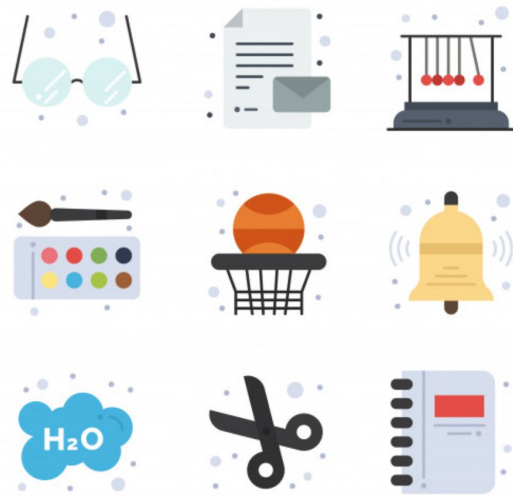


Classes and Objects



What we did:

- Design different classes corresponding to the different game objects.
- Create objects using the above class blueprints
- Make the bird object move with the mouse

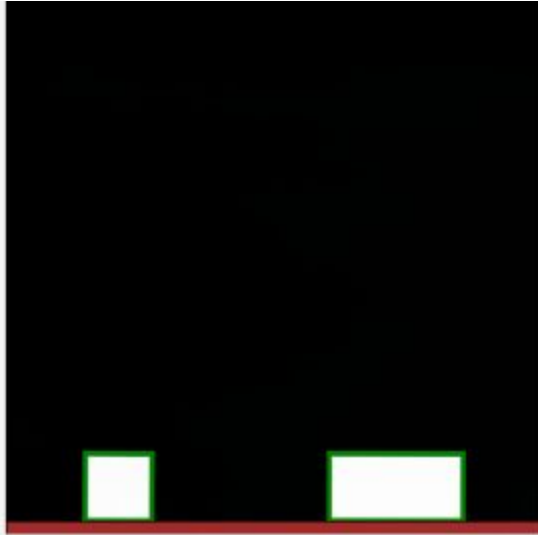
How we did it:



Step 1: Change the colour of the Ground class to brown and add a stroke of green to our box class, so that each object's boundary looks visibly separate

```
JS Ground.js > Ground > display
1  class Ground {
2    constructor(x,y,width,height) {
3      var options = {
4        | isStatic: true
5      }
6      this.body = Bodies.rectangle(x,y,width,height,options);
7      this.width = width;
8      this.height = height;
9      World.add(world, this.body);
10   }
11   display(){
12     var pos =this.body.position;
13     rectMode(CENTER);
14     fill("brown");
15     rect(pos.x, pos.y, this.width, this.height);
16   }
17 };
```

```
JS Box.js > Box > display
1  class Box {
2    constructor(x, y, width, height) {
3      var options = {
4        'restitution':0.8,
5        'friction':0.3,
6        'density':1.0
7      }
8      this.body = Bodies.rectangle(x, y, width, height, options);
9      this.width = width;
10     this.height = height;
11
12     World.add(world, this.body);
13   }
14   display(){
15     var pos =this.body.position;
16     var angle = this.body.angle;
17     push();
18     translate(pos.x, pos.y);
19     rotate(angle);
20     rectMode(CENTER);
21     strokeWeight(4);
22     stroke("green");
23     fill(255);
24     rect(0, 0, this.width, this.height);
25     pop();
26   }
27 };
28
```



Step 2: Create the Pig class. It will be similar to the Box class for now

```
1 class Pig {
2   constructor(x, y) {
3     var options = {
4       'restitution':0.8,
5       'friction':0.3,
6       'density':1.0
7     }
8     this.body = Bodies.rectangle(x, y, 50, 50, options);
9     this.width = 50;
10    this.height = 50;
11
12    World.add(world, this.body);
13  }
14  display(){
15    var pos =this.body.position;
16    var angle = this.body.angle;
17    push();
18    translate(pos.x, pos.y);
19    rotate(angle);
20    rectMode(CENTER);
21
22    fill("red");
23    rect(0, 0, this.width, this.height);
24    pop();
25  }
26 };
27
```

