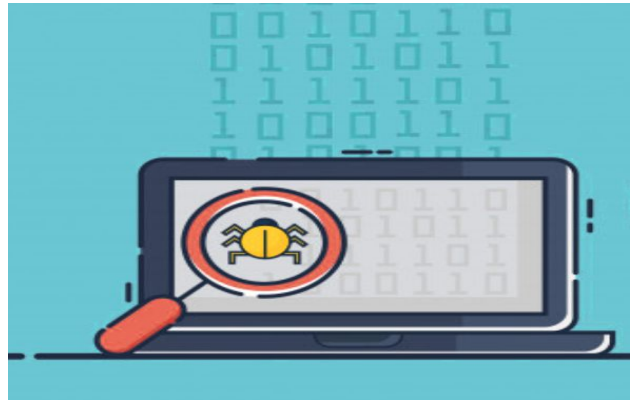


INFINITE GAME WORLD



What is our GOAL for this MODULE?

We created a Trex Game similar to what we see in the Google Chrome browser when it is not connected to the internet.

What did we ACHIEVE in the class TODAY?

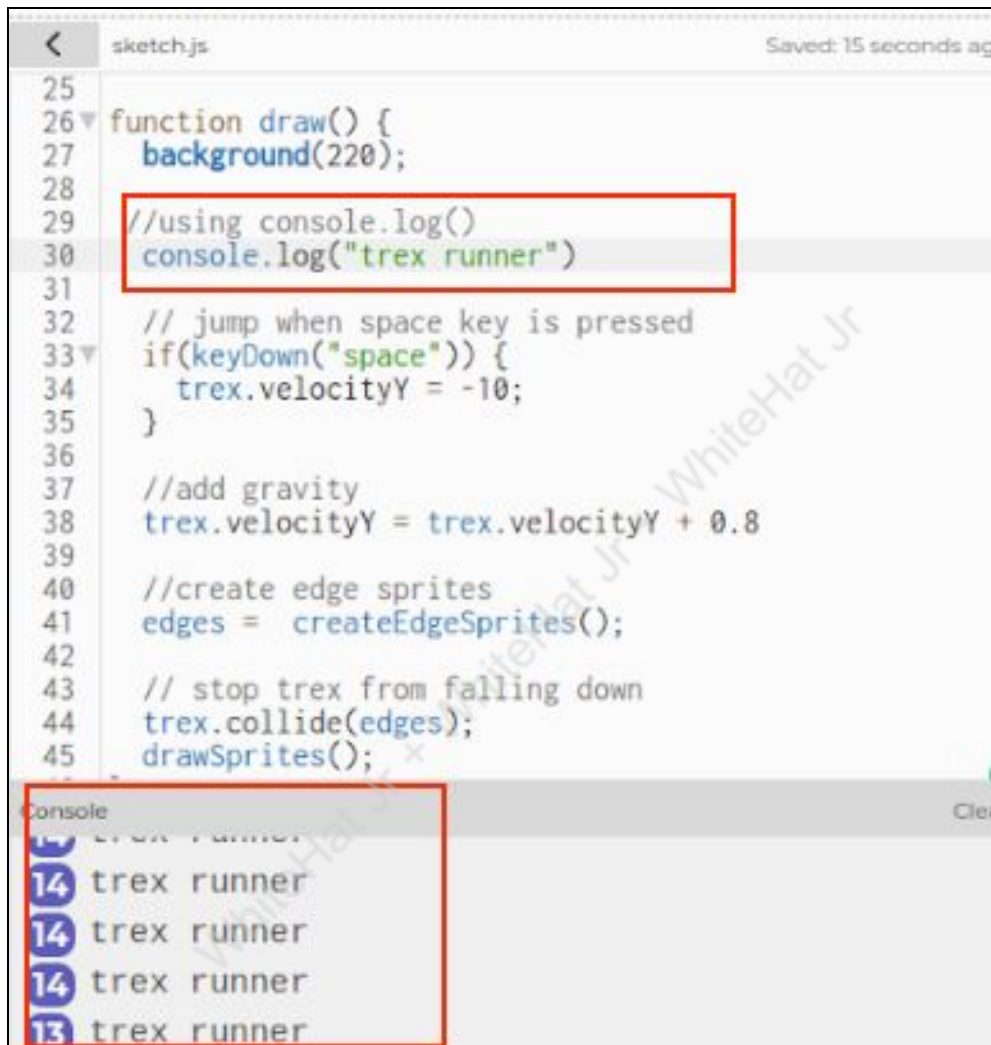
- Learned to log messages/outputs from the program into the console for testing purposes.
- Learned to create an infinitely scrolling ground for the dinosaur to run on.
- Learned to identify an additional condition needed in the program to stop the T rex from jumping again while it is in the air.
- Created an invisible ground sprite to make the T rex run below the ground.

