JavaScript Essentials

Arrays

Table of Contents

Lesson Objectives

- Understand array – a neat way of storing a list of data items
- Able to create an array, retrieve, add and remove items stored in an array

Section 1

Overview – What is an array?

Overview – What is an array?

- Arrays are generally described as "list-like objects"
- Basically single objects that contain multiple values stored in a list
- Array objects can be stored in variables and dealt with in much the same way as any other type of value
- The difference being that we can access each value inside the list individually
- Do super useful and efficient things with the list, like loop through it and do the same thing to every value

Overview – Create an array

- Syntax: Arrays consist of square brackets and elements that are separated by commas.
- Suppose we want to store a shopping list in an array. Paste the following code into the console:
  let shopping = ['bread', 'milk', 'cheese', 'hummus', 'noodles']; shopping;
- In the above example, each element is a string, but in an array we can store various data types — strings, numbers, objects, and even other arrays. We can also mix data types in a single array — we do not have to limit ourselves to storing only numbers in one array, and in another only strings. For example:
  let sequence = [1, 1, 2, 3, 5, 8, 13]; let random = ['tree', 795, [0, 1, 2]];

Overview – Access and modify items

- Syntax: access individual items in the array using bracket
- Modify an item in an array by simply giving a single array item a new value
- **Computers start counting from 0!**
- Note that an array inside an array is called a multidimensional array.
- Access an item inside an array that is itself inside another array by chaining two sets of square brackets together

Overview – Find the length of an array

- Syntax: by using the **length** property
- Modify an item in an array by simply giving a single array item a new value
- Most commonly used to tell a loop to keep going until it has looped through all the items in an array
- Remove item from the array by set the length of the array to be smaller than the current value

Overview – Summary

- Array provide a neat way of storing a list of data items under a single variable
- Arrays are generally described as "list-like objects"; they are basically single objects that contain multiple values stored in a list
- Array objects can be stored in variables and dealt with in much the same way as any other type of value
- The difference being that we can access each value inside the list individually
- Always remember: Computer starts at 0 (not 1)

Section 2

Useful methods

Useful methods – Convert string to array

- Often you'll be presented with some raw data contained in a big long string, and you might want to separate the useful items out into a more useful form and then do things to them, like display them in a data table
- To do this, we can use the split() method. In its simplest form, this takes a single parameter, the character you want to separate the string at, and returns the substrings between the separator as items in an array.
- **Note**: Okay, this is technically a string method, not an array method, but we've put it in with arrays as it goes well here.

Useful methods – Convert string to array

1. Let's play with this, to see how it works. First, create a string in your console:

```
let myData = 'Manchester,London,Liverpool,Birmingham,Leeds,
```

2. Now let's split it at each comma:

```
let myArray = myData.split(',');
myArray;
```

3. Finally, try finding the length of your new array, and retrieving some items from it:

```
myArray.length;
myArray[0]; // the first item in the array
myArray[1]; // the second item in the array
myArray[myArray.length-1]; // the last item in the array
```

4. You can also go the opposite way using the `join()` method. Try the following:

```
let myNewString = myArray.join(',');
myNewString;
```

Useful methods – Add or remove items

- We've not yet covered adding and removing array items — let us look at this now. We'll use the myArray array we ended up with in the last section. If you've not already followed that section, create the array first in your console:
- `let myArray = ['Manchester', 'London', 'Liverpool', 'Birmingham', 'Leeds', 'Carlisle'];`
  First of all, to add or remove an item at the end of an array we can use push() and pop() respectively.

Useful methods – Add or remove items

1. Let's use push() first — note that you need to include one or more items that you want to add to the end of your array. Try this: myArray.push('Cardiff'); myArray; myArray.push('Bradford', 'Brighton'); myArray;
2. The new length of the array is returned when the method call completes. If you wanted to store the new array length in a variable, you could do something like this:let newLength = myArray.push('Bristol'); myArray; newLength;
3. Removing the last item from the array is as simple as running pop() on it. Try this:myArray.pop();
4. The item that was removed is returned when the method call completes. To save that item in a new variable, you could do this:let removedItem = myArray.pop(); myArray; removedItem;

Useful methods – Add or remove items

- unshift() and shift() work in exactly the same way as push() and pop(), respectively, except that they work on the beginning of the array, not the end.

1. First unshift() — try the following commands:myArray.unshift('Edinburgh'); myArray;
2. Now shift(); try these!let removedItem = myArray.shift(); myArray; removedItem;

Useful methods - Summary

- Use strings.split to convert a strings into an array
- To add and remove item (at the end) use push and pop
- To add and remove item (at the beginning) use shift and unshift
- Note: shift and unshift might affect the performance of the program

Section 3

Practice time

Practice Arrays manipulation

Practice Arrays manipulation 2

Practice Template Strings

Practice time - Summary

- When working we array, first think about single item then generalize to all items

Thank you

Q&A