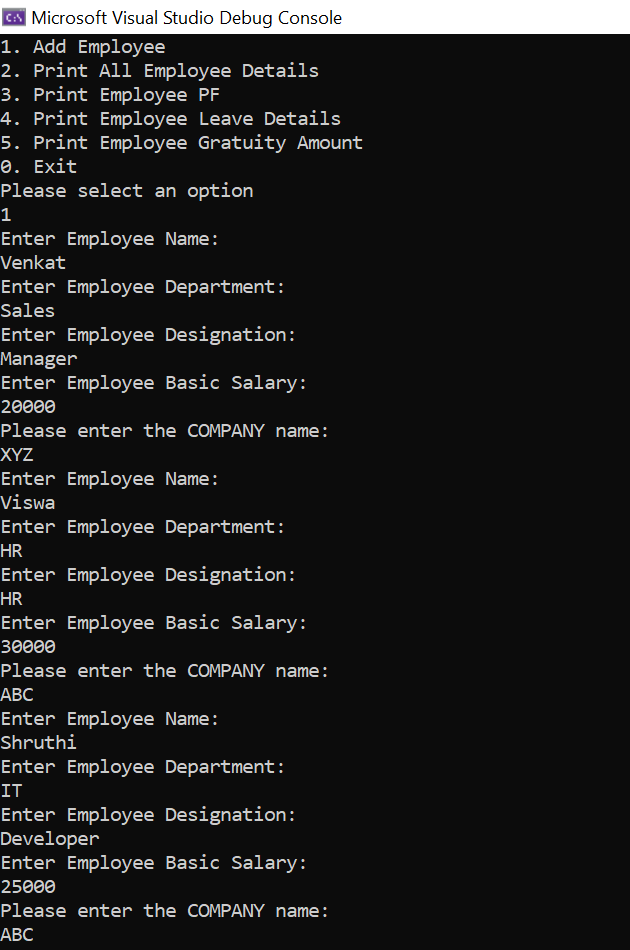
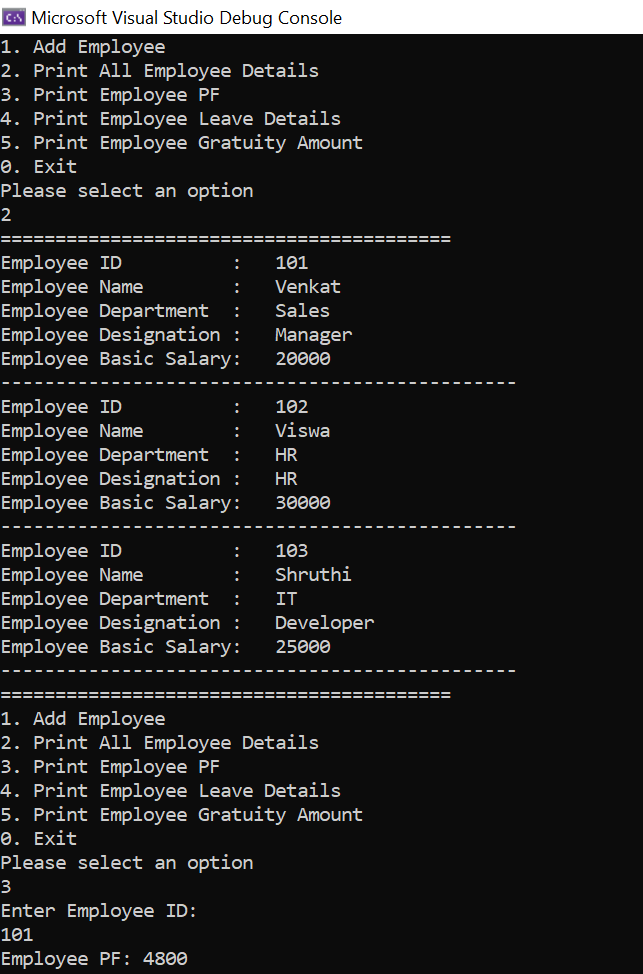
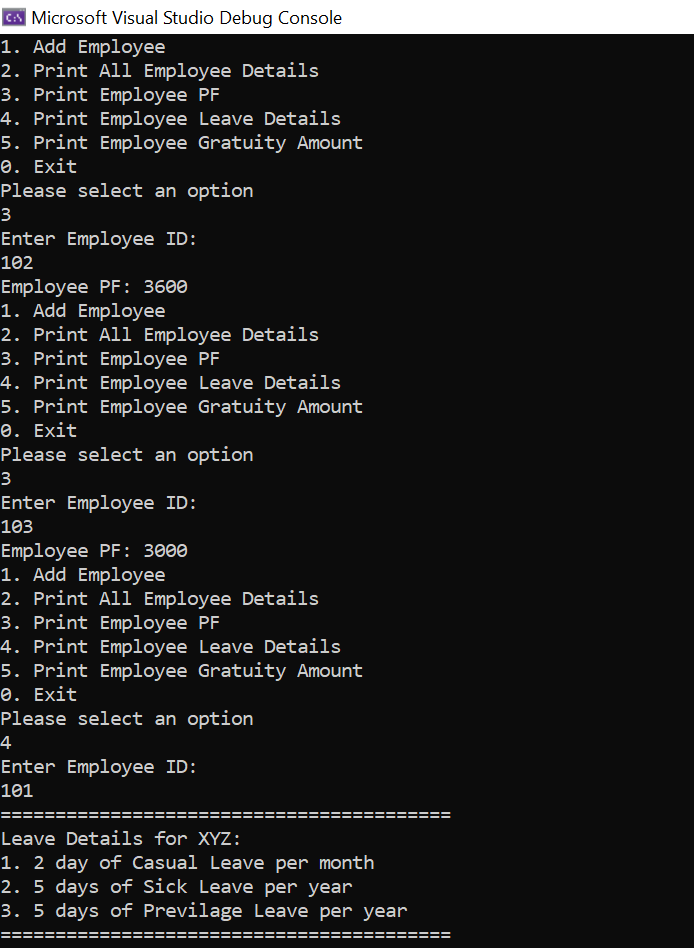
