



Let's make a web game!

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: 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

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

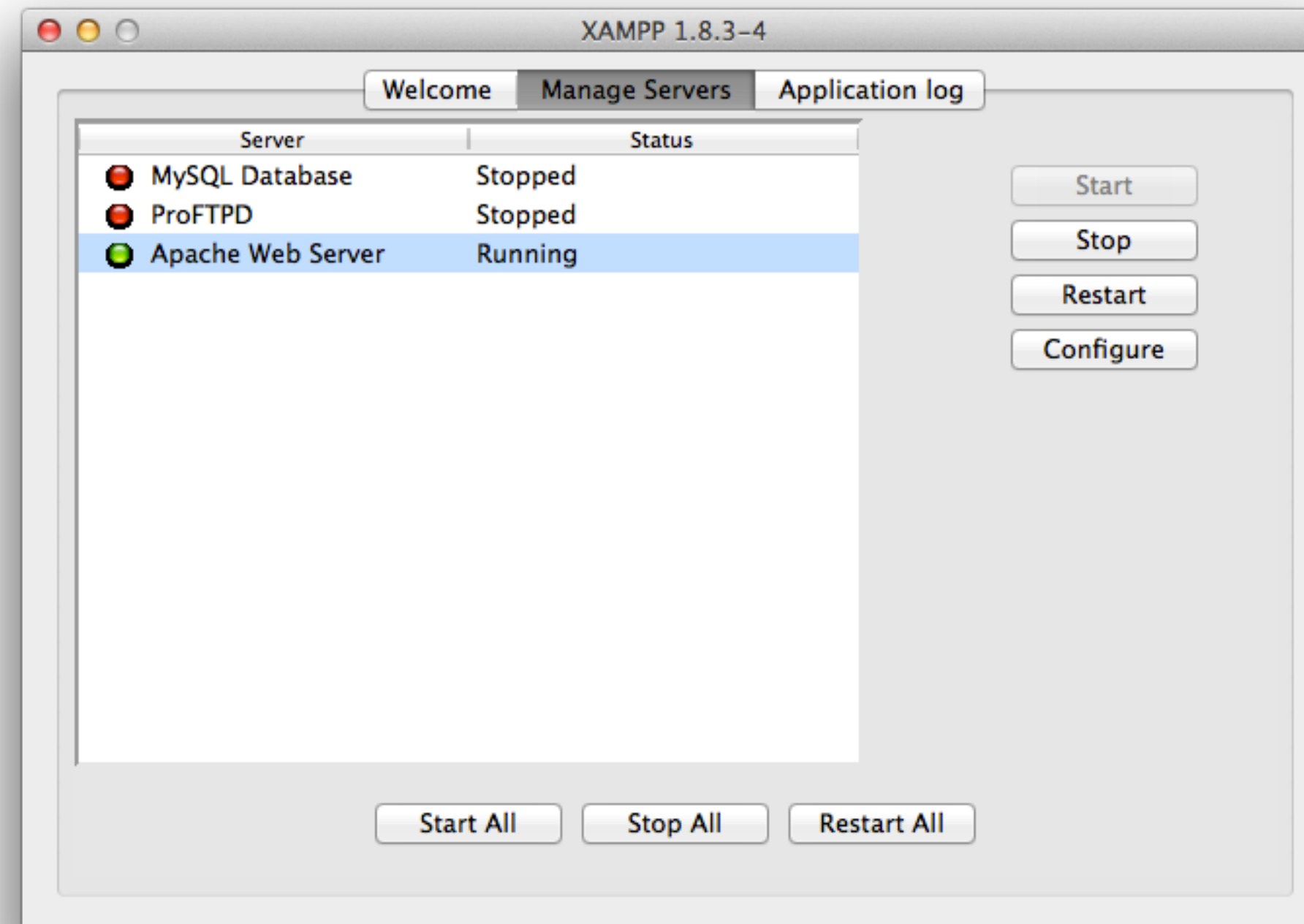
Local servers

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

Phaser requires a server to run for security reasons.



