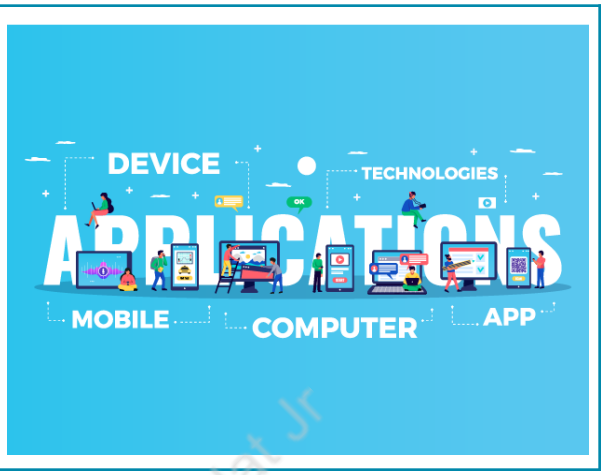
FORCE APPLICATION ON A BODY



What is our GOAL for this MODULE?

In this class, we learned to apply force to a body. We also used OOP concepts to create a class for the ground body and to create multiple objects out of this Ground class.

What did we ACHIEVE in the class TODAY?

- Installed the physics engine library called **matter.min.js**.
- Created a new file named **ground.js**.
- Created a **Ground** class.
- Created a function to display the ground on the canvas.
- Created the body for the ball.
- Created a button for the user to click and apply force on the ball in different directions.
- Created four walls around canvas using the Ground class.

Which CONCEPTS/ CODING BLOCKS did we cover today?

- OOP concepts
- Matter.Body.applyForce()
- mouseClicked()
- createImg()

© 2019 The content of this email is confidential and intended for the recipient specified in message only. It is strictly forbidden to share any part of this message with any third party without a written consent of the sender. If you received this message by mistake, please reply to this message and follow with its deletion, so that we can ensure such a mistake does not occur in the future.

How did we DO the activities?

To create the four walls around canvas:

1. Create a new file named **ground.js** and add this to **index.html**.

```
<!DOCTYPE html><html><head>
  <script src="p5.min.js"></script>
  <script src="p5.dom.min.js"></script>
  <script src="p5.sound.min.js"></script>
  <link rel="stylesheet" type="text/css" href="style.css">
  <meta charset="utf-8">

</head>

<body>
  <script src="matter.min.js"></script>
  <script src="ground.js"></script>
  <script src="sketch.js"></script>

</body></html>
```

2. Create a **Ground class** to create four walls around Canvas :
    - Create a **constructor()** to declare the properties and create a rectangle body.
    - Add it into the **World**.

© 2019 The content of this email is confidential and intended for the recipient specified in message only. It is strictly forbidden to share any part of this message with any third party without a written consent of the sender. If you received this message by mistake, please reply to this message and follow with its deletion, so that we can ensure such a mistake does not occur in the future.

```
class Ground
{
  constructor(x, y, w, h)
  {
    let options = {
      isStatic:true
    };

    this.body = Bodies.rectangle(x, y, w, h, options);
    this.w = w;
    this.h = h;
    World.add(world, this.body);
  }

}
```

3. Create a function **show()** to display the ground on the canvas.
    - Functions inside the class are called methods, therefore there is no need to add the function keyword.

```
show() {
   var pos = this.body.position;
   push();
   rectMode(CENTER);
   stroke(255);
   fill(127);
   rect(pos.x, pos.y, this.w, this.h);
   pop();
 }
```

4. Create the objects from the **Ground** class inside function **setup()** in **sketch.js**.

© 2019 The content of this email is confidential and intended for the recipient specified in message only. It is strictly forbidden to share any part of this message with any third party without a written consent of the sender. If you received this message by mistake, please reply to this message and follow with its deletion, so that we can ensure such a mistake does not occur in the future.

```
var ground;
var left;
var right;
var top_wall;

function setup() {
  createCanvas(400,400);
  engine = Engine.create();
  world = engine.world;

  ground =new Ground(200,390,400,20);
  right = new Ground(390,200,20,400);
  left = new Ground(10,200,20,400);
  top_wall = new Ground(200,10,400,20);

  rectMode(CENTER);
  ellipseMode(RADIUS);
}
```

5. Next, call the **show()** function inside function **draw()** to display objects.

```
function draw()
{
  background(51);
  ground.show();
  top_wall.show();
  left.show();
  right.show();
  Engine.update(engine);
}
```

Output:

© 2019 The content of this email is confidential and intended for the recipient specified in message only. It is strictly forbidden to share any part of this message with any third party without a written consent of the sender. If you received this message by mistake, please reply to this message and follow with its deletion, so that we can ensure such a mistake does not occur in the future.

Now to create a ball within four walls:

6. Declare a **var ball**, create a ball using **bodies.circle(),** and set its **restitution** to **0.5** to bounce the ball.

