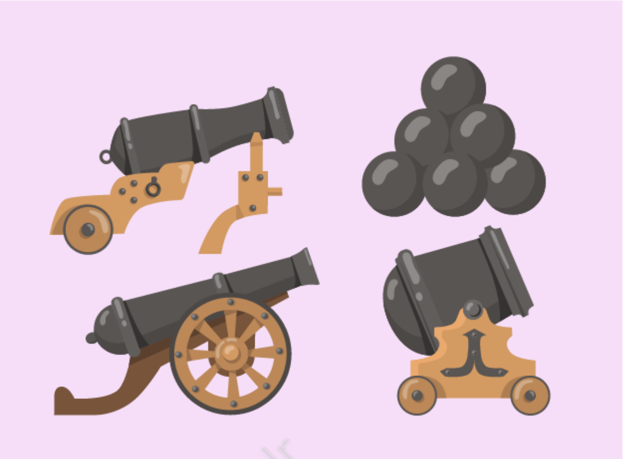
CS-PRO-C24(V3)

MULTIPLE CANNONBALLS



WhiteHat Jr + WhiteHat Jr + WhiteHat Jr

What is our GOAL for this MODULE?

In this class, we created more cannonball objects and shot them from the cannon on the press of a button. We also traced and showed the trajectory taken by the cannonball when it was shot.

What did we ACHIEVE in the class TODAY?

- Created multiple cannonballs.
- Created **showCannonBalls().**

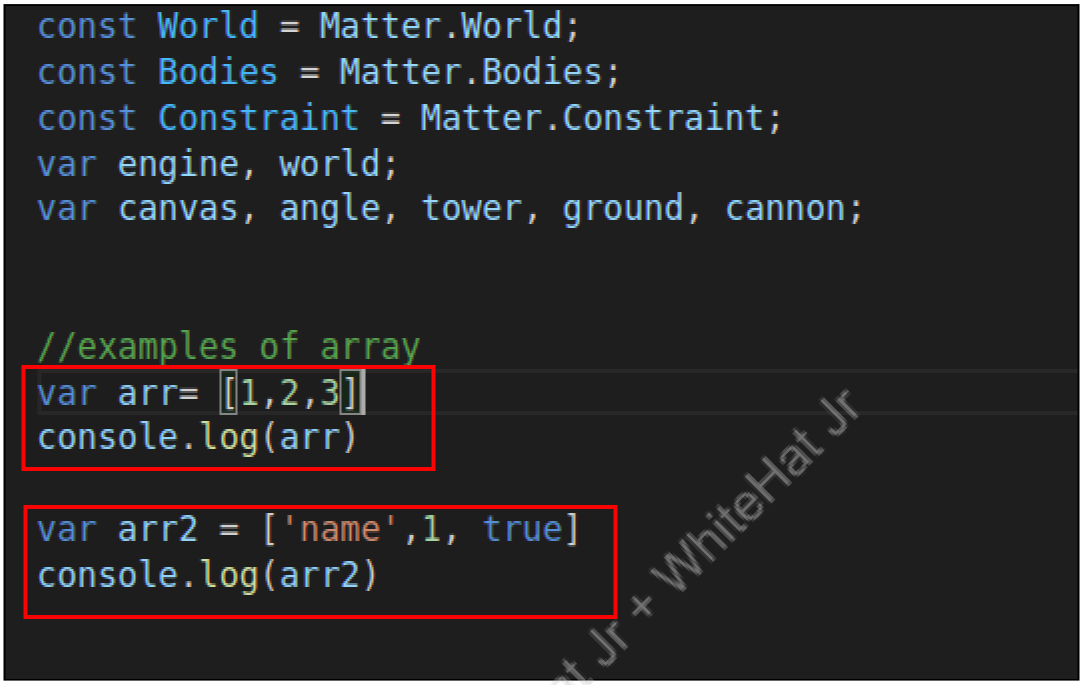
Which CONCEPTS/ CODING BLOCKS did we cover today?

- **Arrays()**
- Create a new cannonball on the press of the down arrow key.

How did we DO the activities?

