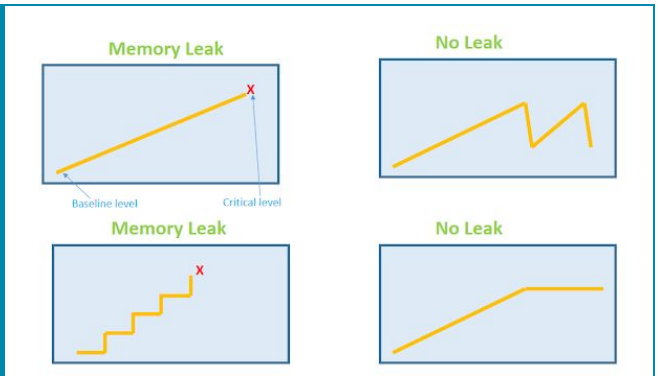
MEMORY LEAK AND SWITCH STATEMENTS



What is our GOAL for this MODULE?

We have solved the memory leak problem and learned the usage of switch statements.

What did we ACHIEVE in the class TODAY?

- Corrected the memory leak problem in code.
- Used switch statements to randomly spawn different kinds of obstacles in the game.
- Designed a simple scoring system.
- Used string concatenation for adding the score

Which CONCEPTS/ CODING BLOCKS did we cover today?

- String concatenation.
- Scoring system.
- Correcting memory leaks.
- Switch statements

How did we DO the activities?

1. Spawn different kinds of obstacles on the way in the Trex runner game.
    - Assign lifetime to each cloud variable which is getting created. (Formula: Time = Distance/ Speed; 400 /3 = 134)

```
function spawnClouds() {
  //write code here to spawn the clouds
  if (frameCount % 60 === 0) {
    var cloud = createSprite(600,300,40,10);
    cloud.addImage(cloudImage)
    cloud.y = Math.round(random(280,320))
    cloud.scale = 0.4;
    cloud.velocityX = -3;

    //assign lifetime to the variable
    cloud.lifetime = 134;
```

2. Print a string on the console
   When any text information is stored in a computer, it is written inside quotes "_" and called a String.

```
  invisibleGround = createSprite(200,190,400,10);
  invisibleGround.visible = false;

  console.log("Hello")
}

function draw() {
  background(180);


  if(keyDown("space")&& trex.y >= 100) {
    trex.velocityY = -10;
  }
```

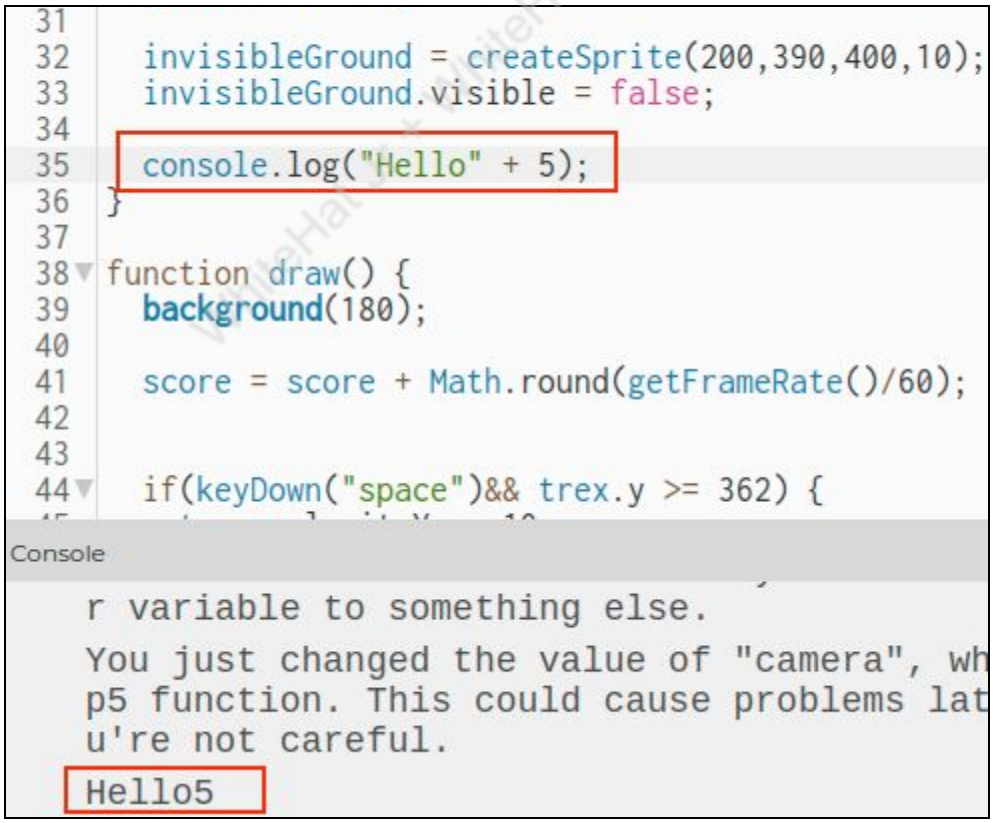
Console Clear

```
p5 had problems creating the global function "Animation", pos
sibly because your code is already using that name as a varia
ble. You may want to rename your variable to something else.

You just changed the value of "camera", which was a p5 functi
on. This could cause problems later if you're not careful.
Hello
```

