# Third Year B. Tech (EL & CE)

**Semester: V Subject:** Object-Oriented Programming Lab

**Name: Shreerang Mhatre Class: SY**

**Roll No: 52 Batch: A3**

# Experiment No: 03

**Name of the Experiment**: **Classes and Objects**

**Performed on: 13/09/2023**

**Submitted on: 27/09/2023**

**Problem Statement:**

Create an Employee Payroll System in C++ using object-oriented programming.

Define a base class Employee with attributes and a pay slip calculator.

Implement derived classes for different employee categories (Programmer, Team Lead,

Assistant Project Manager, and Project Manager) inheriting from Employee.

In the main function, instantiate employees of each category, input their details

and basic pay, and display their pay slips, demonstrating inheritance,

constructors, and polymorphism.

**Output:**

Employee: Viraj Parmar

Employee ID: EMP123

Address: 123 Main St.

Mail ID: john@example.com

Mobile Number: 555-1234

Basic Pay: 50000

Dearness Allowance (DA): 48500House Rent Allowance (HRA): 5000

Provident Fund (PF): 6000

Staff Club Fund: 50

Gross Salary: 103500

Net Salary: 97450

----------------------------------

Employee: Sagar Acharya

Employee ID: EMP456

Address: 456 Elm St.

Mail ID: jane@example.com

Mobile Number: 555-5678

Basic Pay: 60000

Dearness Allowance (DA): 58200House Rent Allowance (HRA): 6000     ce (HRA): 6000

Provident Fund (PF): 7200

Staff Club Fund: 60

Gross Salary: 124200

Net Salary: 116940

----------------------------------

Employee: Sandeep Patil

Employee ID: EMP789

Address: 789 Oak St.

Mail ID: michael@example.com

Mobile Number: 555-9876

Basic Pay: 70000                                           000

Dearness Allowance (DA): 67900House Rent Allowance (HRA): 7ce (HRA): 7000

Provident Fund (PF): 8400

Staff Club Fund: 70

Gross Salary: 144900

Net Salary: 136430

----------------------------------

Employee: Sourabh Shah

Employee ID: EMP101

Address: 101 Pine St.

Mail ID: sarah@example.com

Mobile Number: 555-1111                                    000

Basic Pay: 80000

Dearness Allowance (DA): 77600House Rent Allowance (HRA): 8ce (HRA): 8000

Provident Fund (PF): 9600

Staff Club Fund: 80

Gross Salary: 165600

Net Salary: 155920

----------------------------------

Code:

#include <iostream>

#include <string>

using namespace std;

class Employee {

protected:

string emp\_name;

string emp\_id;

string address;

string mail\_id;

string mobile\_no;

public:

Employee(string name, string id, string addr, string mail, string mobile)

: emp\_name(name), emp\_id(id), address(addr), mail\_id(mail), mobile\_no(mobile) {}

void display\_pay\_slip(double basic\_pay) {

cout << "Employee: " << emp\_name << endl;

cout << "Employee ID: " << emp\_id << endl;

cout << "Address: " << address << endl;

cout << "Mail ID: " << mail\_id << endl;

cout << "Mobile Number: " << mobile\_no << endl;

double da = 0.97 \* basic\_pay;

double hra = 0.10 \* basic\_pay;

double pf = 0.12 \* basic\_pay;

double staff\_club\_fund = 0.001 \* basic\_pay;

double gross\_salary = basic\_pay + da + hra;

double net\_salary = gross\_salary - pf - staff\_club\_fund;

cout << "Basic Pay: " << basic\_pay << endl;

cout << "Dearness Allowance (DA): " << da << endl;

cout << "House Rent Allowance (HRA): " << hra << endl;

cout << "Provident Fund (PF): " << pf << endl;

cout << "Staff Club Fund: " << staff\_club\_fund << endl;

cout << "Gross Salary: " << gross\_salary << endl;

cout << "Net Salary: " << net\_salary << endl;

cout << "----------------------------------" << endl;

}

};

class Programmer : public Employee {

public:

Programmer(string name, string id, string addr, string mail, string mobile, double basic\_pay)

: Employee(name, id, addr, mail, mobile) {

display\_pay\_slip(basic\_pay);

}

};

class TeamLead : public Employee {

public:

TeamLead(string name, string id, string addr, string mail, string mobile, double basic\_pay)

: Employee(name, id, addr, mail, mobile) {

display\_pay\_slip(basic\_pay);

}

};

class AssistantProjectManager : public Employee {

public:

AssistantProjectManager(string name, string id, string addr, string mail, string mobile, double basic\_pay)

: Employee(name, id, addr, mail, mobile) {

display\_pay\_slip(basic\_pay);

}

};

class ProjectManager : public Employee {

public:

ProjectManager(string name, string id, string addr, string mail, string mobile, double basic\_pay)

: Employee(name, id, addr, mail, mobile) {

display\_pay\_slip(basic\_pay);

}

};

int main() {

Programmer programmer("Viraj Parmar", "EMP123", "123 Main St.", "john@example.com", "555-1234", 50000);

TeamLead team\_lead("Sagar Acharya", "EMP456", "456 Elm St.", "jane@example.com", "555-5678", 60000);

AssistantProjectManager assistant\_pm("Sandeep Patil", "EMP789", "789 Oak St.", "michael@example.com", "555-9876", 70000);

ProjectManager project\_manager("Sourabh Shah", "EMP101", "101 Pine St.", "sarah@example.com", "555-1111", 80000);

return 0;

}







