#web developer tips

OOPS CONCEPT YOU MUST KNOW

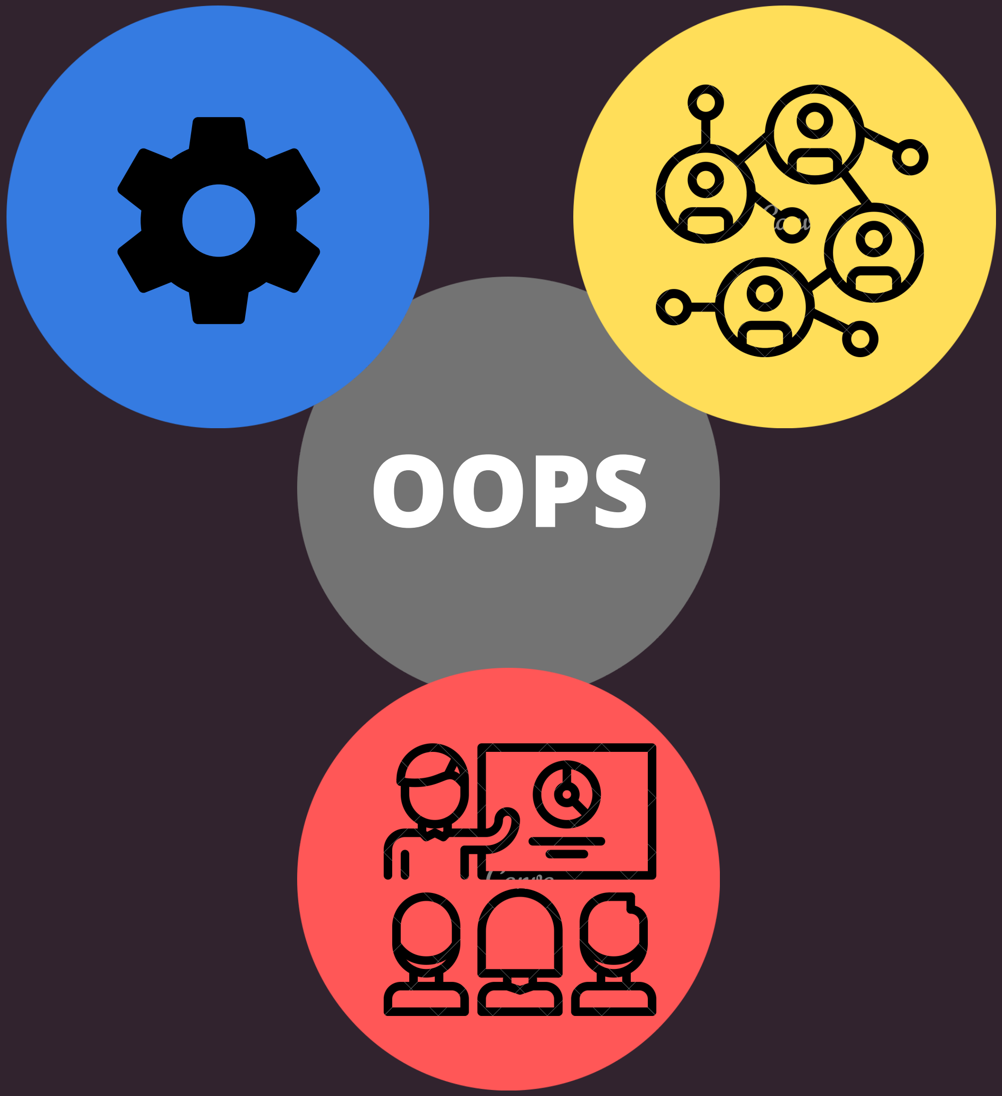
Hans.Dev

What is object-oriented programming (OOP)?

Object-oriented programming (OOP) is a programming paradigm based on the concept of "objects", which may contain data, in the form of fields, often known as attributes; and code, in the form of procedures, often known as methods. For example, a person is an object which has certain properties such as height, gender, age, etc. It also has certain methods such as move, talk, and so on.

Main Concepts of Object Oriented Programming OOPS .