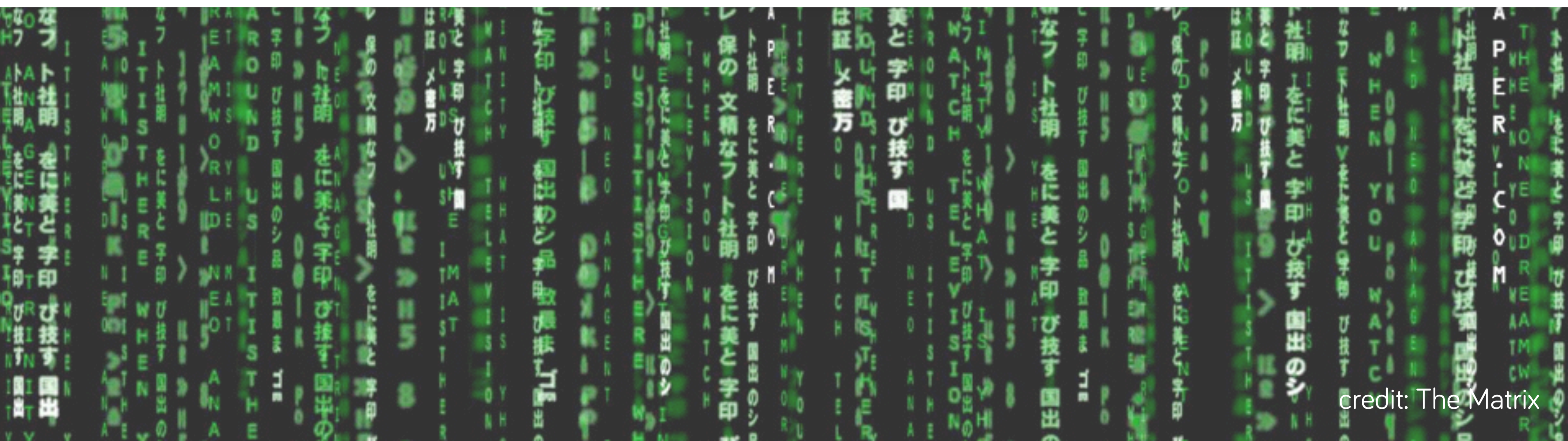
Start your Apache server



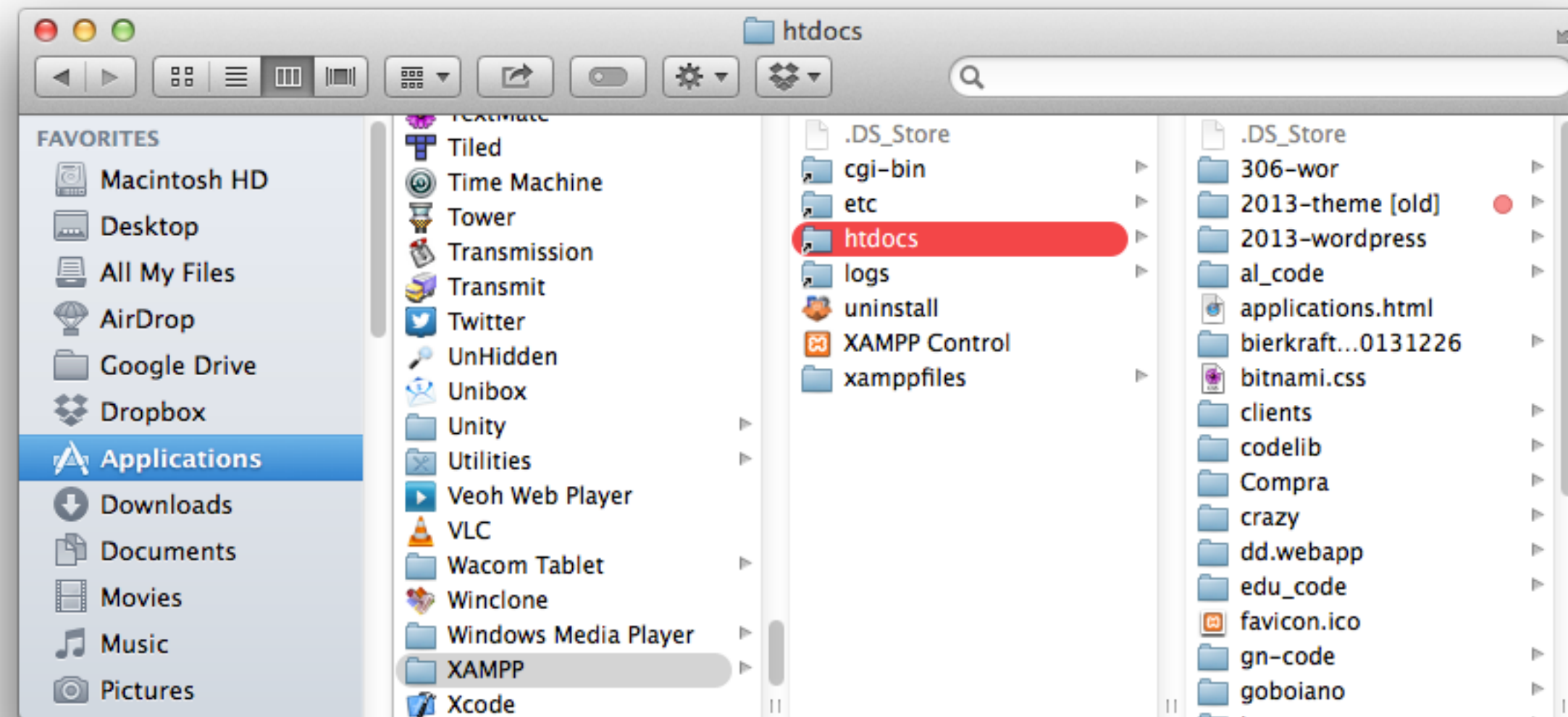
Open XAMPP and turn on Apache



Making an HTML5 page

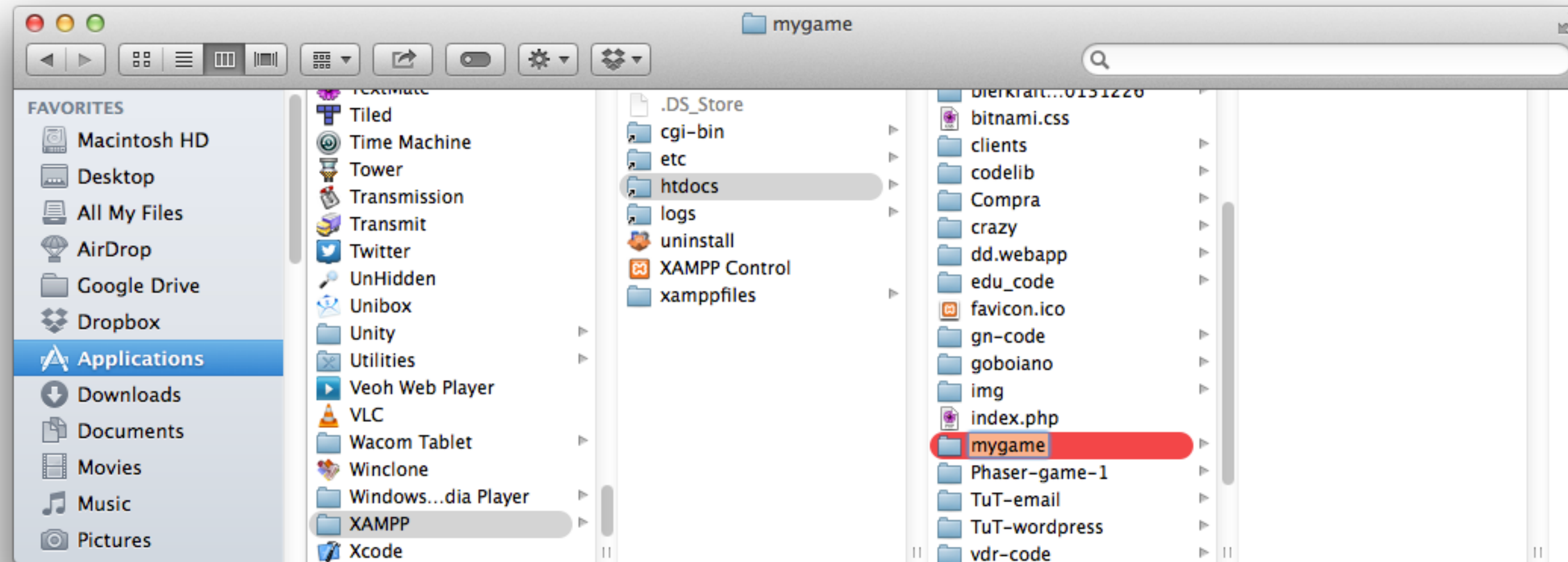


