Chapter 15 Records and Random-Access Files (Main Page)

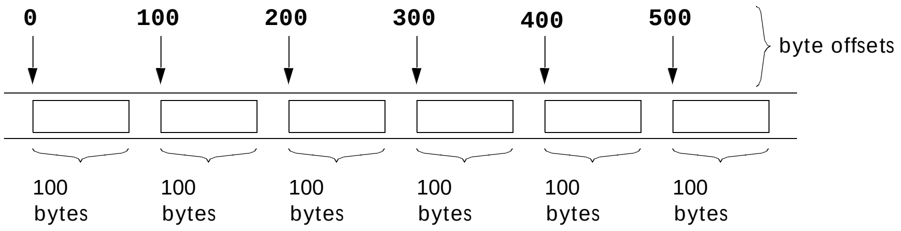


Fig. 15.1 Visual Basic's view of a random-access file.

Access type	Description
`Read`	Open the file in read-only mode.
`Write`	Open the file in write-only mode.
`Read Write`	Open the file for reading and writing.

Fig. 15.2 Access types available for random-access files.

Byte	0	1	2	3
	01100001		00000000	01100001

Fig. 15.3 A possible storage alignment for a variable of type **`Example`** showing an undefined area in memory.

```
' Fig. 15.4
' Creating a blank random-access file
Option Explicit

Private Type ClientRecord
   accountNumber As Integer
   lastName As String * 15
   firstName As String * 15
   balance As Currency
End Type

Sub cmdInitialize_Click()
   Dim recordLength As Long, x As Integer
   Dim udtBlankClient As ClientRecord   ' user-defined type
   Dim filename As String

   ' Determine number of bytes in a ClientRecord object
   recordLength = LenB(udtBlankClient)

   dlgOpen.ShowOpen
   filename = dlgOpen.filename

   If dlgOpen.FileTitle <> "" Then
      ' Open clients.rnd for writing
      Open filename For Random Access Write As #1 _
         Len = recordLength

      For x = 1 To 100
         Put #1, x, udtBlankClient  ' Write empty records
      Next

      Close #1   ' Close file

      cmdInitialize.Enabled = False  ' Disable button
      MsgBox ("File initialized. Click Exit to terminate.")
   Else
      MsgBox ("You must specify a file name")
   End If
End Sub

Sub cmdExit_Click()
   End
End Sub
```

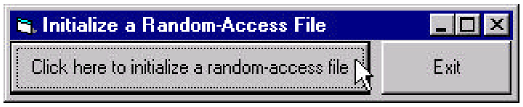
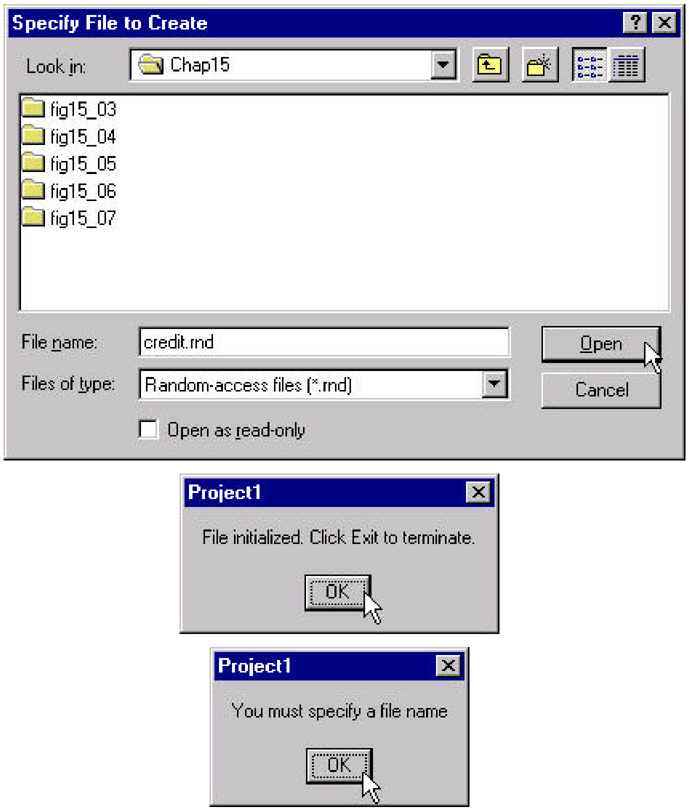

Initialize a Random-Access File
Click here to initialize a random-access file
Exit

Fig. 15.4 Writing empty records to a file (part 1 of 2).



Look in: Chap15

File name: credit.rnd

Files of type: Random-access files (*.rnd)

File initialized. Click Exit to terminate.

