

Devanshu Surana

PC-23, 1032210755

Batch C1

Page No.	
Date	

Lab Assignment - 2B

Problem statement:

Write a statement in Java showing hierarchical inheritance with base class as employee and derived classes as FullTimeEmployee and InternEmployee with methods Displaysalary in Base class and CalculateSalary in derived classes. Calculate salary method will calculate as per increment given to full time and intern Employees. Full time employee - 50% hike, InternEmployees - 25% hike. Display salary before and after hike.

Objective:

1. To study inheritance in Java
2. To study why to use inheritance
3. To study types of inheritance

Theory:

Inheritance In Java: It is a mechanism in which one object acquires all the properties and behaviours of a parent object. The idea behind inheritance in Java is that we can create new classes that are built upon existing classes. When we inherit from an existing class, we can reuse methods and fields of the parent class. Moreover we can add new methods and fields in your current class also. We use inheritance in Java for method overriding.

code reusability and for Abstraction also

Types:

1. Single Inheritance:

Subclasses inherit the features of one super

2. Multilevel Inheritance: A derived class inheriting a base class, and as well as another class also acts as the base class for

3. Hierarchical Inheritance: One class serves as superclass for more than one subclass.

4. Multiple Inheritance: Java does not support multiple inheritance with classes. In Java, we can achieve multiple inheritance only through interfaces.

5. Hybrid Inheritance: Combination of more than one types of inheritances, single and multiple inheritance can be achieved through interfaces only as multiple inheritance is not supported by Java.

Conclusion:

Thus, we have successfully implemented inheritance in Java.

FAQ's

1) Java does not support multiple inheritance with classes. In Java we can achieve multiple inheritance only through interfaces.

An interface is a blueprint of a class that provides a set of abstract methods that a class must implement.

2) Whenever one class inherits another class, it is called as IS-A relationship.

IS-A relationship is ~~not~~ completely related to inheritance.

IS-A relationship is additionally used for code reusability in Java and to avoid code redundancy it can simply be achieved by extending an interface or class by using extend keyword.

3) Constructors and Initializers both static initialization block and instance initialization block are not inherited to the subclass but they get executed during the instantiation of the subclass.

Algorithm

Step 1: Start

Step 2: create base class employee

Step 3: create derived classes using extend keyword

Step 4: Calculate salary by giving 50% hike to fulltime and 25% to intern

Step 5: Display salary after hike

Step 6: End