Which CONCEPTS/ CODING BLOCKS did we cover today?

- Logging messages on the console using `console.log()` to test programs.
- Using logic to create a ground that gives the perception of scrolling infinitely.

How did we DO the activities?

1. Learn to use **console.log()**. The P5 editor has a console window where we can log any message while the program is running. We do this using **console.log()** instruction.

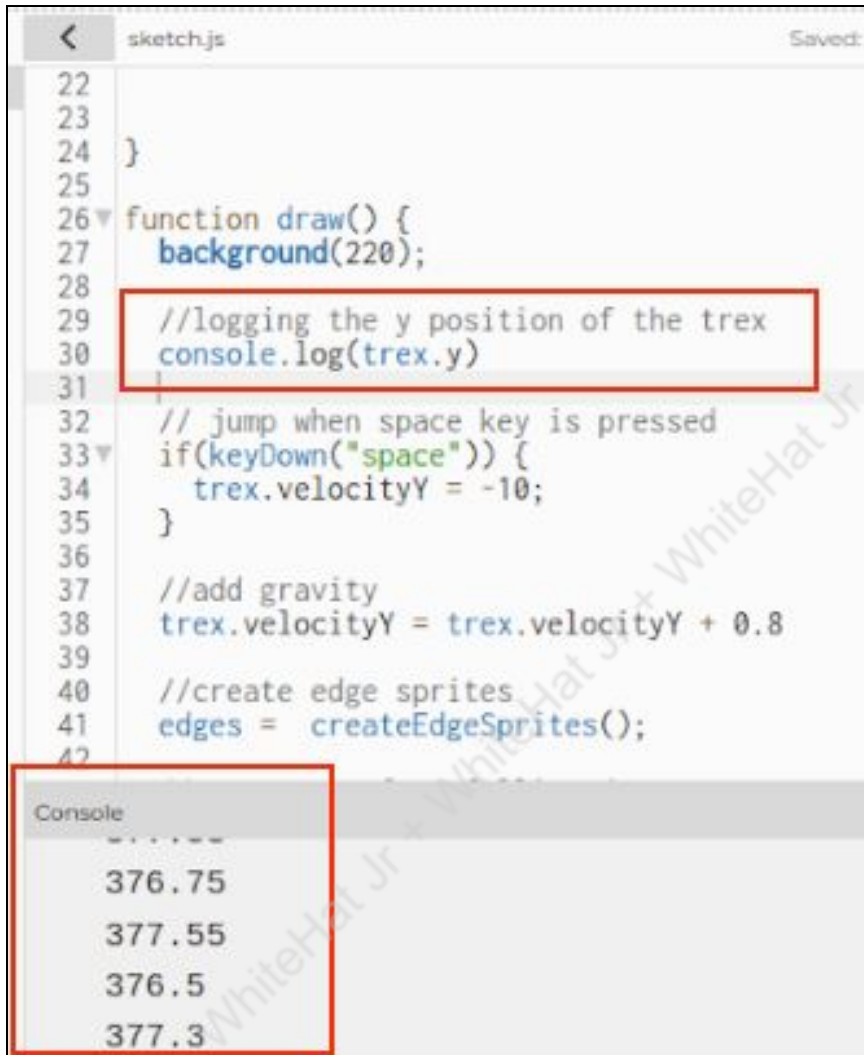


The screenshot shows the P5.js editor interface. The code editor displays the following JavaScript code:

```
25
26 function draw() {
27   background(220);
28
29   //using console.log()
30   console.log("trex runner")
31
32   // jump when space key is pressed
33   if(keyDown("space")) {
34     trex.velocityY = -10;
35   }
36
37   //add gravity
38   trex.velocityY = trex.velocityY + 0.8
39
40   //create edge sprites
41   edges = createEdgeSprites();
42
43   // stop trex from falling down
44   trex.collide(edges);
45   drawSprites();
```

The console window at the bottom shows the output of the `console.log()` statement, displaying the message "trex runner" multiple times, indicating that the code is running and logging the message successfully.

