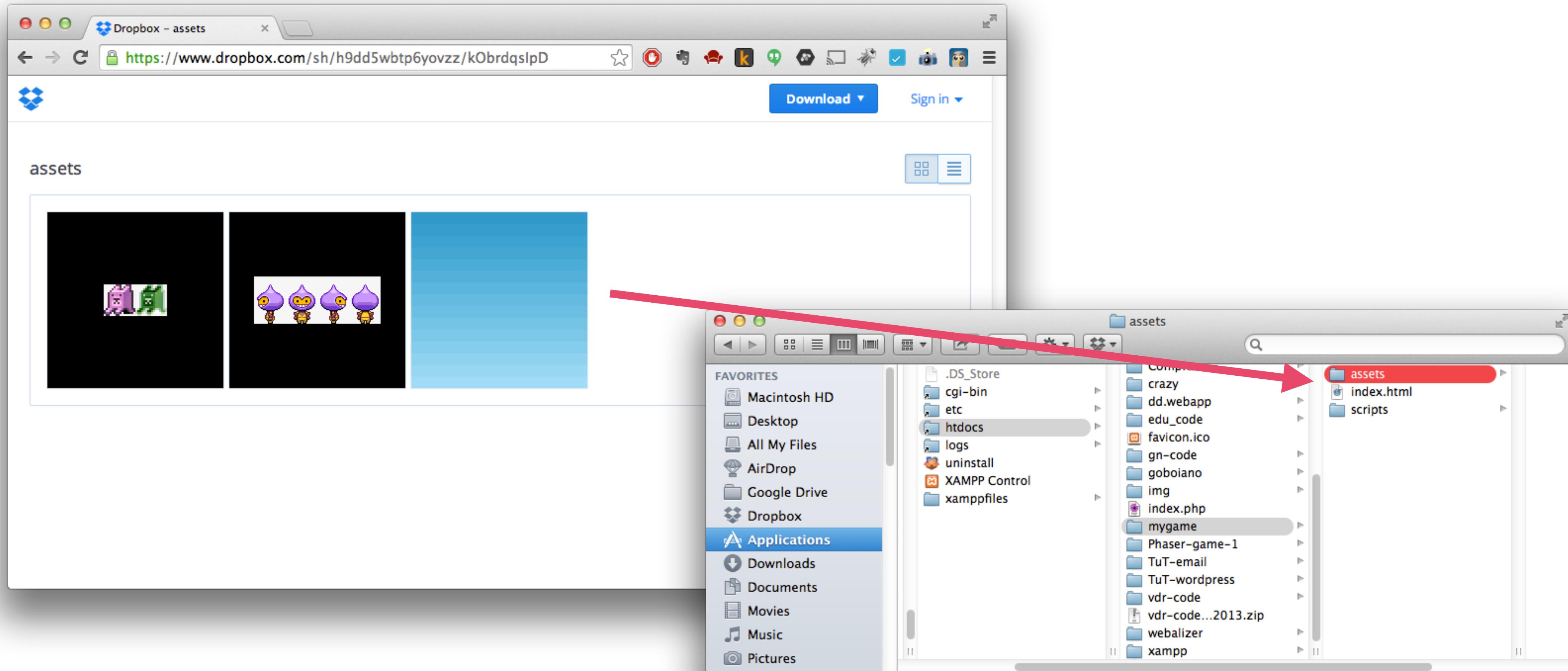
```
var addition = 13 + 22;  
var division = 100/15;
```

