6) Object-Oriented Programming (OOPs) Concepts

1) Basics of OOP: Encapsulation, Inheritance, Polymorphism, Abstraction

Answer: Object-Oriented Programming (OOP) is a programming paradigm based on the concept of "objects," which encapsulate data and behavior. The four core principles of OOP are **Encapsulation**, **Inheritance**, **Polymorphism**, and **Abstraction**.

Encapsulation: wrapping up of data into single unit i.e. : data hiding

: Private your data member and member function

Inheritance: properties of parent class extends into child class
: Properties of super class extends into subclass
: Main purpose is : Reusability , extendsibility
: To used "extends" keyword through create inheritance
: Always called last child class to create object with access the properties of parent class except private

: There are mainly 5 types
1) Single: only one parent having only one child
2) Multilevel: single inheritance having one another child
3) Hierarchical: one parent having 2 or more child
4) Multiple: java does not support directly
5) Hybrid: java does not support directly

Polymorphism: Ability to take one name having many forms or different forms
: There are mainly 2 types

1) Method overloading(compile time) : the two or more method name should be same in a single class but its behaviors(data types, arguments) are different is known as Method overloading.

2) method overriding(run time) : the whole signature of the method should be same in super class as well as in subclass but its behaviors (body part of the method) are different

Abstract: Only essential part should be display rest of the part will be hidden: data hiding
1) Using with class: we cannot create object of that class
: Must inherit into your child class

2) Using with method: do not specify body part of the method
: Your class must be also abstract
: Must override your abstract method into your child class

2) Inheritance: Single, Multilevel, Hierarchical

Answer:

Inheritance: properties of parent class extends into child class
: Properties of super class extends into subclass
: Main purpose is : Reusability , extendibility
: To used "extends" keyword through create inheritance
: Always called last child class to create object with access the properties of parent class except private

: There are mainly 5 types
1) Single: only one parent having only one child
2) Multilevel: single inheritance having one another child
3) Hierarchical: one parent having 2 or more child

3) Method Overriding and Dynamic Method Dispatch

Answer:

1. Method Overriding

Method overriding occurs when a **subclass** provides a specific implementation of a method that is already defined in its **super class**. The method in the subclass must have the **same name, return type, and parameters** as in the super class.

2. Dynamic Method Dispatch (Runtime Polymorphism)

Dynamic Method Dispatch is a process in which a call to an overridden method is resolved at **runtime** rather than compile time. It is achieved using **method overriding** and is the foundation of **runtime polymorphism**.

```
====================================================================
====================================================================
====================================================================
====================================================================
```