Find htdocs



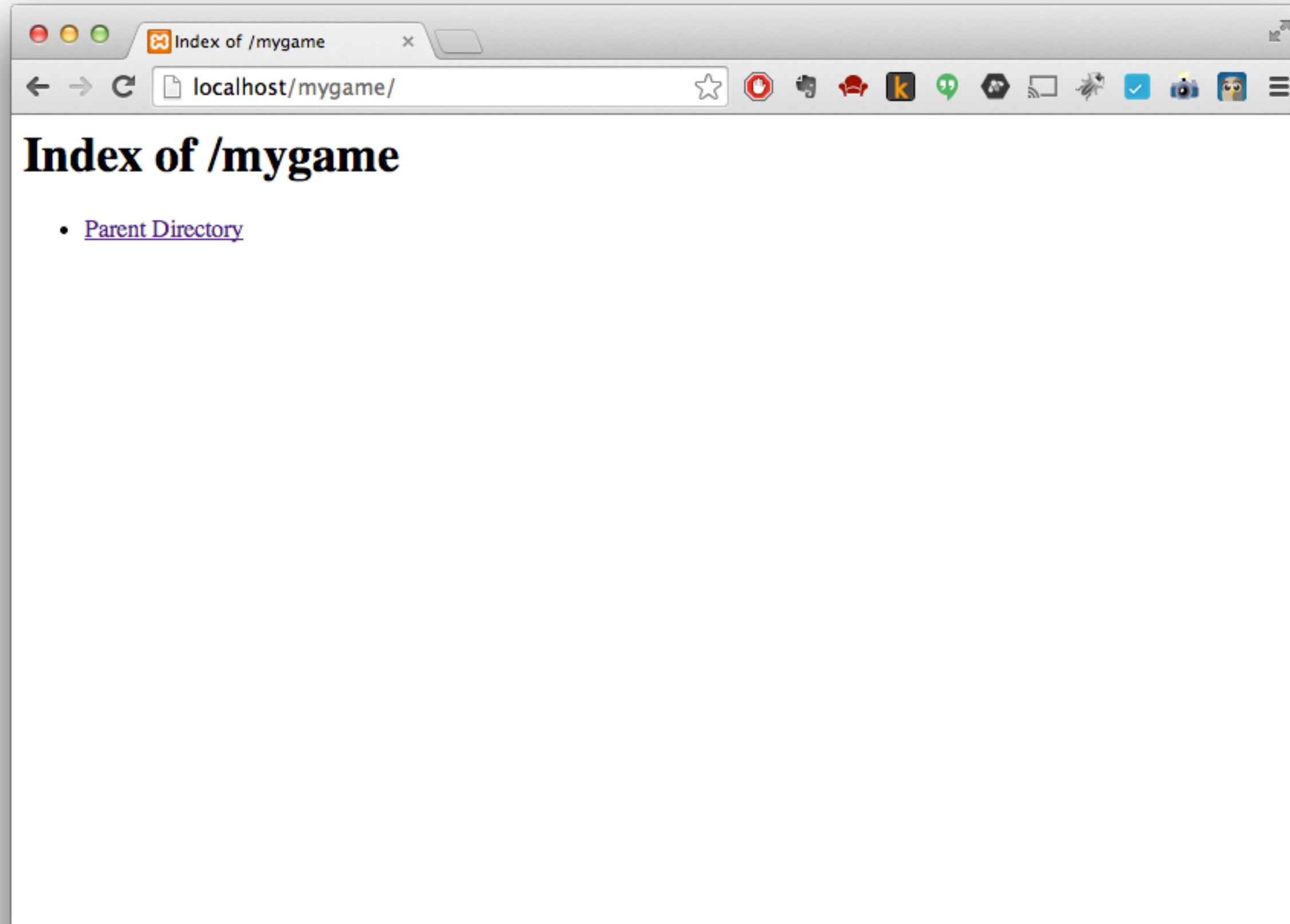
Open your XAMPP folder, then htdocs

Create a folder to work in



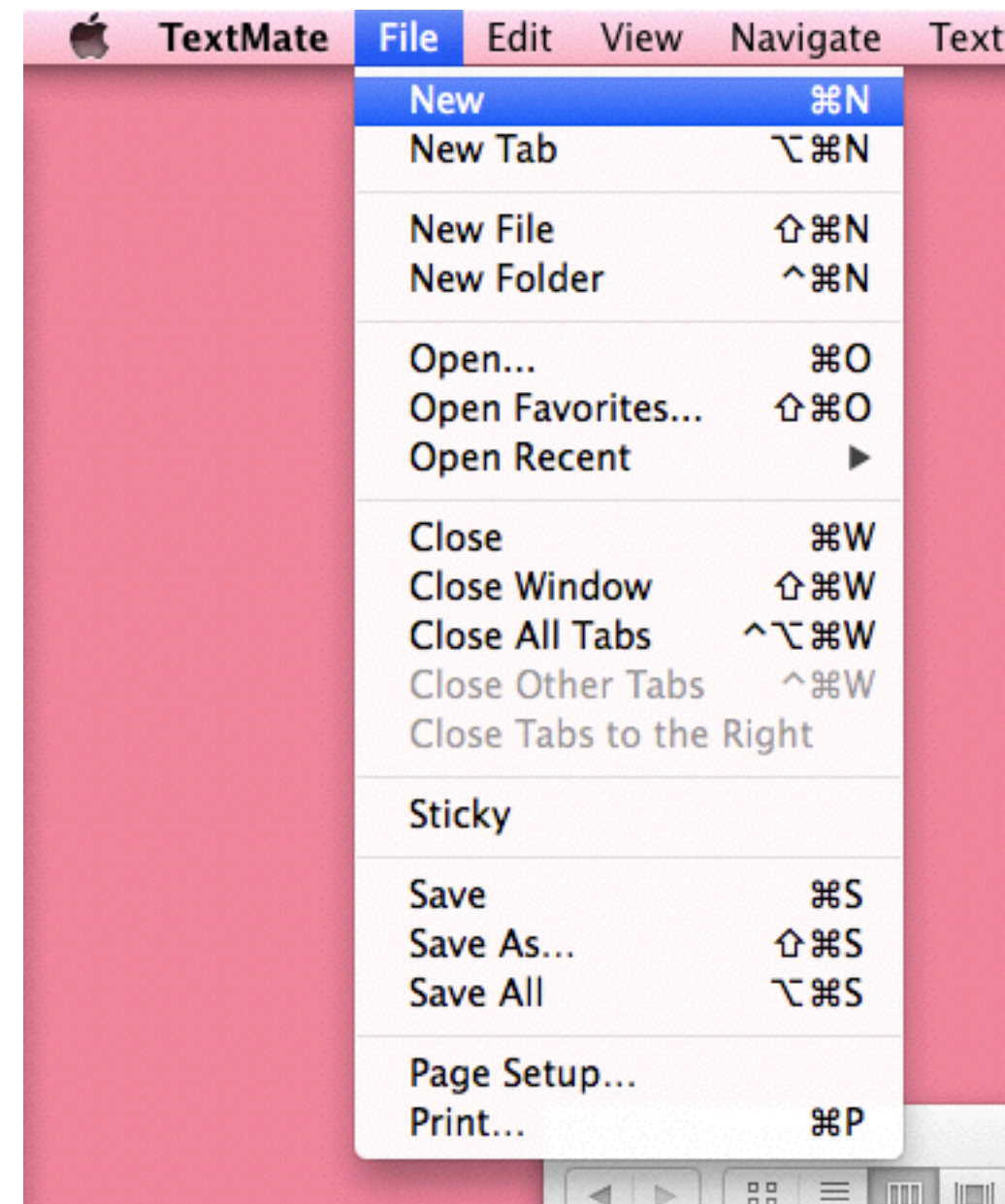
Name your folder **mygame**

Check if it worked!



