Game States and Groups

What is our GOAL for this MODULE?

We created 2 game states and assigned different behaviours 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 behaviour for the different states.
- Grouped similar game objects together in a group and assigned the same behaviour to all of them.
- Created colliders for the T-Rex and each obstacle.

Which CONCEPTS/ CODING BLOCKS did we cover today?

- Creating game states and assigning different behaviours.
- Colliders

How did we DO the activities?

Step 1: Group all objects into a single group (Cloud and obstacle(cactus))
Using group properties program the behaviour of all the objects in a single stroke.

```
33
34    ground = createSprite(200,380,400,20);
35    ground.addImage("ground",groundImage);
36    ground.x = ground.width /2;
37    ground.velocityX = -4;
38
39    invisibleGround = createSprite(200,390,400,10);
40    invisibleGround.visible = false;
41
42    //create Obstacle and Cloud Groups
43    obstaclesGroup = createGroup();
44    cloudsGroup = createGroup();
45
46    console.log("Hello" + 5);
47
48    score = 0;
49 }
50
51 ▼ function draw() {
52    background(180);
53    text("Score: "+ score, 500,50);
54    score = score + Math.round(getFrameRate()/60);
```

Step 2: Add sprites to the groups.

```
8          case 2: obstacle.addImage(obstacle2);
9                  break;
0          case 3: obstacle.addImage(obstacle3);
1                  break;
2          case 4: obstacle.addImage(obstacle4);
3                  break;
4          case 5: obstacle.addImage(obstacle5);
5                  break;
6          case 6: obstacle.addImage(obstacle6);
7                  break;
8          default: break;
9      }
0
1      //assign scale and lifetime to the obstacle
2      obstacle.scale = 0.5;
3      obstacle.lifetime = 300;
4
5      //add each obstacle to the group
6      obstaclesGroup.add(obstacle);
7
```

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

Step 3: Introduce a variable which will hold the value of the game state. Game State could be PLAY or END.

```
1   var PLAY = 1;
2   var END = 0;
3   var gameState = PLAY;
4   |
5   var trex, trex_running, trex_collided;
6   var ground, invisibleGround, groundImage;
7
8   var cloudsGroup, cloudImage;
9   var obstaclesGroup, obstacle1, obstacle2, obstacle3,
    obstacle4, obstacle5, obstacle6;
10
11  var score;
12
```

Step 4: Add an IF and ELSE IF condition inside the function draw()

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

Step 5: Add behaviours inside the game state.

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

Step 6: Move the ground, in PLAY state, stop the movement in END state.

```
55▼ function draw() {
56     background(180);
57     text("Score: "+ score, 500,50);
58     score = score + Math.round(getFrameRate()/60);
59
60
61▼   if(gameState === PLAY){
62       //move the ground
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
74     trex.velocityY = trex.velocityY + 0.8
75
76▼   if (ground.x < 0){
77       ground.x = ground.width/2;
78     }
```

Step 7: Display score at all times.

```
54
55▼ function draw() {
56     background(180);
57     //displaying score
58     text("Score: "+ score, 500,50);
59
60
61
62▼   if(gameState === PLAY){
63       //move the ground
64       ground.velocityX = -4;
65       //scoring
66       score = score + Math.round(getFrameRate()/60);
67     }
68▼     else if (gameState === END) {
69         ground.velocityX = 0;
70     }
71
72▼   if(keyDown("space")&& trex.y >= 362) {
73       trex.velocityY = -10;
74     }
```

Step 8: Reset ground during play state.

```
61
62▼    if(gameState === PLAY){
63        //move the ground
64        ground.velocityX = -4;
65        //scoring
66        score = score + Math.round(getFrameRate()/60);
67
68▼        if (ground.x < 0){
69            ground.x = ground.width/2;
70        }
71
72
```

Make T-Rex jump only during the play state.

```
61
62▼    if(gameState === PLAY){
63        //move the ground
64        ground.velocityX = -4;
65        //scoring
66        score = score + Math.round(getFrameRate()/60);
67
68▼        if (ground.x < 0){
69            ground.x = ground.width/2;
70        }
71
72        //jump when the space key is pressed
73▼        if(keyDown("space")&& trex.y >= 362) {
74            trex.velocityY = -10;
75        }
76
77        //add gravity
78        trex.velocityY = trex.velocityY + 0.8
79        |
80
81    }
82▼    else if (gameState === END) {
83        ground.velocityX = 0;
84    }
```

Step 9: Make the invisible ground support the T-Rex at all times.

```
81      }
82 ▼    else if (gameState === END) {
83          ground.velocityX = 0;
84      }
85
86
87      //stop trex from falling down
88      trex.collide(invisibleGround);
89
90      //spawn the clouds
91      spawnClouds();
92
93      //spawn obstacles on the ground
94      spawnObstacles();
95
96      drawSprites();
97  }
98
```

Step 10: Spawn the cloud and the obstacles In PLAY state.

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

Step 11: Write code to END the game when the T-Rex collides with the obstacles/ cactus.

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

Give '0' velocity to all the obstacles and the clouds in the game when the T-Rex collides with an obstacle.

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

What's next:
We will fix the remaining bugs in the game.

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