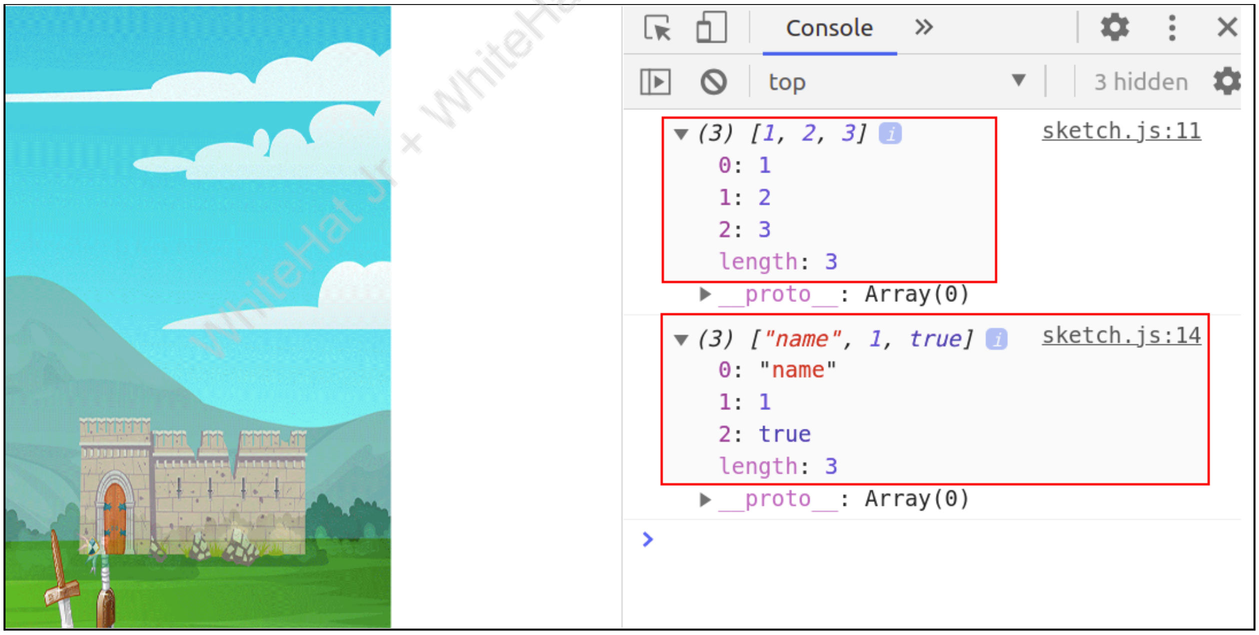
© 2019 The content of this email is confidential and intended for the recipient specified in message only. It is strictly forbidden to share any part of this message with any third party without a written consent of the sender. If you received this message by mistake, please reply to this message and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

1. Explore different ways to iterate on a 2D array.



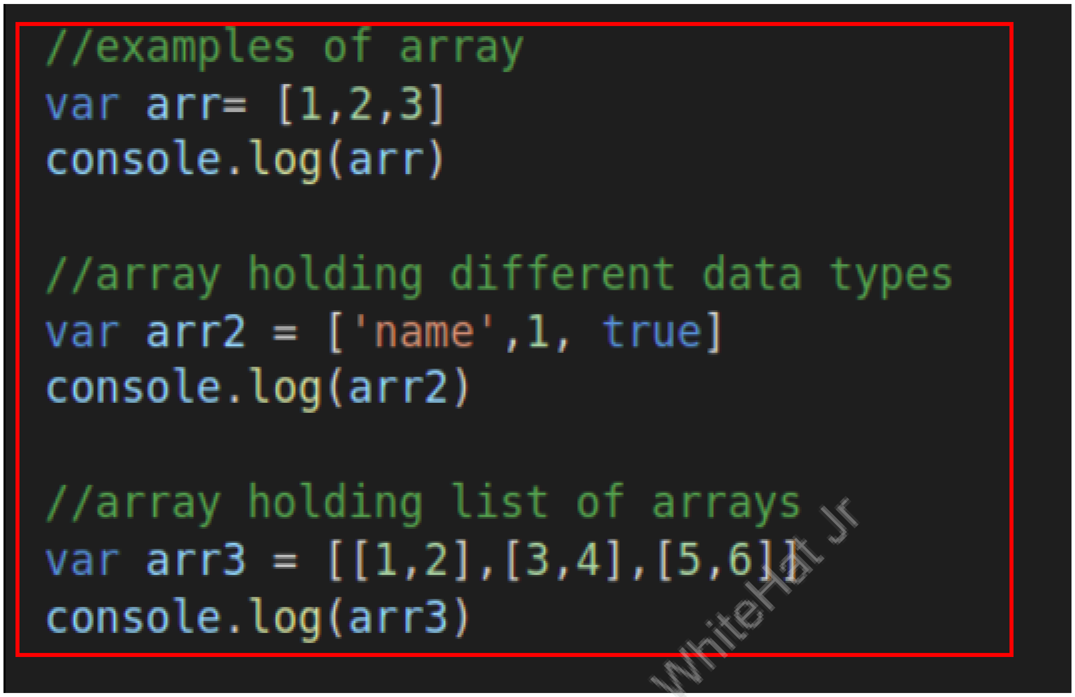
```
const World = Matter.World;
const Bodies = Matter.Bodies;
const Constraint = Matter.Constraint;
var engine, world;
var canvas, angle, tower, ground, cannon;


//examples of array
var arr= [1,2,3]
console.log(arr)

var arr2 = ['name',1, true]
console.log(arr2)
```

