Learn Hash Maps: JavaScript

Intro to Hash Maps

3 min

Hash maps are data structures that serve as efficient key-value stores. They are capable of assigning and retrieving data in the fastest way possible. This is because the underlying data structure that hash maps use is an array.

A value is stored at an array index determined by plugging the key into a hash function. Because we always know exactly where to find values in a hash map, we have constant access to any of the values it contains.

This quick access to values makes a hash map a good choice of data structure whenever we need to store a lot of values but need fast look-up time.

In JavaScript, objects are often used to map keys to values as in a hash map, but in this lesson, you'll create your own implementation of a hash map by building out a HashMap class. You'll build methods to hash and compress a given key, assign an index at which to store a value, and retrieve that value.

To implement a hash map, the HashMap constructor method will create an empty array that will hold values. A hashing function will return an index in the array where the value will be stored. While going through the following exercises, remember that the purpose of this lesson is to gain a deeper understanding of the data structure rather than creating a production-quality data structure.

Instructions

1. Checkpoint 1 Passed

1.

The hash map class we will be completing is stored in the **HashMap.js** file. Look over the code you've been given, the constructor method.

Hash maps are based on arrays, so whenever we create a new hash map instance, we'll create an array filled with null values that can be replaced with key-value pairs later.

HashMap.js

```
class HashMap {
  constructor(size = 0) {
    this.hashmap = new Array(size)
      .fill(null);
  }
}

module.exports = HashMap;
```