PHYSICS ENGINE

What is our GOAL for this MODULE?

We used our knowledge about physics engines and matter.js to make the ball bounce.

What did we ACHIEVE in the class TODAY?

- Used a physics engine to create a world and the objects in them.
- Integrated physics engine with the p5 code to create interactive objects following the rules of physics in this world.
- Changed the behavior of the objects in this world by tuning the physics engine.

Which CONCEPTS/ CODING BLOCKS did we cover today?

- Physics engine.
- Matter.js
- Restitution property.

© 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.

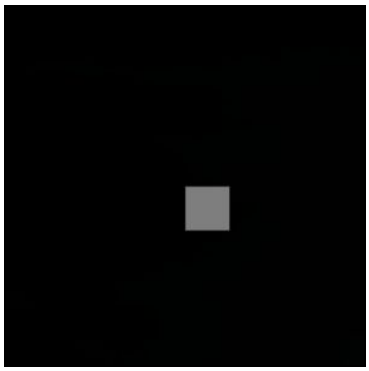
How did we DO the activities?

1. Get the boilerplate from the GitHub.
2. Modify the index.html file by adding a script tag with src as the link for the matter.js library.
   <script src="https://unpkg.com/matter-js@0.14.2/build/matter.min.js"></script>

```
index.html ▸ html ▸ head
<!DOCTYPE html><html><head>
    <script src="p5.js"></script>
    <script src="p5.dom.min.js"></script>
    <script src="p5.sound.min.js"></script>
    <script src="https://unpkg.com/matter-js@0.14.2/build/matter.min.js"></script>
    <script src="./p5.play.js"></script>
    <link rel="stylesheet" type="text/css" href="style.css">
    <meta charset="utf-8">

  </head>
  <body>
      <script src="script.js"></script>


</body></html>
```

3. Open the script.js file, go through code available for creating a canvas, and draw a rectangle at the center.

```
js script.js ▸ draw
function setup(){
    var canvas = createCanvas(400,400);
}

function draw(){
    background(0);
    rect(200,200,50,50);
}
```



4. Create a ground using the physics engine. Write code to namespace Matter.World, Matter.Engine and Matter.Bodies.

© 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.

```
JS script.js ▸ draw
const Engine = Matter.Engine;
const World= Matter.World;
const Bodies = Matter.Bodies;


function setup(){
    var canvas = createCanvas(400,400);
}

function draw(){
    background(0);
    rectMode(CENTER);
    rect(200,200,50,50);
}
```

5. Create a physics engine.

```
JS script.js ▸ setup
const Engine = Matter.Engine;
const World= Matter.World;
const Bodies = Matter.Bodies;

var engine, world;

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

function draw(){
    background(0);
    rectMode(CENTER);
    rect(200,200,50,50);
}
```

6. Make an object in this world. Use Bodies to create a body in this world—create a rectangular body just above the previous rectangle.

© 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.

```
JS script.js ▸ setup
const Engine = Matter.Engine;
const World= Matter.World;
const Bodies = Matter.Bodies;

var engine, world;
var object;

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

    object = Bodies.rectangle(200,100,50,50);

}

function draw(){
    background(0);
    rectMode(CENTER);
    rect(200,200,50,50);
}
```

7. Write the code to add the body to the world.

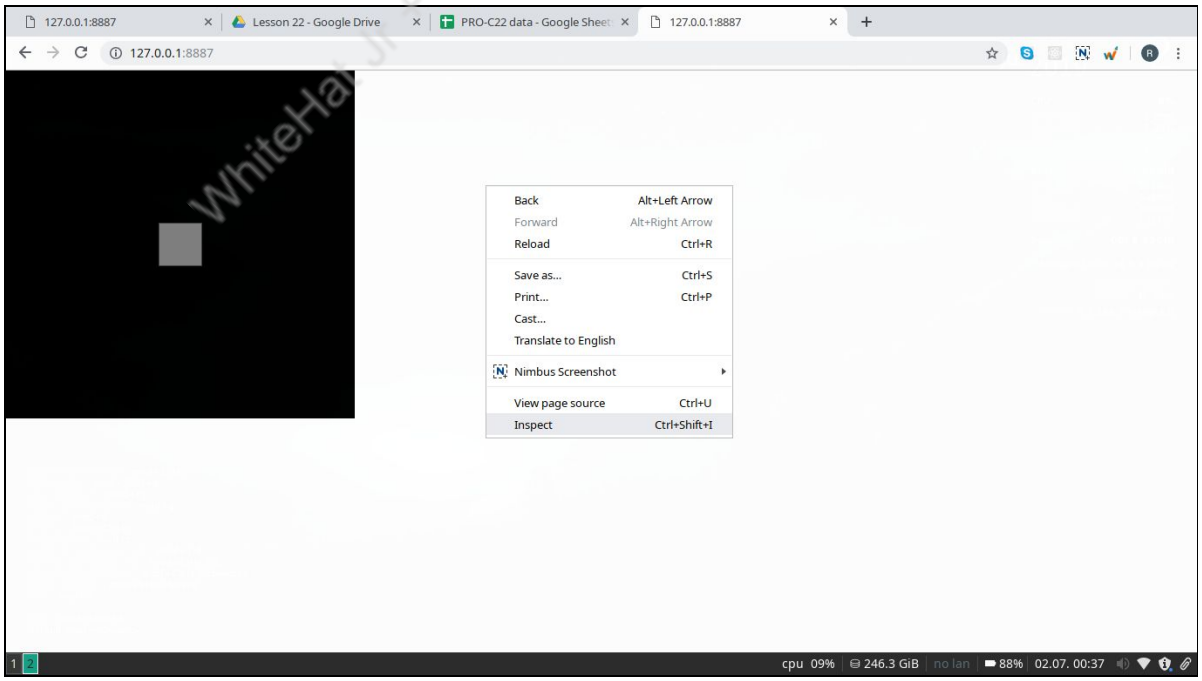
```
JS script.js ▸ setup
const Engine = Matter.Engine;
const World= Matter.World;
const Bodies = Matter.Bodies;

var engine, world;
var object;

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

    object = Bodies.rectangle(200,100,50,50);
    World.add(world,object);
}

function draw(){
    background(0);
    rectMode(CENTER);
    rect(200,200,50,50);
}
```