WhiteHat Jr + WhiteHat Jr + WhiteHat Jr + WhiteHat Jr

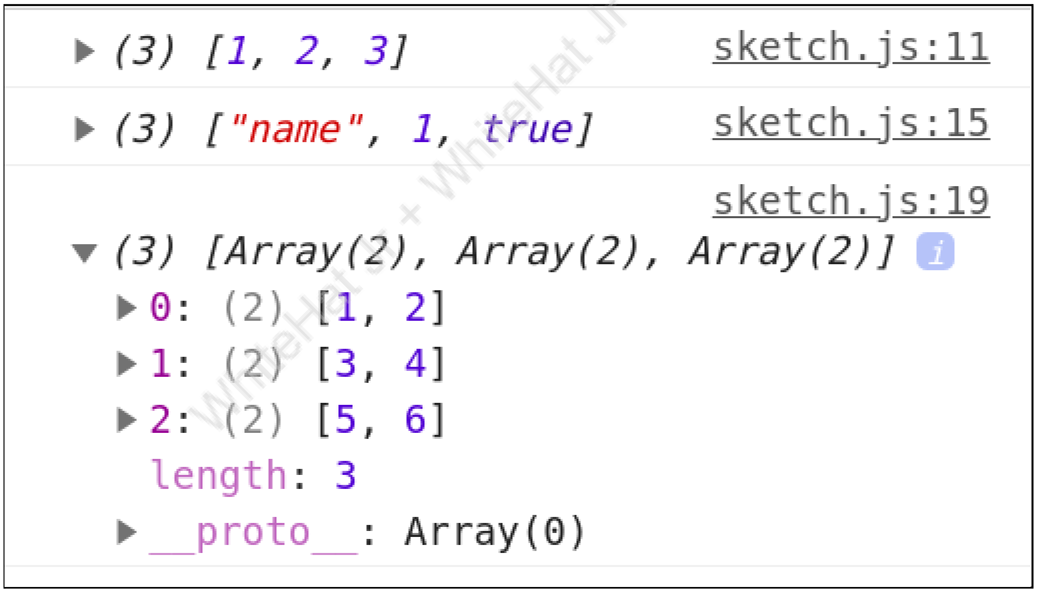


```
▼ (3) [1, 2, 3]                    sketch.js:11
    0: 1
    1: 2
    2: 3
    length: 3
  ▶ __proto__: Array(0)
▼ (3) ["name", 1, true]            sketch.js:14
    0: "name"
    1: 1
    2: true
    length: 3
  ▶ __proto__: Array(0)
>
```

© 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.



```
//examples of array
var arr= [1,2,3]
console.log(arr)

//array holding different data types
var arr2 = ['name',1, true]
console.log(arr2)

//array holding list of arrays
var arr3 = [[1,2],[3,4],[5,6]]
console.log(arr3)
```

