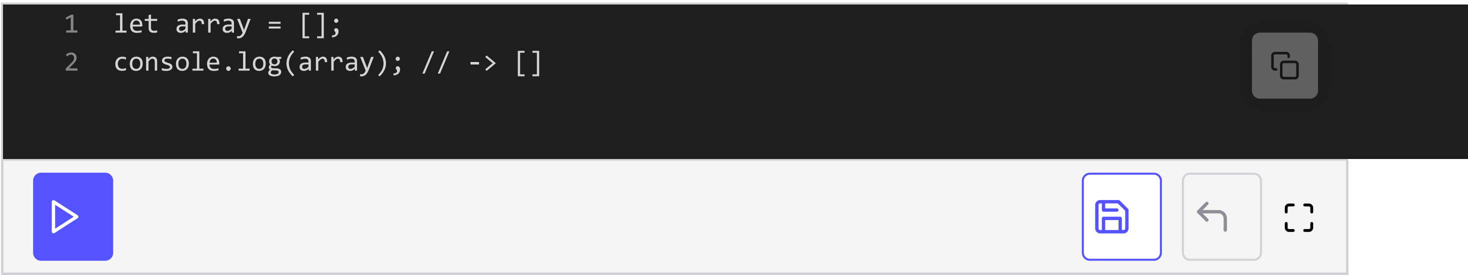
Arrays, Part 0

Learn a more complex data type: the array. Arrays are collections of other variables. They're used to store items and work with them collectively. All items in an array are ordered and indexed, with the first index being 0.

We've seen how to create booleans, numbers, and strings. We're now going to create a new type of variable called an array.

Creating an Array

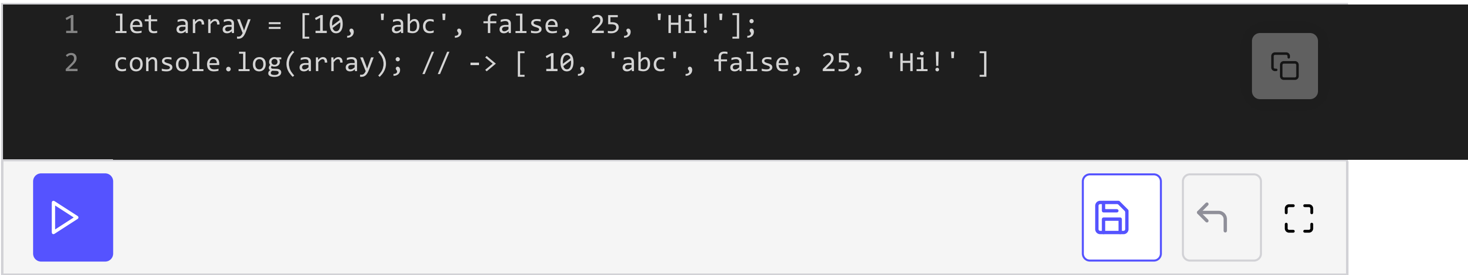
Arrays can be thought of as containers. They contain multiple other variables inside them in an ordered way. They're created with the `[]` characters.



```
let array = [];
console.log(array); // -> []
```

Adding Items

The array above is empty. To create an array with items inside it, we write items inside the brackets, separated by commas.



```
let array = [10, 'abc', false, 25, 'Hi!'];
console.log(array); // -> [ 10, 'abc', false, 25, 'Hi!' ]
```

We can see that `array` is now a collection of items that contains values.

Accessing Items

We can access each of these items individually. Each item inside an array has an **index**. Array indexes start at 0.

To access an item at a certain index, we use the name of the array followed by the index in brackets. Look at the following example.

```
let array = [ 10, 'abc', false, 25, 'Hi!'];
// indexes:   0     1      2     3    4

console.log(array[0]); // -> 10
console.log(array[1]); // -> abc
console.log(array[2]); // -> false
console.log(array[3]); // -> 25
console.log(array[4]); // -> Hi!
```

In this code block, `array` is an array with 5 items. The first item is at index 0 and the last item is at index 4.

Length

We can see how many items an array contains by using `.length` after the array's name.

```
let array = [10, 'abc', false, 25, 'Hi!'];
console.log(array.length); // -> 5
```

We can see that the length is always 1 greater than the last index. In other words, the last item in an array is always at index `array.length - 1`. This means that to get the last item, we can write `array[array.length - 1]`.

Changing Items

We can change an item at a certain index using `=`, like we would on a normal variable.

```
let array = ['a', 'b', 'c'];
console.log(array); // -> ['a', 'b', 'c']

array[0] = 0;
```

```
console.log(array); // -> [0, 'b', 'c']

array[1] = 1;

console.log(array); // -> [0, 1, 'c']

array[2] = 2;
console.log(array); // -> [0, 1, 2]
```

We can also add an item using this method. For the array above, we would place it at the next avaible index, 3.

```
let array = ['a', 'b', 'c'];
array[3] = 3;
console.log(array); // -> ['a', 'b', 'c', 3]
```

We'll get deeper into arrays in the next lesson.

Quiz

Feel free to test your understanding.

1 What will the following code print?

```
let arr = [1, 2, 3, 4, 5];
console.log(arr[5]);
```

○ A) 2

○ B) 3

○ C) 4

○ D)

5

○ E) None of the above

2

What will the following code print?

```
let arr = [1, 2, 3, 4, 5];
arr[5] = 6;

console.log(arr.length);
```

○ A) 3

○ B) 4

○ C) 5

○ D) 6

○ E) None of the above

3

What will the following code print?

```
let arr = [1, 2, 3];
console.log(arr.length + arr[2]);
console.log(arr.length + arr[2]);
```

○ A) 3

○ B) 4

○ C) 5

○ D) 6

○ E) None of the above

4

What will the following code print?

```
let arr = [1, 2, 3];
arr[3] = 4;

console.log(arr.length + arr[4]);
```

○ A) 4

○ B) 5

○ C) 6

○ D) 7

E)

○ None of the above

CHECK ANSWERS