preload() function

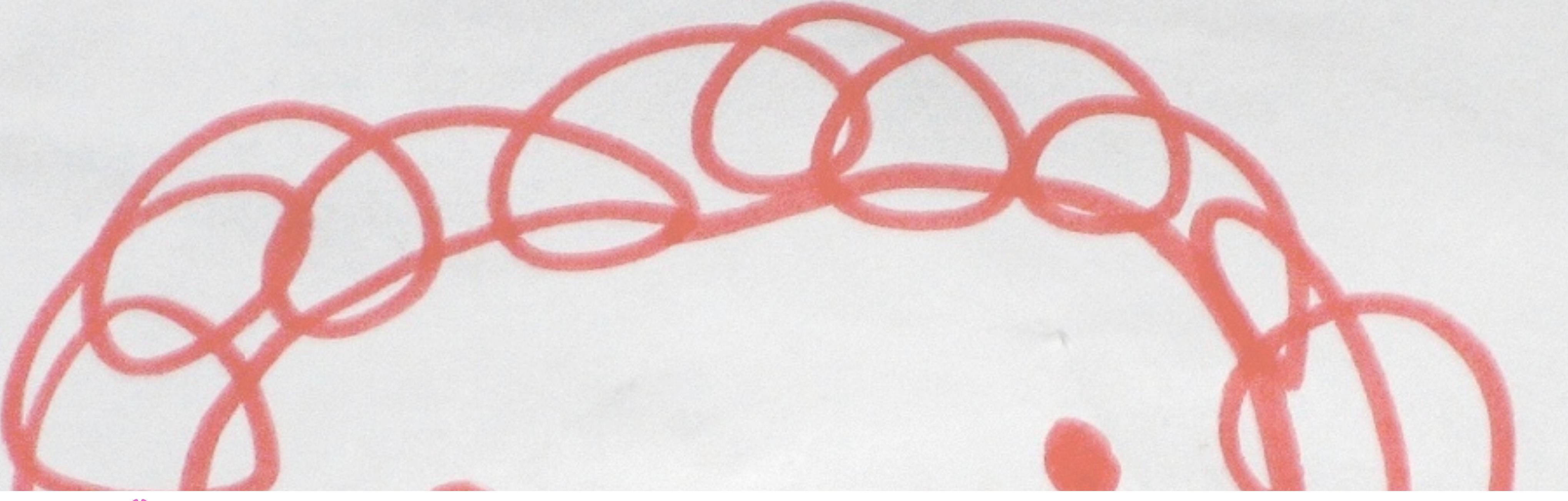


LOADING

# Create assets folder



Download & save images from [tinyurl.com/clf-html5-2014](http://tinyurl.com/clf-html5-2014)

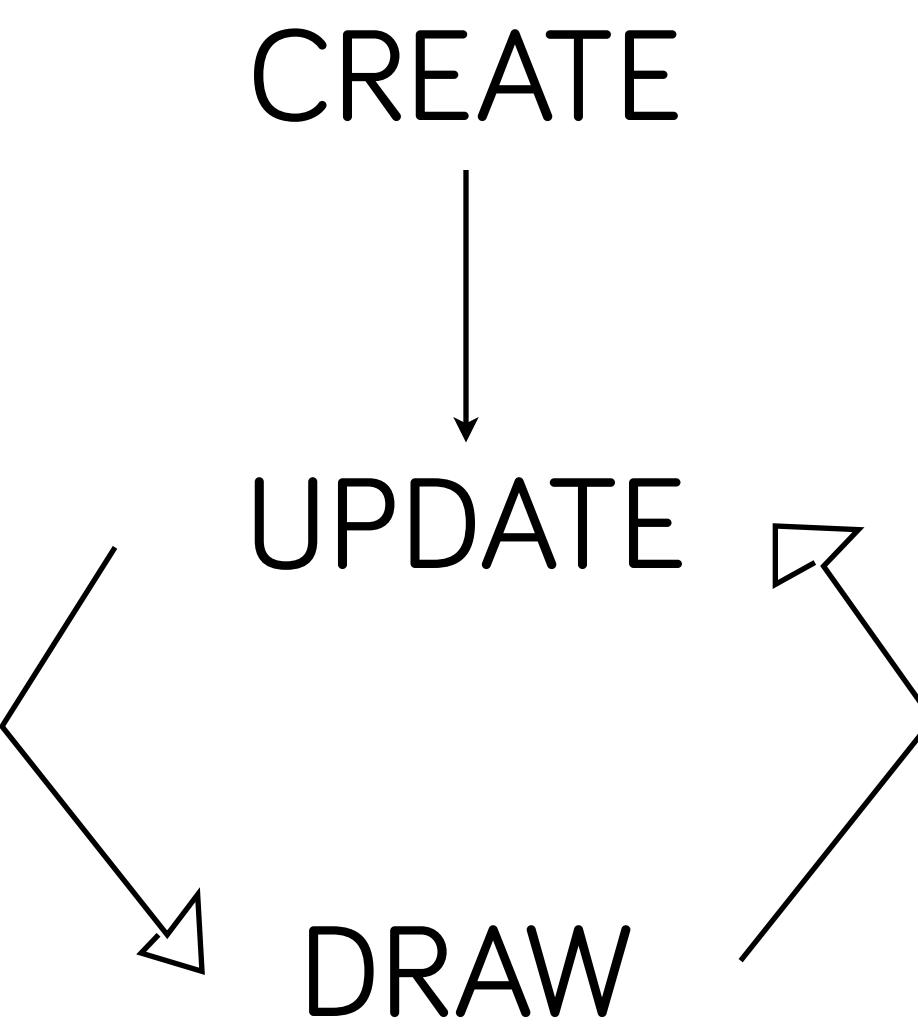


create() function



Updating. Please wait...

update() function



# How does positioning work?

$x=0$

$y=0$

$x=WIDTH$

$y=HEIGHT$



# How do I move my player?

---

# game.input

---

game.input.keyboard.createCursorKeys();

+

conditional statements

# Conditional statements

```
if (squirrel){  
    console.log("squirrel!");  
} else {  
    console.log("bark!");  
}
```

- Check if something is true or false
- Provide backup option if necessary

How do we create a bunch of  
the same objects?

# Groups

