Arrays Lab

In this lab, let's work with arrays. We'll keep using our testing structure.

Setup

1. Create a new test file called arrays.js and include it in your testing web page.
2. Run your tests to make sure they still work alright.

Creating a people array

3. Create a people variable that has five person objects in it. You can make the person look any way you like but they should have a first and last name at minimum.
4. Write a new testing function. It should look like this:

```
const testPerson = { first: "Jo", last: "Bennett"};
// Add the testPerson to the end of the array here
if (people.length !== 6) throw "Should have six people";
if (people[5].first !== "Jo") throw "Jo should be the last person";
// Add a person to the front of the array here
if (people.length !== 7) throw "Should have seven people";
if (people[0].last !== "NewPerson") throw "New person should be first";
// Remove a person from the end of the array here
if (people.length !== 6) throw "Should have deleted a person";
if (people[5].first === "Jo") throw "Removed the wrong person";
// Remove a person from the front of the array here
if (people.length !== 5) throw "Should have deleted another person";
if (people[0].last === "NewPerson") throw "Removed the wrong person";
```

5. Now fill in all of those comments above. You'll know you've got it right when you throw no errors.

Options for looping

6. Write a new test function called *canIterateTwoWays*.
7. It should have these two comments in the test file:

```
// loop through the persons array using "for in", console.logging each.
// loop through the persons array using "for of", console.logging each.
```

8. Make the tests console.log each person as they should.

Destructuring

9. Write a test function called *canDestructure*. It should ...

```
// Create five variables called p1, p2, p3, p4, and p5. Each is a person
// from your array of persons. Use destructuring to do that.
if (p1 !== people[0]) throw "Destructuring failed";
if (p2 !== people[1]) throw "Destructuring failed";
if (p3 !== people[2]) throw "Destructuring failed";
if (p4 !== people[3]) throw "Destructuring failed";
if (p5 !== people[4]) throw "Destructuring failed";
```

10. Make the test pass by using destructuring to create those five variables from the people array.

Converting people to an array of strings

11. WrIte a test function called canConvertYourArrayOfPeopleIntoAnArrayOfStrings.
12. It should do this...

```
// Convert it here
if (arrayOfStrings[0] !== `name: ${people[0].first} ${people[0].last}`)
  throw "Converting failed".
```

```
if (arrayOfStrings[1] !== `name: ${people[1].first} ${people[1].last}`)
  throw "Converting failed".
if (arrayOfStrings[2] !== `name: ${people[2].first} ${people[2].last}`)
  throw "Converting failed".
if (arrayOfStrings[3] !== `name: ${people[3].first} ${people[3].last}`)
  throw "Converting failed".
if (arrayOfStrings[4] !== `name: ${people[4].first} ${people[4].last}`)
  throw "Converting failed".
```

13. Write your JavaScript code in the first line; the one where it says "// Convert it here". See if you can make it a single line of JavaScript without looping imperatively.
14. You'll know you did it right when you can run it to completion without any exceptions being thrown.