C# .Net Programming Assignment 4

- Create single visual Studio project.
- For Business logic write separate class.
- Use Object Oriented concepts while writing the program.

1. Write C# application which contains following things as :

One abstract class named as NumTemplate.
Which contains one integer as a characteristics.

Abstract Class should contains following behaviours as

1. Accept() - Accept elements to change existing value.
2. Display() - Display contents of element on screen.
3. ChkEven() - Check whether number is even or not
4. DisplayFactors() - Display all factors of number.
5. SumFactors() - Return summation of all factors.

Design one class NumOperation which inherit NumTemplate class.

Example :

```
abstract class NumTemplate
{
        public its No;

        public NumTemplate()
        {

        }

        public NumTemplate(int value)
        {

        }

        public abstract void Accept();
        public abstract void Display();
        public abstract bool ChkEven();
        public abstract int DisplayFactors();
        public abstract int SumFactors();
}

class NumOperation : NumTemplate
{
        public void Accept()
        {
                // Logic
        }
```

```
        public void Display()
        {
                // Logic
        }

        public bool ChkEven()
        {
                // Logic
        }

        public int DisplayFactors()
        {
                // Logic
        }

        public int SumFactors()
        {
                // Logic
        }
}


class Marvellous
{
        public static void Main(String []arg)
        {
                NumOperation obj1 = new NumOperation(25);

                // Logic
        }
}
```

2. Write C# application which contains following things as :

Design one class named as NumberActivity which derives above class i.e. NumOperation.

Class should contains following behaviours as

1. ChkPrime() - Check whether number is prime or not.
2. ChkPerfect() - Check whether number is perfect or not.
3. Factorial()- Return factorial of elements.

Example :

```
class NumberActivity : NumOperation
{
        public bool ChkPrime()
        {
                // Logic
        }
```

```
        public bool ChkPerfect()
        {
                // Logic
        }

        public int Factorial()
        {
                // Logic
        }
}
```