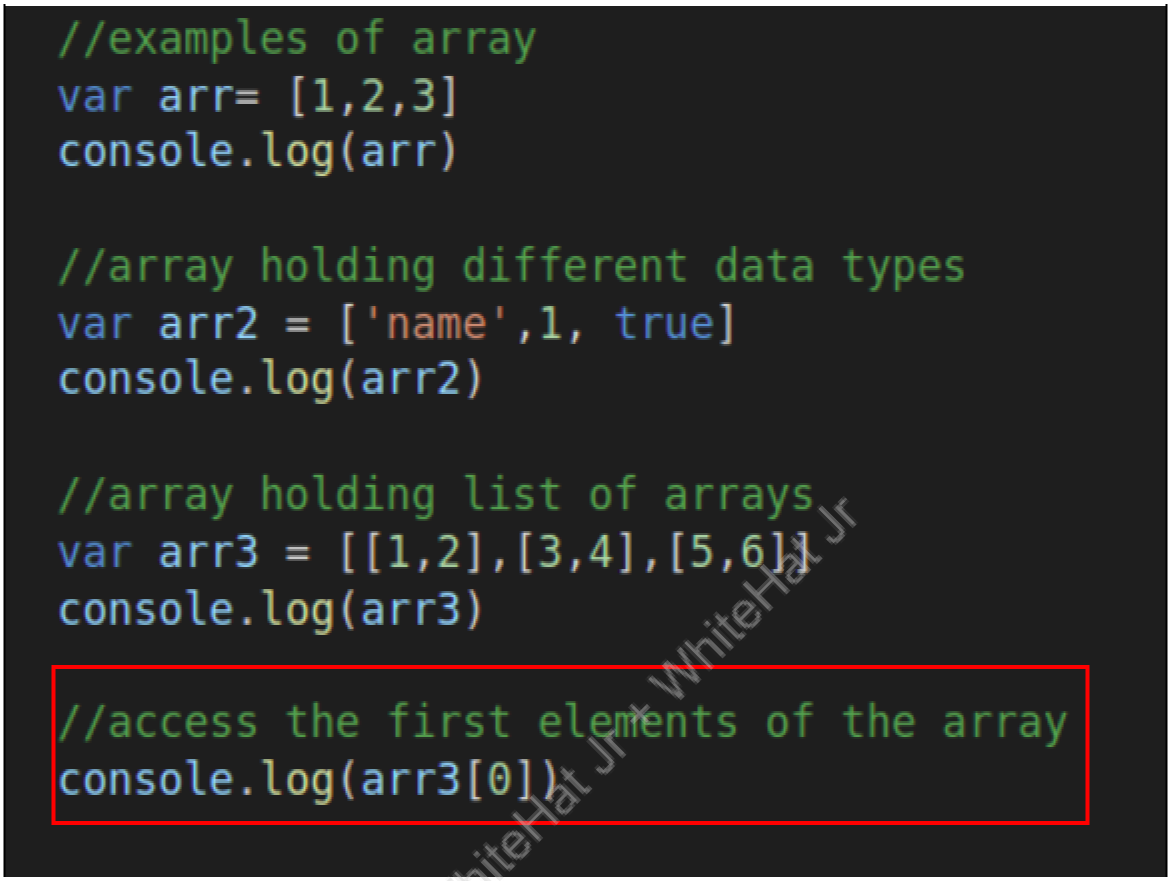


```
▶ (3) [1, 2, 3]                              sketch.js:11
▶ (3) ["name", 1, true]                      sketch.js:15
                                             sketch.js:19
▼ (3) [Array(2), Array(2), Array(2)] i
   ▶ 0: (2) [1, 2]
   ▶ 1: (2) [3, 4]
   ▶ 2: (2) [5, 6]
     length: 3
   ▶ __proto__: Array(0)
```

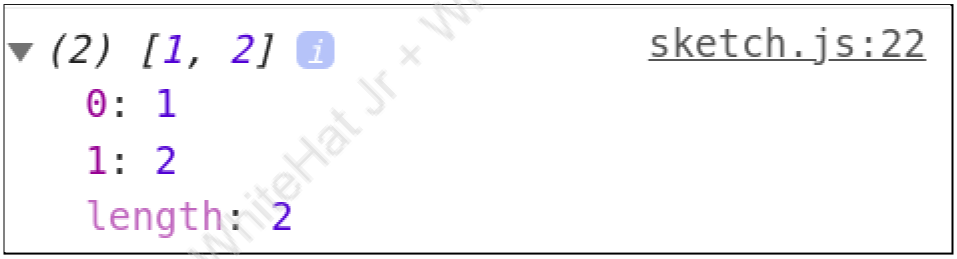
WhiteHat Jr + WhiteHat Jr + WhiteHat Jr

2. Try to access the first element of the array.

© 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.



