Scope

What we did:
- Change the scope of some variables from local to global to be used anywhere in the code.
- Write a reset function to restart the game when the reset icon is pressed.
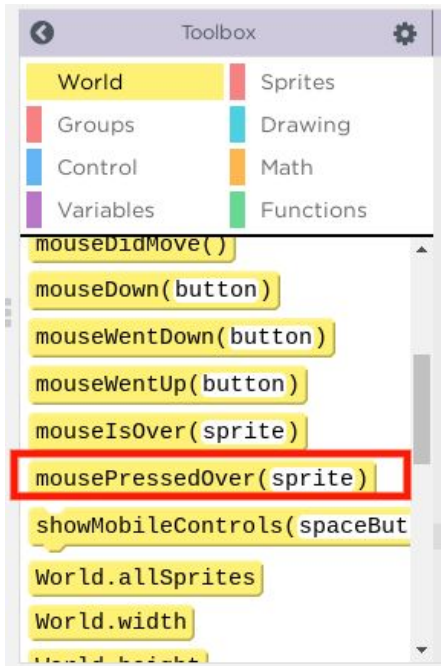
How we did it:
Step 1: T-Rex game— To reset the game, we have to use code.org's reset button.
We wrote code so that we can press on the reset icon in the game to reset the game.

```
102      restart.setAnimation("restart");
103      restart.scale = 0.5;
104    }
105
106 ▾  if(mousePressedOver(restart)) {
107      reset();
108    }
109
110    //console.log(trex.y);
111
112    //stop trex from falling down
113    trex.collide(invisibleGround);
114
115    drawSprites();
116  }
117
118  function reset(){
119
120  }
121
122 ▾ function spawnObstacles() {
123 ▾   if(World.frameCount % 60 === 0) {
```

Toolbox

World	Sprites
Groups	Drawing
Control	Math
Variables	Functions

```
mouseDidMove()
mouseDown(button)
mouseWentDown(button)
mouseWentUp(button)
mouseIsOver(sprite)
mousePressedOver(sprite)
showMobileControls(spaceBut
World.allSprites
World.width
```

You will see a warning because the scope of 'restart' sprite is not set right.

```
        Workspace.              Version History    Show Bloc
102       restart.setAnimation("restart");
103       restart.scale = 0.5;
104     }
105
106 ▾   if(mousePressedOver(restart)) {
107       reset();
108     }
109
110     //console.log(trex.y);
111
112     //stop trex from falling down
113     trex.collide(invisibleGround);
114
115     drawSprites();
116   }
117
118 ▾ function reset(){
119
120   }
121
122 ▾ function spawnObstacles() {
123 ▾   if(World.frameCount % 60 === 0) {
124
```

Step 2:

Create these variables on the top outside function draw(). The variables created outside { } will have a global scope - that is, they can be accessed anywhere in our code.

```
19  ground.setAnimation("ground2");
20  ground.x = ground.width /2;
21
22  //invisible Ground to support Trex
23  var invisibleGround = createSprite(200,385,400,5);
24  invisibleGround.visible = false;
25
26  //create Obstacle and Cloud Groups
27  var ObstaclesGroup = createGroup();
28  var CloudsGroup = createGroup();
29
30  //place gameOver and restart icon on the screen
31  var gameOver = createSprite(200,300);
32  var restart = createSprite(200,340);
33  gameOver.setAnimation("gameOver");
34  gameOver.scale = 0.5;
35  restart.setAnimation("restart");
36  restart.scale = 0.5;
37
38  //set text
39  textSize(18);
40  textFont("Georgia");
41
```

