

```

        End Get
    End Property
End Class    'CashRegister

Class FastTrackRegister
    Inherits CashRegister

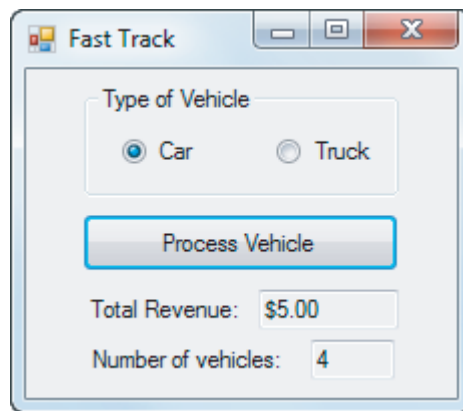
    Private m_count As Integer

    Public ReadOnly Property Count() As Integer
        Get
            Return m_count
        End Get
    End Property

    Sub ProcessCar()
        m_count += 1    'Process a car: $1.00
        Deposit(1)
    End Sub

    Sub ProcessTruck()
        m_count += 1    'Process a truck: $2.00
        Deposit(2)
    End Sub
End Class    'FastTrackRegister

```



## CHAPTER 12

### EXERCISES 12.1

1. Protected Sub btnFind\_Click(...) Handles btnFind.Click
 

```

          Dim max As Double = CDb1(txtFirstNum.Text)
          If CDb1(txtSecondNum.Text) > max Then
              max = CDb1(txtSecondNum.Text)
          End If
          If CDb1(txtThirdNum.Text) > max Then
              max = CDb1(txtThirdNum.Text)
          End If
          txtLargestNum.Text = CStr(max)
      End Sub
      
```



3. Protected Sub btnDisplay\_Click(...) Handles btnDisplay.Click  
 Dim num As Integer = CInt(txtNumber.Text)  
 For i As Integer = 1 To num - 1  
 lstOutput.Items.Add(i & " potato")  
 Next  
 lstOutput.Items.Add(num)  
 End Sub
5. Protected Sub btnCalculate\_Click(...) Handles btnCalculate.Click  
 Dim purchPrice As Double = CDb1(txtPurPrice.Text)  
 Dim redemValue As Double = CDb1(txtRedValue.Text)  
 txtDiscRate.Text = FormatPercent((redemValue - purchPrice) / redemValue)  
 txtIntRate.Text = FormatPercent((redemValue - purchPrice) / purchPrice)  
 End Sub

## EXERCISES 12.2

1. Protected Sub Page\_Load(...) Handles Me.Load  
 If Not IsPostBack Then  
 Dim states() As String =  
 IO.File.ReadAllLines(MapPath("App\_Data\States.txt"))  
 Dim query = From state In states  
 Order By state  
 Select state  
 lstStates.DataSource = query  
 lstStates.DataBind()  
 txtNumStates.Text = CStr(lstStates.Items.Count)  
 End If  
 End Sub
- Protected Sub btnDelete\_Click(...) Handles btnDelete.Click  
 lstStates.Items.Remove(lstStates.Text)  
 txtNumStates.Text = CStr(lstStates.Items.Count)  
 End Sub

## 12.2 Exercise 3

Model

☐ Deluxe ☐ Super

☒ Upgraded Video Card

☐ Internal Modem + Wi-Fi

☐ 1 GB additional memory

Total cost:

**You must select a model!**

3. 'Note: The AutoPostBack and CausesValidation properties were  
 'set to True for each check box.  
 Protected Sub rblModel\_SelectedIndexChanged(...) Handles \_  
 rblModel.SelectedIndexChanged, chkUpgradedVideo.CheckedChanged,  
 chkModem.CheckedChanged, chkMemory.CheckedChanged  
 Dim cost As Double = 0  
 'Add amounts to the cost based upon selections.  
 If rblModel.Text = "Deluxe" Then  
 cost += 1000

```

Else
    cost += 1500
End If
If chkUpgradedVideo.Checked Then
    cost += 200
End If
If chkModem.Checked Then
    cost += 30
End If
If chkMemory.Checked Then
    cost += 120
End If
txtOutput.Text = FormatCurrency(cost)
End Sub