Go to `localhost/mygame`

Create a new file in your text editor



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

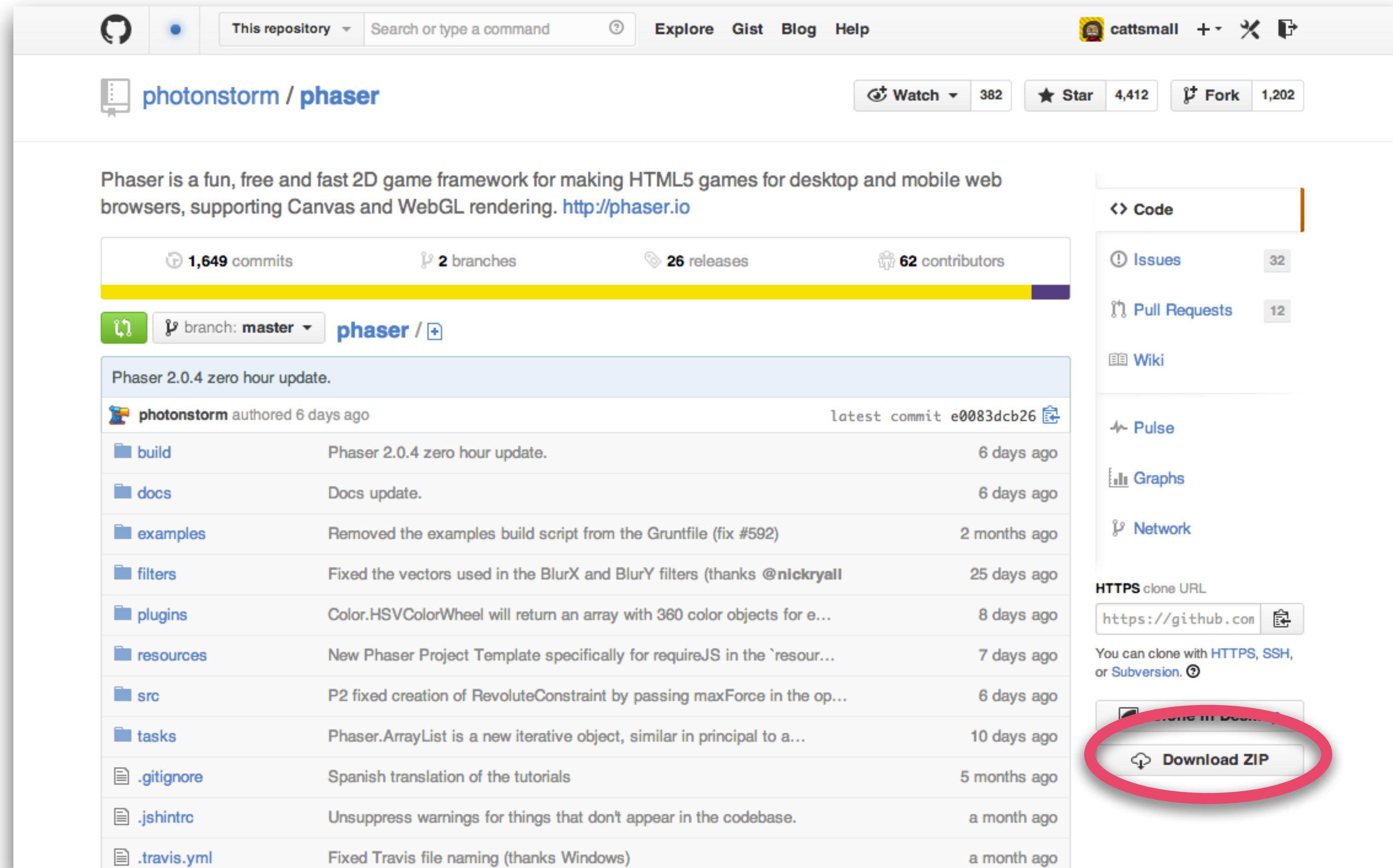
Components of an HTML5 page

- `<!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

Download the latest version



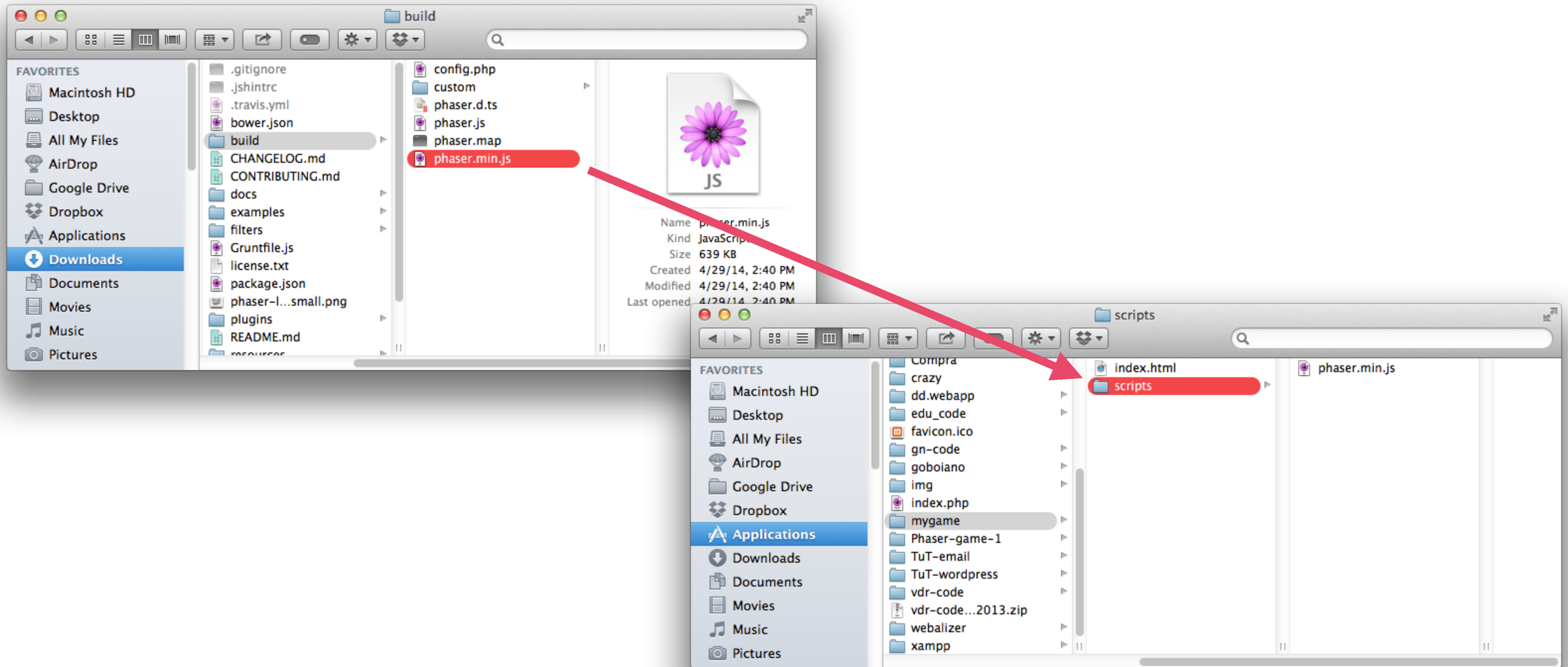
The screenshot shows the GitHub repository page for `photonstorm / phaser`. The repository has 382 watchers, 4,412 stars, and 1,202 forks. It contains 1,649 commits, 2 branches, 26 releases, and 62 contributors. The main content area shows a commit history table with the following entries:

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

On the right side of the page, the 'Code' section is expanded, showing options for cloning the repository. The 'Download ZIP' button is circled in red, indicating the recommended method for downloading the latest version.

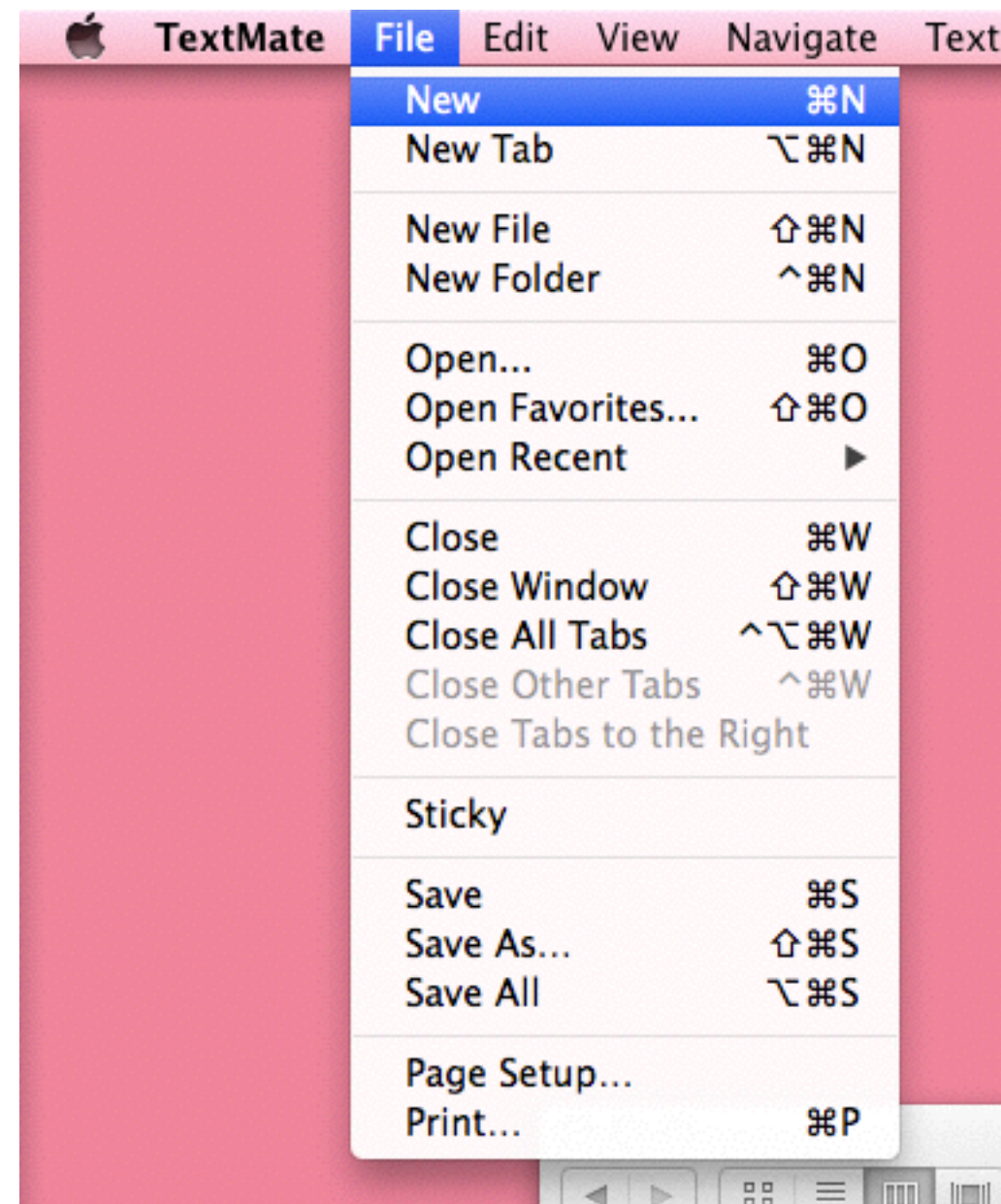
github.com/photonstorm/phaser

Move phaser.min.js



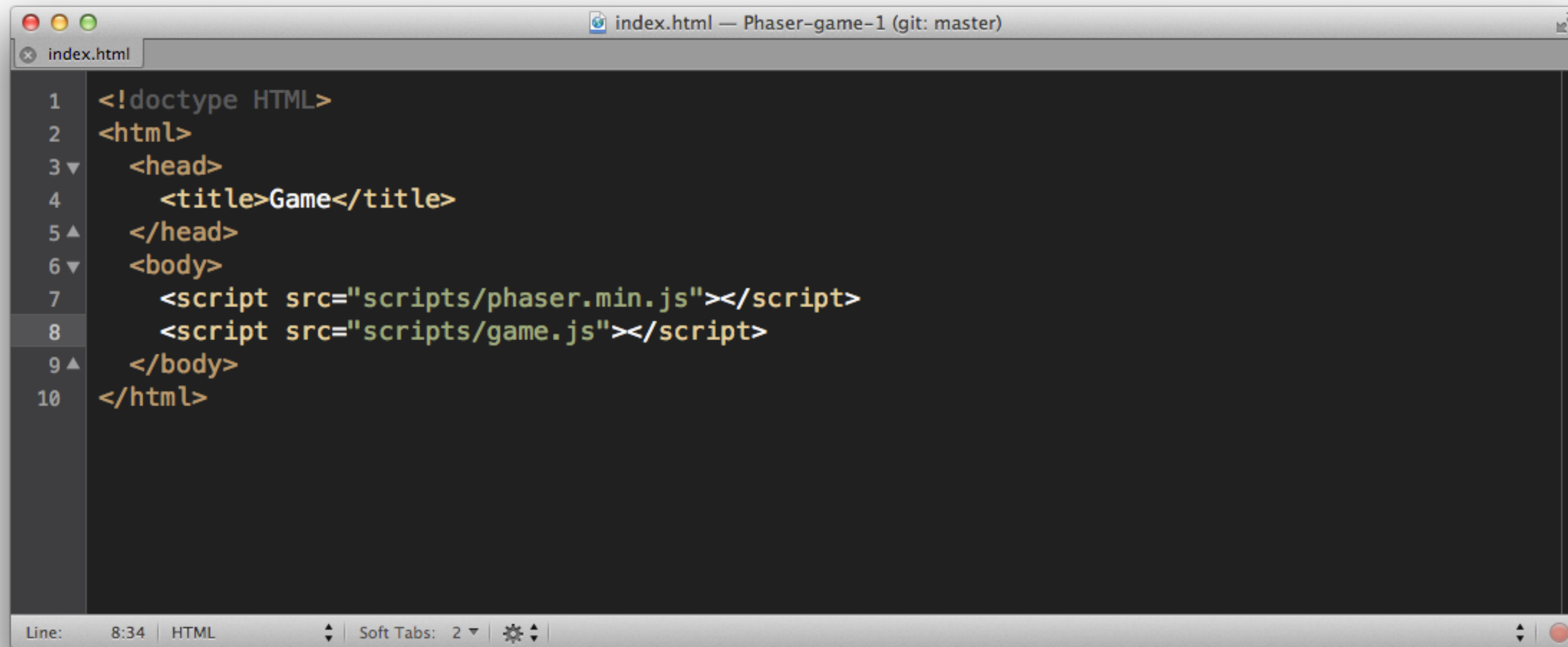
create a scripts folder & move phaser.min.js there

