

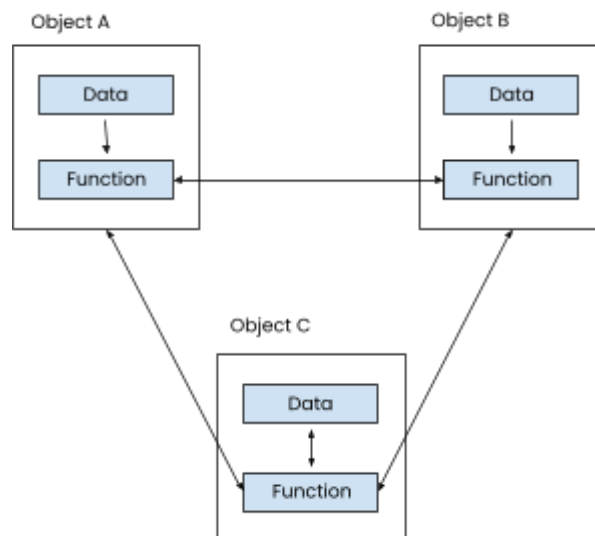
## Unit-5

### Object-Oriented Programming

#### Object-oriented Programming(OOP):

Object-Oriented Programming is a programming approach or methodology that uses the concept of object and class to design computer programs. It is a collection of cooperative objects where each object is capable of receiving, processing, and sending the data to other objects.

The fundamental idea behind OOP is to combine data and functions into a single unit (i.e., object). It is the most popular programming model among developers. It is also known as event-driven programming.

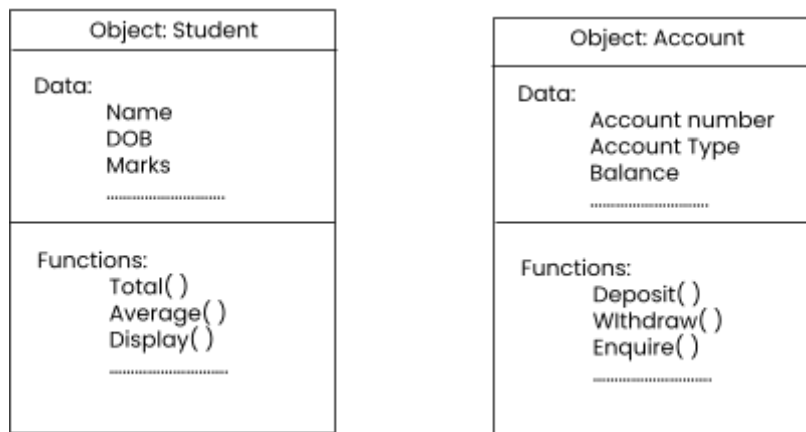


#### Features of OOP:

The main features of Object-oriented programming are as follows:

##### 1. Object:

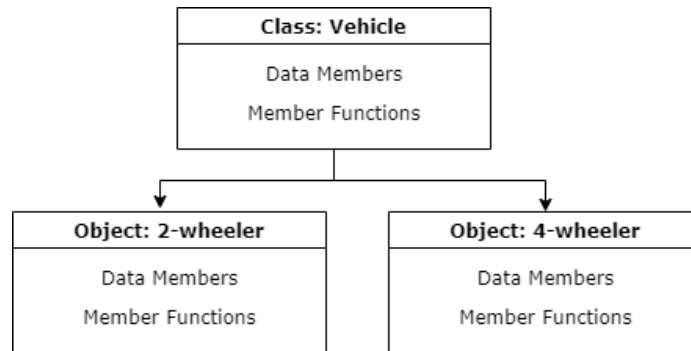
An object is a real-world entity that has some attributes, behaviors, and properties. It is also referred to as an instance of the class. It contains data and member functions to perform the operations upon data. It occupies certain memory of a computer. Each object is capable of receiving messages, processing data and sending messages to other objects.



2. **Class:**

A class is a collection or a group of similar objects that have the same properties and common behavior. It is also called a blueprint of an object. It binds data and functions together into a single unit which allows the data to be hidden from external use. It is a user-defined data type in OOP.

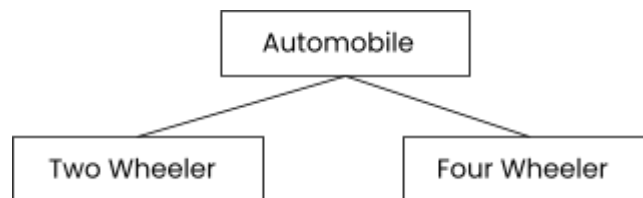
Examples: Class of a person, Class of animals, Class of vehicles.



3. **Inheritance:**

It is a process of creating a new class from the existing class. The newly formed class is called a derived class or subclass whereas the class from which a class is derived is called a base class. It allows an object of one class to acquire the characteristics of objects of another class which helps in code reusability.

For example, car is a part of class Four wheeler and also a part of automobile. It has all the features of four wheelers and also have some features of automobile.



4. **Encapsulation:**

The process of wrapping data and functions in a single unit called class is known as encapsulation. It is one of the most important features of OOP which ensures data security and makes data safe from accidental alterations as well as outside interference.

5. **Polymorphism:**

The term polymorphism means "having multiple forms". It is a concept in OOP that allows us to create methods/functions with the same name but with different method signatures. It enables the same function to behave differently in different classes.

For example, The operator symbol '+' is used for arithmetic operations between two numbers, however, by overloading the same operator '+' can also be used for the concatenation of strings.

6. **Data Abstraction:**

It is a concept in OOP that hides unnecessary information or implementation details from the user only by showing essential information. It helps in reducing programming complexity.

**Advantages of OOP:**

Object-oriented programming contributes to greater programmer productivity, better quality of software, and lesser maintenance cost. The main advantages are:

- Program complexity can be easily managed.
- Making the use of inheritance, the existing class can be extended.
- Beneficial to collaborative development in which a large project is divided into groups.
- Through data hiding, programmers can build secure programs.
- It is possible to have multiple instances of an object to co-exist without any interference.
- System can be easily upgraded from small to large systems.
- Code reusability is much easier than in conventional programming languages.

**Disadvantages of OOP:**

- Not suitable for small problems.
- Takes more time to solve problems.
- Object-oriented programs have larger program sizes.
- Object-oriented programs require more instructions to be executed so are slower.
- Requires more planning before developing programs using OOP.

**Applications of OOP:**

- Real time systems
- Designing user interface for applications
- Simulation & modeling
- Object-Oriented databases
- AI & expert system
- Neural Networks & parallel programming
- Decision support & Office automation system

**Differences between OOP and Structured Programming Language:**

The differences between Object-oriented programming and structured programming language or Procedural-oriented programming (POP) are as follows:

<b>Object-Oriented Programming (OOP)</b>	<b>Structured Programming Language</b>
OOP is a programming paradigm that uses the concept of object and classes.	It is a collection of instructions that are executed by the computer sequentially.
It uses a bottom-up approach.	It uses a top-down approach.
Programs are divided into a number of entities called objects.	Programs are divided into a number of functions.

Emphasis is on data rather than on procedures.	Emphasis is on procedures rather than data.
The concept of encapsulation provides a high level of security.	It doesn't support encapsulation.
Easy to reuse existing program codes because of the inheritance concept.	No proper mechanism for reusing program codes.
New data and functions can be easily added whenever required.	Adding new data and functions is difficult and time-consuming.
It does not allow data to move freely around the program from one object to another.	It allows data to move freely around the program from one function to another.
Examples of OOP languages are C++, JAVA, C#, etc.	Examples of Structured programming languages are C, FORTRAN, COBOL, etc.