- > Class
- > Object
- > Polymorphism
- > Encapsulation
- > Inheritance
- > Abstraction



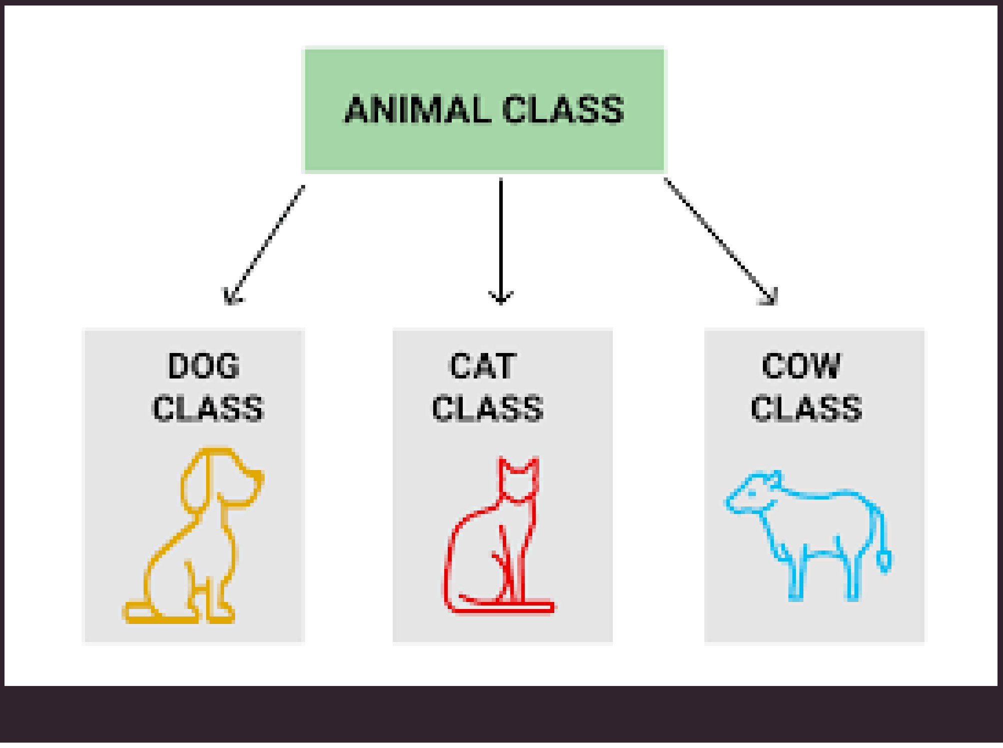
. Class

A class is a data type that has its members i.e. data members and member functions. It is the blueprint for an object in an object-oriented programming language. It is the basic building block of object-oriented programming. The members of a class are accessed in a programming language by creating an instance of the class.

Some important properties of the class are –

- Class is a user-defined data type.
- A class contains members like data members and member functions.
- Data members are variables of the class.
- Member functions are the methods that are used to manipulate data members.
- Data members define the properties of the class whereas the member functions define the behavior of the class.

