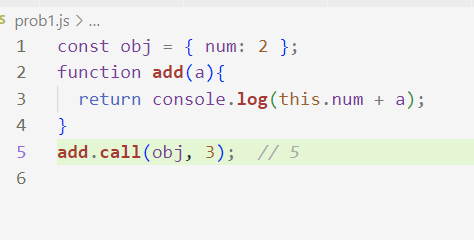
**When to Use Bind(), Call(), and Apply() in JavaScript**

Traditionally in JavaScript, you can have objects that have their own properties and methods. For example, object1 cannot use the methods of object2 and vice versa

But there’s a way to overcome this restriction. You can use call(), apply(), and bind() methods to tie a function into an object and call the function as if it belonged to that object.

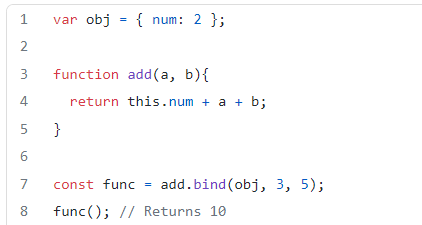
we can write function outside the object but we can use as a object method if we use call()



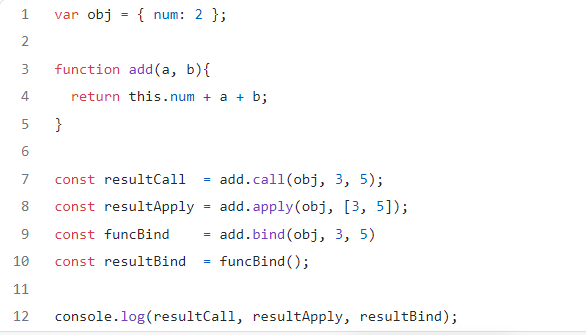


**Bind() Method in JavaScript**

**It works like as call(), apply(). It doesn’t invoke immediately, it calls later. Way of writing syntax is little different.**

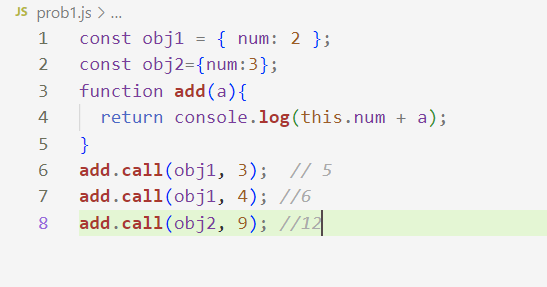
****

**Here summery of 3 methods**

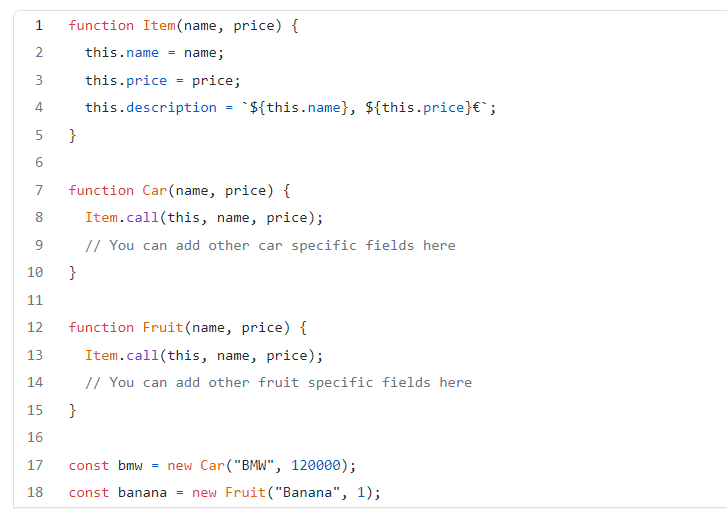
****

**Advantages of call()**

**Look at the above example if we want to use same function in different object, we should not write again and again. And if we use different argument in same object it gives us facility.**

****

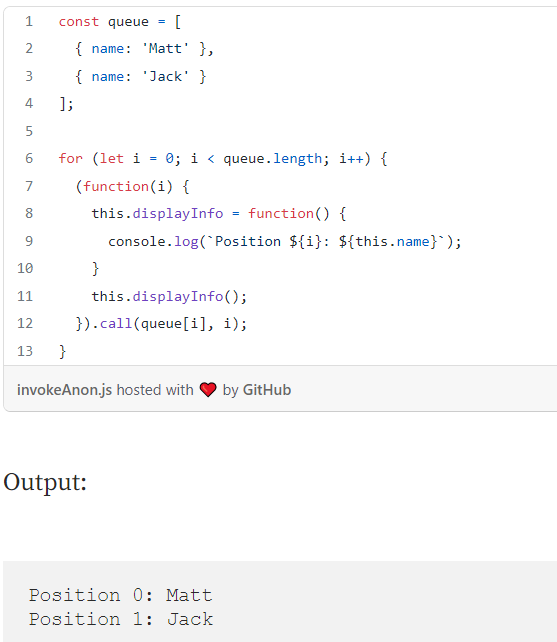
**Use Call() to Chain Object Constructors function**

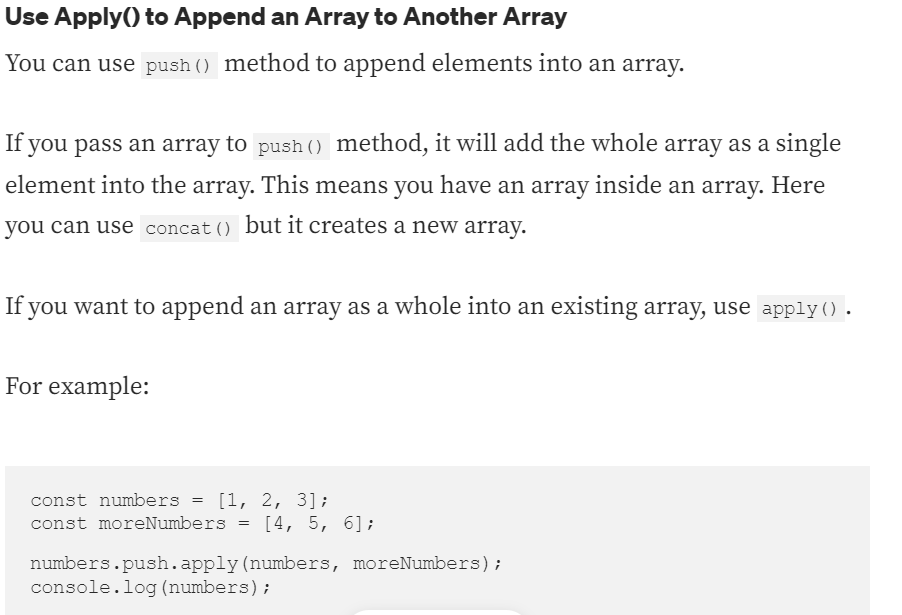
****

**Use Call() to Invoke an Anonymous Function**

**Let’s create an anonymous function and use call() to invoke it for each object of an array.**

**The anonymous function adds a displayInfo() function for each array object. This is to make it print the correct position of each person in the queue:**

****



**Use Bind() to Make SetTimeout Work**

* **1.** pass the bind function fun with this value set to the person object to the setTimeout() function.



2. without bind()

