Inheritance

in JavaScript

What is Inheritance?

- Inheritance allows one object to access the properties and methods of another.

- Just like kids inherit traits from their parents.

- JavaScript supports prototypal inheritance, and with ES6, also class-based inheritance for easier readability.

Prototypal Inheritance (Old School)

```javascript
const animal = {
  speak() {
    console.log("Animal speaks");
  }
};


const dog = Object.create(animal);
dog.speak(); // Output: Animal speaks
```

- Here, dog inherits from animal using Object.create.

ES6 Class Inheritance (Modern Way)

```javascript
class Animal {
  speak() {
    console.log("Animal makes a sound");
  }
}

class Dog extends Animal {
  bark() {
    console.log("Dog barks");
  }
}

const tommy = new Dog();
tommy.speak(); // Animal makes a sound
tommy.bark();  // Dog barks
```

Why Use Inheritance?

1 Avoid code duplication

2 Reuse and extend functionality

3 Structure large codebases better

4 Build scalable object hierarchies

Keep Exploring Javascript with us!

Share this with a friend who needs it and make sure to practice these in scribbler.

Scribbler.live

Free and Open Interface to experiment JavaScript