**Lab Assignment 5**

|  |  |
| --- | --- |
| **Roll No.: A055** | **Name: Ibrahim Shaikh** |
| **Program: B. Tech**-**CSBS (2ND YEAR)** | **Date of Release: 25th August 2021** |
| **Batch:**  1 | **Date of Submission: 30th August 2021** |

**PROBLEM STATEMENT:**

**Problem Statement:** Write a class “Employee” to read and print information of an employee with following details:

* + Data members: Name of the employee, Id of the employee, Department of the employee, basic salary

Calculate gross salary and its components (**declare them as data members**) based on following condition:

Dearness allowance (DA) = 50 % of basic salary

HRA = 30 % basic salary

TA = 10 % basic salary

Gross Salary = basic salary + DA + HRA + TA.

Write a member function to display employee details along with all the components of the salary.

**CODE:**

**Using Default Constructor, Destructor**

#include<iostream>

#include<string.h>          //for using 'string'

using namespace std;

class Employee             //Class Declaration

{

    string name, dept;

    int id, bsal;

    float da, hra, ta, gsal;        //Private data members

    public:                         //Public member functions

    Employee()          //Constructor

    {

        name = "NA";

        dept = "NA";

        id = -1;

        bsal = -1;

    }

    Employee(string a, string d, int i, int sal) //Constructor with Parameters

    {

        name= a;

        dept= d;

        id= i;

        bsal = sal;

        da = 0.5\*sal;

        hra = 0.3\*sal;

        ta = 0.1\*sal;

        gsal = bsal+da+hra+ta;

        return;

    }

    void display(int n)         //Display function to show output

    {

        int i = n;

        cout<<"\n\nName of the Employee "<<i<<": "<<name;

        cout<<"\nID of the Employee "<<i<<": "<<id;

        cout<<"\nDepartment of the Employee "<<i<<": "<<dept;

        cout<<"\nBasic Salary of the Employee "<<i<<": "<<bsal;

        cout<<"\nDearness Allowance of Employee "<<i<<": "<<da;

        cout<<"\nHRA of Employee "<<i<<": "<<hra;

        cout<<"\nTA of Employee "<<i<<": "<<ta;

        cout<<"\nGross Salary of Employee "<<i<<": "<<gsal;

    }

    ~Employee()             //Destructor

    {

        cout<<" ";

    }

};

int main()

{

    int n, i, sal;

    string a, d;

    Employee e[10];            // Declaration of array of objects

    cout<<"Enter the number of Employees: ";

    cin>>n;

    for(int x=1; x<=n; x++)     //for loop for taking user input

    {

        cout<<"\nEnter the name of Employee "<<x<<": ";

        cin>>a;

        cout<<"Enter the ID of Employee "<<x<<": ";

        cin>>i;

        cout<<"Enter the Department name to which Employee "<<x<<" belongs: ";

        cin>>d;

        cout<<"Enter the Basic Salary of Employee "<<x<<": ";

        cin>>sal;

        e[x] =Employee(a, d, i, sal);

    }

    for(int x=1; x<=n; x++)         // for loop for displaying

    {

    e[x].display(x);

    }

    return 0;

}

**Using Constructor Overloading**

#include<iostream>

#include<string.h>          //for using 'string'

using namespace std;

class Employee             //Class Declaration

{

    string name, dept;

    int id, bsal;

    float da, hra, ta, gsal;        //Private data members

    public:                         //Public member functions

    Employee(string a, string d, int i, int sal) //Constructor with Parameters

    {

        name= a;

        dept= d;

        id= i;

        bsal = sal;

        da = 0.5\*sal;

        hra = 0.3\*sal;

        ta = 0.1\*sal;

        gsal = bsal+da+hra+ta;

        return;

