**G. H. RAISONI COLLEGE OF ENGG., NAGPUR**

**(An Autonomous Institute)**

**Department of Computer Science & Engg.**



**Date: 12-08-2021**

**Practical Subject: Java Programming**

**Session: 2021-22**

**Student Details:**

| **Roll Number** | 44 |
| --- | --- |
| **Name** | Anand Suralkar |
| **Semester** | 9th |
| **Section** | A |
| **Batch** | CSE |

**Practical Details: Practical Number-3**

| Practical Aim | Performing the concept of Inheritance in java. |
| --- | --- |
| Theory & Syntax | Inheritance in Java  Inheritance  Types of Inheritance  Why multiple inheritance is not possible in Java in case of class?  Inheritance in Java is a mechanism in which one object acquires all the properties and behaviors of a parent object. It is an important part of OOPs (Object Oriented programming system).  The idea behind inheritance in Java is that you can create new classes that are built upon existing classes. When you inherit from an existing class, you can reuse methods and fields of the parent class. Moreover, you can add new methods and fields in your current class also.  Inheritance represents the IS-A relationship which is also known as a parent-child relationship.  Why use inheritance in java  For Method Overriding (so runtime polymorphism can be achieved).  For Code Reusability.  Terms used in Inheritance  Class: A class is a group of objects which have common properties. It is a template or blueprint from which objects are created.  Sub Class/Child Class: Subclass is a class which inherits the other class. It is also called a derived class, extended class, or child class.  Super Class/Parent Class: Superclass is the class from where a subclass inherits the features. It is also called a base class or a parent class.  Reusability: As the name specifies, reusability is a mechanism which facilitates you to reuse the fields and methods of the existing class when you create a new class. You can use the same fields and methods already defined in the previous class.  The syntax of Java Inheritance  class Subclass-name extends Superclass-name  {  //methods and fields  }  The extends keyword indicates that you are making a new class that derives from an existing class. The meaning of "extends" is to increase the functionality.  In the terminology of Java, a class which is inherited is called a parent or superclass, and the new class is called child or subclass.    Java Inheritance Example  Inheritance in Java  As displayed in the above figure, Programmer is the subclass and Employee is the superclass. The relationship between the two classes is Programmer IS-A Employee. It means that Programmer  In the above example, Programmer object can access the field of own class as well as of Employee class i.e. code reusability.  Types of inheritance in java  On the basis of class, there can be three types of inheritance in java: single, multilevel and hierarchical.  In java programming, multiple and hybrid inheritance is supported through interface only. We will learn about interfaces later.  Types of inheritance in Java  Note: Multiple inheritance is not supported in Java through class.  When one class inherits multiple classes, it is known as multiple inheritance. For Example:  Multiple inheritance in Java  Single Inheritance Example  When a class inherits another class, it is known as a single inheritance.  When there is a chain of inheritance, it is known as multilevel inheritance |
| Program | public class Employee {  float salary=40000;  }  public class Programmer extends Employee{  int bonus=10000;  public static void main(String[] args){  Programmer p=new Programmer ();  System.out.println("Programmers Salary is "+p.salary);  System.out.println("Programmers Bonus is "+p.bonus);  }  } |
| Output |  |
| Conclusion | Learned the concept of Inheritance in java. |