Prototype in Javascript

Simplified

-Radhika Sahebrao sathe

Prototype:

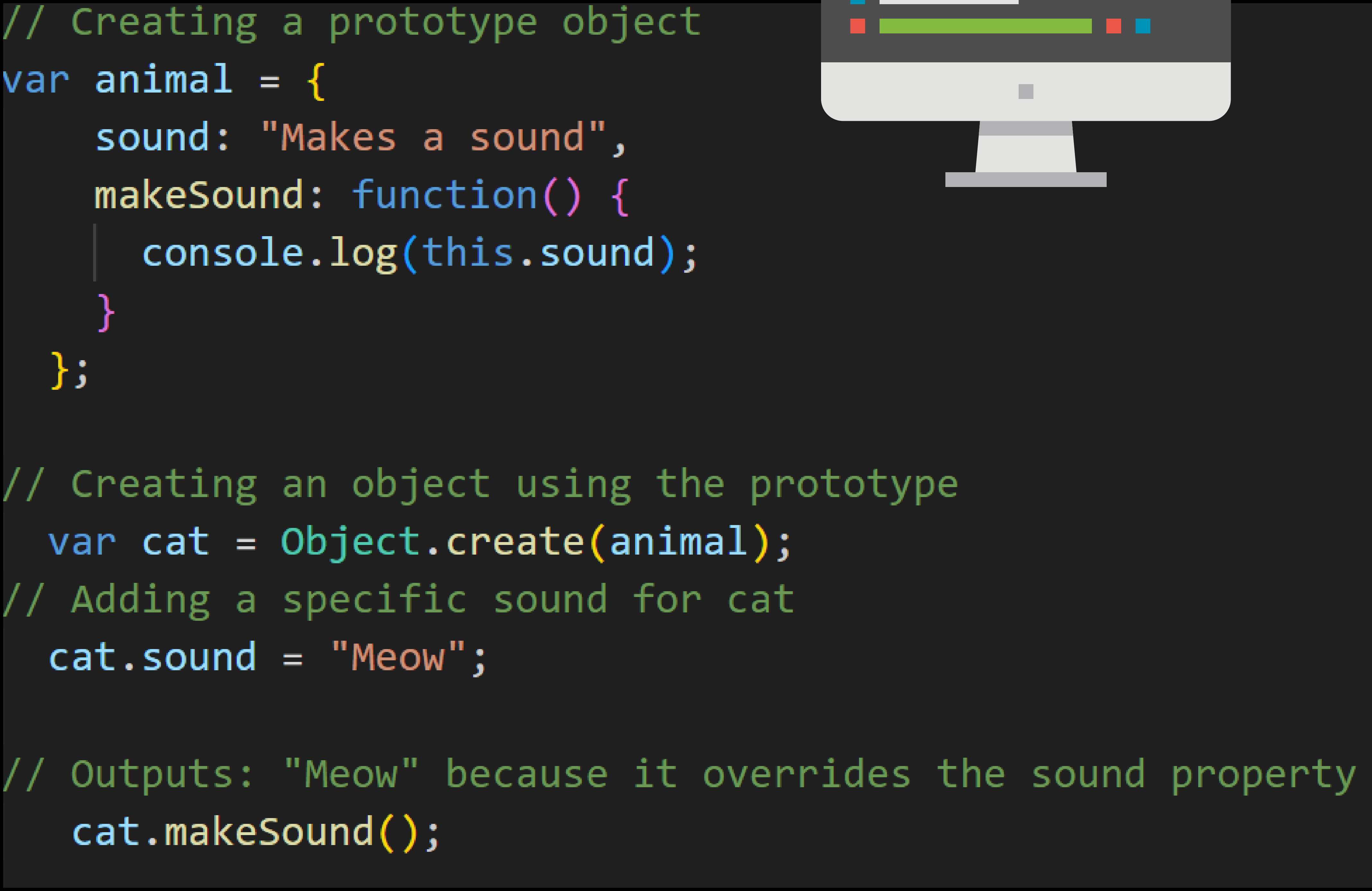
In JavaScript, the term "prototype" refers to the mechanism by which objects inherit properties and methods from other objects.

Each object in JavaScript has an associated prototype, and the prototype is effectively a template from which the object inherits.

Prototype Chain:

When you access a property or method on an object, JavaScript looks for that property or method on the object itself. If it doesn't find it, it looks at the object's prototype, and if it's still not found, it looks at the prototype of the prototype, and so on. This chain of prototypes is known as the "prototype chain."

-Radhika Sahebrao sathe



```javascript
// Creating a prototype object
var animal = {
    sound: "Makes a sound",
    makeSound: function() {
      console.log(this.sound);
    }
  };

// Creating an object using the prototype
  var cat = Object.create(animal);
// Adding a specific sound for cat
  cat.sound = "Meow";

// Outputs: "Meow" because it overrides the sound property
   cat.makeSound();
```

-Radhika Sahebrao sathe

Why Prototypes are important ?

Efficient Memory Use:

Prototypes enable shared methods and properties among objects, reducing memory overhead by not duplicating these elements for each instance.

Dynamic Updates:

Modifications to prototypes reflect instantly across all linked objects, allowing for easy updates and changes in behavior or structure.

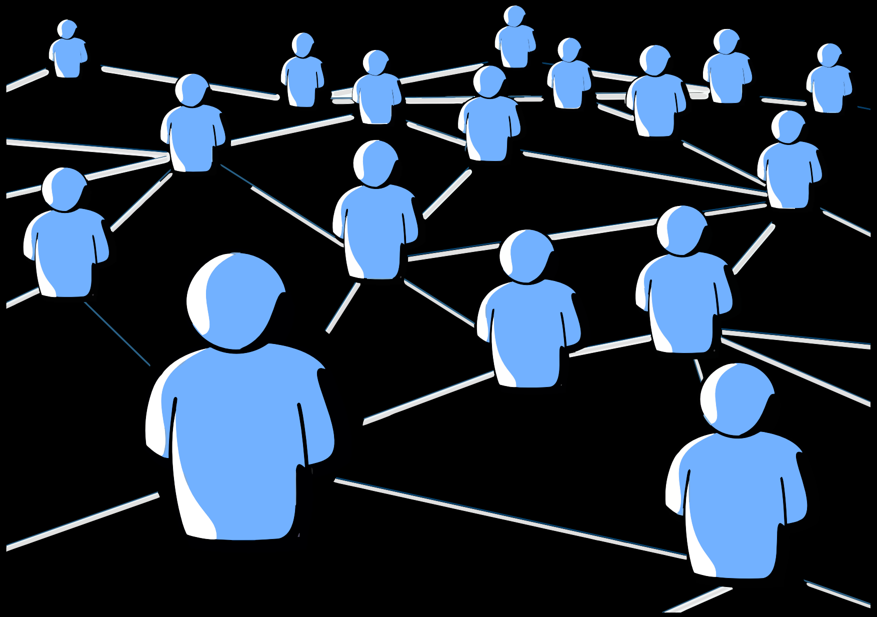
Inheritance and Reusability:

Facilitates a simpler way to implement inheritance, allowing objects to inherit properties and methods from prototypes, promoting code reusability and organization.

-Radhika Sahebrao sathe

In simple terms, Prototypes in JavaScript Shared templates that let objects inherit properties and behaviors.

-Radhika Sahebrao sathe



Radhika Sahebrao Sathe

Don't forget to connect