Arrays I

Overview

```
1  /*
2    - what's an array?
3
4    - typeof []
5
6    - Array.isArray
7
8    - bracket access, bracket assignment
9
10   - .length property
11
12   - basic methods
13 */
14
```

What is an array?

```
1  /* An array is a list-like data structure in JavaScript */
2
3  let numbers = [1, 2, 3];
4
5  let names = ['George', 'John', 'Thomas'];
6
7  let aVariable = 'a value';
8
9  let mixedBag = [30, true, 'apples', null, aVariable];
10
11 /* the values inside of an array are called elements */
12
13
14
```

What is the typeof an array?

```
1   let names = ['George', 'John', 'Thomas'];
2
3   console.log(typeof names);
4
5
6
7
8
9
10
11
12
13
14
```

Array.isArray

```
1   let names = ['George', 'John', 'Thomas'];
2
3   console.log(Array.isArray(names));
4
5   console.log(Array.isArray('i am not an array'));
6
7
8
9
10
11
12
13
14
```

Bracket access

```
1   /* Access elements in an array the same way you'd access a character in a
2      string: using brackets and the index number corresponding to the
3      position of the element inside the array */
4
5   let names = ['George', 'John', 'Thomas'];
6
7   console.log(names[0]);
8
9   console.log(names[1]);
10
11  console.log(names[2]);
12
13  console.log(names[3]);
14
```

Bracket assignment

```
1   /* Use brackets and the assignment operator to assign new values to index
2       positions in an array */
3
4   let names = ['George', 'John', 'Thomas'];
5
6   names[0] = 'Washington';
7
8   names[1] = 'Adams';
9
10  names[2] = 'Jefferson';
11
12  console.log(names);
13
14
```

.length property

```
1    /* Arrays, like strings, have a length property */
2
3    let names = ['George', 'John', 'Thomas'];
4
5    console.log(names.length)
6
7
8
9
10
11
12
13
14
```

.push method

```
1   /* .push takes one or more elements and adds them to the end of the array.
2      .push returns the new length of the array. */
3
4   let names = ['George', 'John', 'Thomas'];
5
6   let new length = names.push('James');
7
8   console.log(names);
9   console.log(newLength);
10
11
12
13
14
```

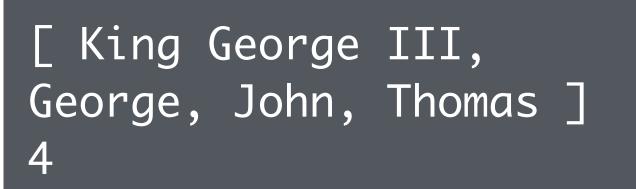# .pop method

```
1    /* .pop removes one element from the end of the array. it returns the
2       removed element */
3
4    let names = ['George', 'John', 'Thomas'];
5
6    let jefferson = names.pop();
7
8    console.log(names);
9    console.log(jefferson)
10
11
12
13
14
```

.shift method

```
1   /* .shift works like .pop, but it removes the first element instead */
2
3   /* it's called shift because the indices for every element in the array
4      are shifted over by one */
5
6   let names = ['George', 'John', 'Thomas'];
7
8   let washington = names.shift();
9
10  console.log(names);
11  console.log(washington);
12
13
14
```
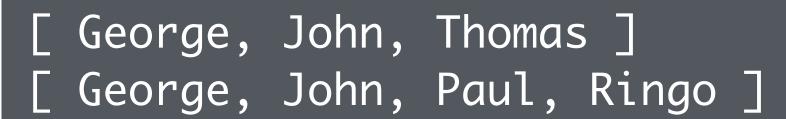
.unshift method

```
1   /* .unshift adds one or more elements to the front of the array */
2
3   let names = ['George', 'John', 'Thomas'];
4
5   let newLength = names.unshift('King George III');
6
7   console.log(names);
8   console.log(newLength);
9
10
11
12
13
14
```

.indexOf method

```
1   /* .indexOf is also an array method, and works the same way as the string
2      method of the same name */
3
4   let names = ['George', 'John', 'Thomas'];
5
6   console.log(names.indexOf('George'));
7   console.log(names.indexOf('Alexander'));
8
9
10
11
12
13
14
```

.slice method

```
1   /* .slice is also an array method, and works the same way as the string
2      method of the same name. */
3
4   let names = ['George', 'John', 'Thomas'];
5
6   let oneTermPresidents = names.slice(1, 2);
7
8   console.log(oneTermPresidents);
9   console.log(names);
10
11
12
13
14
```

.slice method

```
1  /* .slice is also an array method, and works the same way as the string
2     method of the same name. */
3
4  let names = ['George', 'John', 'Thomas'];
5
6  let namesCopy = names.slice();
7
8  namesCopy[2] = 'Paul';
9  namesCopy.push('Ringo');
10
11 console.log(names);
12 console.log(namesCopy);
13
14
```

.includes method

```
1    /* .includes takes a value, and returns true if the value is an element in
2       the array */
3
4    let names = ['George', 'John', 'Thomas'];
5
6    console.log(names.includes('George'));
7    console.log(names.includes('Alexander'));
8
9
10
11
12
13
14
```

.reverse method

```
1   /* .reverse mutates (changes) the original array, reversing the order of
2       its elements */
3
4   let names = ['George', 'John', 'Thomas'];
5
6   names.reverse();
7
8   console.log(names);
9
10
11
12
13
14
```

Recap

```
1   /*
2     - what's an array?
3
4     - typeof []
5
6     - Array.isArray
7
8     - bracket access, bracket assignment
9
10    - .length property
11
12    - basic methods
13  */
14
```