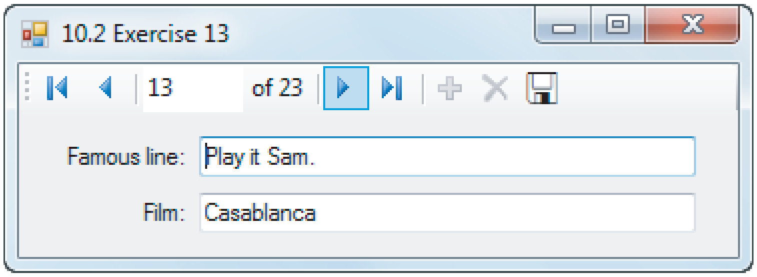
13.
```
Private Sub frmMovies_Load(...) Handles MyBase.Load
  Me.LinesTableAdapter.Fill(Me.MoviesDataSet.Lines)
End Sub

Private Sub btnUpdate_Click(...) Handles btnUpdate.Click
  'These two lines update the database file on the hard drive.
  BindingSource1.EndEdit()
  LinesTableAdapter.Update(MoviesDataSet.Lines)
End Sub
```



Famous line: Play it Sam.

Film: Casablanca

17. Apollo 13 does not appear in the key field of the Actors table. Thus the Rule of Referential Integrity would be violated.

19. Replace the table with two tables. The first table should contain the fields *name*, *address*, and *city*. The second table should contain the fields *city*, *state*, and *stateCapital*.

Chapter 11

Exercises 11.1

1. Any negative grade will be recorded as 0 and any grade greater than 100 will be recorded as 100.

3. Remove the keyword `WriteOnly` from the `Midterm` property block and add the following Get property procedure to it:

```
Get
  Return m_midterm
End Get
```

5. The properties `Midterm` and `Final` are write only.

7. The property `SocSecNum` is initialized to the value `999-99-9999`.

9. The keyword `New` is missing from the third line of the event procedure.

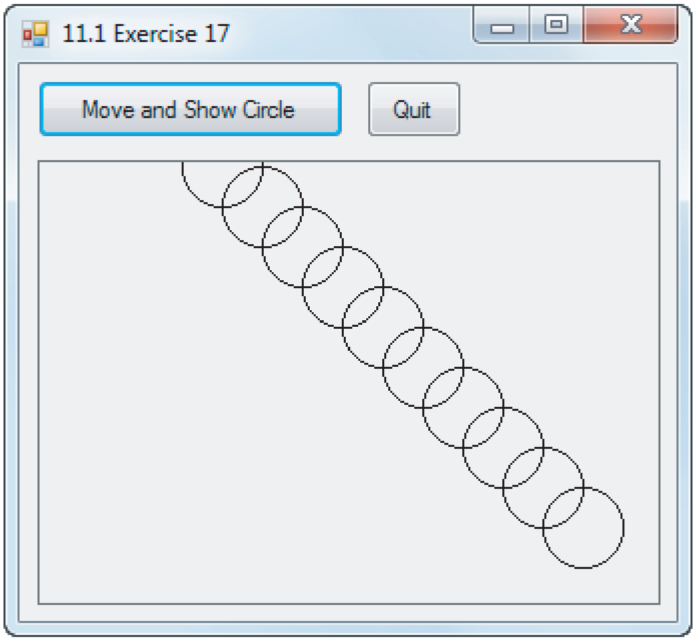
11. The statement `nom = m_name` is not valid. *m_name* would need to be Public and referred to by *scholar.m_name*.

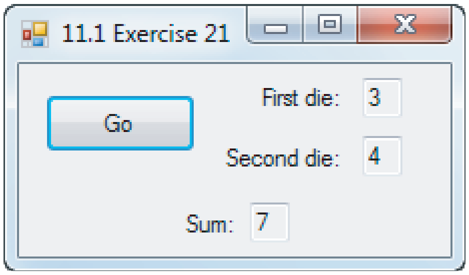
13. The statements `pupil.Midterm = scholar.Midterm` and `lstGrades.Items.Add(pupil.Midterm)` are not valid. The Midterm property is write only; it can be set, but cannot return a value.

