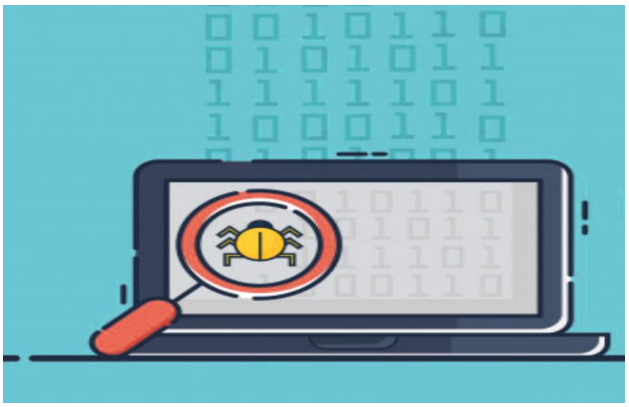
Code Debugging and Code Indentation



What is our GOAL for this MODULE?

Create the Trex Game similar to what we see in chrome browsers when not connected to the internet.

What did we ACHIEVE in the class TODAY?

- Learn to indent the code correctly to make it more readable.
- Learn to identify an additional condition needed in the program to stop the T-Rex from jumping again while it is in the air.
- Create an invisible ground sprite to make the T-Rex run below the ground.

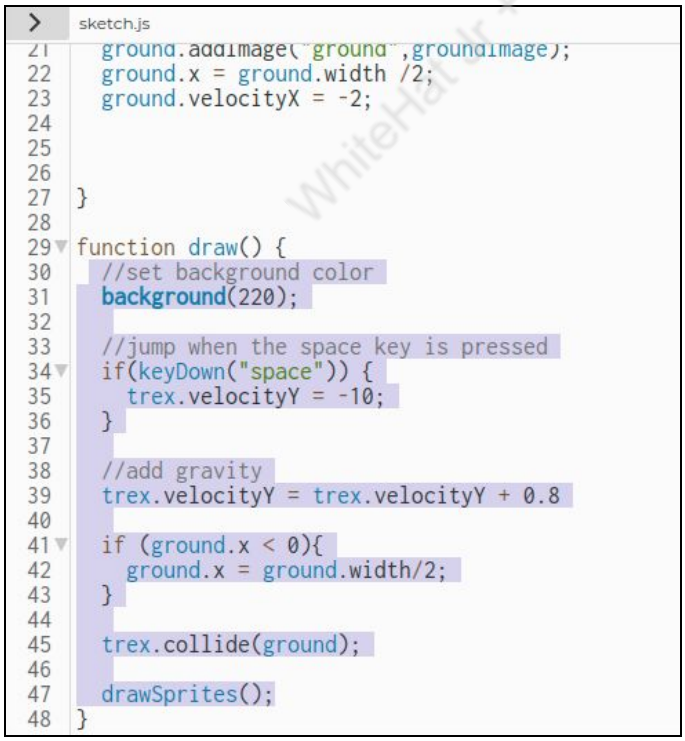
Which CONCEPTS/ CODING BLOCKS did we cover today?

- Code indentation
- Identifying bugs in the program
- Debugging the code

How did we DO the activities?

Step 1: Code Indentation— Leave a space after every meaningful line of code.

Leave an even space after every instruction contained inside another block of code.



Step 2: Fix bugs

Bug 1: The dinosaur is running above the ground: Let us create an invisible ground sprite just below the actual ground sprite.

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
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
```

Console

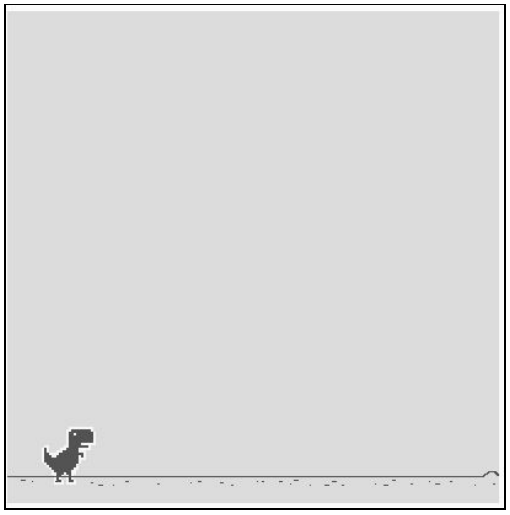
Instead of supporting the T-Rex on the ground, collide it with the invisible ground.

sketch.js

```
  ground.velocityX = -2;

