

HW6 Code

```
Imports System.Data
Imports System.Data.SqlClient
Partial Class HW5
    Inherits System.Web.UI.Page

    Public Shared Con As New SqlConnection("Data Source=cb-ot-devst04.ad.wsu.edu;Initial
Catalog=MF81zachary.seibel;Persist Security Info=True;User
ID=zachary.seibel;Password=c961a639")
    Protected Sub btnCalculate_Click(sender As Object, e As EventArgs) Handles
btnCalculate.Click
        Page.MaintainScrollPositionOnPostBack = True

        'Basic local variables calculating costs of miles traveled, days rented, and
total cost
        Dim decTotalCost As Decimal
        Dim decTotalMiles As Decimal
        Dim decMilesCost As Decimal
        Dim decDaysRentedCost As Decimal
        'Timespan variable
        Dim ts As TimeSpan

        'Error checking for vehicle selected
        If rblVehicle.SelectedIndex = -1 Then
            tbOutput.Text = "Please select the vehicle rented."
            Exit Sub
        End If

        'Error checking for start miles
        If IsNumeric(tbStart.Text) = False OrElse CDec(tbStart.Text) < 0 Then
            tbOutput.Text = "Please enter a numeric starting mileage."
            Exit Sub
        End If

        'Error checking for end miles
        If IsNumeric(tbEnd.Text) = False OrElse CDec(tbEnd.Text) <= 0 Then
            tbOutput.Text = "Please enter an ending mileage greater than 0."
            Exit Sub
        End If

        'Error checking to make sure end mileage is greater than start mileage
        If CDec(tbStart.Text) > CDec(tbEnd.Text) OrElse CDec(tbStart.Text) =
CDec(tbEnd.Text) Then
            tbOutput.Text = "Please ensure start mileage is less than end mileage."
            Exit Sub
        End If

        'Calendar error checking for selected dates
        If calRented.SelectedDate = Nothing OrElse calReturned.SelectedDate = Nothing
Then
            tbOutput.Text = "Please select both start and end dates."
            Exit Sub
        End If

        'Calendar error checking to make sure rent date is before return date
        If calRented.SelectedDate >= calReturned.SelectedDate Then
```

```

        tbOutput.Text = "Please ensure rent date is prior to return date."
    Exit Sub
End If

'Error checking to make sure damage selection is made
If ddlDamage.SelectedIndex = 0 OrElse ddlDamage.SelectedIndex = -1 Then
    tbOutput.Text = "Please select a level of damage."
    Exit Sub
End If

'Defining total miles variable
decTotalMiles = CDec(tbEnd.Text) - CDec(tbStart.Text)

'Defining cost of mileage variable
decMilesCost = decTotalMiles * 0.25

'Defining timespan function
ts = calReturned.SelectedDate - calRented.SelectedDate

'Defining cost of rented days variable
decDaysRentedCost = (rblVehicle.SelectedValue * ts.TotalDays)

'Defining total cost variable
decTotalCost = decDaysRentedCost + decMilesCost + ddlDamage.SelectedValue

'Textbox output response
tbTotalCost.Text = decTotalCost.ToString("C2")
tbDayRentedCost.Text &= vbNewLine & vbNewLine & decDaysRentedCost.ToString("C2")
tbMileageCost.Text &= vbNewLine & vbNewLine & decMilesCost.ToString("C2")
tbCostForDamage.Text &= vbNewLine & vbNewLine & ddlDamage.SelectedValue

End Sub

Protected Sub btnClear_Click(sender As Object, e As EventArgs) Handles btnClear.Click
    'Clear button functions
    rblVehicle.SelectedIndex = -1
    tbStart.Text = Nothing
    tbEnd.Text = Nothing
    calRented.SelectedDate = Nothing
    calReturned.SelectedDate = Nothing
    ddlDamage.SelectedIndex = 0
    tbOutput.Text = Nothing
End Sub

Protected Sub btnInsert_Click(sender As Object, e As EventArgs) Handles
btnInsert.Click
    Dim cmdInsertReservations As New SqlCommand("INSERT Reservation
(Cost_For_Days_Rented, Cost_For_Mileage, Cost_For_Damage, Total_Cost) Values (@p1, @p2,
@p3, @p4)", Con)

    With cmdInsertReservations.Parameters
        .Clear()
        .AddWithValue("@p1", CDec(tbDayRentedCost.Text))
        .AddWithValue("@p2", CDec(tbMileageCost.Text))
    End With

```

```

        .AddWithValue("@p3", CDec(tbCostForDamage.Text))
        .AddWithValue("@p4", CDec(tbTotalCost.Text))

    End With

    Try
        'this is where the command is run to insert the row of data
        If Con.State = ConnectionState.Closed Then Con.Open()
        cmdInsertReservations.ExecuteNonQuery()
        Response.Write("record added")
        Call GetReservations()
    Catch ex As Exception
        Response.Write(ex.Message)
    Finally
        Con.Close()
    End Try
End Sub

Private Sub GetReservations()
    'here we retrieve the full table of customers

    Dim ReservationsDA As New SqlDataAdapter("SELECT TOP 1 * FROM Reservation ORDER
by ReservationID DESC", Con)
    Dim ReservationsTable As New DataTable
    ' Dim NewestReservation As New SqlDataAdapter("SELECT TOP 1 * from
[seibel].[Reservation] ORDER by ReservationsID DESC", Con)

    'if the table was already filled once then clear it out
    If ReservationsTable.Rows.Count > 0 Then
        ReservationsTable.Rows.Clear()
    End If

    Try
        'so here the red sql statement gets run over the connection tot the database
the retrieved data is the displayed
        ReservationsDA.Fill(ReservationsTable)
        GridView1.DataSource = ReservationsTable
        GridView1.DataBind()

        Catch ex As Exception
            Response.Write(ex.Message)
        End Try
    End Sub

End Class

```