. Example



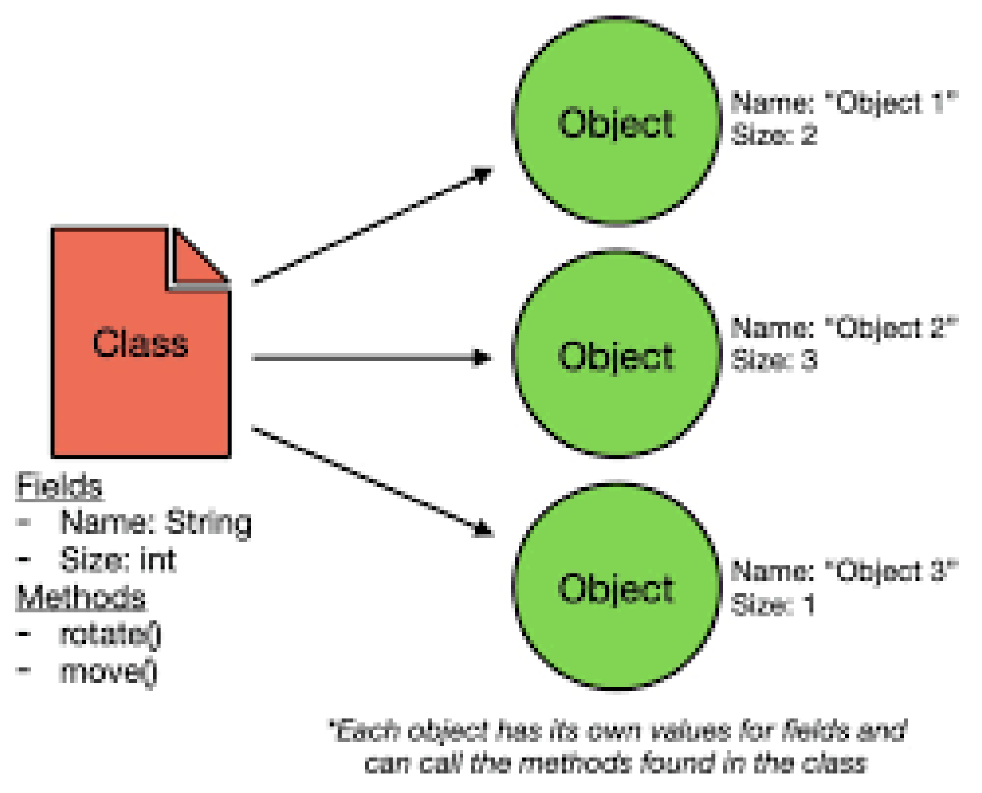
. Object

An object is an instance of a class. It is an entity with characteristics and behaviour that are used in the object oriented programming.

An object is the entity that is created to allocate memory.

A class when defined does not have memory chunk itself which will be allocated as soon as objects are created.

. Example

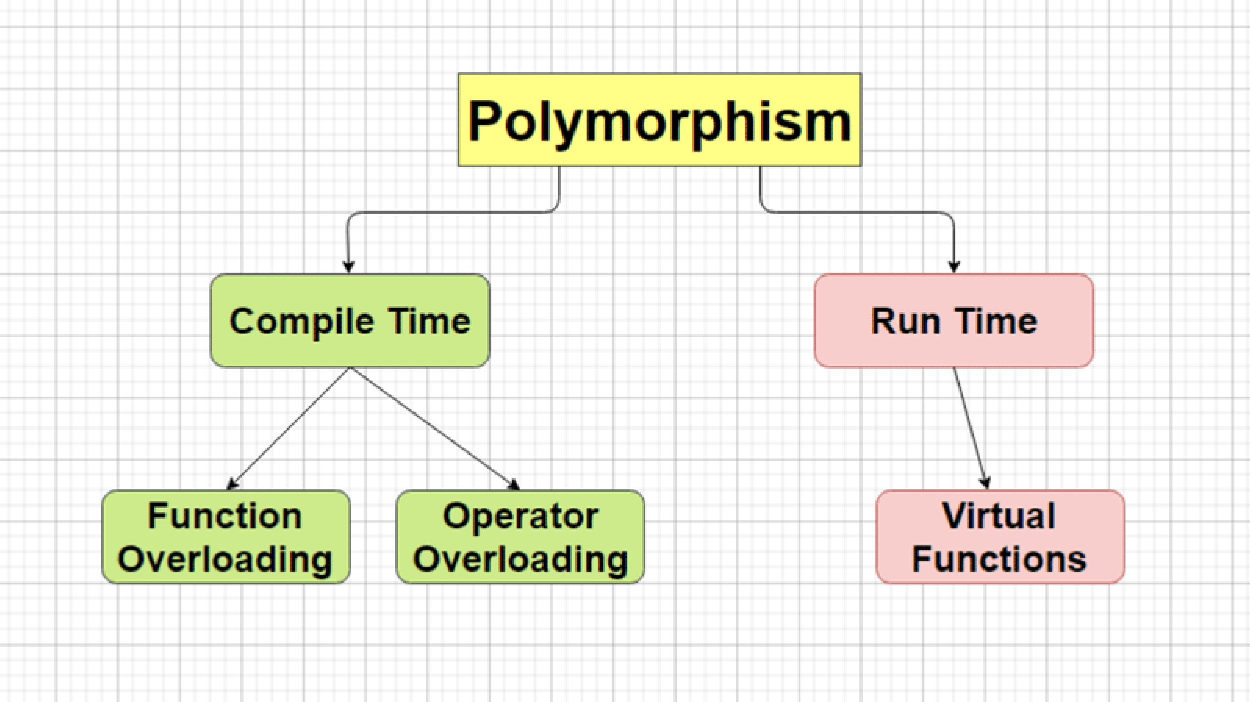


Each object has its own values for fields and can call the methods found in the class

. Polymorphism

Polymorphism is the ability of an object to perform different actions (or, exhibit different behaviors) based on the context.

