UNIT 5: OOP IN JAVA

UNIT STRUCTURE

5.1 Learning Objectives

5.2 Introduction

5.3 Java Inheritance

5.4 Java Super-Class

5.5 Multilevel Inheritance

5.6 Final and Abstract Keyword

5.7 Static Members

5.8 Let Us Sum Up

5.9 Further Reading

5.10 Answer to Check Your Progress

5.11 Model Questions

5.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- describe about Java inheritance.
- explain the Java Super Class
- describe the multilevel Inheritance
- describe about the final and abstract keyword

5.2 INTRODUCTION

In the earlier unit, we have discussed about class fundamentals in Java. Topics covering accessing class members and declaring objects were discussed in the previous unit. The previous unit also included topics like concepts related to passing arguments to methods and returning multiple values from methods. Loop control statements with their types were also discussed in the previous unit.

In this unit, you will be introduced to OOP concepts in Java. Topics like inheritance and Java super class are discussed in detail in this unit. In addition to these, multilevel inheritance, final and abstract keywords in

addition to topics like static members are also described in this unit. In the next unit, we will explore the topics related to arrays, strings and vectors in Java.

5.3 JAVA INHERITANCE

Inheritance is the deriving of a new class from an old class. The old class that is inherited is known as the superclass or base class or parent class and the new class that does the inheriting is called the subclass or derived class or child class. In this mechanism, one Java object acquires all the properties of a parent object.

Types of inheritance are mentioned below.

There are different forms of inheritances:

- **Single inheritances:** Deriving a subclass from only one superclass is called single inheritance.
- **Hierarchical inheritance:** Deriving several subclasses from a single superclass is called hierarchical inheritance.
- **Multilevel inheritance:** Deriving a subclass from another derived subclass is called multilevel inheritance.
- **Hybrid inheritance:** Deriving a class using more than one form of inheritance is called hybrid inheritance.

The following is the procedure for defining a subclass

The syntax defining subclass is -

Syntax :

```
class subclass-name extends superclass-name
{
        variable declaration;
        method declaration;
}
```

The keyword extends denotes that the properties are extended to the subclass name from the superclass name, which means the subclass will now contain the variable and methods of the superclass along with its own.

For example,

```
class SIH extends AICTE_SIH
{
  char team_name[20];
  void display();
}
Another example of Java inheritance is :
Class KKHSOU{
void display()
{
    System.out.println("kkhsou_course");
}
}
class KKHSOU_1 extends KKHSOU{
void display1()
{
    System.out.println("CourseWork_2");}
}
class KKHSOU_2{
public static void main(String args[]){
KKHSOU_1 d=new KKHSOU_1();
d.display1();
d.display();
}}
```

5.4 JAVA SUPER CLASS

The base class from which a subclass is derived is the superclass.

Given an example, the Car class (subclass) inherits the variables and methods from the Vehicle class (superclass) :

```
class Vehicle
 {
    protected String brand = "Ford";   // Vehicle
```

```
attribute
public void honk()    // Vehicle method
 {
    System.out.println("Tuut, tuut!");
 }
}
class Car extends Vehicle
{
   private String modelName = "Mustang";
   // Car attribute
   public static void main(String[] args)
   {
      // Create a myCar object
      Car myCar = new Car();
     // Call the honk() method (from the
     //Vehicle class) on the myCar object
    myCar.honk();
// Display the value of the brand attribute (from
the Vehicle // class) and the value of the modelName
from the Car class
     System.out.println(myCar.brand + " "
     + myCar.modelName);
  }
}
```



CHECK YOUR PROGRESS

Q 1: What is Java inheritance?

.....

Q 2: What keyword is used to inherit a class?

.....

Q 3: A class member declared protected becomes a member of a subclass of which type?

.....

Q 4: Does class B extend A {} is the correct way of inheriting class A by class B?

.....

5.5 JAVA MULTILEVEL INHERITANCE

Deriving a subclass from another derived subclass is called multilevel inheritance. The use of a derived class as a superclass is a common concept in object-oriented programming like Java. A chain of classes is built using this concept.

A derived class with multilevel inheritance is declared as :

```
class A
{
        -----
        -----
}
class B extends A       //First level
{
        -----
        -----
}
class C extends B  // Second level
{
        ------
        ------
}
```

Example:

```
class Shape
{
   public void display()
 {
      System.out.println("Inside display");
   }
}
```

```
class Rectangle extends Shape
{
   public void area()
{
      System.out.println("Inside area");
   }
}
class Cube extends Rectangle
{
     public void volume()
     {
        System.out.println("Inside volume");
     }
}
public class Tester
{
   public static void main(String[] arguments)
   {
      Cube cube = new Cube();
      cube.display();
      cube.area();
      cube.volume();
   }
}
```

5.6 FINAL AND ABSTRACT CLASS

The Final keyword is declared to finalize variables and methods used in the class. A class with the Final keyword is called a Final Class. No other class can inherit a class with the Final keyword. Using the Final keyword means the prevention of inheriting any properties or methods of that class.

