

Q) What is Object-Oriented Programming (OOP)?

A) Object Oriented Programming (OOP) is a programming methodology that models real-world entities using classes and objects. It provides a structured way to organize code, making it reusable, maintainable, and easy to extend. OOP relies on key principles:

Encapsulation:- It binds data (variables) and behaviors (methods) into a class and restricts direct access, ensuring data integrity.

Inheritance:- Enables a new class to inherit properties and methods from an existing class, promoting code reuse.

Polymorphism:- Allows methods to take different forms - method ~~overloading~~ overloading (compile-time) and method overriding (runtime) - enhancing flexibility.

Abstraction:- Focuses on showing only essential details while hiding complex implementation.

By following these principles, OOP enhances software design, supports real-world modeling, and reduces code redundancy, making it a preferred approach for large projects.

2) What are classes and objects in OOP?

A) In OOP, a class is a template that defines the structure and behavior of an object. It outlines the attributes (data members) and methods (functions) that the object will have. It acts like a blueprint - for example, a 'car' class may define properties like 'brand' and 'speed' and behaviors like 'drive'.

An object is an individual instance of a class, holding specific values for the class's attributes and using its methods.

3) What is polymorphism?

A) Polymorphism means "many forms" and allows objects to take different forms depending on the context. It enhances flexibility and reusability by letting methods behave differently based on parameters or object types.

Types of polymorphism:-

Compile-Time Polymorphism (method overloading) :- Same method name, different parameters.

Run-Time Polymorphism (method overriding) :- Child class overrides parent's method.

4) What is encapsulation?

A) Encapsulation is the practice of bundling data (variables) and behavior (methods) into a single unit (class) while restricting direct access to the data. It is achieved using access specifiers.

Private:- Data is accessible only within the class.

Public:- Data is accessible from anywhere.

Protected:- Data is accessible within the class and subclasses.

Default (Package-private):- Data is accessible within the same package.

5) What are constructors in OOP?

A) A constructor is a special method used to initialize objects when they are created. It has the same name as the class and does not have a return type.

Types of constructors:

Default constructor:- No parameters, assigns default values.

Parameterized constructor:- Accepts arguments to set specific values.

Copy constructor:- Creates a new object as a copy of an existing object.