15.
```
Country: Canada
Capital: Ottawa
Pop: 31 million
```

17. Change 20 to –20 in the btnMove_Click event procedure, and create a `frmCircle_Load` event procedure with the following lines,

```
round.Xcoord = picCircle.Width – 40
round.Ycoord = picCircle.Height – 40
```





First die: 3

Second die: 4

Sum: 7

21.

```
Class PairOfDice
  Private m_die1, m_die2 As Integer
  Dim randomNum As New Random()

  Public ReadOnly Property Die1() As Integer
    Get
      Return m_die1
    End Get
  End Property

  Public ReadOnly Property Die2() As Integer
    Get
      Return m_die2
    End Get
  End Property

  Public ReadOnly Property SumOfFaces() As Integer
    Get
      Return m_die1 + m_die2
    End Get
  End Property
  Sub Roll()
    m_die1 = randomNum.Next(1, 7)
    m_die2 = randomNum.Next(1, 7)
  End Sub
End Class   'PairOfDice
```

25.

```
Public Class frmCashRegister
  Dim register As New CashRegister()

  Private Sub btnAdd_Click(...) Handles btnAdd.Click
    'Add an amount to the balance.
    register.Add(CDbl(txtAmount.Text))
    txtBalance.Text = FormatCurrency(register.Balance)
    txtAmount.Clear()
    txtAmount.Focus()
  End Sub

  Private Sub btnSubtract_Click(...) Handles btnSubtract.Click
    'Subtract an amount from the balance.
    register.Subtract(CDbl(txtAmount.Text))
    txtBalance.Text = FormatCurrency(register.Balance)
    txtAmount.Clear()
    txtAmount.Focus()
  End Sub
End Class   'frmCashRegister

Class CashRegister
  Private m_balance As Double

  Public ReadOnly Property Balance() As Double
    Get
      Return m_balance
    End Get
  End Property

  Sub Add(ByVal amount As Double)
    If (m_balance + amount) >= 0 Then  'Ensure balance stays nonnegative
      m_balance += amount
    End If
  End Sub

  Sub Subtract(ByVal amount As Double)
    If (m_balance — amount) >= 0 Then  'Ensure balance stays nonnegative
      m_balance = m_balance — amount
    End If
  End Sub
End Class    'CashRegister
```

Exercises 11.2

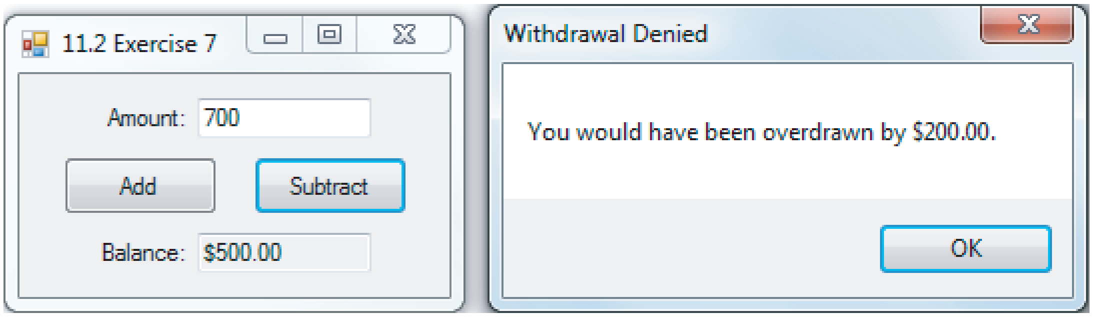
1.

```
Sub btnDisplay_Click(...) Handles btnDisplay.Click
  ReDim Preserve students(lastStudentAdded)
  Dim query = From pupil In students
              Let name = pupil.Name
              Let ssn = pupil.SocSecNum
              Let semGrade = pupil.CalcSemGrade
              Where semGrade = "A"
              Select pupil.Name, pupil.SocSecNum, pupil.CalcSemGrade
```

```
    dgvGrades.DataSource = query.ToList
    dgvGrades.CurrentCell = Nothing
    dgvGrades.Columns("Name").HeaderText = "Student Name"
    dgvGrades.Columns("SocSecNum").HeaderText = "SSN"
    dgvGrades.Columns("CalcSemGrade").HeaderText = "Grade"
    ReDim Preserve students(50)
    txtName.Focus()
  End Sub
```

