**//Experiment no-07**

//Method overloading:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication7

{

public class Employee

{

double ysal, incsal;

public void PrintInfo(string name)

{

Console.WriteLine("Name: " + name);

}

public void PrintInfo(string dept, int id)

{

Console.WriteLine("Department:" + dept + ",Id: " + id);

}

public void PrintInfo(double sal)

{

Console.WriteLine("Monthly Salary :" +sal);

ysal = sal \* 12;

Console.WriteLine("Yearly Salary :" + ysal);

incsal = ysal + ((ysal \* 10) / 100);

Console.WriteLine("Netsalary: " + incsal);

}

}

class Program

{

static void Main(string[] args)

{

Employee emp = new Employee();

// Call the method with different parameter lists

emp.PrintInfo("John");

emp.PrintInfo("Marketting", 1206);

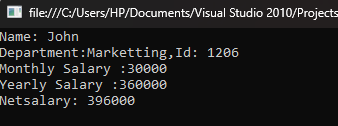
emp.PrintInfo(30000);

Console.ReadLine();

}

}

}



**//Method overriding:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication8

{

class baseClass

{

string name="Ansh";

string std = "ThirdYear";

int roll\_no = 12;

public virtual void showdata()

{

Console.WriteLine("-------Base class------");

Console.WriteLine("Name is :" + name);

Console.WriteLine("Class is :" + std);

Console.WriteLine("Roll no is:" + roll\_no);

Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

}

}

// derived class name 'derived'

// 'baseClass' inherit here

class derived : baseClass

{

float result;

int p = 60, c = 50, b = 70, m = 80;

int sum = 0;

// overriding

public override void showdata()

{

base.showdata();

Console.WriteLine("--------Derived class-------");

Console.WriteLine("Physics :" + p);

Console.WriteLine("chemistry:" + c);

Console.WriteLine("Biology:" + b);

Console.WriteLine("Maths:" + m);

sum = p + c + b + m;

result = sum / 4;

Console.WriteLine("Result is :" + result);

}

}

class GFG

{

// Main Method

public static void Main()

{

derived d = new derived(); // d is object of derived class // also works as object of baseclass

d.showdata();// it first invokes 'showdata()'of baseclass then it invokes 'showdata()' of derived class

Console.ReadLine();

}

}

}

