**Java OOPS Assignments by gaurav diwan**

**Java with DSA and System design**

Q1] what is inheritance in java?

Ans. Inheritance in Java allows a class (subclass or derived class) to inherit properties and behaviors from another class (superclass or base class).

Q2] what is subclass and superclass?

Ans. Subclass: A class that inherits from another class.

Superclass: The class that is inherited from by another class.

Q3] how is inheritance implement in java?

Ans. In Java, inheritance is implemented using the **extends** keyword.

Q4] what is polymorphism?

Ans. Polymorphism in Java allows objects to be treated as instances of their parent class, enabling multiple classes to be used interchangeably.

Q5] differentiate b/w method overloading and method overriding?

Ans. Method Overloading: Same method name but different parameters within the same class.

Method Overriding: Same method name and parameters in both the superclass and the subclass.

Q6] what is abstraction explain with example.

Ans. **Abstraction** in Java involves hiding complex implementation details and showing only the essential features of an object.

Example : abstract class Shape {

abstract void draw(); // Abstract method (no implementation details)

}

class Circle extends Shape {

void draw() {

System.out.println("Drawing a circle");

}

}

class Square extends Shape {

void draw() {

System.out.println("Drawing a square");

}

}

Q7] what is the difference between abstract method and final method in java? Explain with example.

Ans. Abstract Method:

- Declared in an abstract class.

- Has no implementation in the abstract class.

- Must be implemented by any concrete (non-abstract) subclass.

Final Method:

- Declared in a class.

- Cannot be overridden by any subclass.

Example:

abstract class Shape {

abstract void draw(); // Abstract method

}

class Circle extends Shape {

void draw() {

System.out.println("Drawing a circle");

}

}

class Rectangle {

final void draw() {

System.out.println("Drawing a rectangle");

}

}

Q8] what is the final class in java?

Ans. A final class in Java cannot be subclassed. It's declared using the `final` keyword.

Q9] differentiate between abstraction and encapsulation?

Ans. Abstraction hides complex implementation details, exposing only essential features.

Encapsulation bundles data and methods into a single unit, restricting direct access to the internal state.

Q10] difference between run time and compile time polymorphism with an example?

Ans. **Compile-time polymorphism** (method overloading) is resolved at compile time based on method signatures.

class Example {

void display(int x) {

// code

}

void display(int x, int y) {

// code

}

}

**Run-time polymorphism** (method overriding) is resolved at runtime based on the actual object type.

class Animal {

void sound() {

System.out.println("Animal makes a sound");

}

}

class Dog extends Animal {

void sound() {

System.out.println("Dog barks");

}

}

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***