Create a JavaScript file



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

Link your files to the HTML

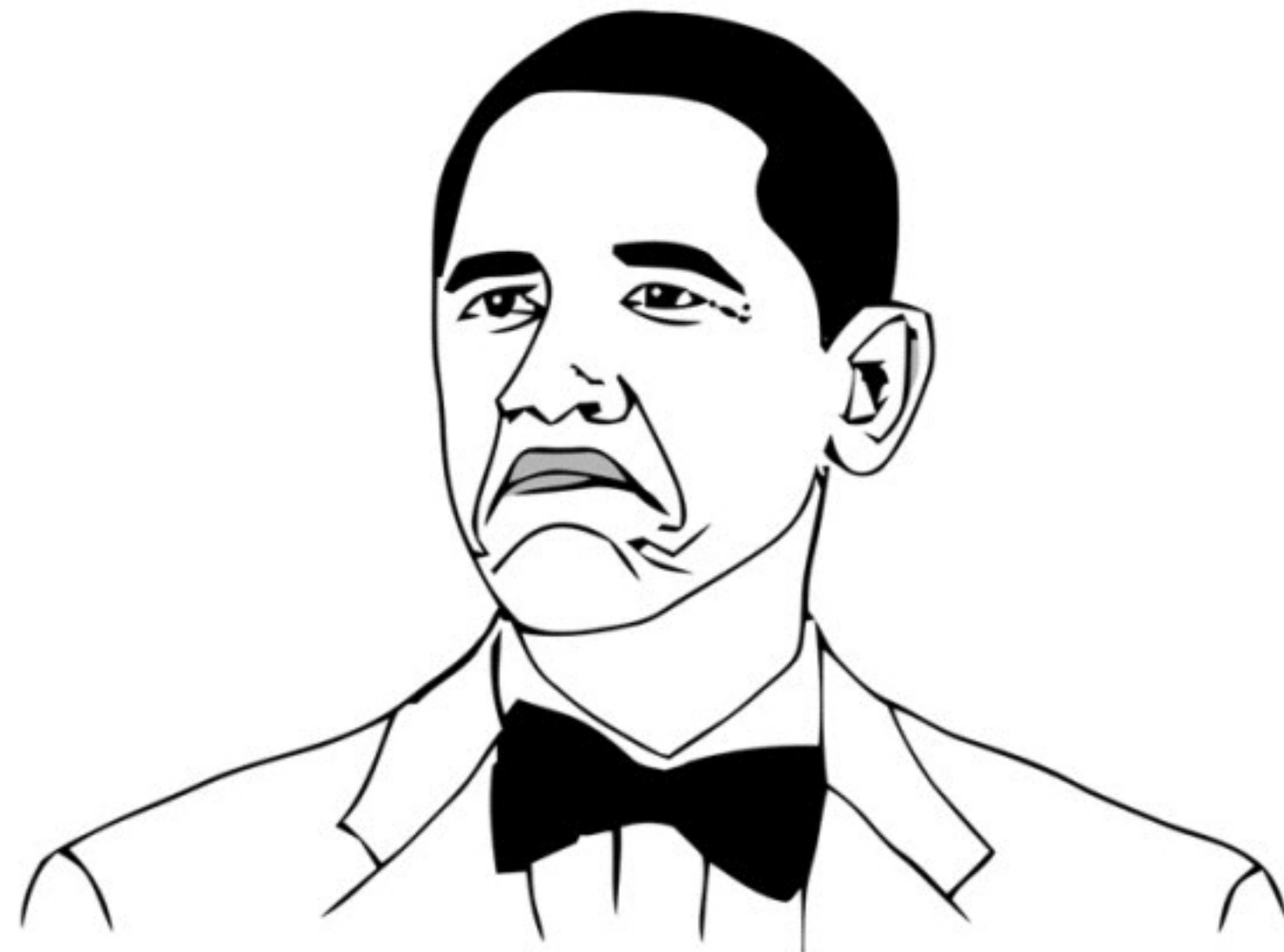


```
1 <!doctype HTML>
2 <html>
3   <head>
4     <title>Game</title>
5   </head>
6   <body>
7     <script src="scripts/phaser.min.js"></script>
8     <script src="scripts/game.js"></script>
9   </body>
10 </html>
```

