

OBJECT ORIENTED PROGRAMMING



WHAT IS OOP?

According to the Merriam-Webster Dictionary, Object Oriented Programming (OOP) is a type of computer programming in which programs are composed of objects which communicate with each other, which may be arranged into hierarchies, and can be combined to form additional objects.

CLASS

- A class is a blueprint or prototype from which objects are created. This section defines a class that models the state and behavior of a real-world object.
- An Object or a Class has Function (method) and States (attribute or data field).
- There is a method called Constructor which the name of the constructor have to be the same as the name of the Class.



CONCEPTS

INHERITANCE

Helps to reuse the code and establish a relationship between different classes. Inheritance allows Child Class to inherit the properties of Parent Class. If the child class wants to inherit the properties from the Parent Class, after name of the Class of Child Class, add "extends <ParentClassName>" to inherit from the Parent Class.

ENCAPSULATION

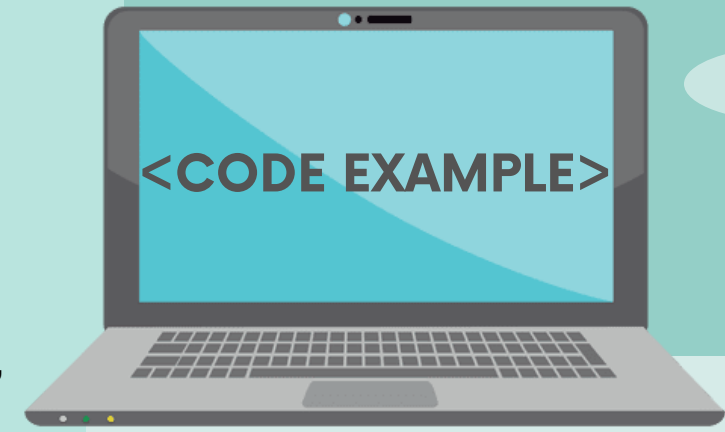
A mechanism which the data and code is binded together as a single unit to make it safe from any modification. It helps us in reusing the code. To achieve encapsulation, we need to declare the variable of a class as private, and provide the public setter (mutator) and getter (accessor) methods to modify and view the variable values.

ABSTRACTION

It is being used to represent complexity from using simple things like objects, classes, and variables. To achieve abstraction, we have to use Abstract Class and Interface.

POLYMORPHISM

According to the dictionary, polymorphism is "the condition of occurring in several different forms". In object oriented programming, polymorphism is when an object takes many forms through a common interface. Upcasting & downcasting are the examples of the implementation.



```
import java.util.*;
public class Laptop{
    //attributes
    private String brand;
    private double price;
    private String os;
    //constructor
    public Laptop(String brand,
                  double price,
                  String os){
        this.brand = brand;
        this.price = price;
        this.os = os;
    }
    //mutator
    public void setPrice(double price){
        this.price = price;
    }
    //accessors
    public String getBrand(){
        return this.brand;
    }
    public double getPrice(){
        return this.price;
    }
    public String getOS(){
        return this.os;
    }
    //string representation
    public String toString(){
        return String.format(
            "Brand: %s, OS: %s, Price: %.2f",
            this.brand, this.os, this.price);
    }
}
```



ADVANTAGES

- Programming gets quite easier, more flexible, and faster
- Easier to test and fix errors
- The objects that are created can be reused by other programs
- Clear modular structures
- Easy to understand

FUN FACT!

99% of major device platforms like Apple, Android, Windows, and Blackberry use object oriented languages for mobile development.



Instructor:

Dr. Fariz Darari

Teaching Assistant:

Samuel Dimas Partogi

SOURCES

<https://docs.oracle.com/javase/tutorial/java/concepts>
<https://stackify.com/oops-concepts-in-java>
<https://edureka.co/blog/object-oriented-programming>
<https://www.petanikode.com/java-oop>