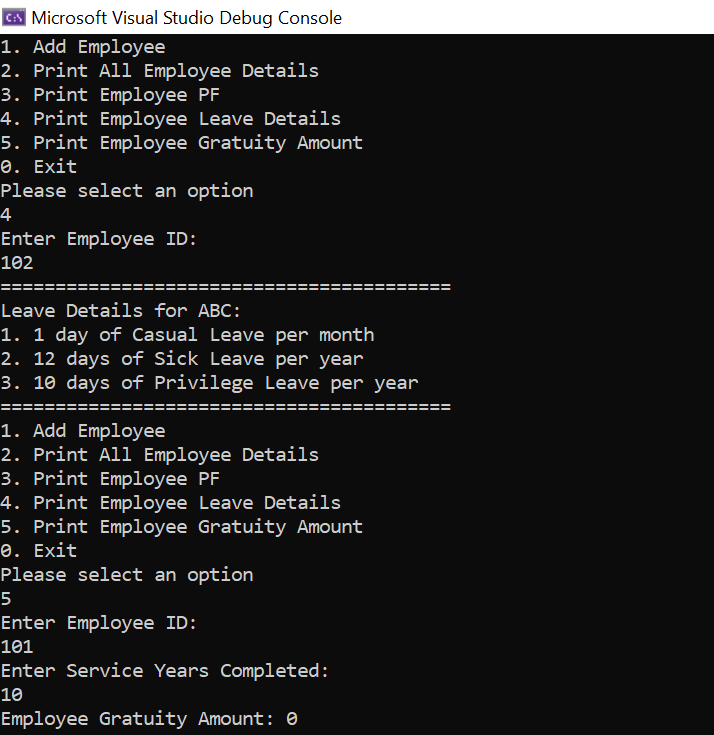
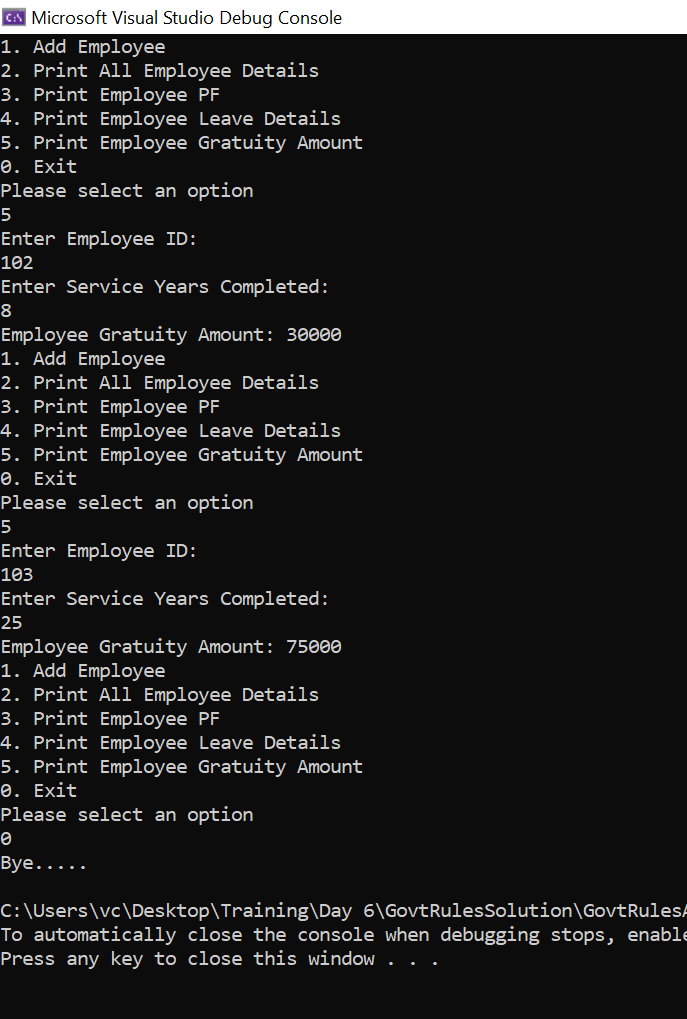
**OUTPUT:**











