


Class

JAVASCRIPT



JS

Hey Everyone 🖐️

JavaScript is a **versatile language** that **powers** many of the **websites** and **web applications** we use every day.

One of the **most important features** of JavaScript is its **ability to create** and use **classes**, which are a fundamental building block of **object-oriented programming**.

In this Post, we will explain **what classes** are in JavaScript, **why** they are **useful**, and **how to create** and use them in your code.

JavaScript Class

In JavaScript, a class is a **blueprint** for creating **objects** with specific **properties** and **methods**.

- It provides a way to **define the structure** and behavior of an object, which can then be **used to create multiple instances** of that object.
- The JavaScript **class contains** various class members **within a body** including methods or **constructor**.
- There is **two steps** to use Class:
 - Create A Class
 - Create individual objects

Create A Class

In JavaScript, we **create a class** using the **class** keyword, followed by the name of the class.

Here's an **example** of how we would create a class **for a Car**:

```
class Car {  
  constructor(make, model, year) {  
    this.make = make;  
    this.model = model;  
    this.year = year;  
  }  
  
  startEngine() {  
    console.log(`${this.make} Engine Starting...`);  
  }  
  
  accelerate() {  
    console.log(`${this.make} Accelerating...`);  
  }  
  
  brake() {  
    console.log(`${this.make} Braking...`);  
  }  
}
```


Constructor Method

A constructor is a **special method** that gets called when an **object is created** from a class.

It allows us to **set the initial values** of the object's properties.

- In previous example, we have **defined a class** called **Car** with a **constructor** method that takes in the **make, model, and year** of the car and **assigns them** to the corresponding **properties** of the object.
- We have also defined **three methods** for the car object: "startEngine", "accelerate", and "brake".

Using Classes In JS

Once we have defined our class, we can use it to create individual objects.

Here's an example of how we would create a new car object using the `Car` class:

```
let myCar = new Car("Toyota", "Corolla", 2022);
```

We have created a new `Car` object called `myCar` using the `new` keyword and passing in the `make`, `model`, and `year` values.

We can then call the methods of the object using the dot notation.

```
myCar.startEngine(); // logs "Toyota Engine Starting..."  
myCar.accelerate(); // logs "Toyota Accelerating..."  
myCar.brake(); // logs "Toyota Braking..."
```


Why Use Classes?

Classes are useful because they **allow us to create objects with similar properties and behaviors** in a more efficient and organized way.

Instead of defining each object individually, we can create a class that **encapsulates the common properties** and behaviors of those objects.

This also makes our code **easier to maintain and update**.

If we need to make a change to the behavior of all car objects, for **example**, we can simply **update the class** instead of having to modify each individual object.

I Have Already Posted On **Important CSS And Javascript Topics, Roadmap, Useful Resources**, And Cheat Sheets.

If You Have Any **Queries** Then Let Me Know In The **Comment Box**.