```

9. 'In RangeValidator controls, Minimum is "a" and Maximum is "zzzzzzz".

```

Protected Sub btnAlphabetize_Click(...) Handles btnAlphabetize.Click
    Dim word1 = txtWord1.Text
    Dim word2 = txtWord2.Text
    DisplayWords(word1, word2)
End Sub

```

```

Sub DisplayWords(ByVal word1 As String, ByVal word2 As String)
    lstOutput.Items.Clear()
    If word1 <= word2 Then
        lstOutput.Items.Add(word1)
        lstOutput.Items.Add(word2)
    Else
        lstOutput.Items.Add(word2)
        lstOutput.Items.Add(word1)
    End If
End Sub

```

12.2 Exercise 9	
First word:	<input type="text"/>
Second word:	<input type="text"/>
<input type="button" value="Alphabetize Words"/>	
<input type="button" value="Unbound"/>	
<p><b>First word missing!    Second word missing!</b></p> <p><b>First letter of first word is not lower-case!    First letter of second word is not lower-case!</b></p>	

11. Protected Sub btnDisplay\_Click(...) Handles btnDisplay.Click
- ```

Dim justices() As String =
    IO.File.ReadAllLines(MapPath("App_Data\Justices.txt"))
Dim stateAbbrev As String = txtState.Text.ToUpper
Dim query = From justice In justices
    Let data = justice.Split(",")
    Let name = data(0) & " " & data(1)
    Let pres = data(2)
    Let state = data(3)
    Let yrsServed = YearsServed(data(4), data(5))
    Where state = stateAbbrev

```

```

        Select name, pres, yrsServed
    grvJustices.DataSource = query
    grvJustices.DataBind()
    grvJustices.HeaderRow.Cells(0).Text = "Justice"
    grvJustices.HeaderRow.Cells(1).Text = "Appointing President"
    grvJustices.HeaderRow.Cells(2).Text = "Years Served"
End Sub

Function YearsServed(ByVal enter As Double,
                    ByVal leave As Double) As Double
    If leave = 0 Then
        Return (Now.Year - enter)
    Else
        Return (leave - enter)
    End If
End Function

```

## 12.2 Exercise 11

State abbreviation:

| Justice        | Appointing President | Years Served |
|----------------|----------------------|--------------|
| Samuel Chase   | George Washington    | 15           |
| Gabriel Duval  | James Madison        | 25           |
| Thomas Johnson | George Washington    | 2            |
| John Roberts   | George W. Bush       | 5            |
| Roger Taney    | Andrew Jackson       | 28           |

### EXERCISES 12.3

7. Protected Sub btnDisplay\_Click(...) Handles btnDisplay.Click
 

```

Dim mcDC As New MegacitiesDataContext
Dim query = From city In mcDC.Cities
             Join country In mcDC.Countries On city.country Equals
             country.name
             Select city.name, country.pop2010
chtMegacities.DataBindTable(query, "name")
chtMegacities.ChartAreas(0).AxisX.Interval = 1
chtMegacities.ChartAreas(0).AxisX.Title = "City"
chtMegacities.ChartAreas(0).AxisY.Title =
"2010 Pop. of Country in Millions"
      
```
17. Protected Sub btnDisplay\_Click(...) Handles btnDisplay.Click
 

```

Dim pizzaDC As New PizzaDataContext
Dim query = From chain In pizzaDC.Pizzerias
             Order By chain.sales2008 Descending
             Let amount = chain.sales2008 / 1000000
             Select chain.name, amount
chtPizzaChains.DataBindTable(query, "name")
chtPizzaChains.ChartAreas(0).AxisX.Interval = 1
chtPizzaChains.ChartAreas(0).AxisX.Title = "Pizzeria"
chtPizzaChains.ChartAreas(0).AxisY.Title = "2008 Sales in Billions"
      
```

```
25. Protected Sub btnDisplay_Click(...) Handles btnDisplay.Click
    Dim pizzaDC As New PizzaDataContext
    Dim query = From chain In pizzaDC.Pizzerias
                Select chain.numStores2008
    txtOutput.Text = FormatNumber(query.Sum, 0)
End Sub
```

### 12.3 Exercise 25

Display Total Number of Stores

Total number of stores: 20,945