Line: 8:34 | HTML | Soft Tabs: 2

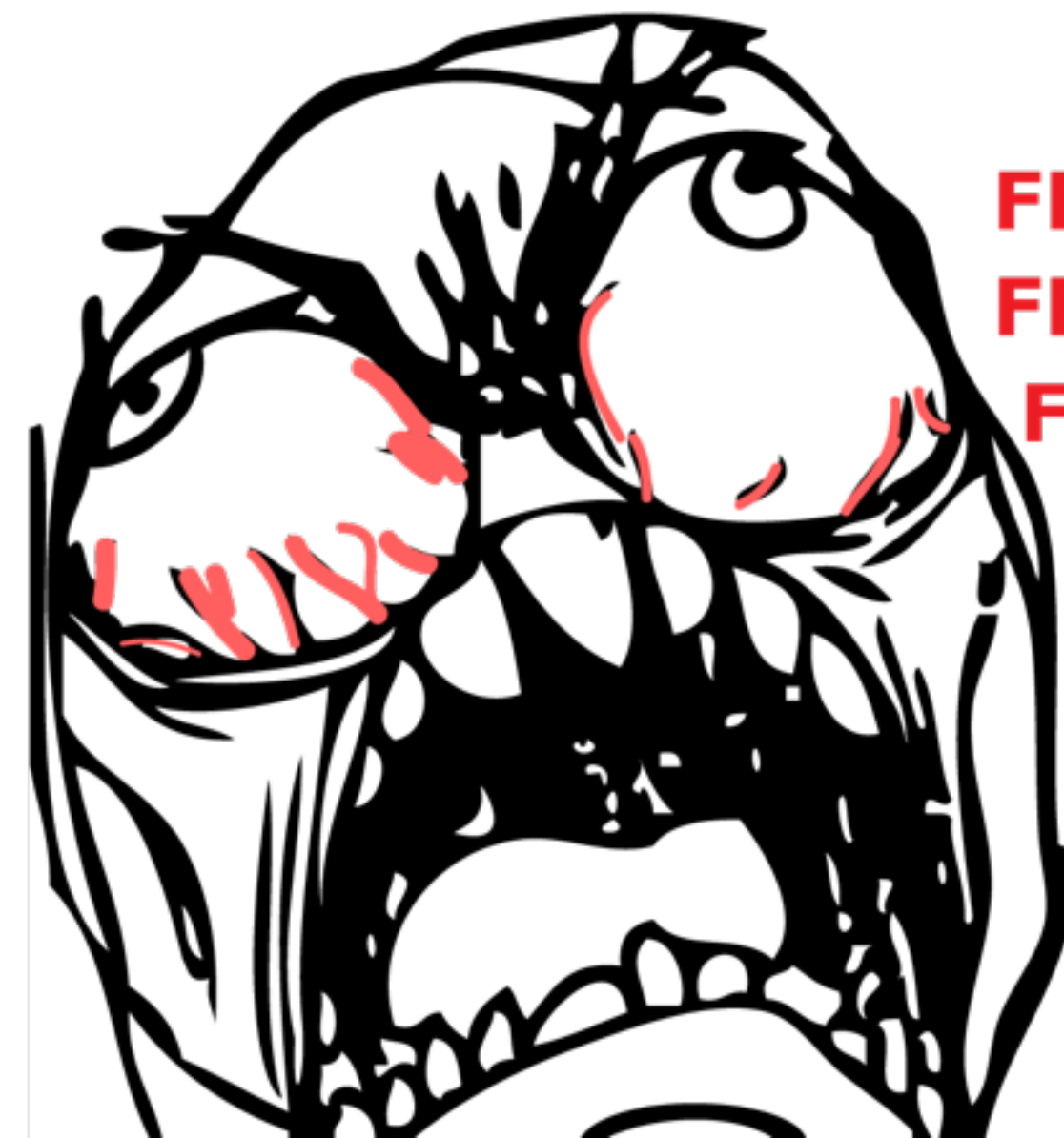
Comment your code

5 minutes after you write code
without comments




NOT BAD

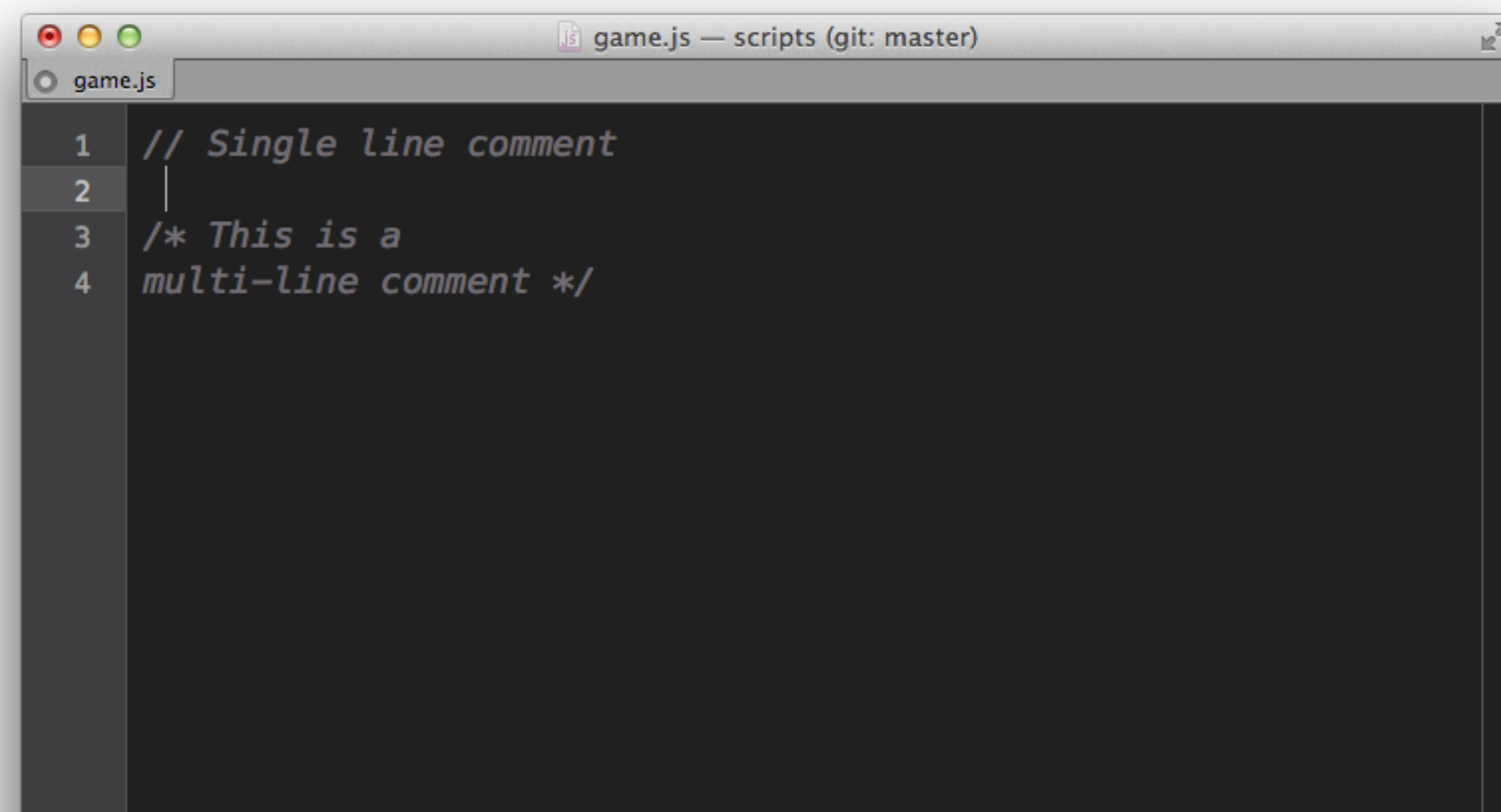
When you come back to it
in 3 weeks



FFFFFFF
FFFFFFF
FFFFFFF
FFFUU
UUUU
UUUU
UUUU
UUUU
UUUU-
UUUU-

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  + /)



```
game.js — scripts (git: master)
game.js
1 // Single line comment
2 |
3 /* This is a
4 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;

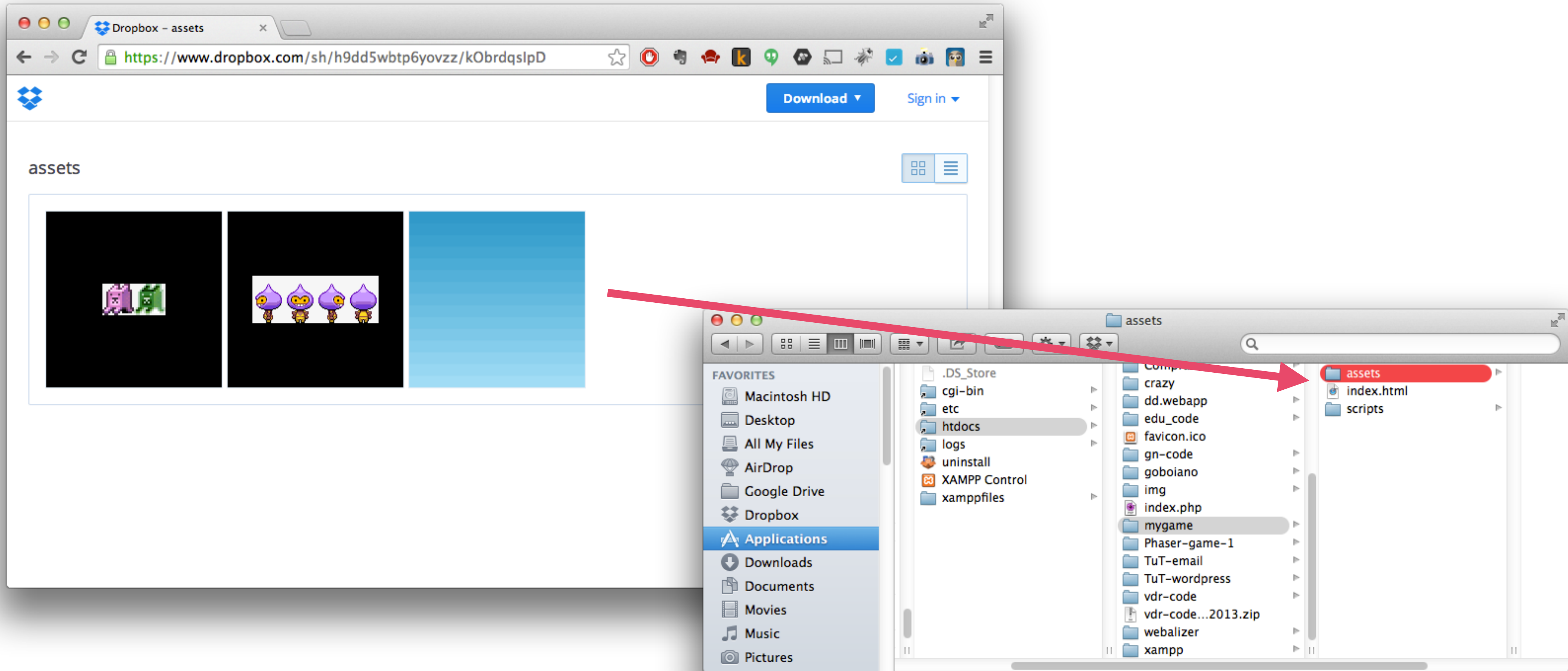