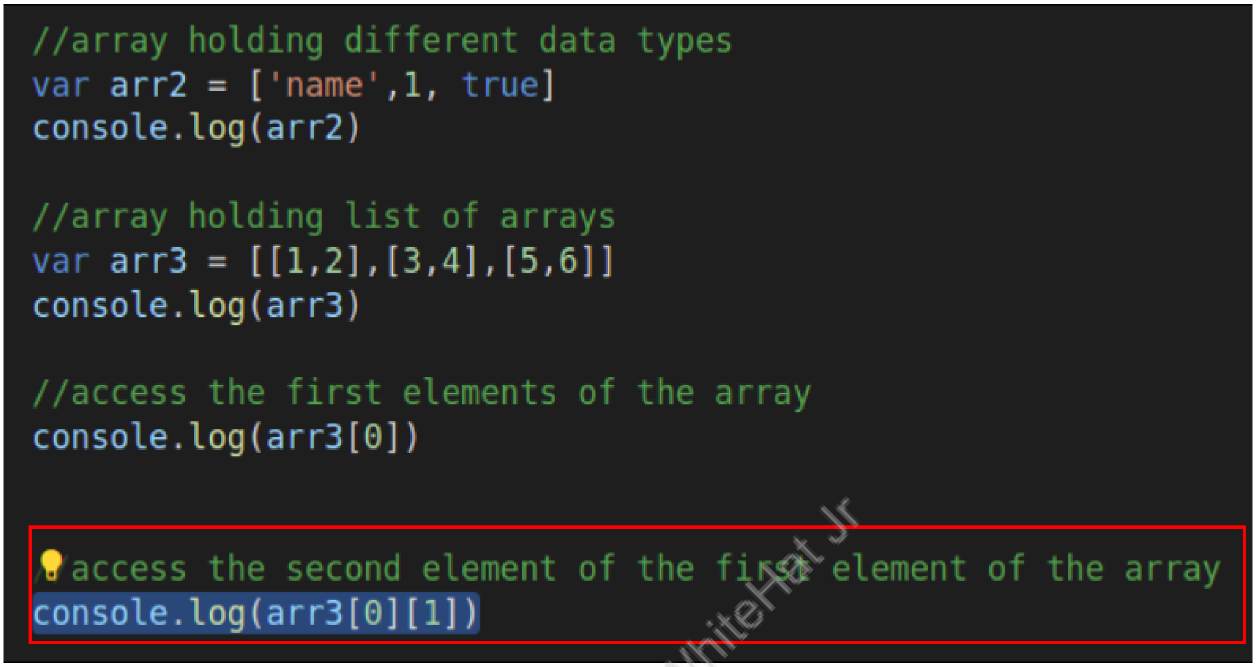
```
//examples of array
var arr= [1,2,3]
console.log(arr)

//array holding different data types
var arr2 = ['name',1, true]
console.log(arr2)

//array holding list of arrays
var arr3 = [[1,2],[3,4],[5,6]]
console.log(arr3)

//access the first elements of the array
console.log(arr3[0])
```



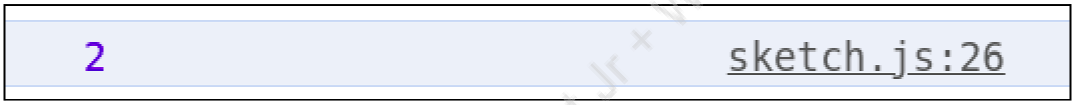
```
▼ (2) [1, 2] i                          sketch.js:22
    0: 1
    1: 2
    length: 2
```

WhiteHat Jr + WhiteHat Jr + WhiteHat Jr

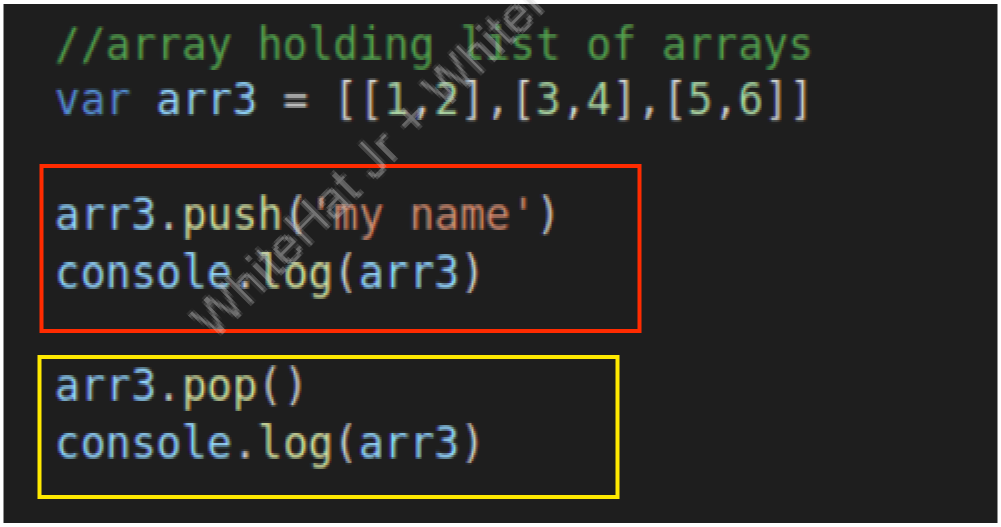
© 2019 The content of this email is confidential and intended for the recipient specified in message only. It is strictly forbidden to share any part of this message with any third party without a written consent of the sender. If you received this message by mistake, please reply to this message and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



```
//array holding different data types
var arr2 = ['name',1, true]
console.log(arr2)

//array holding list of arrays
var arr3 = [[1,2],[3,4],[5,6]]
console.log(arr3)

//access the first elements of the array
console.log(arr3[0])


//access the second element of the first element of the array
console.log(arr3[0][1])
```