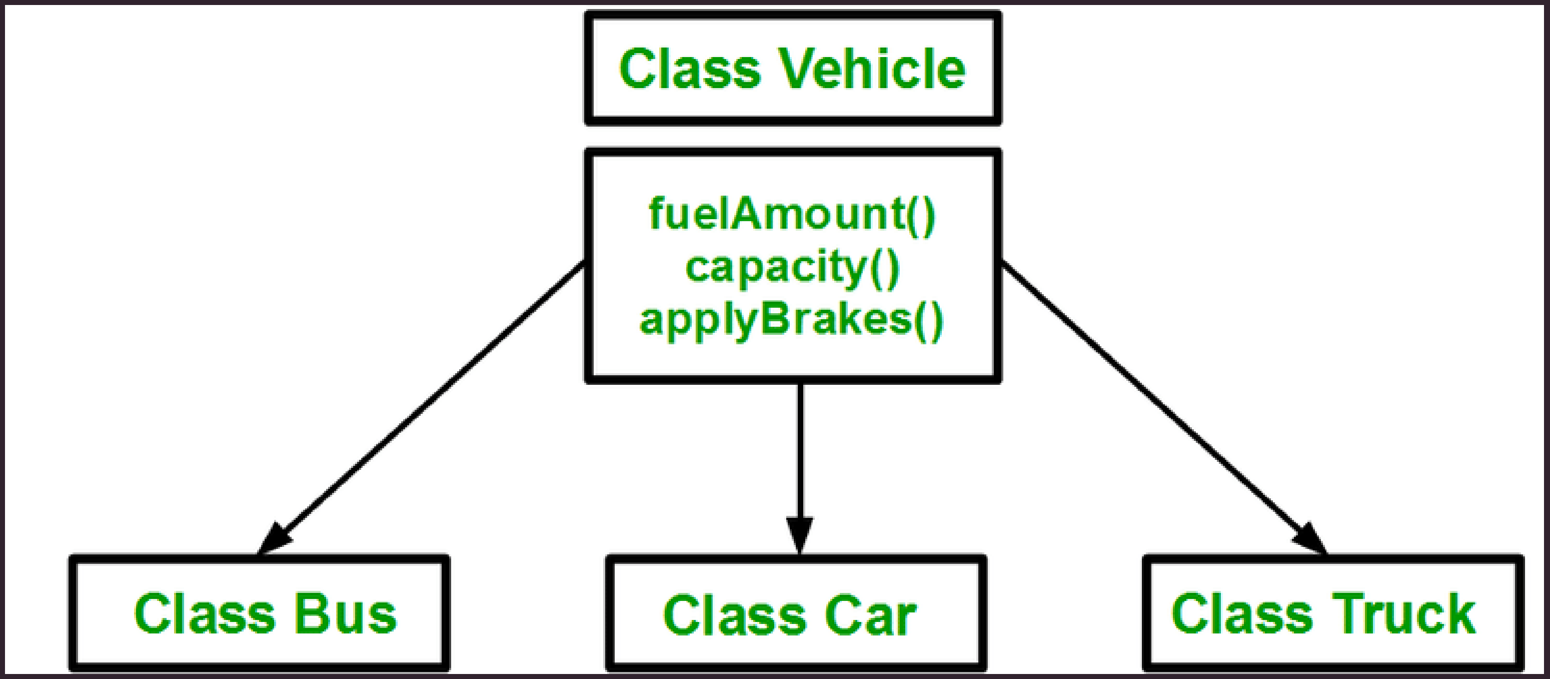
. Example



. Inheritance

Inheritance can be defined as the process where one (parent/super) class acquires the properties (methods and fields) of another (child/sub). With the use of inheritance, the information is made manageable in a hierarchical order.