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4   <script src="p5.min.js"></script>
5   <script src="p5.dom.min.js"></script>
6   <script src="p5.sound.min.js"></script>
7   <script src="matter.js"></script>
8   <script src="Ground.js"></script>
9   <script src="Box.js"></script>
10  <script src="Pig.js"></script>
11  <link rel="stylesheet" type="text/css" href="style.css">
12  <meta charset="utf-8">
13 </head>
14 <body>
15   <script src="sketch.js"></script>
16 </body>
17 </html>
18

```

```

1 const Engine = Matter.Engine;
2 const World = Matter.World;
3 const Bodies = Matter.Bodies;
4
5 var engine, world;
6 var box1, pig1;
7
8 function setup(){
9   var canvas = createCanvas(400,400);
10   engine = Engine.create();
11   world = engine.world;
12
13   box1 = new Box(200,300,50,50);
14   box2 = new Box(240,100,50,100);
15   ground = new Ground(200,height,400,20)
16
17   pig1 = new Pig(200,200);
18 }
19
20 function draw(){
21   background(0);
22   Engine.update(engine);
23   console.log(box2.body.position.x);
24   console.log(box2.body.position.y);
25   console.log(box2.body.angle);
26   box1.display();
27   box2.display();
28   ground.display();
29   pig1.display();
30 }

```



Step 3: Write code for the log class, to pass an angle parameter while creating the log object. The logs in the object also have a fixed width.

```

1 class Log {
2   constructor(x, y, height, angle) {
3     var options = {
4       'restitution':0.8,
5       'friction':0.3,
6       'density':1.0
7     }
8     this.body = Bodies.rectangle(x, y, 20, height, options);
9     this.width = 20;
10    this.height = height;
11    Matter.Body.setAngle(this.body, angle);
12    world.add(world, this.body);
13  }
14  display(){
15    var pos =this.body.position;
16    var angle = this.body.angle;
17    push();
18    translate(pos.x, pos.y);
19    rotate(angle);
20    rectMode(CENTER);
21    strokeWeight(4);
22    stroke("green");
23    fill(255);
24    rect(0, 0, this.width, this.height);
25    pop();
26  }
27 };
28

```

```

35 sketch.js ▶ draw
1  const Engine = Matter.Engine;
2  const World= Matter.World;
3  const Bodies = Matter.Bodies;
4
5  var engine, world;
6  var box1, pig1;
7
8  function setup(){
9    var canvas = createCanvas(400,400);
10    engine = Engine.create();
11    world = engine.world;
12
13    box1 = new Box(200,300,50,50);
14    box2 = new Box(240,100,50,100);
15    ground = new Ground(200,height,400,20)
16
17    pig1 = new Pig(200,200);
18    log1 = new Log(100,300,100, PI/2);
19  }
20
21  function draw(){
22    background(0);
23    Engine.update(engine);
24    console.log(box2.body.position.x);
25    console.log(box2.body.position.y);
26    console.log(box2.body.angle);
27    box1.display();
28    box2.display();
29    ground.display();
30    pig1.display();
31    log1.display();
32  }

```

```
index.html > html > head > script
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <script src="p5.min.js"></script>
5      <script src="p5.dom.min.js"></script>
6      <script src="p5.sound.min.js"></script>
7      <script src="matter.js"></script>
8      <script src="Ground.js"></script>
9      <script src="Box.js"></script>
10     <script src="Pig.js"></script>
11     <script src="Log.js"></script>
12     <link rel="stylesheet" type="text/css" href="style.css">
13     <meta charset="utf-8">
14 </head>
15 <body>
16     <script src="sketch.js"></script>
17 </body>
18 </html>
19
```

Note: Most computer programs use the unit "radians" for degrees.

In the unit of radians:

$\pi = 180$ degrees;

$\pi/2 = 90$ degrees;

$\pi/4 = 45$ degrees and so on

***Note π stands for the mathematical pi symbol
(π in radians = 180 degrees)**

Step 4: Increase the size of our canvas to 1200,400. The ground size will also increase.

```
1  const Engine = Matter.Engine;
2  const World= Matter.World;
3  const Bodies = Matter.Bodies;
4
5  var engine, world;
6  var box1, pig1;
7
8  function setup(){
9    var canvas = createCanvas(1200,400);
10   engine = Engine.create();
11   world = engine.world;
12
13   box1 = new Box(200,300,50,50);
14   box2 = new Box(240,100,50,100);
15   ground = new Ground(600,height,1200,20)
16
17   pig1 = new Pig(200,200);
18   log1 = new Log(100,300,100, PI/2);
19 }
20
21 function draw(){
22   background(0);
23   Engine.update(engine);
24   console.log(box2.body.position.x);
25   console.log(box2.body.position.y);
26   console.log(box2.body.angle);
27   box1.display();
28   box2.display();
29   ground.display();
30   pig1.display();
31   log1.display();
32 }
```

Step 5: First position the two boxes and the pigs in between



Step 6: Create a log object just above these boxes. Remember we need our log at 90 degrees. In radians, 90 degrees = $\pi/2$

Guide the student to create the two boxes, the pig and a log above the first layer.

```

13
14     ground = new Ground(600,height,1200,20)
15
16     box1 = new Box(700,320,70,70);
17     box2 = new Box(920,320,70,70);
18     pig1 = new Pig(810, 350);
19     log1 = new Log(810,260,300, PI/2);
20
21     box3 = new Box(700,240,70,70);
22     box4 = new Box(920,240,70,70);
23     pig3 = new Pig(810, 220);
24
25     log3 = new Log(810,180,300, PI/2);
26
27 }
28
29 function draw(){
30     background(0);
31     Engine.update(engine);
32     console.log(box2.body.position.x);
33     console.log(box2.body.position.y);
34     console.log(box2.body.angle);
35     box1.display();
36     box2.display();
37     ground.display();
38     pig1.display();
39     log1.display();
40
41     box3.display();
42     box4.display();
43     pig3.display();
44     log3.display();
45 }
46
  
```



Step 7: Create the box on the top and two sloping logs to experiment with the angles for the logs.

```

18  pig1 = new Pig(810, 220);
19  log1 = new Log(810, 260, 300, PI/2);
20
21  box3 = new Box(700, 240, 70, 70);
22  box4 = new Box(920, 240, 70, 70);
23  pig3 = new Pig(810, 220);
24
25  log3 = new Log(810, 180, 300, PI/2);
26
27  box5 = new Box(810, 160, 70, 70);
28  log4 = new Log(760, 120, 150, PI/7);
29  log5 = new Log(870, 120, 150, -PI/7);
30
31  }
32
33  function draw(){
34    background(0);
35    Engine.update(engine);
36    console.log(box2.body.position.x);
37    console.log(box2.body.position.y);
38    console.log(box2.body.angle);
39    box1.display();
40    box2.display();
41    ground.display();
42    pig1.display();
43    log1.display();
44
45    box3.display();
46    box4.display();
47    pig3.display();
48    log3.display();
49
50    box5.display();
51    log4.display();
52    log5.display();

```

The logs keep sliding off to the ground. Increase friction to stop the sliding from happening. In the Log and Box class, add "friction = 1.0" in our options to prevent the sliding.

```
1 class Log {
2   constructor(x, y, height, angle) {
3     var options = {
4       'restitution':0.8,
5       'friction':1.0,
6       'density':1.0
7     }
8     this.body = Bodies.rectangle(x, y, 20, height, options);
9     this.width = 20;
10    this.height = height;
11    Matter.Body.setAngle(this.body, angle);
12    World.add(world, this.body);
13  }
14  display(){
15    var pos =this.body.position;
16    var angle = this.body.angle;
17    push();
18    translate(pos.x, pos.y);
19    rotate(angle);
20    rectMode(CENTER);
21    strokeWeight(4);
22    stroke("green");
23    fill(255);
24    rect(0, 0, this.width, this.height);
25    pop();
26  }
27 };
28
```

```
1 class Box {
2   constructor(x, y, width, height) {
3     var options = {
4       'restitution':0.8,
5       'friction':1.0,
6       'density':1.0
7     }
8     this.body = Bodies.rectangle(x, y, width, height, options);
9     this.width = width;
10    this.height = height;
11
12    World.add(world, this.body);
13  }
14  display(){
15    var pos =this.body.position;
16    var angle = this.body.angle;
17    push();
18    translate(pos.x, pos.y);
19    rotate(angle);
20    rectMode(CENTER);
21    strokeWeight(4);
22    stroke("green");
23    fill(255);
24    rect(0, 0, this.width, this.height);
25    pop();
26  }
27 };
28
```

Step 8: Create a Bird Class

Bird Class will be similar to Box Class but we will give it a fixed size (width and height)

Make the Bird Class move with the Mouse. Bird's position will be the same as mouseX and mouseY. Make the Bird red to make it look angry.

```
1  class Bird {
2      constructor(x, y) {
3          var options = {
4              'density':1.5,
5              'friction': 1.0,
6              'restitution':0.5
7          };
8          this.body = Bodies.rectangle(x, y, 50, 50, options);
9          this.width = 50;
10         this.height = 50;
11         World.add(world, this.body);
12     };
13     display(){
14         var pos = this.body.position;
15         pos.x = mouseX;
16         pos.y = mouseY;
17         var angle = this.body.angle;
18
19         push();
20         translate(pos.x, pos.y);
21         rotate(angle);
22         strokeWeight(3);
23         stroke('blue')
24         fill('green')
25         rectMode(CENTER)
26         rect(0, 0, this.width, this.height);
27         pop();
28     };
29 };
30
```

```

js sketch.js ▶ draw
27   box5 = new Box(810,160,70,70);
28   log4 = new Log(760,120,150, PI/7);
29   log5 = new Log(870,120,150, -PI/7);
30
31   bird = new Bird(100,100);
32
33 }
34
35 function draw(){
36   background(0);
37   Engine.update(engine);
38   console.log(box2.body.position.x);
39   console.log(box2.body.position.y);
40   console.log(box2.body.angle);
41   box1.display();
42   box2.display();
43   ground.display();
44   pig1.display();
45   log1.display();
46
47   box3.display();
48   box4.display();
49   pig3.display();
50   log3.display();
51
52   box5.display();
53   log4.display();
54   log5.display();
55
56   bird.display();
57 }

```

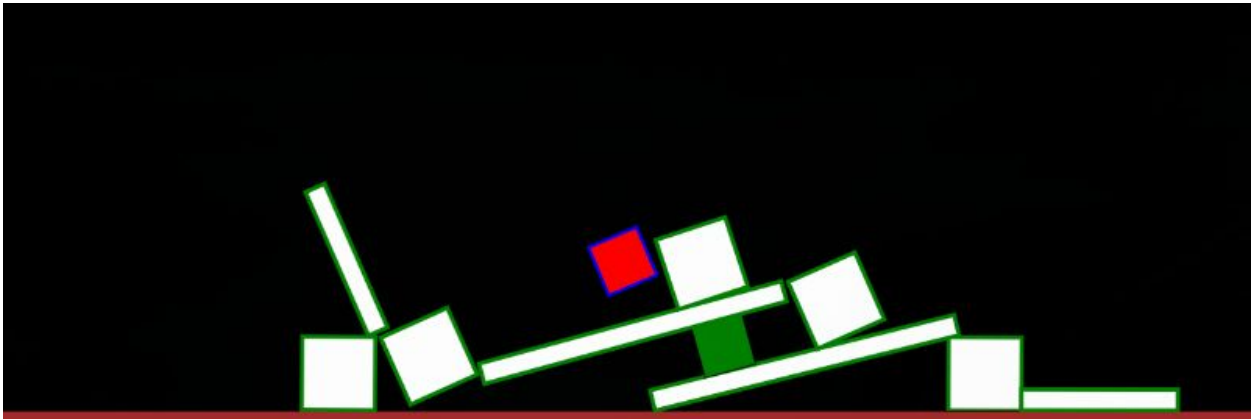
```

index.html ▶ html
1  <!DOCTYPE html>
2  <html>
3  <head>
4    <script src="p5.min.js"></script>
5    <script src="p5.dom.min.js"></script>
6    <script src="p5.sound.min.js"></script>
7    <script src="matter.js"></script>
8    <script src="Ground.js"></script>
9    <script src="Box.js"></script>
10   <script src="Pig.js"></script>
11   <script src="Log.js"></script>
12   <script src="Bird.js"></script>
13   <link rel="stylesheet" type="text/css" href="style.css">
14   <meta charset="utf-8">
15 </head>
16 <body>
17   <script src="sketch.js"></script>
18 </body>
19 </html>
20

```



Step 9: Beat the pigs by moving your mouse.



What's next?: Add images to our angry Birds game!