**CODE:**

**Program.cs**

using CompanyModelClassLibrary;

namespace GovtRulesApp

{

public class Program

{

Employee[] employees = new Employee[3];

string GetStringInput()

{

string inp;

do

{

inp = Console.ReadLine();

if (string.IsNullOrWhiteSpace(inp))

{

Console.WriteLine("Invalid Input! Try again..");

}

} while (string.IsNullOrWhiteSpace(inp));

return inp;

}

double GetDoubleInput()

{

double inp;

while (!(double.TryParse(Console.ReadLine(), out inp)))

{

Console.WriteLine("Invalid Entry! Try Again..");

}

return inp;

}

int GetIntInput()

{

int inp;

while (!(int.TryParse(Console.ReadLine(), out inp)))

{

Console.WriteLine("Invalid Entry! Try Again..");

}

return inp;

}

void PrintMenu()

{

Console.WriteLine("1. Add Employee");

Console.WriteLine("2. Print All Employee Details");

Console.WriteLine("3. Print Employee PF");

Console.WriteLine("4. Print Employee Leave Details");

Console.WriteLine("5. Print Employee Gratuity Amount");

Console.WriteLine("0. Exit");

}

void EmployeeInteraction()

{

int choice = 0;

do

{

PrintMenu();

Console.WriteLine("Please select an option");

choice = Convert.ToInt32(Console.ReadLine());

switch (choice)

{

case 0:

Console.WriteLine("Bye.....");

break;

case 1:

AddEmployees();

break;

case 2:

PrintAllEmployees();

break;

case 3:

PrintEmployeePF();

break;

case 4:

PrintEmployeeLeaveDetails();

break;

case 5:

PrintEmployeeGratuity();

break;

default:

Console.WriteLine("Invalid choice. Try again");

break;

}

} while (choice != 0);

}

Employee CreateEmployee(int id)

{

string name;

string dept;

string desig;

string type;

double salary;

Employee employee = new Employee();

Console.WriteLine("Enter Employee Name:");

name = GetStringInput();

Console.WriteLine("Enter Employee Department:");

dept = GetStringInput();

Console.WriteLine("Enter Employee Designation:");

desig = GetStringInput();

Console.WriteLine("Enter Employee Basic Salary:");

salary = GetDoubleInput();

Console.WriteLine("Please enter the COMPANY name:");

type = GetStringInput();

if (type == "XYZ")

employee = new XYZ(101 + id, name, dept, desig, salary);

else if (type == "ABC")

employee = new ABC(101 + id, name, dept, desig, salary);

return employee;

}

void AddEmployees()

{

if (employees[employees.Length - 1] != null)

{

Console.WriteLine("Sorry we have reached the maximum number of employees");

return;

}

for (int i = 0; i < employees.Length; i++)

{

if (employees[i] == null)

{

employees[i] = CreateEmployee(i);

}

}

}

void PrintAllEmployees()

{

Console.WriteLine("=========================================");

for (int i = 0; i < employees.Length; i++)

{

if (employees[i] != null)

{

employees[i].PrintAllEmployeeDetails();

}

}

Console.WriteLine("=========================================");

}

void PrintEmployeePF()