2. Write the **console.log()** instruction inside the **draw()** function. Try logging the y-position of the Trex sprite when it jumps.



```
22
23
24 }
25
26 function draw() {
27   background(220);
28
29   //logging the y position of the trex
30   console.log(trex.y)
31
32   // jump when space key is pressed
33   if(keyDown("space")) {
34     trex.velocityY = -10;
35   }
36
37   //add gravity
38   trex.velocityY = trex.velocityY + 0.8
39
40   //create edge sprites
41   edges = createEdgeSprites();
42
```

Console

```
.....
376.75
377.55
376.5
377.3
```

3. Move the dinosaur. Give a backward velocity to the ground; add the code to reset the ground.

```
> sketch.js
18 //adding scale and position to trex
19 trex.scale = 0.5;
20 trex.x = 50
21
22 //create ground sprite
23 ground = createSprite(200,380,400,20);
24 ground.addImage("ground",groundImage);
25 ground.x = ground.width /2;
26 }
27
28 function draw() {
29   background(220);
30
31   ground.velocityX = -2
32   console.log(ground.x)
33
34   if (ground.x<0){
35     ground.x = ground.width/2;
36   }
37
38   //jumping the trex on space key press
39   if(keyDown("space")) {
40     trex.velocityY = -10;
41   }
42
43   trex.velocityY = trex.velocityY + 0.8
44 }
```

4. Use an actual ground image.

```
> sketch.js Saved: just now
18 //adding scale and position to trex
19 trex.scale = 0.5;
20 trex.x = 50
21
22 //create ground sprite
23 ground = createSprite(200,380,400,20);
24 ground.addImage("ground",groundImage);
25 ground.x = ground.width /2;
26 }
27
28 function draw() {
29   background(220);
30
31   ground.velocityX = -2
32   console.log(ground.x)
33
34   if (ground.x<0){
35     ground.x = ground.width/2;
36   }
37 }
```

5. Create an invisible ground sprite just below the actual ground sprite since the dinosaur is running above the ground during bug fixes.

```
sketch.js Saved: 18 minutes ago
12 createCanvas(400, 400);
13
14 //create a trex sprite
15 trex = createSprite(50,380,20,50);
16 trex.addAnimation("running", trex_running);
17 trex.scale = 0.5;
18
19 //create a ground sprite
20 ground = createSprite(200,380,400,20);
21 ground.addImage("ground",groundImage);
22 ground.x = ground.width /2;
23 ground.velocityX = -2;
24
25 //creating invisible ground
26 invisibleGround = createSprite(200,390,400,10);
27
28 }
29
```

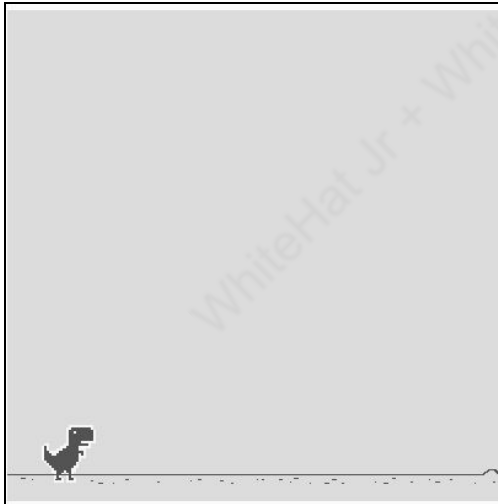
6. Collide the Trex with the invisible ground.

```
sketch.js Sav
23 ground.velocityX = -2;
24
25 //creating invisible ground
26 invisibleGround = createSprite(200,390,400,10);
27
28 }
29
30 function draw() {
31 //set background color
32 background(220);
33
34 //jump when the space key is pressed
35 if(keyDown("space")) {
36 trex.velocityY = -10;
37 }
38
39 //add gravity
40 trex.velocityY = trex.velocityY + 0.8
41
42 if (ground.x < 0){
43 ground.x = ground.width/2;
44 }
45
46 //stop trex from falling down
47 trex.collide(invisibleGround);
48
49 drawSprites();
50 }
Console
```