3. Join two strings together using the '+' sign.

- Use a word and number together too.



4. Create an empty function called **spawnObstacles()** and use it inside the **draw()** function.

```
62
63    trex.collide(invisibleGround);
64
65    //spawn the clouds
66    spawnClouds();
67
68    //spawn obstacles on the ground
69    spawnObstacles();
70
71    drawSprites();
72  }
73
74  function spawnObstacles(){
75
76  }
```

5. Create an obstacle sprite every 60 frames or so and give the obstacle the same velocity as the ground.

```
55
56    //spawn the clouds
57    spawnClouds();
58
59    //spawn obstacles on the ground
60    spawnObstacles();
61
62    drawSprites();
63  }
64
65  function spawnObstacles(){
66   if (frameCount % 60 === 0){
67     var obstacle = createSprite(400,365,10,40);
68     obstacle.velocityX = -6;
69   }
70  }
71
72  function spawnClouds() {
73    //write code here to spawn the clouds
```

6. Generate and store a random number between 1 to 6.
    - Use switch statement to assign different obstacle animations for the obstacle sprites randomly.

```
73  if (frameCount % 60 === 0){
74    var obstacle = createSprite(400,365,10,40);
75    obstacle.velocityX = -6;
76
77    //generate random obstacles
78    var rand = Math.round(random(1,6));
79    switch(rand) {
80      case 1: obstacle.addImage(obstacle1);
81              break;
82      case 2: obstacle.addImage(obstacle2);
83              break;
84      case 3: obstacle.addImage(obstacle3);
85              break;
86      case 4: obstacle.addImage(obstacle4);
87              break;
88      case 5: obstacle.addImage(obstacle5);
89              break;
90      case 6: obstacle.addImage(obstacle6);
91              break;
92      default: break;
93    }
94
95    //assign scale and lifetime to the obstacle
96    obstacle.scale = 0.5;
97    obstacle.lifetime = 300;
98  }
```