  //creating invisible ground
  invisibleGround = createSprite(200,390,400,10);

}

function draw() {
  //set background color
  background(220);

  //jump when the space key is pressed
  if(keyDown("space")) {
    trex.velocityY = -10;
  }

  //add gravity
  trex.velocityY = trex.velocityY + 0.8

  if (ground.x < 0){
    ground.x = ground.width/2;
  }

  //stop trex from falling down
  trex.collide(invisibleGround);

  drawSprites();
}
```

Console

Step 3: Add the following line of code anywhere outside the function draw() and after creating the invisible Ground Sprite: invisibleGround.visible = false;



Step 4:
Bug 2: The Trex jumps even when it is in the air!
Add an additional condition inside the IF block where we make the T_Rex jump only when it is on the ground.

```
sketch.js                                                    Saved: jus
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    }
49
50    //stop trex from falling down
51    trex.collide(invisibleGround);
52
53    drawSprites();
54  }
```

There are several other ways consoles can be used.

1. We can use console.count() to count how many times a particular program is called.

```
27    //creating invisible ground
28    invisibleGround = createSprite(200,390,400,10);
29    invisibleGround.visible = false;
30
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
```

2. We use console.time() to start keeping 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
36    console.time();
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
49    if (ground.x < 0){
```

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
```

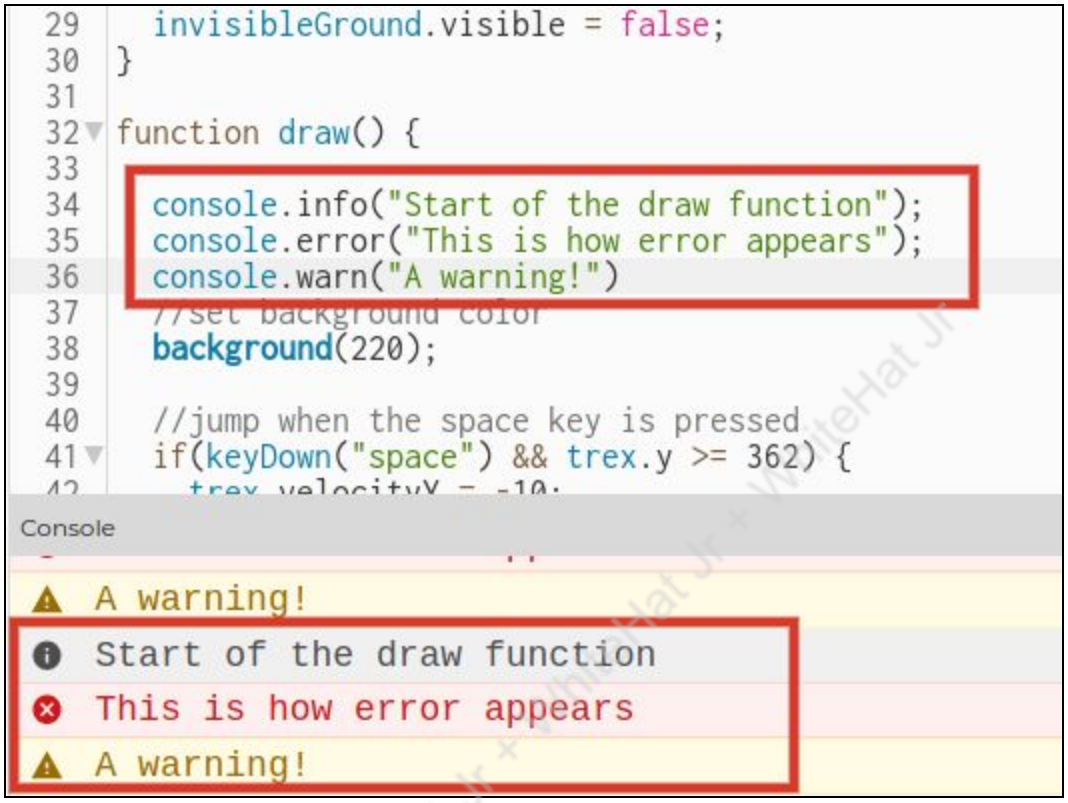
3. You can also find out how long it takes for function setup or function preload to run before your game can start.

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

Console

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

4. Console.log() is used to print a simple message. You can use console.warn() to print a warning. The warning message is formatted differently. Similarly, you can use console.error() to print an error(). the error message is formatted differently. You can also use console.info() to print any information.



What's next?

We will start creating floating clouds at different heights.