

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;
}
```

```
File Edit Selection View Go Run Terminal Help exp3
exp3.c++ X
exp3.c++ > ...
1  /*
2  Problem statement:
3  Create an Employee Payroll System in C++ using object-oriented programming.
4  Define a base class Employee with attributes and a pay slip calculator.
5  Implement derived classes for different employee categories (Programmer, Team Lead,
6  Assistant Project Manager, and Project Manager) inheriting from Employee.
7  In the main function, instantiate employees of each category, input their details
8  and basic pay, and display their pay slips, demonstrating inheritance,
9  constructors, and polymorphism.
10
11 Author: Shreerang P. Mhatre
12 Date: 13/09/2023
13
14 */
15
16 #include <iostream>
17 #include <string>
18
19 using namespace std;
20
21 class Employee {
22 protected:
23     string emp_name;
24     string emp_id;
25     string address;
26     string mail_id;
27     string mobile_no;
28
29 public:
30     Employee(string name, string id, string addr, string mail, string mobile)
31         : emp_name(name), emp_id(id), address(addr), mail_id(mail), mobile_no(mobile) {}
32
33     void display_pay_slip(double basic_pay) {
34         cout << "Employee: " << emp_name << endl;
35         cout << "Employee ID: " << emp_id << endl;
36         cout << "Address: " << address << endl;
37         cout << "Mail ID: " << mail_id << endl;
38     }
39
40     PS D:\Object Oriented Programing\exp3> cd "d:\Object Oriented Programing\exp3" ; if ($?) { g++ exp3.c++ -o exp3 } ; if ($?) { .\exp3 }
41 Employee: Viraj Parmar
42 Employee ID: EMP123
43 Address: 123 Main St.
44 Mail ID: john@example.com
45 Mobile Number: 555-1234
46 Basic Pay: 50000
47 Dearness Allowance (DA): 48500
48 House Rent Allowance (HRA): 5000
49 Provident Fund (PF): 6000
50 Staff Club Fund: 50
51 Gross Salary: 103500
52 Net Salary: 97450
53 -----
54 Employee: Sagar Acharya
55 Employee ID: EMP456
56 Address: 456 Elm St.
57 Mail ID: jane@example.com
58 Mobile Number: 555-5678
59 Basic Pay: 60000
60 Dearness Allowance (DA): 58200
61 House Rent Allowance (HRA): 6000
62 Provident Fund (PF): 7200
63 Staff Club Fund: 60
64 Gross Salary: 124200
65 Net Salary: 116940
66 -----
67 Employee: Sandeep Patil
68 Employee ID: EMP789
69 Address: 789 Oak St.
70 Mail ID: michael@example.com
71 Mobile Number: 555-9876
72 Basic Pay: 70000
73 Dearness Allowance (DA): 67900
74 House Rent Allowance (HRA): 7000
75 Provident Fund (PF): 8400
76 Staff Club Fund: 70
77 Gross Salary: 144900
78 Net Salary: 136430
79
80 Type here to search 24°C 10:03 AM 27-09-2023
```

```
File Edit Selection View Go Run Terminal Help exp3
exp3.c++ X
exp3.c++ > ...
28
29 public:
30     Employee(string name, string id, string addr, string mail, string mobile)
31         : emp_name(name), emp_id(id), address(addr), mail_id(mail), mobile_no(mobile) {}
32
33     void display_pay_slip(double basic_pay) {
34         cout << "Employee: " << emp_name << endl;
35         cout << "Employee ID: " << emp_id << endl;
36         cout << "Address: " << address << endl;
37         cout << "Mail ID: " << mail_id << endl;
38         cout << "Mobile Number: " << mobile_no << endl;
39
40         double da = 0.97 * basic_pay;
41         double hra = 0.10 * basic_pay;
42         double pf = 0.12 * basic_pay;
43         double staff_club_fund = 0.001 * basic_pay;
44
45         double gross_salary = basic_pay + da + hra;
46         double net_salary = gross_salary - pf - staff_club_fund;
47
48         cout << "Basic Pay: " << basic_pay << endl;
49         cout << "Dearness Allowance (DA): " << da << endl;
50         cout << "House Rent Allowance (HRA): " << hra << endl;
51         cout << "Provident Fund (PF): " << pf << endl;
52         cout << "Staff Club Fund: " << staff_club_fund << endl;
53         cout << "Gross Salary: " << gross_salary << endl;
54         cout << "Net Salary: " << net_salary << endl;
55         cout << "-----" << endl;
56     }
57
58     class Programmer : public Employee {
59     public:
60         Programmer(string name, string id, string addr, string mail, string mobile, double basic_pay)
61             : Employee(name, id, addr, mail, mobile) {
62             display_pay_slip(basic_pay);
63         }
64     }
65
66 PS D:\Object Oriented Programing\exp3> cd "d:\Object Oriented Programing\exp3" ; if ($?) { g++ exp3.c++ -o exp3 } ; if ($?) { .\exp3 }
67 Employee: Viraj Parmar
68 Employee ID: EMP123
69 Address: 123 Main St.
70 Mail ID: john@example.com
71 Mobile Number: 555-1234
72 Basic Pay: 50000
73 Dearness Allowance (DA): 48500
74 House Rent Allowance (HRA): 5000
75 Provident Fund (PF): 6000
76 Staff Club Fund: 50
77 Gross Salary: 103500
78 Net Salary: 97450
79 -----
80 Employee: Sagar Acharya
81 Employee ID: EMP456
82 Address: 456 Elm St.
83 Mail ID: jane@example.com
84 Mobile Number: 555-5678
85 Basic Pay: 60000
86 Dearness Allowance (DA): 58200
87 House Rent Allowance (HRA): 6000
88 Provident Fund (PF): 7200
89 Staff Club Fund: 60
90 Gross Salary: 124200
91 Net Salary: 116940
92 -----
93 Employee: Sandeep Patil
94 Employee ID: EMP789
95 Address: 789 Oak St.
96 Mail ID: michael@example.com
97 Mobile Number: 555-9876
98 Basic Pay: 70000
99 Dearness Allowance (DA): 67900
100 House Rent Allowance (HRA): 7000
101 Provident Fund (PF): 8400
102 Staff Club Fund: 70
103 Gross Salary: 144900
104 Net Salary: 136430
105
106 Type here to search 24°C 10:03 AM 27-09-2023
```

