Day 5
C#

1) Practice
ElectricReading.cs

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

namespace SimplePrograms
{
   internal class ElectricReading
   {
      private int consumernumber,previousreading,currentreading;
      private string consumername,consumertype;

      public ElectricReading(int consumernumber, string consumername, int
currentreading, int previousreading, string consumertype)
      {
         Consumernumber = consumernumber;
         Previousreading = previousreading;
         Currentreading = currentreading;
         Consumername = consumername;
         Consumertype = consumertype;
      }

      public int Consumernumber { get => consumernumber; set =>
consumernumber = value; }
      public int Previousreading { get => previousreading; set =>
previousreading = value; }
      public int Currentreading { get => currentreading; set => currentreading =
value; }
      public string Consumername { get => consumername; set =>
consumername = value; }
      public string Consumertype { get => consumertype; set => consumertype
= value; }

      public int CalculateBill()
      {
         int consumption = Currentreading - Previousreading;
```

```
    int billamt = 0;
    if (Consumertype.Equals("Domestic"))
    {

        if (consumption <= 100)
        {
            billamt = 0;
        }
        else if (consumption>100 && consumption<=200){
            billamt = (consumption - 100) * 2;
        }
        else if (consumption > 200 && consumption <= 500)
        {
            billamt = (consumption - 100) * 5;
        }
        else if (consumption > 500)
        {
            billamt = (consumption - 100) * 10;
        }


    }
    else if (Consumertype.Equals("Commercial"))
    {
        if (consumption <= 100)
        {
            billamt = 10;
        }
        else if (consumption > 100 && consumption <= 200)
        {
            billamt = (consumption - 100) * 20;
        }
        else if (consumption > 200 && consumption <= 500)
        {
            billamt = (consumption - 100) * 50;
        }
        else if (consumption > 500)
        {
            billamt = (consumption - 100) * 100;
        }
    }
    return billamt;
}
```

```
      /* public void DisplayBillDetails()
       {
          int billamt=CalculateBill();
          Console.WriteLine("Bill : " + consumernumber + " " + consumername +
" " + billamt);
       }*/

   }
}
```

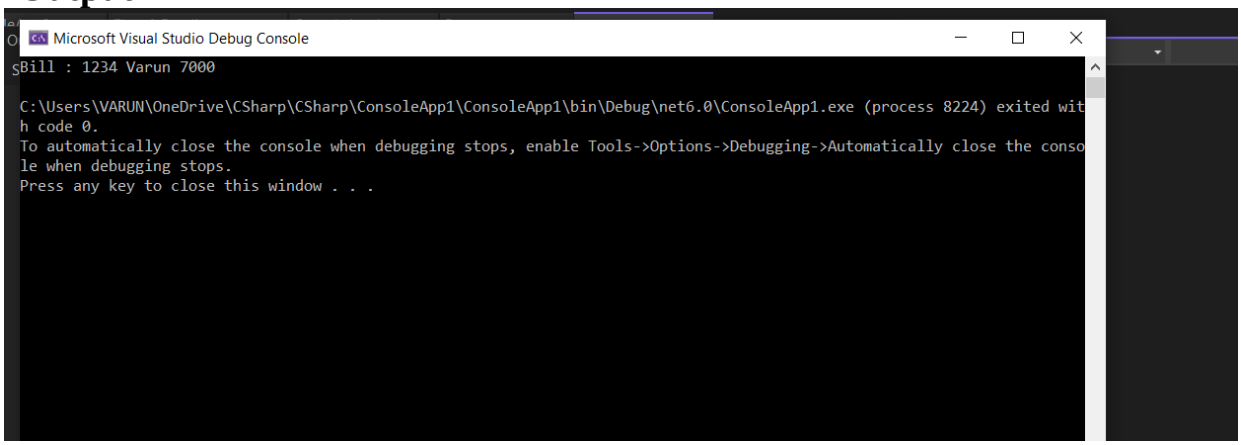
Program.cs

```
using SimplePrograms;

/*int consumernumber = Convert.ToInt32(Console.ReadLine());
string? consumername = Console.ReadLine();
int currentreading = Convert.ToInt32(Console.ReadLine());
int previousreading = Convert.ToInt32(Console.ReadLine());
string? consumertype = Console.ReadLine();*/

ElectricReading electricReading = new ElectricReading(1234, "Varun", 2000,
1200, "Domestic");
int billamt = electricReading.CalculateBill();
Console.WriteLine($"Bill : " + $"{electricReading.Consumernumber}
{electricReading.Consumername} {billamt}");
```

Output



2)Assignment 1

Program.cs

BankAccount.cs

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

namespace ConsoleApp1
{
    internal class BankBook
    {

        private readonly int acc_number;
        private int balance;
        private string acc_holder_name;

        public BankBook(string acc_holder_name)
        {
            acc_number = 123456;
            Acc_holder_name = acc_holder_name;
            Balance = 0;

        }

        public int Acc_number => acc_number;

        public int Balance { get => balance; set => balance = value; }
        public string Acc_holder_name { get => acc_holder_name; set =>
acc_holder_name = value; }

        public void Deposit(int dep_amount)
        {
            if (dep_amount <= 0)
            {
                Console.WriteLine("inadequate amount");
            }
            else
            {
                Balance = dep_amount + Balance;
```

```
            }
        }
        public void Withdraw(int withdraw_Amount)
        {
            if (withdraw_Amount <= 0)
            {
                Console.WriteLine("Zero balance");
            }
            else if (Balance >= withdraw_Amount)
            {
                Balance = Balance - withdraw_Amount;
            }
            else
            {
                Console.WriteLine("please enter amount correctly");
            }
        }
        public void Display()
        {
            Console.WriteLine("Acc Number : " + Acc_number);
            Console.WriteLine("Acc Holder Name:" + Acc_holder_name);
            Console.WriteLine("Balance:" + Balance);

        }
    }
}


Console.WriteLine("acc_name : ");
string acc_holder_name = Console.ReadLine();
Console.WriteLine("dep_amount ");
int Deposit_Amount = Convert.ToInt32(Console.ReadLine());
Console.WriteLine("withdraw amount :  ");
int withdraw_Amount = Convert.ToInt32(Console.ReadLine());


BankBook bankAccount = new BankBook(acc_holder_name);

bankAccount.Deposit(Deposit_Amount);
```

```
bankAccount.Withdraw(withdraw_Amount);
bankAccount.Display();
Console.ReadLine();
```

Output