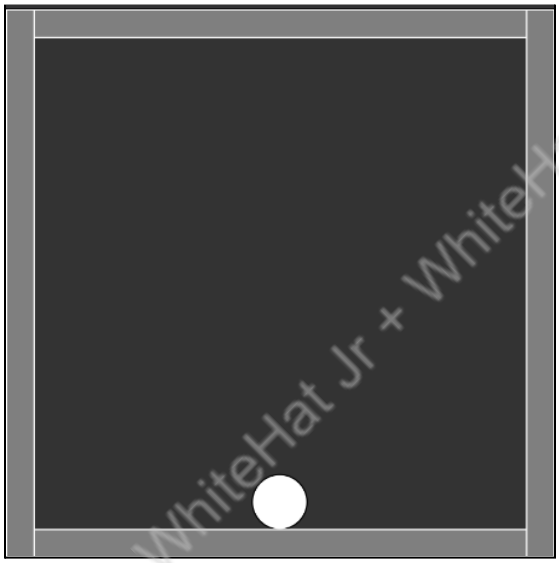
```
var ball;

function setup() {
  createCanvas(400,400);
  engine = Engine.create();
  world = engine.world;

  ground =new Ground(200,390,400,20);
  right = new Ground(390,200,20,400);
  left = new Ground(10,200,20,400);
  top_wall = new Ground(200,10,400,20);

  var ball_options = {
    restitution: 0.95
  }

  ball = Bodies.circle(200,100,20,ball_options);
  World.add(world,ball);

  rectMode(CENTER);
  ellipseMode(RADIUS);
}
```

© 2019 The content of this email is confidential and intended for the recipient specified in message only. It is strictly forbidden to share any part of this message with any third party without a written consent of the sender. If you received this message by mistake, please reply to this message and follow with its deletion, so that we can ensure such a mistake does not occur in the future.

Use **ellipse ()** to display the object in function **draw()**.

```
function draw()
{
  background(51);

  ellipse(ball.position.x,ball.position.y,20);

  ground.show();
  top_wall.show();
  left.show();
  right.show();
  Engine.update(engine);
}
```

Output:



Now, to apply force on the ball using buttons:

7. Create two functions to apply force in horizontal & vertical force.
    - Use **Matter.Body.applyForce(ball,{x:0,y:0},{x:0,y:0})**;.
      Here the first argument is the body, the second argument is the initial amount of force and the third argument is the amount and direction of the force.
    - For Horizontal Force:

© 2019 The content of this email is confidential and intended for the recipient specified in message only. It is strictly forbidden to share any part of this message with any third party without a written consent of the sender. If you received this message by mistake, please reply to this message and follow with its deletion, so that we can ensure such a mistake does not occur in the future.

```
function hForce()
{
  Matter.Body.applyForce(ball,{x:0,y:0},{x:0.05,y:0});
}
```

- For Vertical Force:

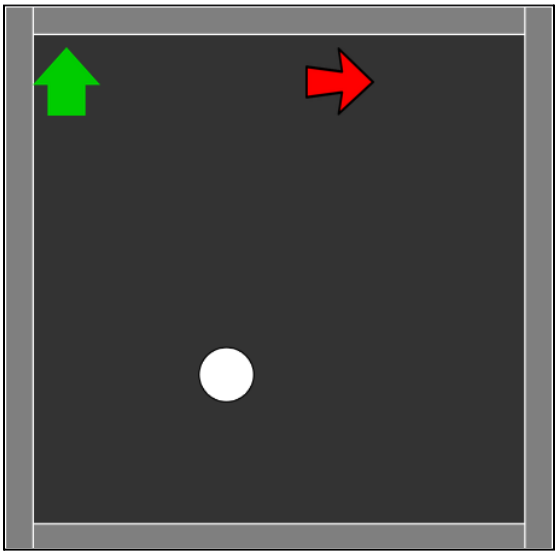
```
function vForce()
{
  Matter.Body.applyForce(ball,{x:0,y:0},{x:0,y:-0.05});
}
```

8. Next to create buttons on the canvas and attach **hForce()** and **vForce()** functions to the buttons.
    - Create the image buttons using **createImg()**, these button images will be displayed on the canvas, and they will work like buttons.
    - Declare size and position using **size()** and **position()**.
    - Call **hForce** and **vForce** using **mouseClicked()** for respective buttons.

```
var btn1;
var btn2;

function setup() {
  createCanvas(400,400);
  engine = Engine.create();
  world = engine.world;

  btn1 = createImg('right.png');
  btn1.position(220,30);
  btn1.size(50,50);
  btn1.mouseClicked(hForce);

  btn2 = createImg('up.png');
  btn2.position(20,30);
  btn2.size(50,50);
  btn2.mouseClicked(vForce);
```

Output:

© 2019 The content of this email is confidential and intended for the recipient specified in message only. It is strictly forbidden to share any part of this message with any third party without a written consent of the sender. If you received this message by mistake, please reply to this message and follow with its deletion, so that we can ensure such a mistake does not occur in the future.



What's next?

In the next class, we will be starting on our quest to create the Pirate Invasion Game, and we will learn about how to use the concept of blueprints in the game.

EXTEND YOUR KNOWLEDGE

Bookmark the following link: to know more about Apply Force in matter.js
https://brm.io/matter-js/docs/classes/Body.html#method_applyForce

WhiteHat Jr + WhiteHat Jr + WhiteHat Jr

© 2019 The content of this email is confidential and intended for the recipient specified in message only. It is strictly forbidden to share any part of this message with any third party without a written consent of the sender. If you received this message by mistake, please reply to this message and follow with its deletion, so that we can ensure such a mistake does not occur in the future.