684 ◆ **Answers** To Selected Odd-Numbered Exercises

```vbnet
      gr.DrawLine(Pens.Black, 0, 0, 0, 98)      'left border
      gr.DrawLine(Pens.Black, 0, 98, 148, 98)   'bottom border
      gr.DrawLine(Pens.Black, 148, 0, 148, 98) 'right border
   End Sub
```
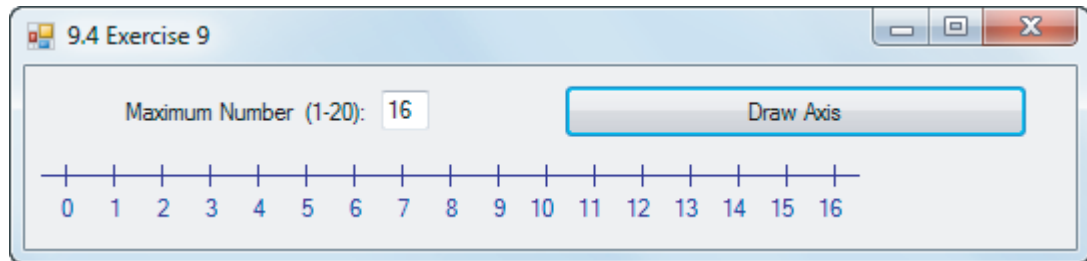
7.
```vbnet
Private Sub btnCreate_Click(...) Handles btnCreate.Click
   Dim gr As Graphics = picFlag.CreateGraphics
   Dim br() As Brush = {Brushes.Orange, Brushes.White, Brushes.Green}
   Dim r As Integer = 12    'radius of circle
   'picFlag.Width = 149; picFlag.Height = 99
   For i As Integer = 0 To 2
     gr.FillRectangle(br(i), 0, 0 + i * 33, 149, 33)
   Next
   gr.FillPie(Brushes.Orange, 75 — r, 49 — r, 2 * r, 2 * r, 0, 360)
   gr.DrawLine(Pens.Black, 0, 0, 148, 0)      'top border
   gr.DrawLine(Pens.Black, 0, 0, 0, 98)      'left border
   gr.DrawLine(Pens.Black, 0, 98, 148, 98)   'bottom border
   gr.DrawLine(Pens.Black, 148, 0, 148, 98) 'right border
   End Sub
```

9.
```vbnet
Private Sub btnDraw_Click(...) Handles btnDraw.Click
   Dim numbers As String = ""
   Dim gr As Graphics = picBox.CreateGraphics
   picBox.Refresh()
   Dim n As Integer = CInt(txtNumber.Text)
   'Font is Microsoft Sans Serif
   For i As Integer = 0 To n
     If i < 9 Then
       numbers &= i & "        "
     Else
       numbers &= i & "     "
     End If
     gr.DrawLine(Pens.Blue, 12 + (24 * i), 5, 12 + (24 * i), 15)
   Next
   gr.DrawLine(Pens.Blue, 0, 10, 24 * (n + 1), 10)
   gr.DrawString(numbers, Me.Font, Brushes.Blue, 8, 20)
   End Sub
```


CHAPTER 10

EXERCISES 10.1

1. (e) **3.** (d) **5.** (b)

7.
```vbnet
Private Sub frmCities_Load(...) Handles MyBase.Load
   Me.CitiesTableAdapter.Fill(Me.MegacitiesDataSet.Cities)
   End Sub
```

```
    Private Sub btnDisplay_Click(...) Handles btnDisplay.Click
      Dim query = From city In MegacitiesDataSet.Cities
                  Let popGrowth = (city.pop2015 — city.pop2010) / city.pop2010
                  Let formattedPopGrowth = FormatPercent(popGrowth)
                  Order By popGrowth Descending
                  Select city.name, city.country, formattedPopGrowth
      dgvOutput.DataSource = query.ToList
      dgvOutput.CurrentCell = Nothing
      dgvOutput.Columns("name").HeaderText = "City"
      dgvOutput.Columns("country").HeaderText = "Country"
      dgvOutput.Columns("formattedPopGrowth").HeaderText =
                            "Projected Population Growth"
    End Sub
```

