Looping through objects in JavaScript

Once in a while, you may need to loop through Objects in JavaScript. The only way to do so before ES6 is with a `for...in` loop.

The problem with a `for...in` loop is that it iterates through properties in the Prototype chain. When you loop through an object with the `for...in` loop, you need to check if the property belongs to the object. You can do this with `hasOwnProperty`.

```
for (var property in object) {
  if (object.hasOwnProperty(property)) {
    // Do things here
  }
}
```

We no longer have to rely on `for...in` and `hasOwnProperty` now. There's a better way.

A better way to loop through objects

The better way to loop through objects is first to convert the object into an array. Then, you loop through the array.

You can convert an object into an array with three methods:

1. `Object.keys`
2. `Object.values`
3. `Object.entries`

Object.keys

`Object.keys` creates an array that contains the properties of an object. Here's an example.

```
const fruits = {
  apple: 28,
  orange: 17,
  pear: 54,
}

const keys = Object.keys(fruits)
console.log(keys) // [apple, orange, pear]
```

Object.values

`Object.values` creates an array that contains the values of every property in an object. Here's an example:

```
const fruits = {
  apple: 28,
  orange: 17,
  pear: 54,
}

const values = Object.values(fruits)
console.log(values) // [28, 17, 54]
```

Object.entries

`Object.entries` creates an array of arrays. Each inner array has two item. The first item is the property; the second item is the value.

Here's an example:

```
const fruits = {
  apple: 28,
  orange: 17,
  pear: 54,
}

const entries = Object.entries(fruits)
console.log(entries)
// [
//   [apple, 28],
//   [orange, 17],
//   [pear, 54]
// ]
```

My favorite of the three is `Object.entries` because you get both the key and property values.

Looping through the array

Once you've converted the object into an array with `Object.keys`, `Object.values`, or `Object.entries`, you can loop through it as if it was a normal array.

```
// Looping through arrays created from Object.keys
const keys = Object.keys(fruits)
for (const key of keys) {
  console.log(key)
}

// Results:
// apple
// orange
// pear
```

If you use `Object.entries` you might want to [destructure](#) the array into its key and property.

```
for (const [fruit, count] of entries) {
  console.log(`There are ${count} ${fruit}s`)
}

// Result
// There are 28 apples
// There are 17 oranges
// There are 54 pears
```

Wrapping up

The better way to loop through objects is first convert it into an array with one of these three methods.

1. `Object.keys`
2. `Object.values`
3. `Object.entries`

Then, you loop through the results like a normal array.

If this lesson has helped you, might enjoy Learn JavaScript, where you'll learn how to build anything you want from scratch. Enrollment for Learn JavaScript opens in July 2018 (in two weeks!).