1. public class Calculator {

// Method to multiply two integers

public int multiply(int a, int b) {

return a \* b;

}

// Method overload to multiply three doubles

public double multiply(double x, double y, double z) {

return x \* y \* z;

}

public static void main(String[] args) {

// Create an instance of Calculator

Calculator calculator = new Calculator();

// Call the multiply method with two integers

int resultInt = calculator.multiply(5, 7);

System.out.println("Multiplication of two integers: " + resultInt);

// Call the multiply method overload with three doubles

double resultDouble = calculator.multiply(2.5, 3.0, 1.5);

System.out.println("Multiplication of three doubles: " + resultDouble);

}

}

2)

class Employee {

private String name;

private int employeeID;

public Employee(String name, int employeeID) {

this.name = name;

this.employeeID = employeeID;

}

public double calculateSalary() {

return 50000; // Basic salary for all employees

}

public String toString() {

return "Employee ID: " + employeeID + "\nName: " + name + "\nSalary: $" + calculateSalary();

}

}

class Manager extends Employee {

private double bonusPercentage;

public Manager(String name, int employeeID, double bonusPercentage) {

super(name, employeeID);

this.bonusPercentage = bonusPercentage;

}

@Override

public double calculateSalary() {

// Adding bonus for managers

return super.calculateSalary() + super.calculateSalary() \* (bonusPercentage / 100);

}

}

class Developer extends Employee {

private String programmingLanguage;

public Developer(String name, int employeeID, String programmingLanguage) {

super(name, employeeID);

this.programmingLanguage = programmingLanguage;

}

@Override

public double calculateSalary() {

// Adding extra allowance for developers

return super.calculateSalary() + 10000; // Extra allowance for developers

}

@Override

public String toString() {

return super.toString() + "\nProgramming Language: " + programmingLanguage;

}

}

public class EmployeeDemo {

public static void main(String[] args) {

Manager manager = new Manager("John Doe", 101, 15);

Developer developer = new Developer("Jane Smith", 102, "Java");

System.out.println("Manager Details:\n" + manager);

System.out.println("\nDeveloper Details:\n" + developer);

}

}

3)

class Vehicle {

protected double speed;

public Vehicle(double speed) {

this.speed = speed;

}

public double calculateSpeed() {

return speed;

}

}

class Car extends Vehicle {

private int numPassengers;

public Car(double speed, int numPassengers) {

super(speed);

this.numPassengers = numPassengers;

}

@Override

public double calculateSpeed() {

return super.calculateSpeed() \* numPassengers;

}

}

class Motorcycle extends Vehicle {

private int numWheels;

public Motorcycle(double speed, int numWheels) {

super(speed);

this.numWheels = numWheels;

}

@Override

public double calculateSpeed() {

return super.calculateSpeed() \* numWheels;

}

}

public class Main {

public static void main(String[] args) {

Car myCar = new Car(60.0, 4);

Motorcycle myMotorcycle = new Motorcycle(80.0, 2);

double carSpeed = myCar.calculateSpeed();

double motorcycleSpeed = myMotorcycle.calculateSpeed();

System.out.println("Car speed: " + carSpeed);

System.out.println("Motorcycle speed: " + motorcycleSpeed);

if (carSpeed > motorcycleSpeed) {

System.out.println("The car has the highest effective speed.");

} else if (motorcycleSpeed > carSpeed) {

System.out.println("The motorcycle has the highest effective speed.");

} else {

System.out.println("Both vehicles have the same effective speed.");

}

}

}