



What is our GOAL for this MODULE?

We created two game states and assigned different behaviors to them. We also set colliders for all the objects of the game.

What did we ACHIEVE in the class TODAY?

- Created two new game states PLAY and END.
- Assigned different game behavior for the different states.
- Grouped similar game objects in a group and assigned the same behavior to all of them.
- Created colliders for the Trex and each obstacle.

Which CONCEPTS/ CODING BLOCKS did we cover today?

- Group()
- Gamestate
- Colliders



How did we DO the activities?

- 1. Group all objects into a single group [clouds and obstacles (cactus)].
 - Program the behavior of all the objects in a single stroke using group properties.

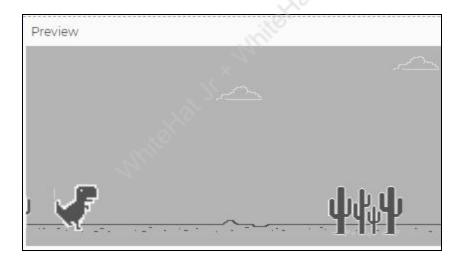
```
34
      ground = createSprite(200,380,400,20);
      ground.addImage("ground",groundImage);
ground.x = ground.width /2;
35
36
      ground.velocityX = -4;
37
38
39
      invisibleGround = createSprite(200,390,400,10);
      invisibleGround.visible = false;
40
41
42
      //create Obstacle and Cloud Groups
      obstaclesGroup = createGroup();
43
      cloudsGroup = createGroup();
44
45
      console.log("Hello" + 5);
46
47
48
      score = 0;
49
```

2. Add sprites to the groups.

```
case 2: obstacle.addImage(obstacle2);
9
                  break;
0
         case 3: obstacle.addImage(obstacle3);
1
                  break;
23
         case 4: obstacle.addImage(obstacle4);
                  break;
4
         case 5: obstacle.addImage(obstacle5);
56789012345
                  break;
         case 6: obstacle.addImage(obstacle6);
                  break;
         default: break;
       //assign scale and lifetime to the obstacle
       obstacle.scale = 0.5:
       obstacle.lifetime = 300;
      //add each obstacle to the group
6
       obstaclesGroup.add(obstacle);
```



```
109
110 ▼ function spawnClouds() {
111
       //write code here to spawn the clouds
       if (frameCount % 60 === 0) {
112♥
         var cloud = createSprite(600,300,40,10);
113
         cloud.addImage(cloudImage)
114
         cloud.y = Math.round(random(280,320))
115
         cloud.scale = 0.4;
116
         cloud.velocityX = -3;
117
118
         //assign lifetime to the variable
119
         cloud.lifetime = 134;
120
121
         //adjust the depth
122
         cloud.depth = trex.depth
123
         trex.depth = trex.depth + 1;
124
125
         //add each cloud to the group
126
         cloudsGroup.add(cloud);
127
128
129
```





3. Introduce a variable that will hold the game state's value and set it to either PLAY or END.

```
var PLAY = 1;
var END = 0;
var gameState = PLAY;

var trex, trex_running, trex_collided;
var ground, invisibleGround, groundImage;

var cloudsGroup, cloudImage;
var obstaclesGroup, obstacle1, obstacle2, obstacle3, obstacle4, obstacle5, obstacle6;

var score;
```

4. Add an if and else-if condition inside the function draw().

```
55 ▼ function draw() {
56
      background(180);
      text("Score: "+ score, 500,50);
57
      score = score + Math.round(getFrameRate()/60);
58
59
60
      if(gameState === PLAY){
61 ₹
62
63
       else if (gameState === END) {
64
65
66
67
68 ₹
      if(keyDown("space")&& trex.y >= 362) {
        trex.velocityY = -10;
69
70
71
      trex.velocityY = trex.velocityY + 0.8
72
73
74▼
      if (ground.x < 0){
        ground.x = ground.width/2;
75
76
```



5. Add behaviors inside gameState.

```
55 ▼ function draw() {
      background(180);
56
57
      text("Score: "+ score, 500,50);
58
      score = score + Math.round(getFrameRate()/60);
59
60
      if(gameState === PLAY){
61 V
62
63
       else if (gameState === END) {
64 ₹
65
66
67
      if(keyDown("space")&& trex.y >= 362) {
68 ₹
69
        trex.velocityY = -10;
70
71
72
      trex.velocityY = trex.velocityY + 0.8
73
      if (ground.x < 0){
74♥
        ground.x = ground.width/2;
75
76
```