    }

    void display(int n)         //Display function to show output

    {

        int i = n;

        cout<<"\n\nName of the Employee "<<i<<": "<<name;

        cout<<"\nID of the Employee "<<i<<": "<<id;

        cout<<"\nDepartment of the Employee "<<i<<": "<<dept;

        cout<<"\nBasic Salary of the Employee "<<i<<": "<<bsal;

        cout<<"\nDearness Allowance of Employee "<<i<<": "<<da;

        cout<<"\nHRA of Employee "<<i<<": "<<hra;

        cout<<"\nTA of Employee "<<i<<": "<<ta;

        cout<<"\nGross Salary of Employee "<<i<<": "<<gsal;

    }

};

int main()

{

    Employee e("Samir", "Sales", 4550, 30000);

    Employee e1("Tom", "R & D", 5091, 41000);

    e.display(1);

    e1.display(2);

    return 0; }

**Using Copy Constructor**

#include<iostream>

#include<string.h>          //for using 'string'

using namespace std;

class Employee             //Class Declaration

{

    string name, dept;

    int id, bsal;

    float da, hra, ta, gsal;        //Private data members

    public:                         //Public member functions

    Employee(string a, string d, int i, int sal) //Constructor with Parameters

    {

        name= a;

        dept= d;

        id= i;

        bsal = sal;

        da = 0.5\*sal;

        hra = 0.3\*sal;

        ta = 0.1\*sal;

        gsal = bsal+da+hra+ta;

        return;

    }

    Employee(Employee &e2)

    {

        name= e2.name;

        dept= e2.dept;

        id= e2.id;

        bsal = e2.bsal;

        da = 0.5\*e2.bsal;

        hra = 0.3\*e2.bsal;

        ta = 0.1\*e2.bsal;

        gsal = bsal+da+hra+ta;

        return;

    }

    void display(int n)         //Display function to show output

    {

        int i = n;

        cout<<"\n\nName of the Employee "<<i<<": "<<name;

        cout<<"\nID of the Employee "<<i<<": "<<id;

        cout<<"\nDepartment of the Employee "<<i<<": "<<dept;

        cout<<"\nBasic Salary of the Employee "<<i<<": "<<bsal;

        cout<<"\nDearness Allowance of Employee "<<i<<": "<<da;

        cout<<"\nHRA of Employee "<<i<<": "<<hra;

        cout<<"\nTA of Employee "<<i<<": "<<ta;

        cout<<"\nGross Salary of Employee "<<i<<": "<<gsal;

    }

};

int main()

{

    Employee e("Samir", "Sales", 4550, 30000);

    Employee e1("Tom", "R & D", 5091, 41000);

    Employee e2 = e1;

    e.display(1);

    e1.display(2);

    cout<<"\n\nCopy Constructor used:";

    e2.display(3);

    return 0;

}

**Using Dynamic Constructor**

#include<iostream>

#include<string.h>          //for using 'string'

using namespace std;

class Employee             //Class Declaration

{

    string \*name, \*dept;

    int \*id, bsal;

    float da, hra, ta, gsal;        //Private data members

    public:                         //Public member functions

    Employee(string a, string d, int i, int sal) //Constructor with Parameters

    {

        name= new string;   //allocating memory at runtime

        \*name= a;

        dept= new string;

        \*dept= d;

        id= new int;

        \*id= i;

        bsal = sal;

        da = 0.5\*sal;

        hra = 0.3\*sal;

        ta = 0.1\*sal;

        gsal = bsal+da+hra+ta;

        return;

    }

    void display(int n)         //Display function to show output

    {

        int i = n;

        cout<<"\n\nName of the Employee "<<i<<": "<<\*name;

        cout<<"\nID of the Employee "<<i<<": "<<\*id;

        cout<<"\nDepartment of the Employee "<<i<<": "<<\*dept;

        cout<<"\nBasic Salary of the Employee "<<i<<": "<<bsal;

        cout<<"\nDearness Allowance of Employee "<<i<<": "<<da;

        cout<<"\nHRA of Employee "<<i<<": "<<hra;

        cout<<"\nTA of Employee "<<i<<": "<<ta;

        cout<<"\nGross Salary of Employee "<<i<<": "<<gsal;

    }

};

int main()