7. Add the following line of code anywhere outside the function draw() and after creating the invisible Ground Sprite: invisibleGround.visible = false;

```
29 //create a ground sprite
30 ground = createSprite(200,180,400,20);
31 ground.addImage("ground",groundImage);
32 ground.x = ground.width /2;
33 ground.velocityX = -4;
34
35 //creating invisible ground
36 invisibleGround = createSprite(200,190,400,10);
37 invisibleGround.visible = false;
38
39 }
40
41 function draw() {
42 //set background color
43 background(220);
44
```

Output:



8. Add an additional condition inside the if block where we make the Trex jump only when it is on the ground while fixing the second bug.

```

> sketch.js
27 //creating invisible ground
28 invisibleGround = createSprite(200,390,400,10);
29 invisibleGround.visible = false;
30 }
31
32 function draw() {
33   //set background color
34   background(220);
35
36   console.log(trex.y)
37
38   //jump when the space key is pressed
39   if(keyDown("space") && trex.y >= 362) {
40     trex.velocityY = -10;
41   }
42
43   //add gravity
44   trex.velocityY = trex.velocityY + 0.8
45
46   if (ground.x < 0){
47     ground.x = ground.width/2;
48   }
  
```

9. Use **console.count()** to count how many times a particular program is called.

```

31 }
32 }
33
34 function draw() {
35
36   //set background color
37   background(220);
38
39
40   console.count("Draw frame is called:");
41
42
43   //jump when the space key is pressed
44   if(keyDown("space") && trex.y >= 362) {
45     trex.velocityY = -10;
46   }
47
  
```

Console

```

Draw frame is called:: 80
Draw frame is called:: 81
Draw frame is called:: 82
Draw frame is called:: 83
  
```

10. Use **console.time()** to keep a log of the time and **console.timeEnd()** to stop and print the time on the console.

- **console.time()** when the **draw()** function starts:

```
> sketch.js
32 }
33
34 function draw() {
35   console.time();
36
37   //set background color
38   background(220);
39
40   //jump when the space key is pressed
41   if(keyDown("space") && trex.y >= 362) {
42     trex.velocityY = -10;
43   }
44
45   //add gravity
46   trex.velocityY = trex.velocityY + 0.8
47
48   if (ground.x < 0){
49     ground.x = ground.width/2;
```

- **console.timeEnd()** when the **draw()** function ends:

```
41   //jump when the space key is pressed
42   if(keyDown("space") && trex.y >= 362) {
43     trex.velocityY = -10;
44   }
45
46   //add gravity
47   trex.velocityY = trex.velocityY + 0.8
48
49   if (ground.x < 0){
50     ground.x = ground.width/2;
51   }
52
53   //stop trex from falling down
54   trex.collide(invisibleGround);
55
56   drawSprites();
57   console.timeEnd();
58
59 }
```

Console

```
default: 0.2800000074785203ms
default: 0.09499999578110874ms
default: 0.14999997802078724ms
default: 0.3549999964889139ms
```


11. Find the time taken by functions **setup()** or **preload()** to run before the game starts.

```
29 invisibleGround.visible = false;
30 }
31
32 function draw() {
33
34   console.time();
35   for(var i=0; i<100; i++){
36     console.log("Running Loop");
37   }
38   //set background color
39   background(220);
40
41   //jump when the space key is pressed
42   if(keyDown("space") && trex.y >= 362) {
43     trex.velocityY = -10;
44   }
45
46   //add gravity
47   trex.velocityY = trex.velocityY + 0.8
48 }
```

Console

```
100Running Loop
default: 4.374999989522621ms
100Running Loop
default: 9.765000024344772ms
```

12. Use **console.log()** to print a simple message; **console.warn()** to print a warning; **console.error()** to print errors; and **console.info()** to print any information.

```
29 invisibleGround.visible = false;
30 }
31
32 function draw() {
33
34   console.info("Start of the draw function");
35   console.error("This is how error appears");
36   console.warn("A warning!");
37   //set background color
38   background(220);
39
40   //jump when the space key is pressed
41   if(keyDown("space") && trex.y >= 362) {
42     trex.velocityY = -10;
  }
```

Console

- ⚠ A warning!
- ℹ Start of the draw function
- ✖ This is how error appears
- ⚠ A warning!

What's next?

We will start creating floating clouds on different heights.

Extend Your Knowledge:

1. [Console in Javascript](#): Read more about the console.