I need to practice mostly with my JavaScript. I want to change CSS properties and create animations using JavaScript. So I thought for this project I would try and make a game similar to the google dinosaur game. I made one similar in C# for ByteJam.

I referenced my JavaScript class project to see how I accomplished the animations on that website and found the setInterval function so I incorporated that. To make the cactus move and the player jump.

First I made a couple div’s to act as the canvas and objects in the game.

A screenshot of a computer screen

Description automatically generated

Then I used the setInterval in JavaScript to automate the game with out any input. I did this to find the minimum and maximum values I needed to get the movement I wanted.

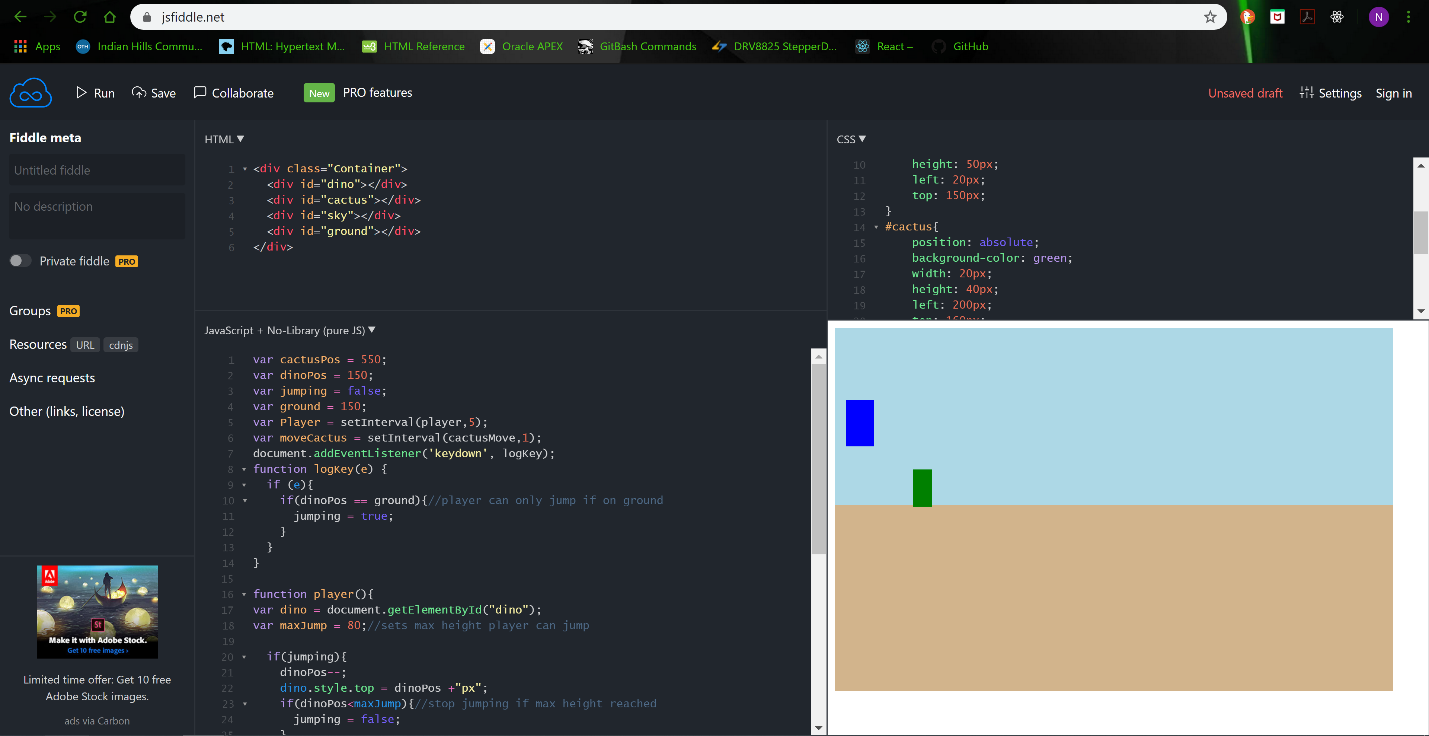
A screenshot of a computer screen

Description automatically generated

I incorporated a keydown event listener to control player jumping.

