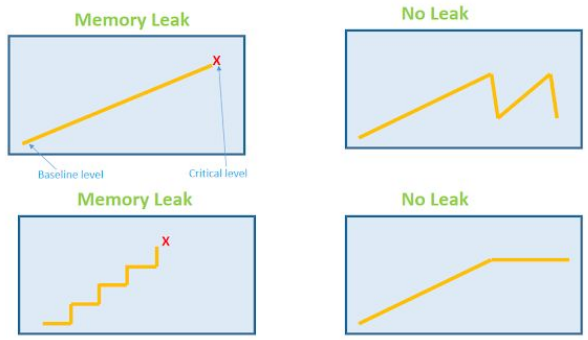
Memory leak and String concatenation



What is our GOAL for this MODULE?

Solve the memory leak problem and learn the usage of string concatenation.

What did we ACHIEVE in the class TODAY?

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

Which CONCEPTS/ CODING BLOCKS did we cover today?

- String concatenation
- Scoring system
- correcting memory leak

How did we DO the activities?

Step 1: Spawn different kinds of obstacles on the way in the T-Rex 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;

    //adjust the depth
    cloud.depth = trex.depth
    trex.depth = trex.depth + 1;
    }
}
```

Step 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.

Step 3: String Concatenation
Two strings can be joined together using + sign. Like this: "Hello" and "World".

```
27    ground = createSprite(200,380,400,20);
28    ground.addImage("ground",groundImage);
29    ground.x = ground.width /2;
30    ground.velocityX = -4;
31
32    invisibleGround = createSprite(200,390,400,10);
33    invisibleGround.visible = false;
34
35    console.log("Hello" + "World");
36  }
37
38  function draw() {
39    background(180);
40
41    score = score + Math.round(getFrameRate()/60);
42
43
44    if(keyDown("space")&& trex.y >= 362) {
```

Console

```
r variable to something else.
You just changed the value of "camera", which
p5 function. This could cause problems later
u're not careful.
HelloWorld
```

A word and number can also be used together.

```
27    ground = createSprite(200,380,400,20);
28    ground.addImage("ground",groundImage);
29    ground.x = ground.width /2;
30    ground.velocityX = -4;
31
32    invisibleGround = createSprite(200,390,400,10);
33    invisibleGround.visible = false;
34
35    console.log("Hello" + 5);
36  }
37
38  function draw() {
39    background(180);
40
41    score = score + Math.round(getFrameRate()/60);
42
43
44    if(keyDown("space")&& trex.y >= 362) {
```

Console

```
r variable to something else.
You just changed the value of "camera", wh
p5 function. This could cause problems lat
u're not careful.
Hello5
```

Step 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  }
```

Step 5: Create an obstacle sprite every 60 frames or so.
Give the obstacle the same velocity as the ground. The obstacles need to move with 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
```

Step 6: Generate and store a random number between 1 to 6.
Use string concatenation to randomly assign different obstacle animations for the obstacle sprites.

```
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  }
```

Scale the obstacles by half and give them a lifetime.

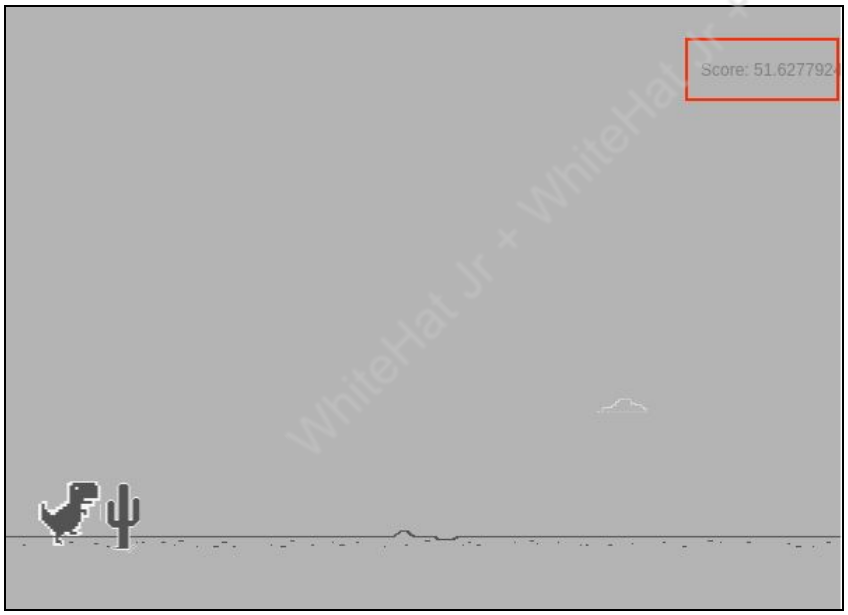
```
function spawnObstacles(){
 if (frameCount % 60 === 0){
   var obstacle = createSprite(400,365,10,40);
   obstacle.velocityX = -6;