17.
```
    Private Sub frmNations_Load(...) Handles MyBase.Load
       Me.NationsTableAdapter.Fill(Me.UNDataSet.Nations)
       Dim query = From nation In UNDataSet.Nations
                   Order By nation.continent Ascending
                   Select nation.continent
                   Distinct
       lstContinents.DataSource = query.ToList
    End Sub

    Private Sub lstContinents_SelectedIndexChanged(...) Handles _
                        lstContinents.SelectedIndexChanged
      Dim queryPop = From nation In UNDataSet.Nations
                     Where nation.continent = lstContinents.Text
                     Order By nation.population Descending
                     Select nation.name
       lstCountriesPop.DataSource = queryPop.ToList
       lstCountriesPop.SelectedItem = Nothing
       Dim queryArea = From nation In UNDataSet.Nations
                       Where nation.continent = lstContinents.Text
                       Order By nation.area Descending
                       Select nation.name
       lstCountriesArea.DataSource = queryArea.ToList
       lstCountriesArea.SelectedItem = Nothing
    End Sub
```
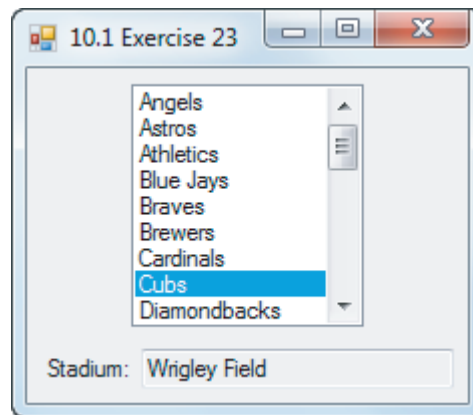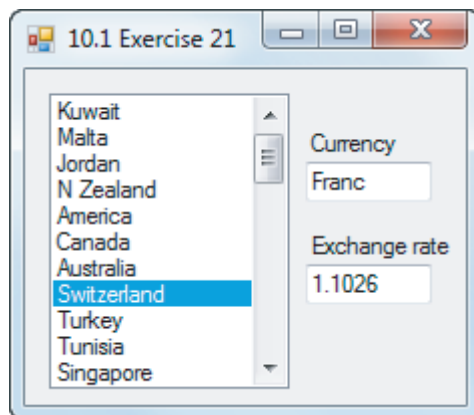

21.
```
    Private Sub frmCurrencies_Load(...) Handles MyBase.Load
       Me.RatesTableAdapter.Fill(Me.ExchrateDataSet.Rates)
       Dim query = From money In ExchrateDataSet.Rates
                   Order By money.dollarRate Ascending
                   Select money.country
```

```
      lstCountries.DataSource = query.ToList
   End Sub

   Private Sub lstCountries_SelectedIndexChanged(...) Handles _
               lstCountries.SelectedIndexChanged
     Dim query = From money In ExchrateDataSet.Rates
               Where money.country = lstCountries.Text
               Select money.monetaryUnit, money.dollarRate
     txtCurrency.Text = query.First.monetaryUnit
     txtRate.Text = CStr(query.First.dollarRate)
   End Sub
```
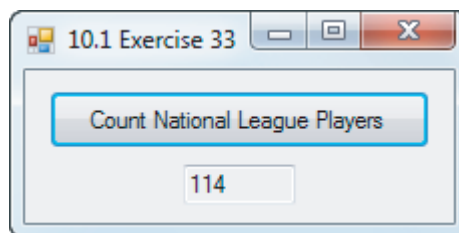
23.
```
   Private Sub frmBaseball_Load(...) Handles MyBase.Load
     Me.TeamsTableAdapter.Fill(Me.BaseballDataSet.Teams)
     Dim query = From team In BaseballDataSet.Teams
               Order By team.name
               Select team.name
       lstTeams.DataSource = query.ToList
     End Sub

   Private Sub lstTeams_SelectedIndexChanged(...) Handles _
                 lstTeams.SelectedIndexChanged
     Dim query = From team In BaseballDataSet.Teams
               Where team.name = lstTeams.Text
               Select team.stadium
     txtStadium.Text = query.First
   End Sub
```

33.
```
   Private Sub frmBaseball_Load(...) Handles MyBase.Load
     Me.TeamsTableAdapter.Fill(Me.BaseballDataSet.Teams)
     Me.PlayersTableAdapter.Fill(Me.BaseballDataSet.Players)
   End Sub

   Private Sub btnCount_Click(...) Handles btnCount.Click
     Dim query = From player In BaseballDataSet.Players
               Join team In BaseballDataSet.Teams
               On player.team Equals team.name
               Where team.league = "National"
               Select player.name
     txtNational.Text = FormatNumber(query.Count, 0)
   End Sub
```