© 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.

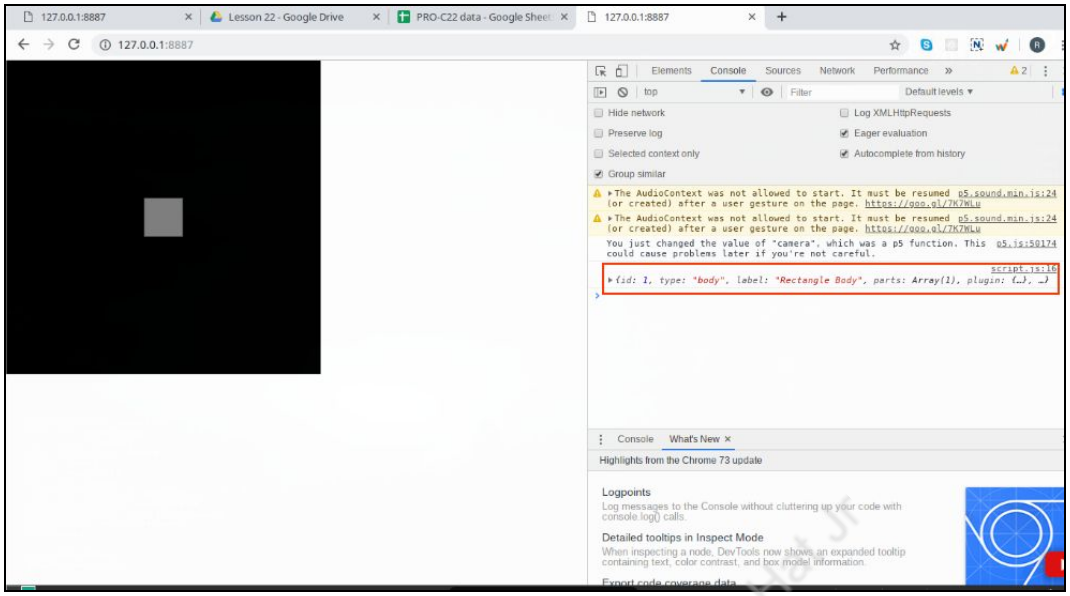
8. Let's give the object inside the console to see another body other than the rectangle we had drawn.

```
const Engine = Matter.Engine;
const World= Matter.World;
const Bodies = Matter.Bodies;

var engine, world;
var object;

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

    object = Bodies.rectangle(200,100,50,50);
    World.add(world,object);

    console.log(object);
}

function draw(){
    background(0);
    rectMode(CENTER);
    rect(200,200,50,50);
}
```

9. Right-click inside the browser and see the console output by pressing on inspect.



© 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.