```

exp3.c++
58
59 class Programmer : public Employee {
60 public:
61     Programmer(string name, string id, string addr, string mail, string mobile, double basic_pay)
62         : Employee(name, id, addr, mail, mobile) {
63         display_pay_slip(basic_pay);
64     };
65 };
66
67 class TeamLead : public Employee {
68 public:
69     TeamLead(string name, string id, string addr, string mail, string mobile, double basic_pay)
70         : Employee(name, id, addr, mail, mobile) {
71         display_pay_slip(basic_pay);
72     };
73 };
74
75 class AssistantProjectManager : public Employee {
76 public:
77     AssistantProjectManager(string name, string id, string addr, string mail, string mobile, double basic_pay)
78         : Employee(name, id, addr, mail, mobile) {
79         display_pay_slip(basic_pay);
80     };
81 };
82
83 class ProjectManager : public Employee {
84 public:
85     ProjectManager(string name, string id, string addr, string mail, string mobile, double basic_pay)
86         : Employee(name, id, addr, mail, mobile) {
87         display_pay_slip(basic_pay);
88     };
89 };
90
91 int main() {
92     Programmer programmer("Viraj Parmar", "EMP123", "123 Main St.", "john@example.com", "555-1234", 50000);
93     TeamLead team_lead("Sagar Acharya", "EMP456", "456 Elm St.", "jane@example.com", "555-5678", 60000);
94     AssistantProjectManager assistant_pm("Sandeep Patil", "EMP789", "789 Oak St.", "michael@example.com", "555-9876", 70000);
95     ProjectManager project_manager("Sourabh Shah", "EMP101", "101 Pine St.", "sarah@example.com", "555-1111", 80000);
96
97     return 0;
98 }
99

```

```

PS D:\Object Oriented Programing\exp3> cd "d:\Object Oriented Programing\exp3" ; if ($?) { g++ exp3.c++ -o exp3 } ; if ($?) { .\exp3 }
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): 48500
House 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): 58200
House Rent Allowance (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
Dearness Allowance (DA): 67900
House Rent Allowance (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
Basic Pay: 80000
Dearness Allowance (DA): 78000
House Rent Allowance (HRA): 8000
Provident Fund (PF): 9600
Staff Club Fund: 80
Gross Salary: 166400
Net Salary: 157400
-----

```

```

exp3.c++
69 TeamLead team_lead(string name, string id, string addr, string mail, string mobile, double basic_pay)
70 : Employee(name, id, addr, mail, mobile) {
71     display_pay_slip(basic_pay);
72 };
73 };
74
75 class AssistantProjectManager : public Employee {
76 public:
77     AssistantProjectManager(string name, string id, string addr, string mail, string mobile, double basic_pay)
78         : Employee(name, id, addr, mail, mobile) {
79         display_pay_slip(basic_pay);
80     };
81 };
82
83 class ProjectManager : public Employee {
84 public:
85     ProjectManager(string name, string id, string addr, string mail, string mobile, double basic_pay)
86         : Employee(name, id, addr, mail, mobile) {
87         display_pay_slip(basic_pay);
88     };
89 };
90
91 int main() {
92     Programmer programmer("Viraj Parmar", "EMP123", "123 Main St.", "john@example.com", "555-1234", 50000);
93     TeamLead team_lead("Sagar Acharya", "EMP456", "456 Elm St.", "jane@example.com", "555-5678", 60000);
94     AssistantProjectManager assistant_pm("Sandeep Patil", "EMP789", "789 Oak St.", "michael@example.com", "555-9876", 70000);
95     ProjectManager project_manager("Sourabh Shah", "EMP101", "101 Pine St.", "sarah@example.com", "555-1111", 80000);
96
97     return 0;
98 }
99

```

```

PS D:\Object Oriented Programing\exp3> cd "d:\Object Oriented Programing\exp3" ; if ($?) { g++ exp3.c++ -o exp3 } ; if ($?) { .\exp3 }
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): 48500
House 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): 58200
House Rent Allowance (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
Dearness Allowance (DA): 67900
House Rent Allowance (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
Basic Pay: 80000
Dearness Allowance (DA): 78000
House Rent Allowance (HRA): 8000
Provident Fund (PF): 9600
Staff Club Fund: 80
Gross Salary: 166400
Net Salary: 157400
-----

```