var division = 100/15;

preload() function

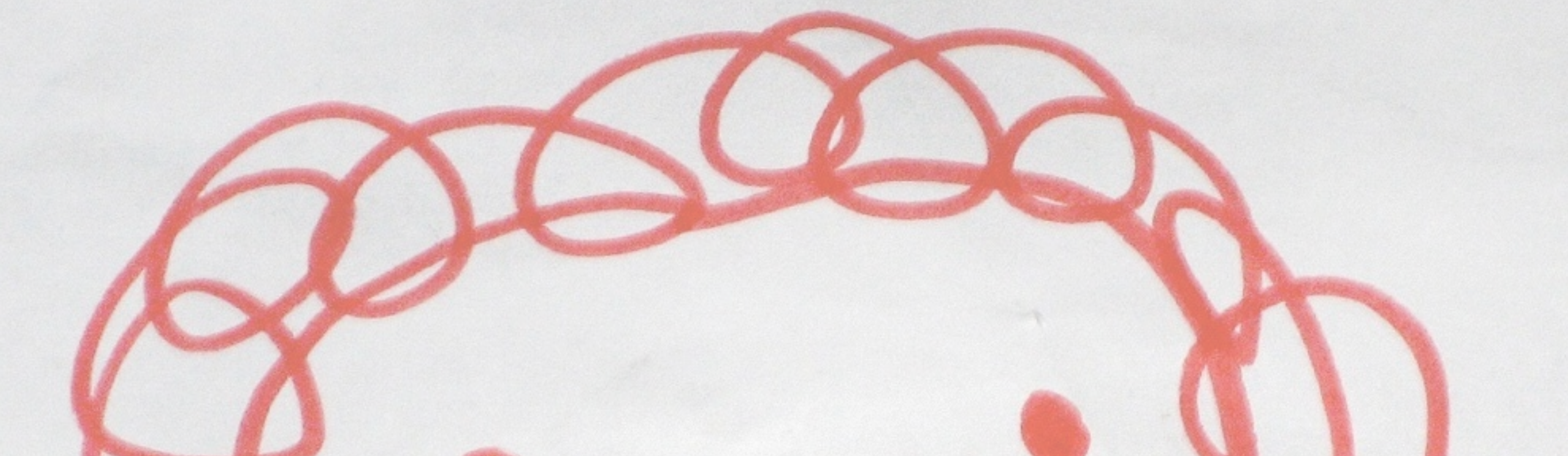


LOADING

Create assets folder



Download & save images from tinyurl.com/clf-html5-2014

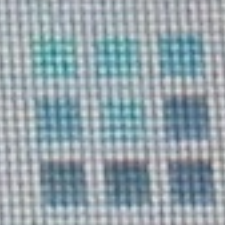


create() function



Updating. Please wait...

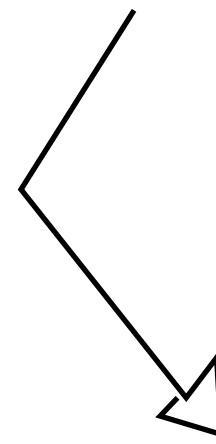
update() function



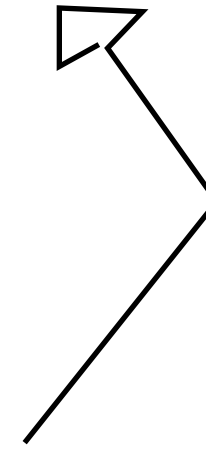
CREATE



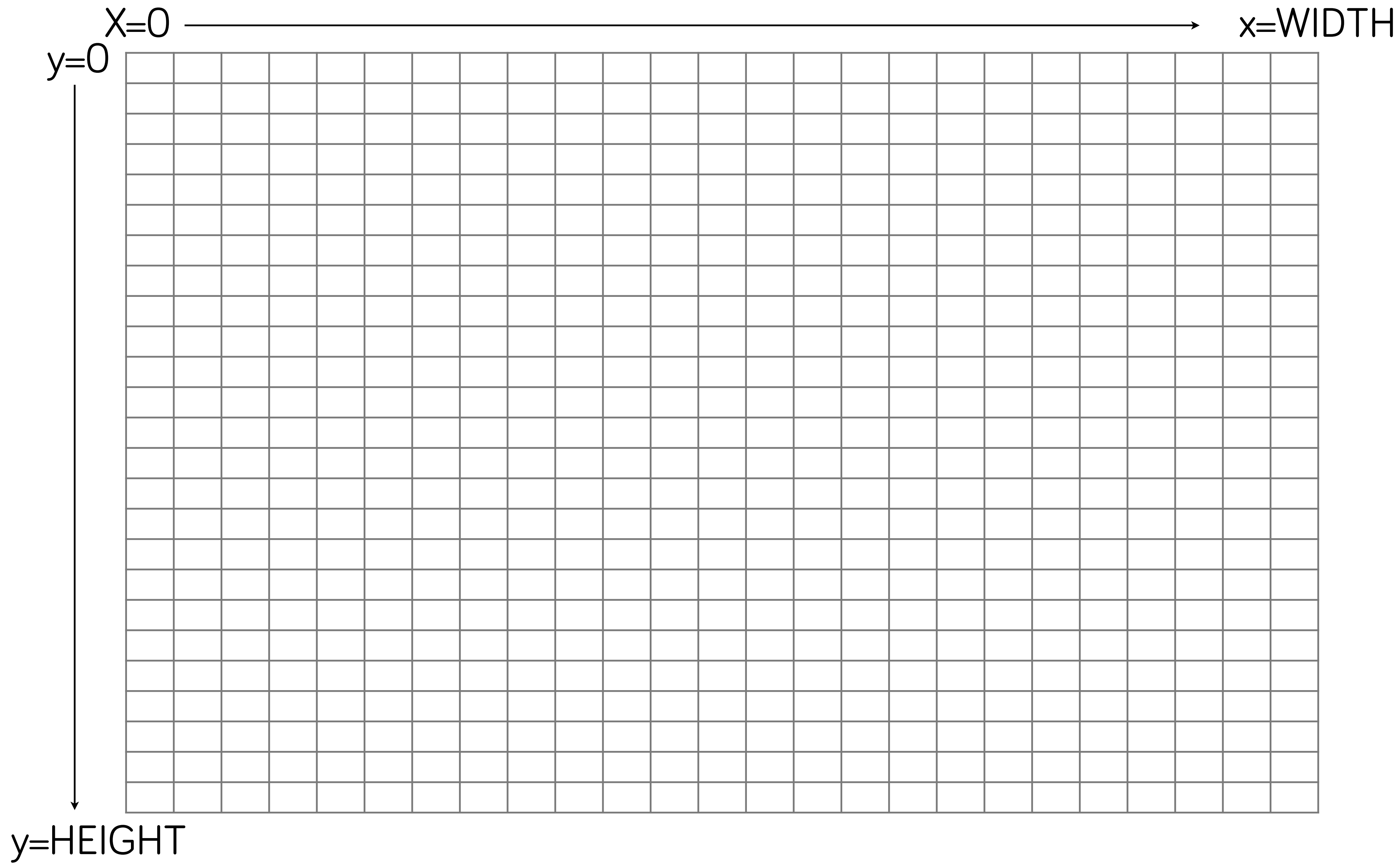
UPDATE



DRAW



How does positioning work?





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