Step 3: Game Over and restart icon would always appear on the screen. We don't want that. To resolve this, we can make these invisible at the top and then make it visible when the game state changes to END

```
30  //place gameOver and restart icon on the screen
31  var gameOver = createSprite(200,300);
32  var restart = createSprite(200,340);
33  gameOver.setAnimation("gameOver");
34  gameOver.scale = 0.5;
35  restart.setAnimation("restart");
36  restart.scale = 0.5;
37
38  gameOver.visible = false;
39  restart.visible = false;
40
41  //set text
42  textSize(18);
43  textFont("Georgia");
44  textStyle(BOLD);
45
46  //score
47  var count = 0;
48
49  function draw() {
50      //set background to white
51      background("white");
52      //display score
```

Step 4: The different things that we need to reset the game are: Change the gameState back to PLAY and make the gameOver and restart invisible again.
The idea is to change the gameState to PLAY and make the gameOver icons invisible

```
115    }
116
117    //console.log(trex.y);
118
119    //stop trex from falling down
120    trex.collide(invisibleGround);
121
122    drawSprites();
123  }
124
125  function reset(){
126    gameState = PLAY;
127
128    gameOver.visible = false;
129    restart.visible = false;
130
131  }
132
133  function spawnObstacles() {
134    if(World.frameCount % 60 === 0) {
135      var obstacle = createSprite(400,365,10,40);
136      obstacle.velocityX = - (6 + 3*count/100);
137
```

Step 5: Destroy all the obstacles and clouds when we reset the game.

```
112
113    if(mousePressedOver(restart)) {
114      reset();
115    }
116
117    //console.log(trex.y);
118
119    //stop trex from falling down
120    trex.collide(invisibleGround);
121
122    drawSprites();
123  }
124
125  function reset(){
126    gameState = PLAY;
127
128    gameOver.visible = false;
129    restart.visible = false;
130
131    ObstaclesGroup.destroyEach();
132    CloudsGroup.destroyEach();
133
134  }
135
136  function spawnObstacles() {
137
```

Step 6: Change the T-Rex animation.

```
121
122    drawSprites();
123  }
124
125 ▾ function reset(){
126    gameState = PLAY;
127
128    gameOver.visible = false;
129    restart.visible = false;
130
131    ObstaclesGroup.destroyEach();
132    CloudsGroup.destroyEach();
133
134    trex.setAnimation("trex");
135
136  }
137
138 ▾ function spawnObstacles() {
139 ▾    if(World.frameCount % 60 === 0) {
140       var obstacle = createSprite(400,365,10,40);
141       obstacle.velocityX = - (6 + 3*count/100);
142
143       //generate random obstacles
144       var rand = randomNumber(1,6);
145       obstacle.setAnimation("obstacle" + rand);
146
147   ◂
```

Step 7: Set count = 0 in the reset to reset the score

```
120    trex.collide(invisibleGround);
121
122    drawSprites();
123  }
124
125 ▾ function reset(){
126    gameState = PLAY;
127
128    gameOver.visible = false;
129    restart.visible = false;
130
131    ObstaclesGroup.destroyEach();
132    CloudsGroup.destroyEach();
133
134    trex.setAnimation("trex");
135
136    count = 0;
137
138  }
139
140 ▾ function spawnObstacles() {
141 ▾    if(World.frameCount % 60 === 0) {
142       var obstacle = createSprite(400,365,10,40);
143       obstacle.velocityX = - (6 + 3*count/100);
144
```

Step 8: Update count using frameRate instead of frameCount: count = count +
Math.round(frameRate/60)

```
40
41  //set text
42  textSize(18);
43  textFont("Georgia");
44  textStyle(BOLD);
45
46  //score
47  var count = 0;
48
49- function draw() {
50     //set background to white
51     background("white");
52     //display score
53     text("Score: "+ count, 250, 100);
54     console.log(gameState);
55
56-    if(gameState === PLAY){
57       //move the ground
58       ground.velocityX = -(6 + 3*count/100);
59       //scoring
60       count = count + Math.round(World.frameRate/60);
61
62-       if (count>0 && count%100 === 0){
63          playSound("checkPoint.mp3");
64       }
65
```

What's next? : Write more code outside code.org.