39. film

41.
```
   Private Sub frmMovies_Load(...) Handles MyBase.Load
     Me.LinesTableAdapter.Fill(Me.MoviesDataSet.Lines)
     Me.ActorsTableAdapter.Fill(Me.MoviesDataSet.Actors)
     Dim query = From line In MoviesDataSet.Lines
               Select line.film
               Distinct
     lstFilms.DataSource = query.ToList
   End Sub
```
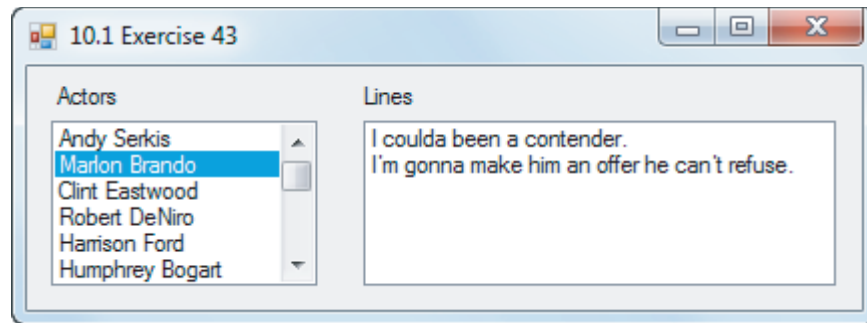
```
    Private Sub lstFilms_SelectedIndexChanged(...) Handles _
                    lstFilms.SelectedIndexChanged
      Dim query = From line In MoviesDataSet.Lines
                  Join flick In MoviesDataSet.Actors
                  On line.film Equals flick.film
                  Where line.film = lstFilms.Text
                  Select flick.maleLead
      txtLead.Text = query.First
    End Sub
```

43.
```
    Private Sub frmFilms_Load(...) Handles MyBase.Load
      Me.LinesTableAdapter.Fill(Me.MoviesDataSet.Lines)
      Me.ActorsTableAdapter.Fill(Me.MoviesDataSet.Actors)
      Dim query = From flick In MoviesDataSet.Actors
                  Select flick.maleLead
                  Distinct
      lstActors.DataSource = query.ToList
    End Sub

    Private Sub lstFilms_SelectedIndexChanged(...) Handles _
                    lstActors.SelectedIndexChanged
      Dim query = From line In MoviesDataSet.Lines
                  Join flick In MoviesDataSet.Actors
                  On line.film Equals flick.film
                  Where flick.maleLead = lstActors.Text
                  Select line.famousLine
      lstLines.DataSource = query.ToList
      lstLines.SelectedItem = Nothing
    End Sub
```


EXERCISES 10.2

7. Add a record to the Cities table whose name field is empty or contains the same name as an already existing record.

9. `(BindingSource1.Find("name", strVar) = 0) And (strVar <> "Bombay")`

11. Create a control named BindingSource2 that has the Countries table as its DataMember.
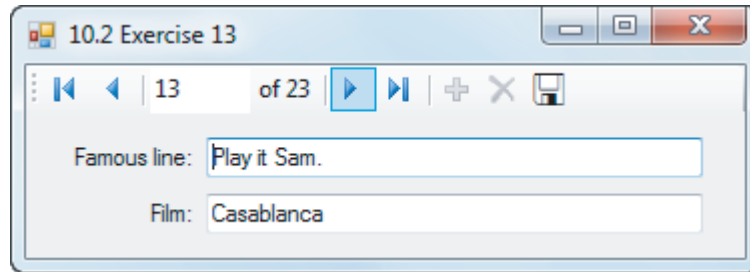
```
    Private Sub CountryTextBox_Leave(...) Handles _
                    CountryTextBox.Leave, BindingNavigator1.Click
      BindingSource2.Position = BindingSource2.Find("name",
                    CountryTextBox.Text)
      If (CountryTextBox.Text <> "") And
         (BindingSource2.Position = 0) And
         (CountryTextBox.Text <> "Argentina") Then
        MessageBox.Show("Not a valid country.", "ERROR")
        CountryTextBox.Focus()
      End If
    End Sub
```

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
```

```
10.2 Exercise 13
⏮ ◀ | 13        of 23 | ▶ ⏭ | ✚ ✕ 🖫
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
```