

```
1 'Nathan Carlson
2 'RCET0265
3 'asg4-6
4 'Rental Form
5 'https://github.com/carlnathF19/NDC-VS-F19/tree/master/assignments/Asg4-6
6
7 Option Strict On
8 Option Explicit On
9
10 Public Class rentalForm
11     Dim cost As Double = 0
12     Dim miles As Double
13     Dim days As Integer
14     Dim totalMiles As Double '= 0
15     Dim totalMileCharge As Double '= 0
16     Dim totalDays As Integer '= 0
17     Dim totalDayCharge As Double '= 0
18     Dim totalDiscount As Double '= 0
19     Dim totalPaid As Double '= 0
20     Dim customers As Integer '= 0
21
22     Dim validValues As Boolean
23     'This Sub sets the label text color to a visible color if validation returns true, to same as
24     'background color if false, so previous values aren't visible.
25     Private Sub SetLabelForeColor()
26         If validValues Then
27             milesDrivenLabel.ForeColor = Color.White
28             mileChargeLabel.ForeColor = Color.White
29             dayChargeLabel.ForeColor = Color.White
30             discountLabel.ForeColor = Color.White
31             youOweLabel.ForeColor = Color.White
32         Else
33             milesDrivenLabel.ForeColor = Color.Black
34             mileChargeLabel.ForeColor = Color.Black
35             dayChargeLabel.ForeColor = Color.Black
36             discountLabel.ForeColor = Color.Black
37             youOweLabel.ForeColor = Color.Black
38         End If
39
40     End Sub
41     'This subroutine checks to see if Discounts apply and modify the global variable cost.
42     'Cost and miles each update a global variable passed to the summary subroutine.
43     Private Sub checkDiscount()
44         Dim discountMultiplier As Double = 0
45         Dim discountAmount As Double = 0
46         If aaaCheckBox.Checked Then
```

```
47         discountMultiplier += 0.05
48     End If
49     If seniorCheckBox.Checked Then
50         discountMultiplier += 0.03
51     End If
52     discountAmount = discountMultiplier * cost
53     'Adds to totals summary
54     totalDiscount += discountAmount
55     discountLabel.Text = discountAmount.ToString("C")
56     cost = (1 - discountMultiplier) * cost
57 End Sub
58 'This subroutine checks each value, starting with the last in the Tab Index ↗
59 'and ending with the
60 'first, if any returns false, then the last one to return false takes ↗
61 'focus. Also uses tries
62 'to see if values parse correctly.
63 Private Sub ValidateFields()
64     Dim summaryLocal As String
65     Dim zipCodeValid As Boolean = True
66
67     summaryLocal = ""
68     validValues = False
69     Try
70         days = Integer.Parse(daysTextBox.Text)
71         If days > 45 Or days < 1 Then
72             MessageBox.Show("Please enter valid number of days, (0 to 45)")
73             daysTextBox.Select()
74         End If
75     Catch ex As Exception
76         MessageBox.Show("Please enter valid number of days, (0 to 45)")
77         daysTextBox.Select()
78     End Try
79
80     Try
81         miles = Double.Parse(endOdometerTextBox.Text) - Double.Parse ↗
82             (beginOdometerTextBox.Text)
83     Catch ex As Exception
84         MessageBox.Show("Please enter valid odometer values")
85         beginOdometerTextBox.Select()
86     End Try
87
88     Try
89         Integer.Parse(zipCodeTextBox.Text)
90     Catch ex As Exception
91         zipCodeValid = False
92     End Try
93
94     If zipCodeTextBox.Text.Length() <> 5 Or zipCodeValid = False Then
95         summaryLocal = "Please enter a 5 digit zip code." & vbNewLine & ↗
```

```
summaryLocal
93     zipCodeTextBox.Select()
94 End If
95
96 If stateTextBox.Text.Length() <> 2 Then
97     summaryLocal = "Please enter a state abbreviation." & vbNewLine &
summaryLocal
98     stateTextBox.Select()
99 End If
100
101 If cityTextBox.Text.Length() = 0 Then
102     summaryLocal = "Please enter a City." & vbNewLine & summaryLocal
103     cityTextBox.Select()
104 End If
105
106 If addressTextBox.Text.Length() = 0 Then
107     summaryLocal = "Please enter an Address." & vbNewLine &
summaryLocal
108     addressTextBox.Select()
109 End If
110
111 If nameTextBox.Text.Length() = 0 Then
112     summaryLocal = "Please enter a Name." & vbNewLine & summaryLocal
113     nameTextBox.Select()
114 End If
115
116 If summaryLocal <> "" Then
117     MessageBox.Show(summaryLocal)
118 Else validValues = True
119 End If
120 End Sub
121 'This subroutine exits if a yes no message box receives a yes click
122 Private Sub ExitButton_Click(sender As Object, e As EventArgs) Handles
exitButton.Click
123     If MessageBox.Show("Do you want to exit?", "Exit?",
MessageBoxButtons.YesNo) _
124         = Windows.Forms.DialogResult.Yes Then
125         Me.Close()
126     End If
127
128 End Sub
129
130 Private Sub CalculateButton_Click(sender As Object, e As EventArgs) Handles
calculateButton.Click
131     Dim dayCharge As Integer
132     ValidateFields()
133     SetLabelForeColor()
134
135     If validValues Then
```

```
136         If miles > 0 Then
137
138         Else
139             MessageBox.Show("Please enter valid odometer values, end
140                             reading must be " &
141                             "greater than beginning reading.")
