**package** oop;

//Base class for Employee

**class** Employee {

// Member variables

String empName;

String empId;

String address;

String mailId;

String mobileNo;

**double** basicPay;

// Constructor to initialize the employee details

**public** Employee(String empName, String empId, String address, String mailId, String mobileNo, **double** basicPay) {

**this**.empName = empName;

**this**.empId = empId;

**this**.address = address;

**this**.mailId = mailId;

**this**.mobileNo = mobileNo;

**this**.basicPay = basicPay;

}

// Method to calculate allowances (DA, HRA, PF, staff club fund)

**public** **double**[] calculateAllowances() {

**double** DA = 0.97 \* basicPay;

**double** HRA = 0.10 \* basicPay;

**double** PF = 0.12 \* basicPay;

**double** staffClubFund = 0.001 \* basicPay;

**return** **new** **double**[]{DA, HRA, PF, staffClubFund};

}

// Method to calculate the gross and net salary

**public** **double**[] calculateSalary() {

**double**[] allowances = calculateAllowances();

**double** DA = allowances[0];

**double** HRA = allowances[1];

**double** PF = allowances[2];

**double** staffClubFund = allowances[3];

**double** grossSalary = basicPay + DA + HRA;

**double** deductions = PF + staffClubFund;

**double** netSalary = grossSalary - deductions;

**return** **new** **double**[]{grossSalary, netSalary};

}

// Method to generate and print the pay slip for the employee

**public** **void** generatePaySlip() {

**double**[] salary = calculateSalary();

**double** grossSalary = salary[0];

**double** netSalary = salary[1];

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

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

System.***out***.println("Basic Pay: " + basicPay);

System.***out***.println("Gross Salary: " + grossSalary);

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

System.***out***.println("----------------------------------------");

}

}

//Derived class for Programmer

**class** Programmer **extends** Employee {

**public** Programmer(String empName, String empId, String address, String mailId, String mobileNo, **double** basicPay) {

**super**(empName, empId, address, mailId, mobileNo, basicPay);

}

}

//Derived class for Team Lead

**class** TeamLead **extends** Employee {

**public** TeamLead(String empName, String empId, String address, String mailId, String mobileNo, **double** basicPay) {

**super**(empName, empId, address, mailId, mobileNo, basicPay);

}

}

//Derived class for Assistant Project Manager

**class** AssistantProjectManager **extends** Employee {

**public** AssistantProjectManager(String empName, String empId, String address, String mailId, String mobileNo, **double** basicPay) {

**super**(empName, empId, address, mailId, mobileNo, basicPay);

}

}

//Derived class for Project Manager

**class** ProjectManager **extends** Employee {

**public** ProjectManager(String empName, String empId, String address, String mailId, String mobileNo, **double** basicPay) {

**super**(empName, empId, address, mailId, mobileNo, basicPay);

}

}

//Main class to test the implementation

**public** **class** Main {

**public** **static** **void** main(String[] args) {

// Create instances of each type of employee

Programmer programmer = **new** Programmer("John Doe", "E001", "123 Street", "john@example.com", "9876543210", 50000);

TeamLead teamLead = **new** TeamLead("Alice Smith", "E002", "456 Avenue", "alice@example.com", "9876543220", 70000);

AssistantProjectManager assistantPM = **new** AssistantProjectManager("Bob Brown", "E003", "789 Boulevard", "bob@example.com", "9876543230", 90000);

ProjectManager projectManager = **new** ProjectManager("Eve White", "E004", "1010 Road", "eve@example.com", "9876543240", 120000);

// Generate pay slips for each employee

programmer.generatePaySlip();

teamLead.generatePaySlip();

assistantPM.generatePaySlip();

projectManager.generatePaySlip();

}

}