|  |  |  |  |
| --- | --- | --- | --- |
| **Branch: Instrumentation & Control Engineering** | | **Year:** Second Year | |
| **Division: C** | **Roll No: 04** | **GR Number: 11911180** | **Subject:** OOPS |
| **Assignment No: 04** | **Date of Submission: 29-04-2021** | **Student Full Name: Shaunak Sudhir Deshpande** | |

Aim: Create an Employee class with data members as employee id, name and date of joining. Derive class WageEmployee with data members as hours

and rate and class Manager with data members as No of subordinates, basic salary and DA from class Employee. Derive SalesPerson class from

WageEmployee class. Use multiple inheritance to derive class SalesManager from classes i.e SalesPerson and Manager. Assume suitable data

members and member functions. Display the summary using inheritance.

Software Used: MinGW, VSCode

Code:

#include <iostream>

#include <cstring>

#include <string>

using namespace std;

class Employee

{

protected:

    int empID;

    char name[15];

    char joinDate[10];

public:

    Employee(int a, char \*b, char\* c)

    {

        empID = a;

        strcpy(name, b);

        strcpy(joinDate,c);

    }

    void Display()

    {

        cout << "\nName:\t" << name << "\nEmployee ID is:\t" << empID << "\nDate of joining:" << joinDate;

    }

};

class WageEmployee : protected Employee

{

protected:

    int hours;

    float rate;

    WageEmployee(int a, char \*b, char\* c, int h, float r) : Employee(a, b, c)

    {

        hours = h;

        rate = r;

    }

    void Display()

    {

        Employee::Display();

        cout << "\nHours:\t" << hours << "\nRate:\t" << rate;

    }

};

class Manager : virtual protected Employee

{

protected:

    int subordinateNo;

    size\_t basic;

    size\_t DA;

public:

    Manager(int a, char \*b, char\* c, int subno, size\_t bas, size\_t da) : Employee(a, b, c)

    {

        subordinateNo = subno;

        basic = bas;

        DA = da;

    }

    void Display()

    {

    //    Employee::Display();

        cout << "\nNumber of Subordinates:\t" << subordinateNo << "\nBasic:\t" << basic << "\nDA:\t" << DA;

    }

};

class SalesPerson : virtual protected WageEmployee

{

protected:

    char itemname[20]; //Name of the item which he is selling

    int itemID;

    float successRate;

public:

    SalesPerson(int a, char \*b, char\* c, int h, float r, char \*iname, int iID, float sRate) : WageEmployee(a, b, c, h, r)

    {

        strcpy(itemname, iname);

        itemID = iID;

        successRate = sRate;

    }

    void Display()

    {

        WageEmployee::Display();

        cout << "\nItemName:\t" << itemname << "\nItem ID:\t" << itemID << "\nSuccess Rate:\t" << successRate;

    }

};

class SalesManager : protected SalesPerson, protected Manager

{

private:

    char area[20]; //Area which he is responsible for

    int generatedRevenue;

public:

    SalesManager(int a, char \*b, char\* c, int h, float r, char \*iname, int iID, float sRate, int subno, size\_t bas, size\_t da, char \*ar, int gnr) : SalesPerson(a, b, c, h, r, iname, iID, sRate), Manager(a, b, c, subno, bas, da), WageEmployee(a, b, c, h, r), Employee(a, b, c)

    {

        strcpy(area, ar);

        generatedRevenue = gnr;

    }

    void Display()

    {

        SalesPerson::Display();

        Manager::Display();

        cout << "\nArea:\t" << area << "\nRevenue Generated:\t" << generatedRevenue;

    }

};

int main()

{

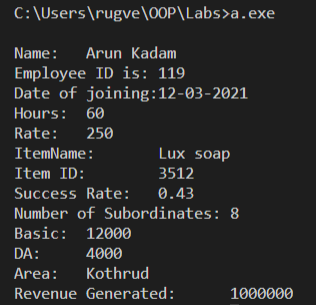
    SalesManager Arun(119, "Arun Kadam","12-03-2021",60,250,"Lux soap",3512,0.43,8,12000,4000,"Kothrud",1000000 );

    Arun.Display();

    return 0;

}

Output:



Analysis of Program:

In this program, we use multilevel & multiple inheritance both, in order to create an accurate representation of hierarchy in a sales firm. The hierarchy is as follows:

Employee

Manager

Wage employee

Salesperson

Sales Manager

Conclusion:

Using multiple & multilevel inheritance, along with concepts such as virtual functions and parameterized constructors, we can successfully create a system in C++ which is representative of real life organizational structures.