Object-Oriented Programming in JavaScript

Object-Oriented Programming is a programming style based on classes and objects. These group data (properties) and methods (actions) inside a box.

OOP was developed to make code more flexible and easier to maintain.

JavaScript is prototype-based procedural language, which means it supports both functional and object-oriented programming.

What are Classes and Objects in JavaScript?

What is a Class?

You can think of a class like a blueprint of a house. A class is not a real world object but we can create objects from a class. It is like an template for an object.

We can create classes using the class keyword which is reserved keyword in JavaScript. Classes can have their own properties and methods. We will study how to create a class in detail shortly. This is just a high level overview of a class.

What is an Object?

An object is an instance of a class. Now with the help of the house class we can construct a house. We can construct multiple houses with the help of same house class.

How Do We Actually Design a Class?

There is no perfect answer to this question. But we can get help from some OOP principles when designing our classes.

There are 4 main principles in OOP, and they are:

- Abstraction

- Encapsulation
- Inheritance
- Polymorphism

We will dive deep into these concepts in JavaScript below. But first, let's get a high level overview of these concepts to understand them better.

What Does Abstraction Mean in OOP?

Abstraction means hiding certain details that don't matter to the user and only showing essential features or functions.

For example, take a cell phone. We don't show details like verifyTemperature(), verifyVolt(), frontCamOn(), frontCamOff() and so on. Instead we provide essential features which matter to user like camera(), volumeBtn(), and others.

What Does Encapsulation Mean in OOP?

Encapsulation means keeping properties and methods private inside a class, so that they are not accessible from outside that class.

This will keep code that's outside the class from accidentally manipulating internal methods and properties.

What Does Inheritance Mean in OOP?

Inheritance makes all properties and methods available to a child class. This allows us to reuse common logic and to model real-world relationships. We will discuss inheritance in further section of this article with pratical example.

What Does Polymorphism Mean in OOP?

Polymorphism means having different and many forms. We can overwrite a method inherited from a parent class.