18. Electricity bill perparation

1. create a front end and back-end program for an Electricity bill preparation system

FRONTEND :

FRONTEND CODING

```vb
Imports System.Data.SqlClient
Imports System.Diagnostics.Eventing.Reader
Imports System.Windows.Forms.VisualStyles.VisualStyleElement

Public Class Form1
    Dim conn As New SqlConnection
    Dim cmd As New SqlCommand
    Dim dt As DataTable
    Dim dr As SqlDataReader
    Dim da As New SqlDataAdapter
    Dim str As String

    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        'TODO: This line of code loads data into the 'BillDataSet.EB' table. You
can move, or remove it, as needed.
        Me.EBTableAdapter.Fill(Me.BillDataSet.EB)
        conn.ConnectionString = "Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=D:\folders\MIS\Electricity_bill\El
ectricity_bill\bill.mdf;Integrated Security=True"
    End Sub

    Private Overloads Sub Insert(sender As Object, e As EventArgs) Handles
Button1.Click
        conn.Open()
        str = "insert into EB values('" & TextBox1.Text & "','" & TextBox2.Text &
"' ,'" & TextBox3.Text & "', '" & RichTextBox1.Text & "','" & TextBox4.Text &
"','" & TextBox5.Text & "','" & TextBox6.Text & "','" & TextBox7.Text & "')"
```

```vbnet
        cmd = New SqlCommand(str, conn)
        If cmd.ExecuteNonQuery() Then
            MsgBox("successfully inserted")
        End If
        cmd.Dispose()
        conn.Close()
    End Sub

    Private Overloads Sub Update(sender As Object, e As EventArgs) Handles
Button2.Click
        conn.Open()
        str = "update EB set Consumer_no='" & TextBox1.Text & "', Name='" &
TextBox2.Text & "' , Phone_no='" & TextBox3.Text & "', Address='" &
RichTextBox1.Text & "',Current_reading='" & TextBox4.Text &
"',Previous_reading='" & TextBox5.Text & "',Total_reading='" & TextBox6.Text &
"',Amount='" & TextBox7.Text & "' where Consumer_no =" & TextBox1.Text & ""
        cmd = New SqlCommand(str, conn)
        If cmd.ExecuteNonQuery() Then
            MsgBox("successfully updated")
        End If
        cmd.Dispose()
        conn.Close()
    End Sub

    Private Overloads Sub Delete(sender As Object, e As EventArgs) Handles
Button3.Click
        conn.Open()
        str = "Delete from EB where Consumer_no ='" & TextBox1.Text & "'"
        cmd = New SqlCommand(str, conn)
        If cmd.ExecuteNonQuery() Then
            MsgBox("successfully deleted")
        End If
        cmd.Dispose()
        conn.Close()
    End Sub

    Private Overloads Sub View(sender As Object, e As EventArgs) Handles
Button4.Click
        DataGridView1.Visible = True
        conn.Open()
        str = "select * from EB"
        cmd = New SqlCommand(str, conn)
        da = New SqlDataAdapter(cmd)
        dt = New DataTable
        da.Fill(dt)
        DataGridView1.DataSource = dt
        cmd.Dispose()
        conn.Close()
    End Sub

    Private Sub Calculate(sender As Object, e As EventArgs) Handles Button5.Click
        TextBox6.Text = Val(TextBox4.Text) - Val(TextBox5.Text)
        If TextBox6.Text < 100 Then
            TextBox7.Text = Val(TextBox6.Text) * 2
        ElseIf TextBox6.Text > 101 And TextBox6.Text < 500 Then
            TextBox7.Text = Val(TextBox6.Text) * 4
        ElseIf TextBox6.Text > 501 And TextBox6.Text < 1000 Then
            TextBox7.Text = Val(TextBox6.Text) * 6
        ElseIf TextBox6.Text > 1001 Then
            TextBox7.Text = Val(TextBox6.Text) * 8
        Else
            TextBox7.Text = "invalid"
        End If
```

```
        End Sub
End Class
```

BACKEND DATABASE SET :

	Name	Data Type	Allow Nulls	Default
🔑	Customer_no	int	☐	
	Name	nchar(10)	☐	
	Phone_no	int	☐	
	Address	nchar(10)	☐	
	Current_reading	float	☐	
	Previous_reading	float	☐	
	Total_reading	float	☐	
	Amount	float	☐	
			☐	

BACKEND SQL CODING :

```sql
CREATE TABLE [dbo].[EB] (
    [Customer_no]      INT        NOT NULL,
    [Name]             NCHAR (10) NOT NULL,
    [Phone_no]         INT        NOT NULL,
    [Address]          NCHAR (10) NOT NULL,
    [Current_reading]  FLOAT (53) NOT NULL,
    [Previous_reading] FLOAT (53) NOT NULL,
    [Total_reading]    FLOAT (53) NOT NULL,
    [Amount]           FLOAT (53) NOT NULL,
    PRIMARY KEY CLUSTERED ([Customer_no] ASC)
);
```