Rev. 08Aug2019

Exercise 32

A LITTLE DEEPER INTO ARRAYS

You know how to access individual members of an array directly using the `arrayName[offsetNumber]` pattern. Most of the time, though, when we work with arrays, we want to do something to *all* the members in the array. This is where arrays shine.

Let's start with a very simple use case: we want to output each array member to the web page. We can start again with our example from Exercise 31:

```
let temps = [84, 89, 78, 79, 91, 95, 90]
```

Arrays come ready-made with some built-in functions for processing every member of an array. All of the functions work the same basic way: they accept a function as an argument.

We saw previously how a function can accept another function as an argument, so there's nothing new about that. Here's the twist: the function we pass into the *array function* accepts an argument which is one of the members of the array. This is a concept that, when you first encounter it, appears mercilessly hard to understand. Once you get it, it will seem completely natural to you.

That could actually be said about most things you'll learn in programming: hard to understand at first; obvious once you "get it"

I'm going to try to illustrate how this works. My graphical skills are "limited", so here goes: think of the members of an array as values existing in little boxes:

84	89	78	79	91	95	90

Now think of a function that you'll write as accepting one of these boxes:

```
let printTemp = (temperature) => {
  console.log(temperature)
}
```

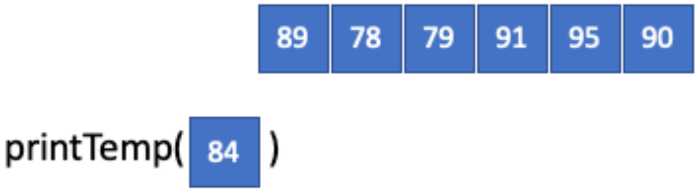
So far so good? Onward then!

I said that arrays have some built-in functions that are particularly well suited to working with *all* the elements in an array. Let's take one of the simplest of these built-in functions: `forEach`. The `forEach` function accepts a function that you write and calls that function once for each member of the array.

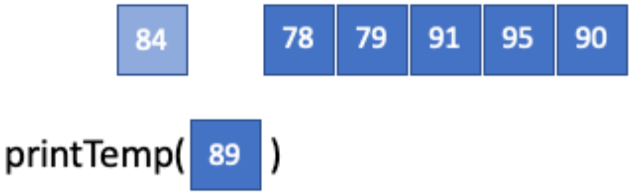
Let's say we pass in the `printTemp` function (from the previous page) into the `temps` array's `forEach` function. Here's what the code for that would look like:

```
temps.forEach(printTemp)
```

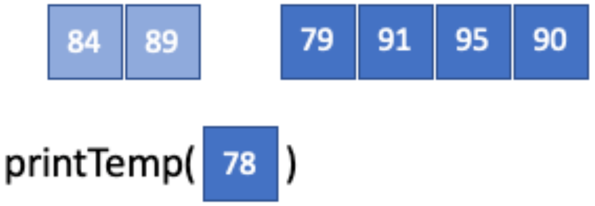
That `printTemp` function will actually be called *seven times* — once for each temperature in our `temps` array. So the first pass, you could imagine it looking like this:



`printTemp` will print the number `84` to the console and be called again — this time with the *second* member of the `temps` array:



`printTemp` now prints `89` to the console and the `forEach` function picks the *next* item in the array:



`printTemp` now prints `78` to the console — and so it goes until all of the members of `temps` have been processed by `printTemp`.

□ With that said, examine the code in `index.js` and run the repl. Make sure to check the **console** tab.