{

    Employee e("Samir", "Sales", 4550, 30000); //parameterized contructor called

    Employee e1("Tom", "R & D", 5091, 41000);  //parameterized contructor called

    e.display(1);

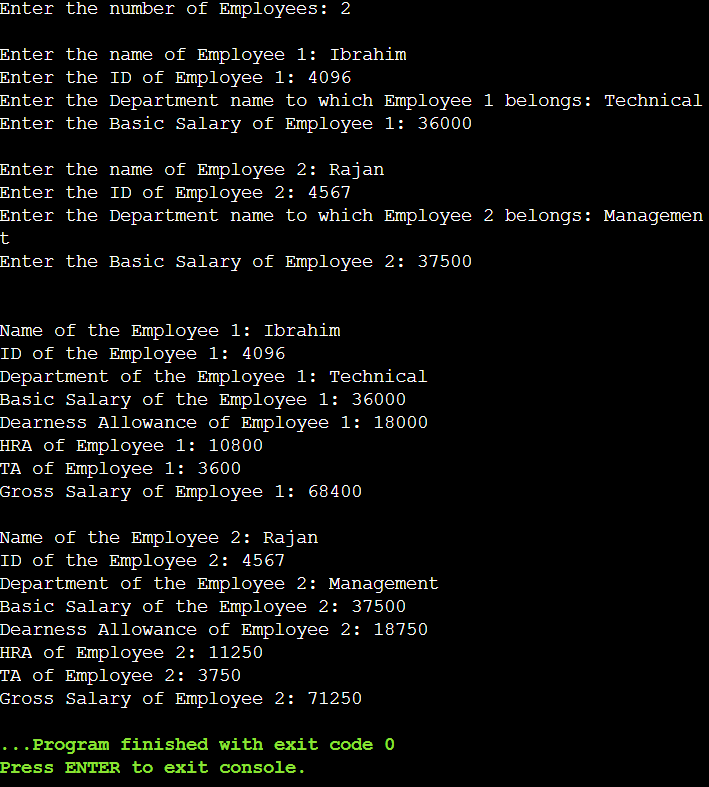
    e1.display(2);

    return 0;

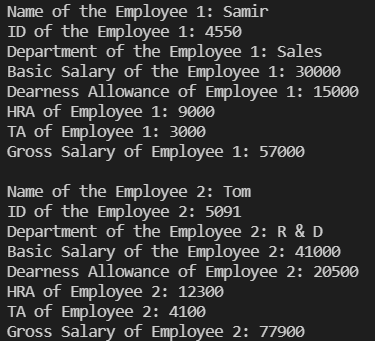
}

**OUTPUT:**

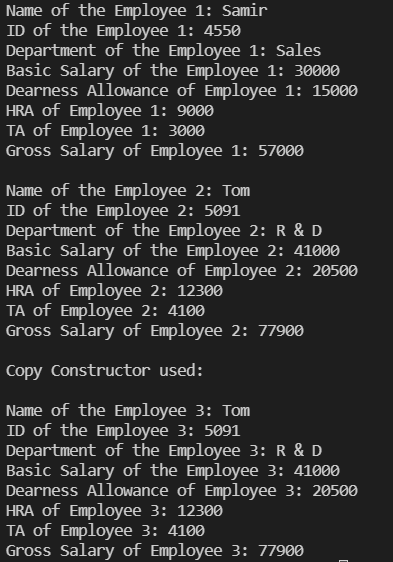
Default Constructor and Destructor



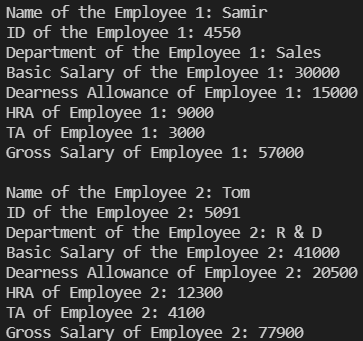
Constructor Overloading



Copy Constructor



Dynamic Constructor

****

**LINK FOR THE CODE:** Done using VSCode and OnlineGDB