01 - OBJECT ORIENTED PROGRAMMING

1.1 Basics of Object Oriented Programming

1.1.1 What is an Object Oriented Programming?

It is the programming paradigm that is defined using objects. It also refers to as OOPs.

1.1.2 What is an Object?

An entity that has a state and behavior is called an Object. Here the state represents properties and behavior represents actions and functionality. Ex: person, chair, pen, keyboard, bike, etc.

The state and properties of a person are hair color, eyes color, skin color, height, weight, etc. The actions are eat, sleep, drink, walk, play, etc.

1.1.3 What is a Class?

It is a user-defined data type or a template of an object. It represents the common properties of an object.
Ex: bus, lorry, car, etc., belong to Vehicle class.
chair, table, cot, etc., belong to Furniture class.

Object takes space in memory but Class doesn't take space in memory. Object exists physically and Class doesn't exist physically.

1.2 Principles / Features of Object Oriented Programming

1.2.1 What is an Abstraction?

It is a process of hiding unnecessary data from the user and exposing only essential data required by the user. It permits to use an object without knowing its internal details.

For example, we know TV displays some image with sound. The user knows to operate the TV.
But he doesn’t know the internal function of TV.

1.2.2 What is an Encapsulation?

The process of data (variables) and corresponding functions (methods) together into a single unit (class) is called encapsulation. Data security in OOPs is achieved using encapsulation.

Encapsulation hides and protects the data from the outside non-member functions of a class.
Only the functions defined in a class will have access to data.

1.2.3 What is an Inheritance?

An ability to acquire the properties of one class to another class is called inheritance. Data and functions of one class can be used by another class using inheritance. It makes code reusable. New features can be added easily.

Ex: Polygon represents several shapes. We can make two distinct sorts of polygons: Rectangle and Triangle.

There are 5 types of Inheritance:

a. Single Inheritance
b. Multiple Inheritance
c. Multilevel Inheritance

d. Hierarchical Inheritance
e. Hybrid Inheritance

1.2.4 What is Polymorphism?

An ability to process any data in more than one form is called Polymorphism. The most common use of Polymorphism in OOPs occurs when a parent class reference is used to refer to a child class object.

Some real life examples of polymorphism are, a person at the same time can have different roles to play in life. A woman at the same time is a mother, wife, an employee, and a daughter. So the same person can have many features but has to implement as per the situation and condition. It is an important feature of OOPs.

1.3 What are Advantages of OOPs?

a. **Security:** Achieved by Encapsulation
b. **Reusability:** Achieved by Inheritance
c. **Effective Communication:** Achieved by message passing
d. **Developing Complex Software:** Complexity is decreased by inheritance
e. **Easily Upgraded:** Easy to upgrade from small to large systems
f. **Easy Partition of Work:** Complicated work can be easily partitioned
g. **Maintenance:** Object oriented code can be easily maintained
h. **Efficiency:** Provides easy and efficient development process