{

Console.WriteLine("Enter Employee ID:");

int id = GetIntInput();

double salary = employees[id - 101].BasicSalary;

double pf = employees[id - 101].EmployeePF(salary);

Console.WriteLine($"Employee PF: {pf}");

}

void PrintEmployeeLeaveDetails()

{

Console.WriteLine("Enter Employee ID:");

int id = GetIntInput();

string leavedetails = employees[id-101].LeaveDetails();

Console.WriteLine("=========================================");

Console.WriteLine(leavedetails);

Console.WriteLine("=========================================");

}

void PrintEmployeeGratuity()

{

Console.WriteLine("Enter Employee ID:");

int id = GetIntInput();

Console.WriteLine("Enter Service Years Completed:");

int years = GetIntInput();

double salary = employees[id - 101].BasicSalary;

double gratuity = employees[id - 101].gratuityAmount(years,salary);

Console.WriteLine($"Employee Gratuity Amount: {gratuity}");

}

static void Main(string[] args)

{

Program program = new Program();

program.EmployeeInteraction();

}

}

}

**XYZ.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace CompanyModelClassLibrary

{

public class XYZ : Employee, GovtRules

{

//public XYZ() : base()

//{

//}

public XYZ(int empID, string name, string department, string designation, double basicSalary) : base(empID, name, department, designation, basicSalary)

{

}

public override double EmployeePF(double basicSalary)

{

//EMPLOYER\_CONTRIBUTION = basicSalary\*0.0833;

return ((basicSalary) \* 0.12)\*2;

}

public override string LeaveDetails()

{

return "Leave Details for XYZ: \r\n1. 2 day of Casual Leave per month\r\n2. 5 days of Sick Leave per year\r\n3. 5 days of Previlage Leave per year";

}

public override double gratuityAmount(float serviceCompleted, double basicSalary)

{

return 0;

}

}

}

**ABC.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace CompanyModelClassLibrary

{

public class ABC : Employee, GovtRules

{

//public ABC() : base()

//{

//}

public ABC(int empID, string name, string department, string designation, double basicSalary) : base(empID, name, department, designation, basicSalary)

{

}

public override double EmployeePF(double basicSalary)

{

//EMPLOYER\_CONTRIBUTION = basicSalary\*0.0833;

return (basicSalary)\*0.12;

}

public override string LeaveDetails()

{

return "Leave Details for ABC: \r\n1. 1 day of Casual Leave per month\r\n2. 12 days of Sick Leave per year\r\n3. 10 days of Privilege Leave per year";

}

public override double gratuityAmount(float serviceCompleted, double basicSalary)

{

if (serviceCompleted > 5 && serviceCompleted <=10)

{

return (basicSalary) \* 1;

}

else if(serviceCompleted > 10 && serviceCompleted <=20) {

return (basicSalary) \* 2;

}

else if(serviceCompleted > 20)

{

return (basicSalary) \* 3;

}

else

{

return 0;

}

}

}

}

**GovtRules.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace CompanyModelClassLibrary

{

public interface GovtRules

{

public double EmployeePF(double basicSalary);

public string LeaveDetails();

public double gratuityAmount(float serviceCompleted, double basicSalary);

}

}

**Employee.cs**

namespace CompanyModelClassLibrary

{

public class Employee

{

public int EmpID { get; set; }

public string Name { get; set; }

public string Department { get; set; }

public string Designation { get; set; }

public double BasicSalary { get; set; }

//double EMPLOYER\_CONTRIBUTION { get; set; }

public Employee()

{

EmpID = 0;

Name = string.Empty;

Department = string.Empty;

Designation = string.Empty;

BasicSalary = 0.0;

}

public Employee(int empID, string name, string department, string designation, double basicSalary)

{

EmpID = empID;

Name = name;

Department = department;

Designation = designation;

BasicSalary = basicSalary;

}

public void PrintAllEmployeeDetails()

{

Console.WriteLine($"Employee ID :\t {EmpID}");

Console.WriteLine($"Employee Name :\t {Name}");

Console.WriteLine($"Employee Department :\t {Department}");

Console.WriteLine($"Employee Designation :\t {Designation}");

Console.WriteLine($"Employee Basic Salary:\t {BasicSalary}");

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

}

public virtual double EmployeePF(double basicSalary)

{

return 0;

}

public virtual string LeaveDetails()

{

return "";

}

public virtual double gratuityAmount(float serviceCompleted, double basicSalary)

{

return 0;

}

}

}