Example:

```
// Final class
```

```
final class Super
{
    private int data = 100;
}
public class Sub extends Super
 {
    public static void main(String args[])
    {
    }
}
```

The Abstract keyword is used for abstraction.A class with the Abstract keyword is called Abstract Class. An Abstract class consists of both Abstract and concrete methods.

Example:

```
// An abstract class
// Abstract parent class
abstract class Book
{
   // Abstract method without body
    public abstract void page();
}
  // shayar class extends Book class
public class shayar extends Book
{
  // Declaring the abstract method
    public void page()
    {
        System.out.println("Geek");
    }
   // Driver code
    public static void main(String args[])
    {
        Book obj = new shayar();
        obj.page();
    }
}
```

5.7 STATIC MEMBERS

Members of a class that can be accessed without creating an object are called static members. The Static keyword is used to create a static method. The Static keyword can be used in methods, classes, fields, and blocks of a Java program.

Example:

```
public class MyClass
 {
   public static void sample()
   {
      System.out.println("Hello");
   }
   public static void main(String args[])
   {
      MyClass.sample();
   }
}
```



CHECK YOUR PROGRESS

Q 5: What is multilevel inheritance?

.....

Q 6: What is a static member?

.....

Q 7: What is the abstract keyword?

.....



5.8 LET US SUM UP

- Inheritance is the deriving of a new class from an old class. The old class that is inherited is known as the superclass or base class or parent class and the new class that does the inheriting is called the subclass or derived class or child class.

- There are different forms of inheritances: Single inheritances: Deriving a subclass from only one superclass is called single inheritance. Hierarchical inheritance: Deriving several subclasses from a single superclass is called hierarchical inheritance. Multilevel inheritance: Deriving a subclass from another derived subclass is called multilevel inheritance. Hybrid inheritance: Deriving a class using more than one form of inheritance is called hybrid inheritance.
- The syntax defining subclass is -

  Syntax : class subclass-name extends superclass-name

  {

  variable declaration;

  method declaration;

  }
- The base class from which a subclass is derived is the superclass. Given an example, the Car class (subclass) inherits the variables and methods from the Vehicle class (superclass) :

```
class Vehicle
{
   protected String brand = "Ford";
   // Vehicle attribute
   public void honk()    // Vehicle method
    {
      System.out.println("Tuut, tuut!");
    }
}
```

- Deriving a subclass from another derived subclass is called multilevel inheritance. The use of a derived class as a superclass is a common concept in object-oriented programming like java. A chain of classes is built using this concept.
- The Final keyword is declared to finalize variables and methods used in the class. A class with the Final keyword is called a Final Class. No other class can inherit a class with the Final keyword. Using the Final keyword means the prevention of inheriting any properties or methods

of that class.

- The Abstract keyword is used for abstraction. A class with the Abstract keyword is called Abstract Class. An Abstract class consists of both Abstract and concrete methods.
- Members of a class that can be accessed without creating an object are called static members. The Static keyword is used to create a static method. The Static keyword can be used in methods, classes, fields, and blocks of a Java program.

5.9 FURTHER READINGS

1) Schildt, H., & Coward, D. (2014). Java: the complete reference (p. 1312). New York: McGraw-Hill Education.

5.10 ANSWERS TO CHECK YOUR PROGRESS

Ans to Q No 1: Inheritance is the deriving of a new class from an old class. The old class that is inherited is known as the superclass or base class or parent class and the new class that does the inheriting is called the subclass or derived class or child class

Ans to Q No 2: The keyword used is : Extends

Ans to Q No 3: private member

Ans to Q No 4: Yes, this is the correct way.

Ans to Q No 5: Deriving a subclass from another derived subclass is called multilevel inheritance. The use of a derived class as a superclass is a common concept in object-oriented programming like Java. A chain of classes is built using this concept. Not directly through class

Ans to Q No 6: Members of a class that can be accessed without creating an object are called static members.

Ans to Q No 7: The Abstract keyword is used for abstraction. A class with the Abstract keyword is called Abstract Class. An Abstract class consists of both Abstract and concrete methods.



5.11 MODEL QUESTIONS

Q 1: What is Inheritance in Java?

Q 2: Why do we need to use inheritance?

Q 3: What is Is-A relationship in Java?

Q 4: What are superclass and subclass?

Q 5: How is Inheritance implemented/achieved in Java?

Q 6: Write the syntax for creating the subclass of a class.

Q 7: What are the advantages of inheritance in Java?

*** ***** ***