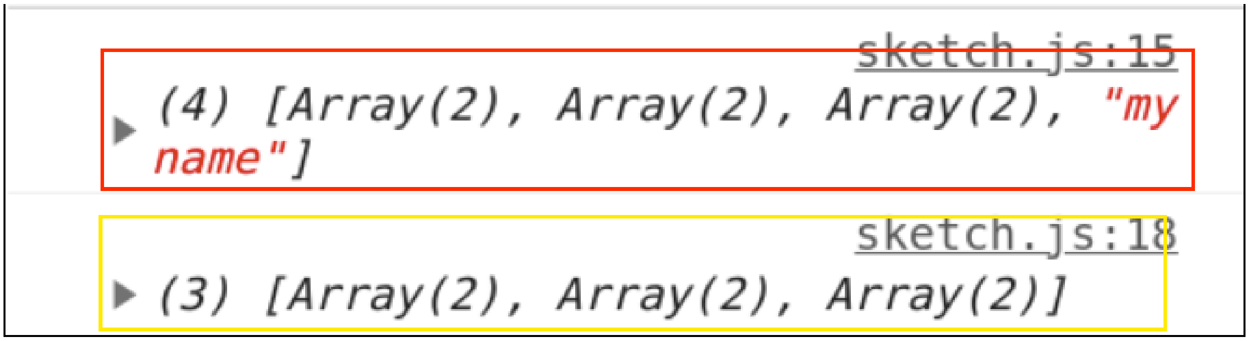
```
  2                                   sketch.js:26
```



```
//array holding list of arrays
var arr3 = [[1,2],[3,4],[5,6]]

arr3.push('my name')
console.log(arr3)

arr3.pop()
console.log(arr3)
```

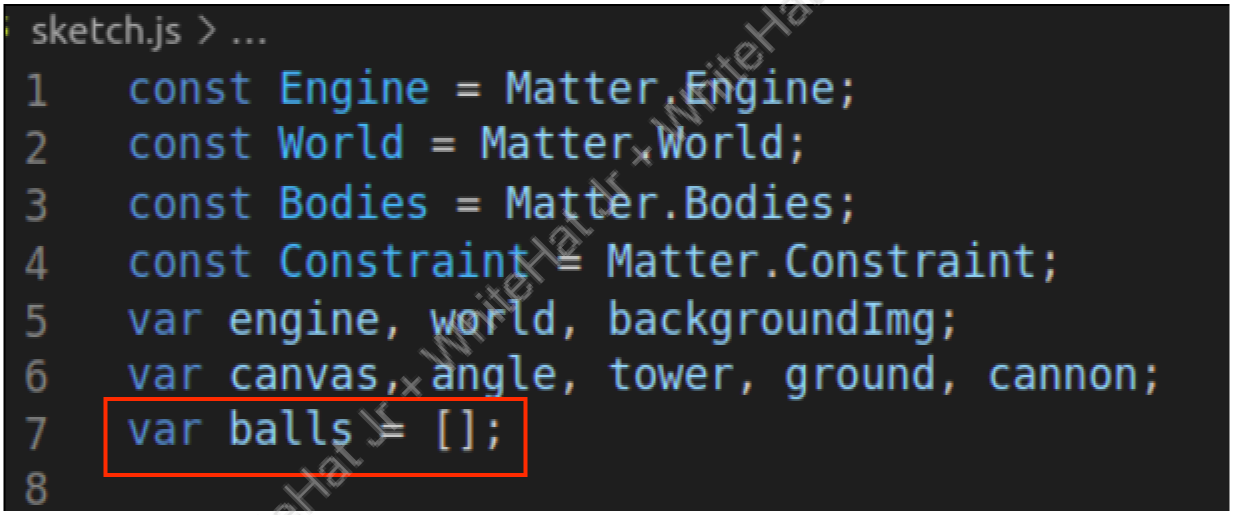
WhiteHat Jr + WhiteHat Jr + WhiteHat Jr

© 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.



```
                                          sketch.js:15
▶ (4) [Array(2), Array(2), Array(2), "my
  name"]
                                          sketch.js:18
▶ (3) [Array(2), Array(2), Array(2)]
```

