

Name: Basmala Mohamed Zain El-Abidine Ibrahim

Track: Open Source – Alexandria

Intake: 44

Course: Javascript

Inheritance in function constructor

In JavaScript, inheritance can be achieved using function constructors and prototypes. The concept of inheritance allows one object to inherit properties and methods from another object, creating a hierarchical relationship between them.

```
56 // Parent constructor function
57 function Animal(name) {
58     this.name = name;
59 }
60 // Adding a method to the prototype of the Animal constructor
61 Animal.prototype.sayHello = function () {
62     console.log("Hello, I'm " + this.name);
63 };
64 // Child constructor function
65 function Dog(name, breed) {
66     // Call the parent constructor using the 'call' method
67     Animal.call(this, name);
68     // Additional property specific to Dog
69     this.breed = breed;
70 }
71 // Inherit from the Animal prototype
72 Dog.prototype = Object.create(Animal.prototype);
73 // Set the constructor property back to Dog (since it was overwritten by Object.create)
74 Dog.prototype.constructor = Dog;
75
76 // Adding a method specific to Dog
77 Dog.prototype.bark = function () {
78     console.log("Woof!");
79 };
80 // Create instances of Animal and Dog
81 var genericAnimal = new Animal("Generic Animal");
82 var myDog = new Dog("Buddy", "Golden Retriever");
83
84 // Test the inherited methods
85 genericAnimal.sayHello(); // Output: Hello, I'm Generic Animal
86 myDog.sayHello();         // Output: Hello, I'm Buddy
87 // Test the Dog-specific method
88 myDog.bark();             // Output: Woof!
89
```