142             beginOdometerTextBox.Select()
143         End If
144
145         'If kilometers is checked then the same variable is used to convert
146         to miles, the reading in km is stored
147         'in the miles button and converted via *= .62 to miles
148         If kilometersRadioButton.Checked = True Then
149             miles *= 0.62
150         End If
151
152         milesDrivenLabel.Text = Convert.ToString(miles)
153         'Adds the total miles to the summary sub through global variable.
154         totalMiles += miles
155
156         If miles >= 201 And miles <= 500 Then
157             cost = (miles - 200) * 0.12
158         ElseIf miles > 500 Then
159             'If miles > 500 then 300 of those miles (500-200) are at
160             the .12 price
161             cost = (miles - 500) * 0.1 + (300 * 0.12)
162         End If
163
164         totalMileCharge += cost
165
166         'cost += days*15 did not work, so a local variable is declared, and
167         then passed
168         'to cost. Programattically, cost += should work, but it didn't?
169
170         Dim costLocal As Double = cost
171
172         mileChargeLabel.Text = cost.ToString("C")
173         dayCharge = days * 15
174         'Adds to totals summary
175         totalDays += days
176         totalDayCharge += dayCharge
177         costLocal = cost + dayCharge
178         cost = costLocal
179         dayChargeLabel.Text = dayCharge.ToString("C")
180
181         checkDiscount()
182         'Adds to totals summary
183         totalPaid += cost
184         customers += 1
```

```
181         youOweLabel.Text = cost.ToString("C")
182         Console.WriteLine(cost)
183         summaryButton.Enabled = True
184     End If
185
186 End Sub
187
188 Private Sub SummaryButton_Click(sender As Object, e As EventArgs) Handles summaryButton.Click
189     Dim totalsSummary As String
190     totalsSummary = "Distance Driven = " & totalMiles & vbNewLine &
191         "Mileage Charge = " & totalMileCharge.ToString("C") & vbNewLine &
192         "Total Days Rented = " & totalDays & vbNewLine &
193         "Total Day Charge = " & totalDayCharge.ToString("C") & vbNewLine &
194         "Total Discounts = " & totalDiscount.ToString("C") & vbNewLine &
195         "Total Paid = " & totalPaid.ToString("C") & vbNewLine &
196         "Customers = " & customers
197     MessageBox.Show(totalsSummary)
198
199 End Sub
200
201 Private Sub RentalForm_Load(sender As Object, e As EventArgs) Handles MyBase.Load
202     summaryButton.Enabled = False
203 End Sub
204
205 Private Sub ClearButton_Click(sender As Object, e As EventArgs) Handles clearButton.Click
206     nameTextBox.Clear()
207     addressTextBox.Clear()
208     cityTextBox.Clear()
209     stateTextBox.Clear()
210     zipCodeTextBox.Clear()
211     beginOdometerTextBox.Clear()
212     endOdometerTextBox.Clear()
213     daysTextBox.Clear()
214 End Sub
215
216 End Class
217
```