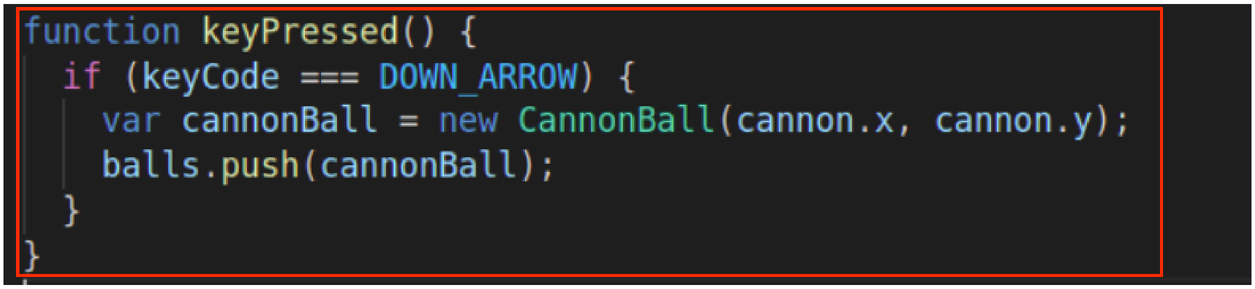
3. To create multiple cannonballs. Start by creating an empty array of balls.



```
sketch.js > ...
const Engine = Matter.Engine;
const World = Matter.World;
const Bodies = Matter.Bodies;
const Constraint = Matter.Constraint;
var engine, world, backgroundImg;
var canvas, angle, tower, ground, cannon;
var balls = [];

```

WhiteHat Jr

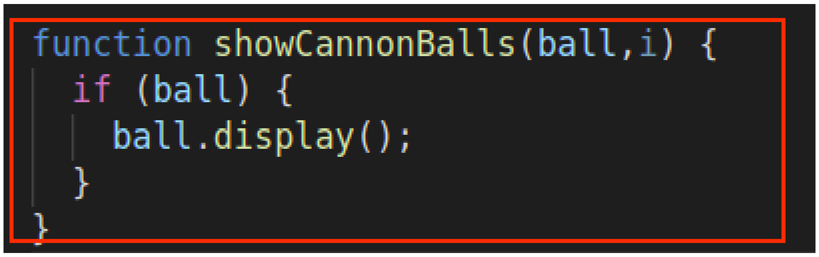
4. Create a cannonball when the **DOWN_ARROW** key is pressed and then push it into the **balls** array.



```
function keyPressed() {
  if (keyCode === DOWN_ARROW) {
    var cannonBall = new CannonBall(cannon.x, cannon.y);
    balls.push(cannonBall);
  }
}
```

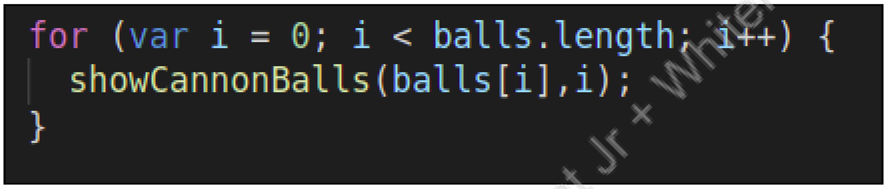
© 2019 The content of this email is confidential and intended for the recipient specified in message only. It is strictly forbidden to share any part of this message with any third party without a written consent of the sender. If you received this message by mistake, please reply to this message and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

5. Write a **showCannonballs()** function to display the cannonballs.



```
function showCannonBalls(ball,i) {
  if (ball) {
    ball.display();
  }
}
```