7.

```
Public Class frmCashRegister
  Dim WithEvents register As New CashRegister()

  Private Sub btnAdd_Click(...) Handles btnAdd.Click
    'Add an amount to the balance.
    register.Add(CDbl(txtAmount.Text))
    txtBalance.Text = FormatCurrency(register.Balance)
    txtAmount.Clear()
    txtAmount.Focus()
  End Sub

  Private Sub btnSubtract_Click(...) Handles btnSubtract.Click
    'Subtract an amount from the balance.
    register.Subtract(CDbl(txtAmount.Text))
    txtBalance.Text = FormatCurrency(register.Balance)
    txtAmount.Clear()
    txtAmount.Focus()
  End Sub

  Private Sub AttemptToOverdraw(ByVal amt As Double) Handles _
                                        register.AttemptToOverdraw
    MessageBox.Show("You would have been overdrawn by " &
                FormatCurrency(amt) & ".", "Withdrawal Denied")
  End Sub
End Class   'frmCashRegister

Class CashRegister
  Private m_balance As Double

  Public Event AttemptToOverdraw(ByVal amt As Double)

  Public ReadOnly Property Balance() As Double
    Get
      Return m_balance
    End Get
  End Property

  Sub Add(ByVal amount As Double)
    If (m_balance + amount) >= 0 Then  'Ensure balance stays nonnegative
      m_balance += amount
     Else
       RaiseEvent AttemptToOverdraw(amount + m_balance)
    End If
  End Sub

  Sub Subtract(ByVal amount As Double)
    If (m_balance – amount) >= 0 Then  'Ensure balance stays nonnegative
      m_balance = m_balance – amount
    Else
      RaiseEvent AttemptToOverdraw(amount – m_balance)
    End If
  End Sub
End Class      'CashRegister
```



11.2 Exercise 7

Amount: 700

Balance: $500.00

Withdrawal Denied

You would have been overdrawn by $200.00.

Exercises 11.3

1. 4 **3.** 64 **5.** Can move Has jointed limbs and no backbone

7. The keyword Overrides should be Inherits.

9. The Hi function should be declared with the Overridable keyword in class Hello and with the keyword Overrides keyword in class Aussie.

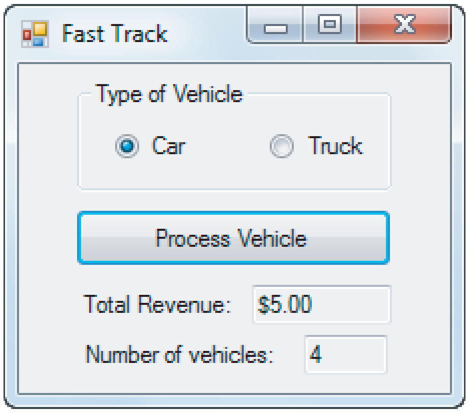
11. The Hi function should be declared with the Overrides keyword in class Cowboy.

13. The Hello class should be declared with the MustInherit keyword, and the function Hi should be declared with the MustOverride keyword.

15. The Hello class should be declared with the MustInherit keyword, not MustOverride.

19.

```
Public Class frmRegister
  Dim tollBooth As New FastTrackRegister()

  Private Sub btnProcess_Click(...) Handles btnProcess.Click
    If radCar.Checked Then
      tollBooth.ProcessCar()
    Else
      tollBooth.ProcessTruck()
    End If
    txtRevenue.Text = FormatCurrency(tollBooth.Balance)
    txtNumVehicles.Text = CStr(tollBooth.Count)
  End Sub
End Class    'frmRegister

Class CashRegister
  Private m_balance As Double

  Sub Deposit(ByVal amount As Double)
    m_balance += amount
  End Sub

  Sub WithDrawal(ByVal amount As Double)
    m_balance = m_balance - amount
  End Sub

  Public ReadOnly Property Balance() As Double
    Get
      Return m_balance
    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
```



● Car ○ Truck

Total Revenue: $5.00

Number of vehicles: 4

Chapter 12

Exercises 12.1

1.
```
Protected Sub btnFind_Click(...) Handles btnFind.Click
  Dim max As Double = CDbl(txtFirstNum.Text)
  If CDbl(txtSecondNum.Text) > max Then
    max = CDbl(txtSecondNum.Text)
  End If
  If CDbl(txtThirdNum.Text) > max Then
    max = CDbl(txtThirdNum.Text)
  End If
  txtLargestNum.Text = CStr(max)
End Sub
```