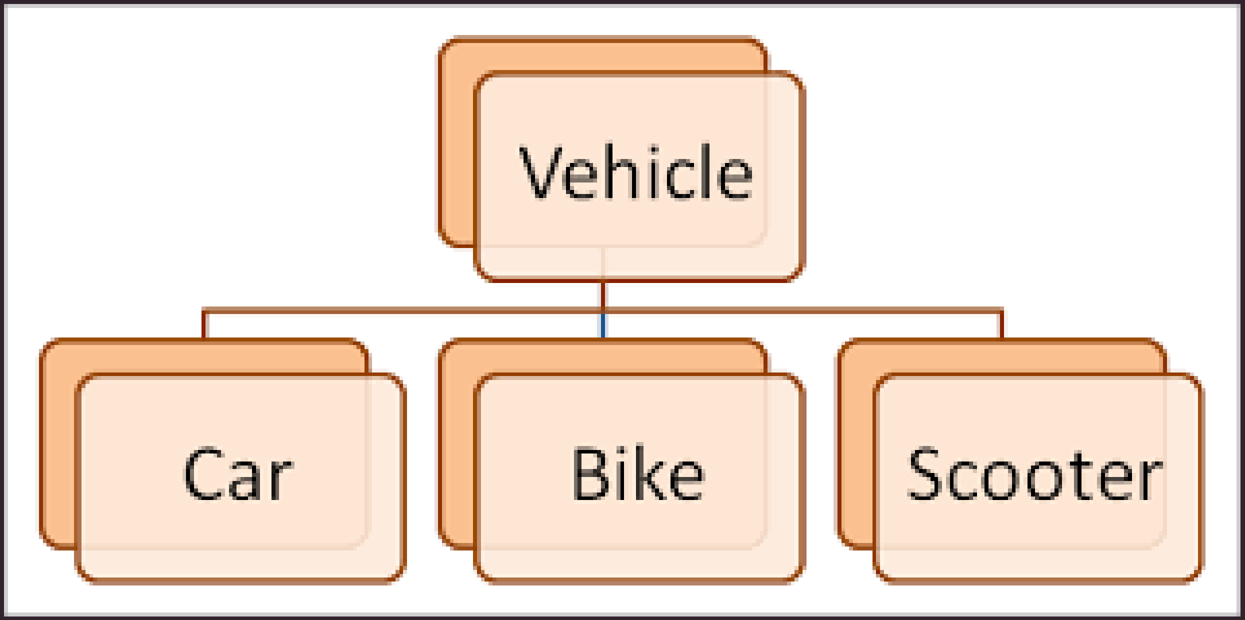
. Example



. Abstraction

Abstraction is a process of hiding the implementation details from the user, only the functionality will be provided to the user. In other words, the user will have the information on what the object does instead of how it does it.

. Example



. Encapsulation

Encapsulation in Java is a mechanism for wrapping the data (variables) and code acting on th data (methods) together as a single unit. In encapsulation, the variables of a class will be hidden from other classes and can be accessed only through the methods of their current class. Therefore, it is also known as data hiding. To achieve encapsulation in Java –

1. Declare the variables of a class as private.
2. Provide public setter and getter methods to modify and view the variables values.

. Example

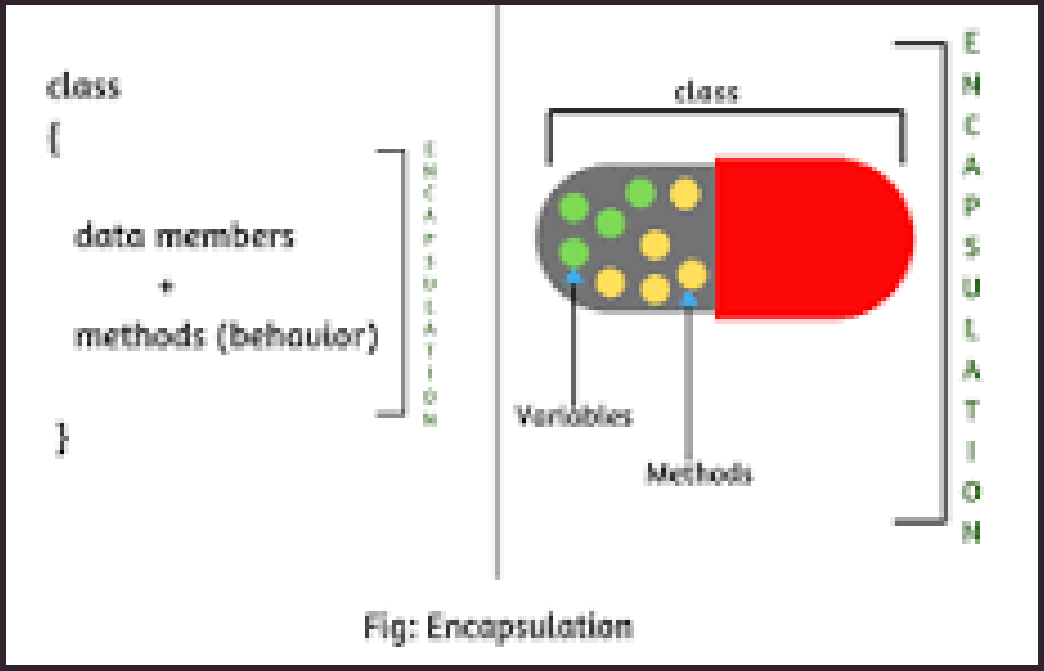


Fig: Encapsulation

Don't forget to Like & Save it for future

Hans raj

hans_dev_1350

FOLLOW FOR MORE