ASSESSMENT 3 **VASANTHABALAN M**

1. Answer all the problems by applying OOP concepts in all places applicable

a. Count the number of vowels in a given string

b. Create a ATM class that would allow the end user to perform cash withdrawal and deposit operations in sequence not by choice and print the balance in the account after every operation.

2. Implement hierarchical inheritance in a Company with employees of different departments segregated in the child classes. Write method to calculate the employees bonus (assume bonus is different for every department and is calculated from basic salary). (basic = 10000 bonus=10% bonus amt=10000*10% = 1000)

3. Create a Calculator Interface and derived class should implement the interface methods

a. Create a class named SpaceStation that is abstract

b. On that abstract class, add a abstract method called FireLaser

c. Create a derived class called DeathStar that implements the FireLaser method to write "Pew Pew" to the Console followed by a new line

Program.cs

```
using System;
using TestAssessment;

namespace MyApp // Note: actual namespace depends on the project name.
{
    internal class Program
    {
        public static void Main(string[] args)
        {
            //1.
            Ques1 ques1 = new Ques1();
            //ques1.vowel();
            ATM aTM = new ATM();
            aTM.Deposit(1000);
            aTM.Withdraw(200);


            //2.
```

```
            /*Company company = new Company();
            Console.WriteLine("Enter the department 1. department1
2.department2");
            int ch = Convert.ToInt32(Console.ReadLine());
            switch (ch)
            {
                case 1:
                    Departmnt1 departmnt1 = new Departmnt1();
                    departmnt1.BonusD1();
                    break;
                case 2:
                    Departmnt2 departmnt2 = new Departmnt2();
                    departmnt2.BonusD2();
                    break;
                default:
                    Console.WriteLine("Enter the correct choice!..");
                    break;
            }*/


            //3.
            //SpaceStation spaceStation = new DeathStar();
            //spaceStation.FireLaser();

        }
    }
}
```

1Ans

a)

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace TestAssessment
{
    internal class Ques1
    {
       public void vowel()
        {
            Console.WriteLine("enter the String:");
            string name = Console.ReadLine();
            int count = 0;

            char[] vowel = { 'a', 'e', 'i', 'o', 'u', 'A', 'E', 'I', 'O', 'U' };

            foreach (char c in name)
            {
                if(Array.Exists( vowel, vowels=>vowels==c))
                {
                    count++;
                }
            }
            Console.WriteLine(count);

        }
    }
}
```

1 b):

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace TestAssessment
{
    internal class ATM
    {
        private double balance;

        public ATM()
        {
           balance = 100;
        }

        public void Withdraw(double amount)
        {
            if (amount > 0)
            {
                if(amount<=balance)
                {
                    balance -= amount;
                    Console.WriteLine(balance);
                }
            }

        }
        public void Deposit(double amount)
        {
            if(amount > 0)
            {
                balance += amount;
                Console.WriteLine($"{amount} {balance}");
            }

        }
        public void Balance()
        {

        }
    }

}
```

Output:

2 Ans:

Parent.class

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace TestAssessment
{
    internal class Company
    {
        public Company()
        {

        }

    }
}
```

Child1:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace TestAssessment
{
    internal class Departmnt1:Company
    {


        public void BonusD1()
        {
            Console.WriteLine("enter the details: basicamount,bonus is5 only");
            int basicAmount = Convert.ToInt32(Console.ReadLine());
            double bounus = Convert.ToDouble(Console.ReadLine());

            double bnsAmt = basicAmount * (bounus/100);

            Console.WriteLine(bnsAmt);


        }
    }
}
```

Child2:

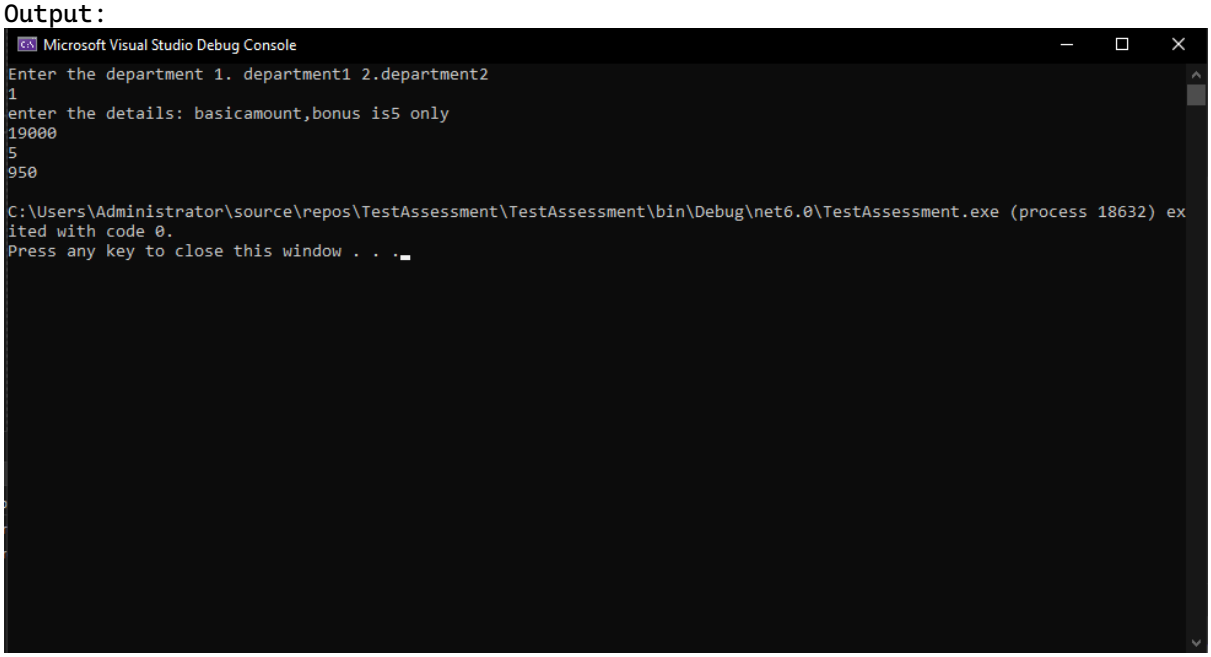
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace TestAssessment
{
    internal class Departmnt2:Company
    {
        public void BonusD2()
        {
            Console.WriteLine("enter the details: basicamount,bonus is 7 only");
            int basicAmount = Convert.ToInt32(Console.ReadLine());
            double bounus = Convert.ToDouble(Console.ReadLine());

            double bnsAmt = basicAmount * (bounus / 100);

            Console.WriteLine(bnsAmt);
        }
    }
}
```

Output:



3)
Parent class:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace TestAssessment
{
```

```
    public abstract class SpaceStation
    {
        public abstract void FireLaser();
    }
}
```

Derved class:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace TestAssessment
{
    public class DeathStar:SpaceStation
    {
        public override void FireLaser()
        {
            Console.WriteLine("Pew pew");
        }
    }
}
```

Output: