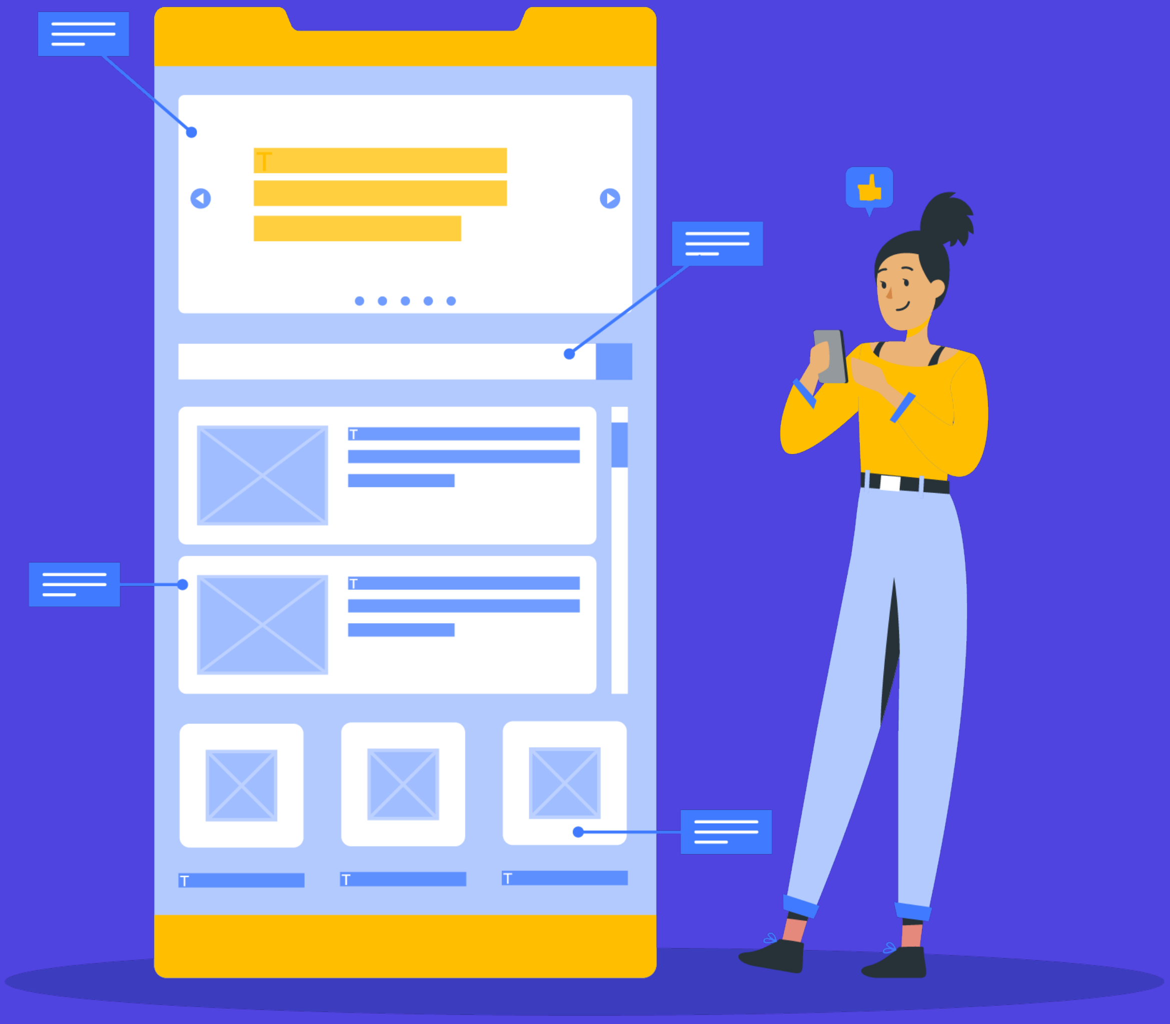
# OOPs: Intro to OOPs in JS

## Pre Read



Relevel

by Unacademy

Things you need to know before we begin this session

This session requires you to have basic knowledge of variables in Javascript. Make sure you have revised the previous concepts.

## Object:

In OOP, objects are instances of a class. When a class is defined, we instantiate the class by creating an object of that class. Objects are individual entities that have their properties and types.

In Js, objects are collections of unordered properties where each property is a key-value pair.

**For eg :**

```
let  emptyObject ={}  ---------> Creates a empty object

let  person =  {
                        firstName : 'ABC' ,
                        lastName :  'XYZ'
                          }
```

## Basic Terminology :

### 1. Declaring:

Creating a new variable is called a variable declaration. The keywords used to declare variables are let, var, const. When a variable is declared if no initial value is given, it will store undefined until a value is assigned.

**Eg :**

```
let a=10;
var b;
const c=20;
```

### 2. Instantiation :

Creating an instance or object of a class is called instantiation

**Eg :**

```
class Sum {
  let a =10;
  let b=20;
        }

let  obj1 = new Sum(); // Instating the class by
creating class object
```

## Classes :

Classes act as templates to create new objects.  Classes are just normal JavaScript functions and can be used without using the class syntax. It is a special syntactic sugar added in ES6 to make it easier to declare and inherit complex objects.

Classes have variables to represent properties of those real-world entities and methods to represent behavior.

Let's consider 'Bird ' as an example of a real-world entity. So, we can represent this Bird using a class. A bird can fly and can walk as well. So these two are methods.

```
class Bird(){
 constructor(){
 // Write something here
}

walk(){
 // Write something here
}


fly(){
 // Write something here
}
}
```

So here Bird is a class that has two methods walk() and fly(); To
access these methods; an object has to be defined;

let birdObj = new Bird(); // Instantiating class i.e creating object of class

```
// Calling the Bird class methods from Bird object
birdObj.walk();
birdObj.fly();
```

## New Keyword:

new is used to create an instance of an object.
It creates a new empty object and then binds the object with the constructor function or Object reference.

**Eg:**

const myLaptop = new Laptop();

this Keyword:

this keyword in Javascript refers to the object of the current scope or context in the code.

**Eg:**

```
function Laptop() {
    this.make = 'Dell';
    this.model = 'Inspiron 1590';
    this.cores = '8';
}
```