    //generate random obstacles
    var rand = Math.round(random(1,6));
    switch(rand) {
      case 1: obstacle.addImage(obstacle1);
              break;
      case 2: obstacle.addImage(obstacle2);
              break;
      case 3: obstacle.addImage(obstacle3);
              break;
      case 4: obstacle.addImage(obstacle4);
              break;
      case 5: obstacle.addImage(obstacle5);
              break;
      case 6: obstacle.addImage(obstacle6);
              break;
      default: break;
    }

    //assign scale and lifetime to the obstacle
    obstacle.scale = 0.5;
    obstacle.lifetime = 300;
 }
}
```

Build a simple scoring system. We can use the frameCount as the score.

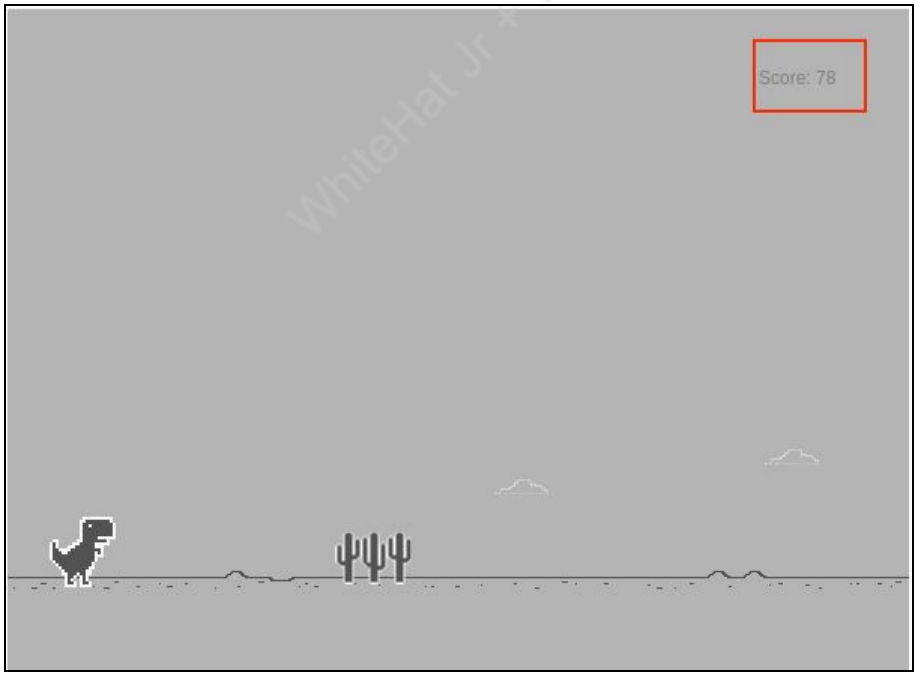
```
41
42    console.log("Hello" + 5);
43
44    score = 0;
45  }
46
47  function draw() {
48    background(180);
49    text("Score: "+ score, 500,50);
50    score = score + (getFrameRate()/60);
51
52
53    if(keyDown("space")&& trex.y >= 362) {
54      trex.velocityY = -10;
55    }
56
```



Step 7: Take the frameCount as the score.
Code:

```
38
39    invisibleGround = createSprite(200,390,400,10);
40    invisibleGround.visible = false;
41
42    console.log("Hello" + 5);
43
44    score = 0;
45  }
46
47  function draw() {
48    background(180);
49    text("Score: "+ score, 500,50);
50    score = score + Math.round(getFrameRate()/60);
51
52
53    if(keyDown("space")&& trex.y >= 362) {
54      trex.velocityY = -10;
55    }
56
```

Output:



What's next?

We will build collision with the obstacles and using game states

Extend Your Knowledge:

You can read more about the different functions of p5.play by exploring the examples in the following link:
https://molleindustria.github.io/p5.play/examples/index.html?fileName=animation.js