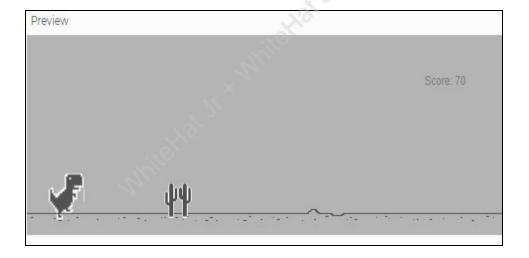
6. Move the ground in PLAY state and stop the movement in END state.

```
55 ▼ function draw() {
      background(180);
56
57
      text("Score: "+ score, 500,50);
58
      score = score + Math.round(getFrameRate()/60);
59
60
61 ₹
      if(gameState === PLAY){
        //move the ground
62
63
        ground.velocityX = -4;
64
65
66 ₹
       else if (gameState === END) {
67
          ground.velocityX = 0;
68
69
70 ▼
      if(keyDown("space")&& trex.y >= 362) {
71
        trex.velocityY = -10;
72
73
      trex.velocityY = trex.velocityY + 0.8
74
75
76 ₹
      if (ground.x < 0){
        ground.x = ground.width/2;
77
78
```



7. Display score at all times.

```
55 ▼ function draw() {
      background(180);
56
57
      //displaying score
      text("Score: "+ score, 500,50);
58
59
60
61
62 ₹
      if(gameState === PLAY){
        //move the ground
63
64
        ground.velocityX = -4;
65
        //scoring
        score = score + Math.round(getFrameRate()/60);
66
67
68 ₹
       else if (gameState === END) {
69
          ground.velocityX = 0;
70
71
      if(keyDown("space")&& trex.y >= 362)
72 ₹
        trex.velocityY = -10;
73
74
```





8. Reset ground during PLAY state.

```
if(gameState === PLAY){
62 ₹
        //move the ground
63
64
        ground.velocityX = -4;
65
        //scoring
        score = score + Math.round(getFrameRate()/60);
66
67
        if (ground.x < 0){
68 V
          ground.x = ground.width/2;
69
70
71
72
```

9. Make Trex jump only during the PLAY state.

```
if(gameState === PLAY){
62 ₹
        //move the ground
63
        ground.velocityX = -4;
64
65
        //scoring
        score = score + Math.round(getFrameRate()/60);
66
67
        if (ground.x < 0){
68 ₹
69
          ground.x = ground.width/2;
70
71
        //jump when the space key is pressed
72
        if(keyDown("space")&& trex.y >= 362) {
73 ₹
          trex.velocityY = -10;
74
75
76
77
        //add gravity
        trex.velocityY = trex.velocityY + 0.8
78
79
80
81
82 ₹
       else if (gameState === END) {
          ground.velocityX = 0;
83
84
```



10. Make the invisible ground support the Trex at all times.

```
81
      }
       else if (gameState === END) {
82 ₹
83
          ground.velocityX = 0;
84
85
86
87
      //stop trex from falling down
      trex.collide(invisibleGround);
88
89
      //spawn the clouds
90
91
      spawnClouds():
92
      //spawn obstacles on the ground
93
      spawnObstacles():
94
95
      drawSprites();
96
97
```

11. Spawn the cloud and the obstacles In PLAY state.

```
//jump when the space key is pressed
72
73 ₹
        if(keyDown("space")&& trex.y >= 362) {
74
            trex.velocityY = -12;
75
76
77
        //add gravity
        trex.velocityY = trex.velocityY + 0.8
78
79
80
        //spawn the clouds
81
        spawnClouds();
82
83
        //spawn obstacles on the ground
84
        spawnObstacles();
85
        if(obstaclesGroup.isTouching(trex)){
86 ▼
87
            gameState = END;
88
89
90 ▼
       else if (gameState === END) {
91
          ground.velocityX = 0;
92
         obstaclesGroup.setVelocityXEach(0);
93
94
         cloudsGroup.setVelocityXEach(0);
95
```



12. Write code to END the game when the Trex collides with the obstacles/cacti.

```
72
        //jump when the space key is pressed
        if(keyDown("space")&& trex.y >= 362) {
73 ♥
74
            trex.velocityY = -10:
75
76
77
        //add gravity
        trex.velocityY = trex.velocityY + 0.8
78
79
80
        if(obstaclesGroup.isTouching(trex)){
81 W
82
            gameState = END;
83
84
       else if (gameState === END) {
85 V
          ground.velocityX = 0;
86
87
```

13. Give zero velocity to all the obstacles and the clouds in the game when the Trex collides with an obstacle.

```
if(obstaclesGroup.isTouching(trex)){
81 ₹
82
            gameState = END;
83
        }
84
       else if (gameState === END) {
85 ₹
          ground.velocityX = 0;
86
87
         obstaclesGroup.setVelocityXEach(0);
88
         cloudsGroup.setVelocityXEach(0);
89
90
91
```

What's next:

We will fix the bugs present in the game.

Extend Your Knowledge:

1. <u>P5 functions and their examples</u>: Read more about the different functions of p5.play via various examples.