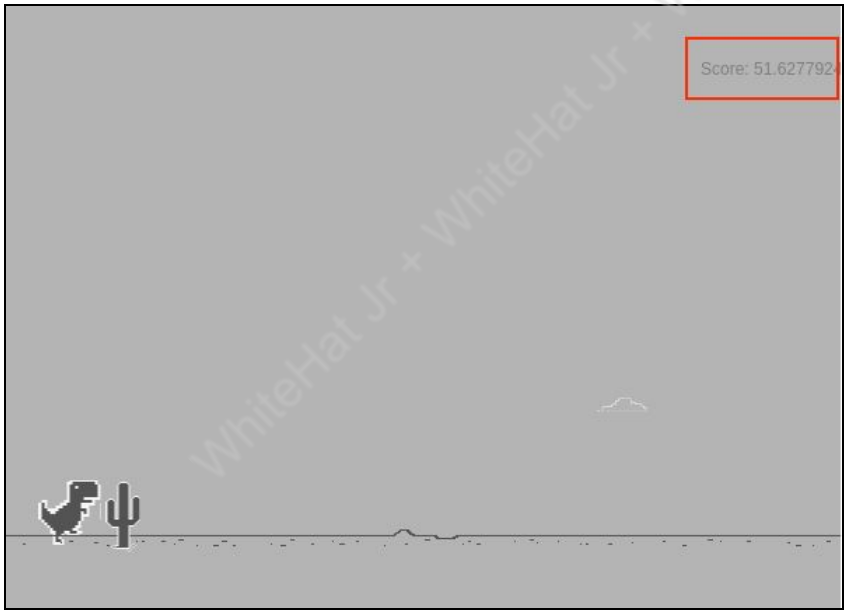
7. Scale the obstacles by half and give them a lifetime.

```
77    //generate random obstacles
78    var rand = Math.round(random(1,6));
79    switch(rand) {
80      case 1: obstacle.addImage(obstacle1);
81              break;
82      case 2: obstacle.addImage(obstacle2);
83              break;
84      case 3: obstacle.addImage(obstacle3);
85              break;
86      case 4: obstacle.addImage(obstacle4);
87              break;
88      case 5: obstacle.addImage(obstacle5);
89              break;
90      case 6: obstacle.addImage(obstacle6);
91              break;
92      default: break;
93    }
94
95    //assign scale and lifetime to the obstacle
96    obstacle.scale = 0.5;
97    obstacle.lifetime = 300;
98  }
99 }
```

8. Build a simple scoring system using the frame count as the score.

```
42
43    console.log("Hello" + 5);
44
45    score = 0;
46  }
47
48  function draw() {
49    background(180);
50    text("Score:  "+ score, 500,50);
51    score = score + (frameCount/60);
52
53
```

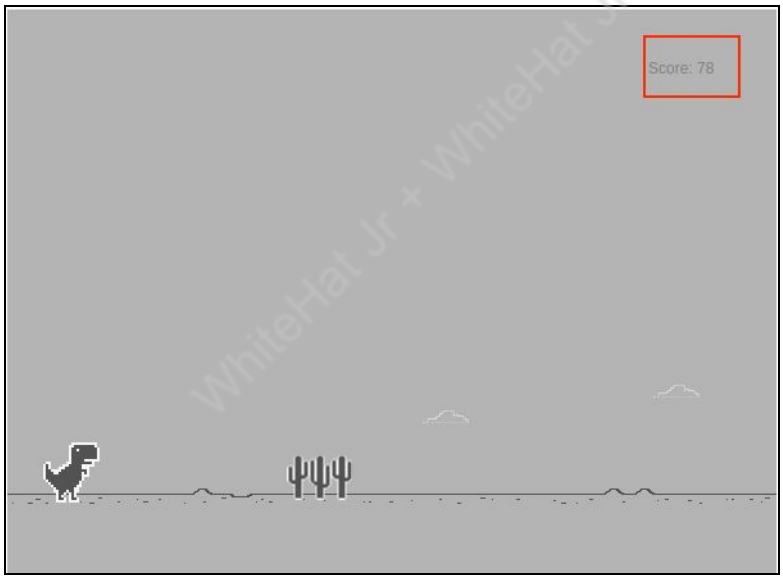
Output:



9. Use Math.round() function for rounding the score.

```
43    console.log("Hello" + 5);
44
45    score = 0;
46  }
47
48  function draw() {
49    background(180);
50    text("Score: "+ score, 500,50);
51    score = score + Math.round(frameCount/60);
52
53
54    if(keyDown("space")&& trex.y < 362) {
55      trex.velocityY = -13;
56    }
```

Output:



What's next?

We will build collisions with the obstacles and use game states.

Extend Your Knowledge:

1. P5 Functions: Read more about the different functions of p5.play