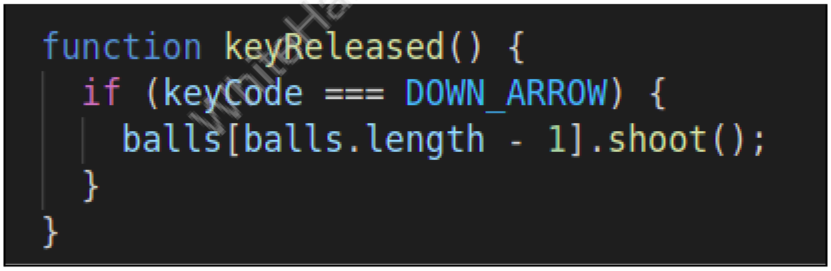
6. Write a for loop on the balls array to get all the balls. Call the **showCannonBalls()** function inside the loop.



```
for (var i = 0; i < balls.length; i++) {
  showCannonBalls(balls[i],i);
}
```

WhiteHat Jr + WhiteHat Jr + WhiteHat Jr

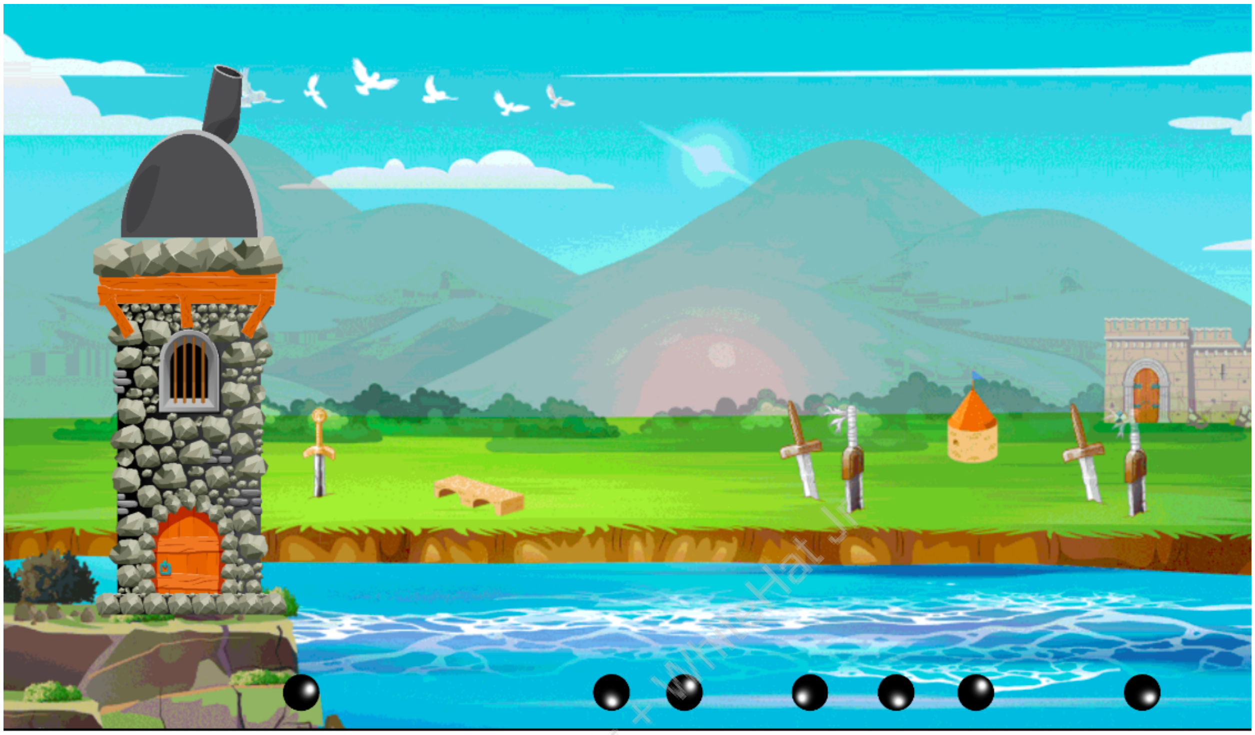
7. Shoot the cannonball when the **DOWN_ARROW** key is released. Call the **shoot()** function by accessing the balls from the array.



```
function keyReleased() {
  if (keyCode === DOWN_ARROW) {
    balls[balls.length - 1].shoot();
  }
}
```

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.



WhiteHat Jr + WhiteHat Jr + WhiteHat Jr

What's next?

In the next class, we will create pirate boats.

EXTEND YOUR KNOWLEDGE

Bookmark the following link to know more about loops and iterations in the following link created by Mozilla and individual contributors:

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Loops_and_iteration

About MDN by Mozilla Contributors is licensed under CC-BY-SA 2.5.

© 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.