interface Employee {

String getEmployeeId();

String getEmployeeName();

}

interface SalaryCalculator extends Employee {

double calculateBasicSalary();

double calculateDeductions();

}

interface BonusCalculator extends SalaryCalculator {

double calculatePerformanceBonus();

double calculateYearlyBonus();

}

class EmployeeImpl implements Employee {

private String employeeId;

private String employeeName;

public EmployeeImpl(String employeeId, String employeeName) {

this.employeeId = employeeId;

this.employeeName = employeeName;

}

@Override

public String getEmployeeId() {

return employeeId;

}

@Override

public String getEmployeeName() {

return employeeName;

}

}

class SalaryManagerImpl extends EmployeeImpl implements SalaryCalculator, BonusCalculator {

public SalaryManagerImpl(String employeeId, String employeeName) {

super(employeeId, employeeName);

}

@Override

public double calculateBasicSalary() {

return 50000.0;

}

@Override

public double calculateDeductions() {

return 5000.0;

}

@Override

public double calculatePerformanceBonus() {

return 2000.0;

}

@Override

public double calculateYearlyBonus() {

return 3000.0;

}

public double calculateNetSalary() {

return calculateBasicSalary()-calculateDeductions()+calculatePerformanceBonus()+calculateYearlyBonus();

}

public void displayEmployeeDetails() {

System.out.println("Employee ID: " + getEmployeeId());

System.out.println("Employee Name: " + getEmployeeName());

System.out.println("Basic Salary: " + calculateBasicSalary());

System.out.println("Deductions: " + calculateDeductions());

System.out.println("Performance Bonus: " + calculatePerformanceBonus());

System.out.println("Yearly Bonus: " + calculateYearlyBonus());

System.out.println("Net Salary: " + calculateNetSalary());

}

}

public class p4 {

public static void main(String[] args) {

SalaryManagerImpl employee = new SalaryManagerImpl("EMP001", "vinay");

employee.displayEmployeeDetails();

}

}