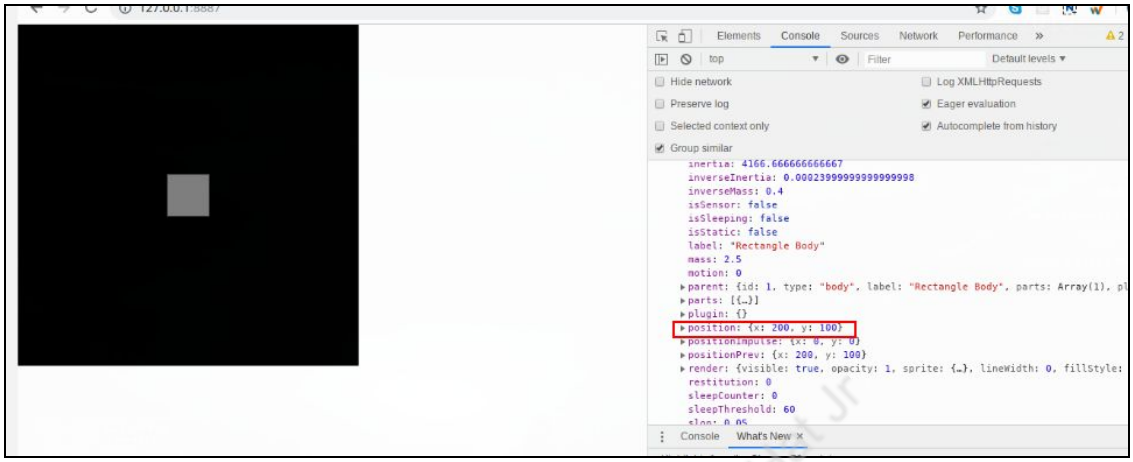


10. Print this object type on the console. You just need to write **console.log(object.type)**.

```
const Engine = Matter.Engine;
const World= Matter.World;
const Bodies = Matter.Bodies;

var engine, world;
var object;

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

    object = Bodies.rectangle(200,100,50,50);
    World.add(world,object);

    console.log(object);
    console.log(object.type);
}

function draw(){
    background(0);
    rectMode(CENTER);
    rect(200,200,50,50);
}
```

© 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.

11. Click on the arrow to the left of the object we have created; you will see it has many attributes. It also has an attribute called position.



12. Print the x and y coordinates of the rectangle object: **console.log(object.position.x)** and **console.log(object.position.y)**.

© 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.

```
const Engine = Matter.Engine;
const World= Matter.World;
const Bodies = Matter.Bodies;

var engine, world;
var object;

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

    object = Bodies.rectangle(200,100,50,50);
    World.add(world,object);

    console.log(object);
    console.log(object.position.x);
    console.log(object.position.y);
}

function draw(){
    background(0);
    rectMode(CENTER);
    rect(200,200,50,50);
}
```

13. In the draw function - instead of drawing a rectangle at any position, draw it at our object's position.

```
const Engine = Matter.Engine;
const World= Matter.World;
const Bodies = Matter.Bodies;

var engine, world;
var object;

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

    object = Bodies.rectangle(200,100,50,50);
    World.add(world,object);

    console.log(object);
}

function draw(){
    background(0);
    Engine.update(engine);
    rectMode(CENTER);
    rect(object.position.x,object.position.y,50,50);
}
```

© 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.

14. For a static rectangle, write this code:

```
const Engine = Matter.Engine;
const World= Matter.World;
const Bodies = Matter.Bodies;

var engine, world;
var object;

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

    var object_options ={
        isStatic: true
    }

    object = Bodies.rectangle(200,100,50,50,object_options);
    World.add(world,object);

    console.log(object);
}

function draw(){
    background(0);
    Engine.update(engine);
    rectMode(CENTER);
    rect(object.position.x,object.position.y,50,50);
}
```

15. Create a ball that bounces on the ground like a tennis ball and then comes to rest.

```
const Engine = Matter.Engine;
const World= Matter.World;
const Bodies = Matter.Bodies;

var engine, world;
var ground;

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

    var ground_options ={
        isStatic: true
    }

    ground = Bodies.rectangle(200,390,200,20,ground_options);
    World.add(world,ground);

    console.log(ground);
}

function draw(){
    background(0);
    Engine.update(engine);
    rectMode(CENTER);
    rect(ground.position.x,ground.position.y,400,20);
}
```

© 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.

16. Create a ball (Rectangle) similar to the “object” created.

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

    var ground_options ={
        isStatic: true
    }

    ground = Bodies.rectangle(200,390,200,20,ground_options);
    World.add(world,ground);

    var ball_options ={
        restitution: 1.0
    }

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

    console.log(ground);
}

function draw(){
    background(0);
    Engine.update(engine);
    rectMode(CENTER);
    rect(ground.position.x,ground.position.y,400,20);


    rect(ball.position.x, ball.position.y, 20, 20);
}
```

17. Add restitution and pass it to make it bounce like a tennis ball.

```
    ground = Bodies.rectangle(200,390,200,20,ground_options);
    World.add(world,ground);

    var ball_options ={
        restitution: 1.0
    }

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

    console.log(ground);
}

function draw(){
    background(0);
    Engine.update(engine);
    rectMode(CENTER);
    rect(ground.position.x,ground.position.y,400,20);


    rect(ball.position.x, ball.position.y, 20, 20);
}
```

© 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.

Output:

What's next?

We will create our own Angry Birds game.

Extend your knowledge:

1. Learn about different kinds of forces that can be used in a game using physics engine: https://p5js.org/examples/simulate-forces.html

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.