A screenshot of a computer screen

Description automatically generated



I added a hits counter. I created a variable using getEmementById and set the innerHTML to the hit counter variable.

A screenshot of a computer screen

Description automatically generated

I added more cactus and assigned them into an array. I used a parallel array to store their left style property and assigned them with javascript using a for loop. I used a similar loop to get all cactus moving to the left and resetting when they reach the end of the window. This has however broken my hits counter so I am going to see what I can do to fix that.

A screenshot of a computer screen

Description automatically generated

I fixed the hit counter by using a for loop to compare all cactus position to a position that would contact the player the function returns true if any of the cactus are in that position.

A screenshot of a computer screen

Description automatically generated

Most of the information I needed I found on w3schools website. This project used multiple setInterval functions, arrays, loops and functions. I used javaScript to set innerHTML values and change CSS properties of elements. It took a little bit to get everything to work together and it won’t be number one on the app store, but it worked in the end.

**JavaScript**

var dinoPos = 150;

var jumping = false;

var ground = 150;

var hitCtr = 0;

var Player = setInterval(player, 5);

var moveCactus = setInterval(cactusMove,1);

var Score = setInterval(keepScore, 120);

document.addEventListener('keydown', logKey);

var cactus = document.getElementsByClassName("cactus");//put all cactus in array

var cactusPos = [610,405,200];

for (var i = 0;i<3;i++){

cactus[i].style.left = cactusPos[i]+"px";

}

function logKey(e) { //listens for keydown

if (e) {

if (dinoPos == ground) { //player can only jump if on ground

jumping = true;

}

}

}

function player() {

var dino = document.getElementById("dino");

var maxJump = 80; //sets max height player can jump

if (jumping) {

dinoPos--;

dino.style.top = dinoPos + "px";

if (dinoPos < maxJump) { //stop jumping if max height reached

jumping = false;

}

} else {

if (dinoPos < ground) { //start falling

dinoPos++;

dino.style.top = dinoPos + "px";

}

}

}

function cactusMove() {

for (var i = 0;i<3;i++){

if (cactusPos[i] > 0) {

cactusPos[i]--;

cactus[i].style.left = cactusPos[i] + "px";

} else {

cactusPos[i] = 610;

cactus[i].style.left = cactusPos[i] + "px";

}

}

}

function keepScore() {

var hits = document.getElementById("hits");

if (hitCactus() && hitPlayer()) {

hitCtr++;

}

hits.innerHTML = hitCtr;

}

function hitCactus() {

retval = false;

for (var i = 0;i<3;i++){

if (cactusPos[i] < 50 && cactusPos[i] > 20)

retval = true;

}

return retval;

}

function hitPlayer() {

return (dinoPos > 120);

}

**HTML**

<div class="Container">

<div id="dino"></div>

<div class="cactus"></div>

<div class="cactus"></div>

<div class="cactus"></div>

<div id="scoreBox">

<span class="lable">Hits: <span id="hits"></span></span>

</div>

<div id="sky"></div>

<div id="ground"></div>

</div>

**CSS**

.container{

width: 100%;

height: 100%;

position: relative;

}

#dino{

position: absolute;

background-color: blue;

width: 30px;

height: 50px;

left: 20px;

top: 150px;

}

.cactus{

position: absolute;

background-color: green;

width: 20px;

height: 40px;

top: 160px;

}

#ground{

width: 600px;

height: 200px;

background-color: tan;

}

#sky{

width: 600px;

height: 170px;

background-color: lightblue;

}

#scoreBox{

position: relative;

top: 40px;

left: 500px;

width: 70px;

}