

Web App House Course HW 6

JavaScript in the browser

There are two parts of this assignment. The first is optional and includes writing about ~15 lines of code to finish Flappy Bird. The second is writing code (without JQuery) that causes images to move around a document.

Flappy Bird (Optional)

Objective: Finish building Flapping Bird, and learn a little bit about the DOM and events.

What you need to know:

This assignment will cover the DOM, events and mixins,

To 308 elitists

This entire game is inside one file. In fact, one function. Yes, this is terrible practice for a real game, which would normally use a separate JavaScript file that houses the game engine. However, my “game engine” is meant to be as simple as possible, so you spend your working on what we learn / will learn in class and not learning a game engine. (If you are interested in a real javascript game engine, check out [crafty.js](#).)

So let's get started.

Canvas and the DOM

Remember all the fun we had with HTML and CSS? Well it's back. If you open up index.html you will see a canvas element. Now, we want to write some javascript that will draw on this canvas. How do we get the canvas into our script? We use the DOM. I recommend reading at least the first of the following two readings (the first one takes about two minutes).

- <http://css-tricks.com/dom/>
- https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Canvas_tutorial

Task One

The first task of this assignment will be to load the canvas into flappy.js using the DOM. In the “canvas and scene variable section” (search for PART ONE) fill in the variables. If you need help go to:

https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Canvas_tutorial/Basic_usage

Task Two

Let’s see how you did. Open up index.html. Hopefully you see Flappy Bird take a nose-dive into the ground. So, it looks like we can’t control Flappy Bird! Let’s fix that. Navigate to PART TWO and add the specified listeners so that Flappy Bird jumps on clicks and space presses.

<https://developer.mozilla.org/en-US/docs/Web/API/EventTarget.addEventListener>

Task Three

Well, Flappy Bird can jump now. But the only way he can die is by hitting the ground. We need to add collision to the pipes. Navigate to PART THREE and use what you learned from class with Mixins to finish making the Pipe constructor by making it have the behavior of a `CollidableEntity`. We recommend reading what the function ‘`add_mixin`’ does.

When a pipe collides with `flappyBird`, we should set the `alive` property of `flappyBird` to false, and set the `dirty` property of `momentum` to true. (This will stop the game moving forward).

Task Four (Optional^2)

You should have a working game now. If you want make the game better. For example play around with the gravity, jump, and momentum forces. Or maybe create some special effects when Flappy Bird collides with the pipe. Or reskin the game by looking the code in `ImageBank` and `SpriteSheet`. I also recommend turning the variable `flappyKill` to true.

You Can’t Catch Me (Not Optional)

Use what you have learned about inheritance from last class, and the DOM and Events from this assignment’s readings to create a website with three pictures. Two of the pictures should be “move away” pictures. When the mouse gets close to the “move away” pictures they should move away from the mouse. The third picture

should “follow” the mouse around. In addition, if the user does not move the mouse for 5 seconds, the other two “move away” pictures will jump to a random new location. We recommend making the pictures funny.

We are mostly concerned about reacting to events and using the DOM in this assignment. If you can’t get a perfect “move away” algorithm that’s okay. Just make sure you are reacting to mouse events differently in the “move away” pictures than you do for the “follow” picture. Also, try to use either the Inheritance or mixin strategy from last class.

Hint: To move the pictures, make their position absolute and change their left, right, top or bottom properties.

Some other helpful readings

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/EventLoop>
<https://developer.mozilla.org/en-US/docs/Web/API/Window.setTimeout>

What to submit?

Submit the finished moveAway.html, and moveAway.js.