You must specify a file name

Fig. 15.4 Writing empty records to a file (part 2 of 2).

```
' Fig. 15.5
' Writing data to a random-access file
Option Explicit

Private Type ClientRecord
   accountNumber As Integer
   lastName As String * 15
   firstName As String * 15
   balance As Currency
End Type

Dim mUdtClient As ClientRecord   ' user-defined type

Private Sub Form_Load()
   cmdEnter.Enabled = False
   cmdDone.Enabled = False
End Sub

Sub cmdOpenFile_Click()
   Dim recordLength As Long
   Dim filename As String

   ' Determine number of bytes in a ClientRecord object
   recordLength = LenB(mUdtClient)

   dlgOpen.ShowOpen
   filename = dlgOpen.filename

   If dlgOpen.FileTitle <> "" Then
      ' Open file for writing
      Open filename For Random Access Write As #1 _
         Len = recordLength

      cmdOpenFile.Enabled = False  ' Disable button
      cmdEnter.Enabled = True
```

Fig. 15.5 Writing data randomly to a random-access file (part 1 of 3).

```
      cmdDone.Enabled = True
   Else
      MsgBox ("You must specify a file name")
   End If
End Sub

Private Sub cmdEnter_Click()
   mUdtClient.accountNumber = Val(txtAccount.Text)
   mUdtClient.firstName = txtFirstName.Text
   mUdtClient.lastName = txtLastName.Text
   mUdtClient.balance = Val(txtBalance.Text)

   ' Write record to file
   Put #1, mUdtClient.accountNumber, mUdtClient

   Call ClearFields
End Sub

Sub cmdDone_Click()
   Close #1
   cmdOpenFile.Enabled = True
   cmdEnter.Enabled = False
   cmdDone.Enabled = False
End Sub
```

```

Private Sub Form_Terminate()
   Close #1
End Sub

Private Sub ClearFields()
   txtAccount.Text = ""
   txtFirstName.Text = ""
   txtLastName.Text = ""
   txtBalance.Text = ""
End Sub
```

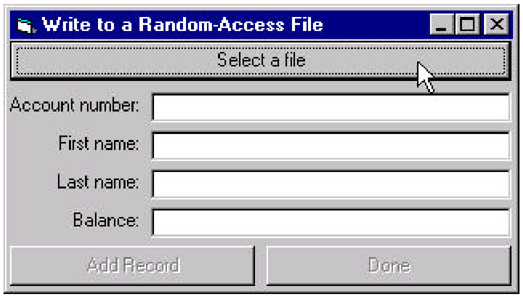
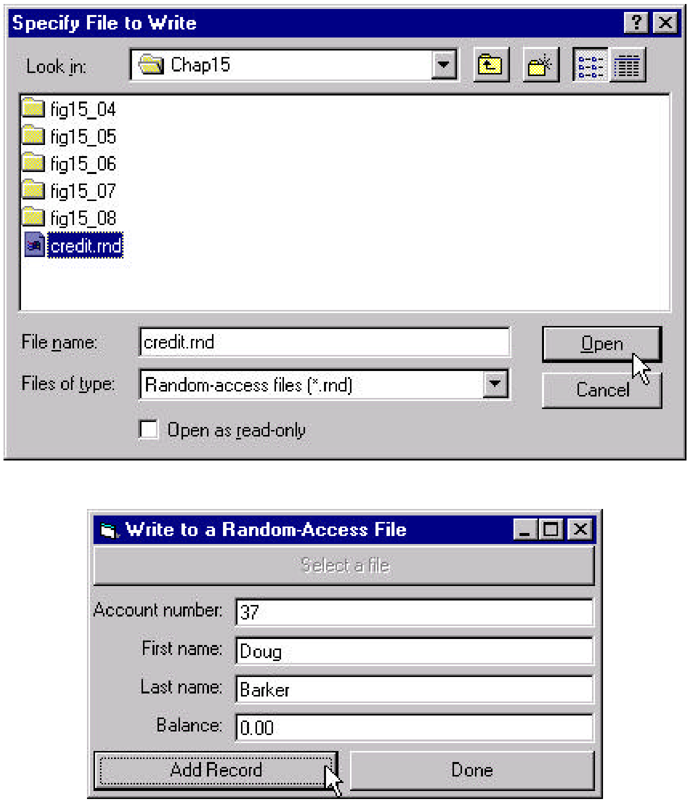


Fig. 15.5 Writing data randomly to a random-access file (part 2 of 3).



Look in: Chap15

File name: credit.rnd

Files of type: Random-access files (*.rnd)

Account number: 37

First name: Doug

Last name: Barker

Balance: 0.00

Fig. 15.5 Writing data randomly to a random-access file (part 3 of 3).

```
' Fig. 15.6
' Reading data sequentially from a random-access file
Option Explicit

Private Type ClientRecord
   accountNumber As Integer
   lastName As String * 15
   firstName As String * 15
   balance As Currency
End Type

Dim mUdtClient As ClientRecord   ' user-defined type

Private Sub Form_Load()
   cmdNext.Enabled = False
   cmdDone.Enabled = False
End Sub

Sub cmdOpenFile_Click()
   Dim recordLength As Long
   Dim filename As String

   ' Determine number of bytes in a ClientRecord object
   recordLength = LenB(mUdtClient)

   dlgOpen.ShowOpen
   filename = dlgOpen.filename

   If dlgOpen.FileTitle <> "" Then
      ' Open file for writing
      Open filename For Random Access Read As #1 _
         Len = recordLength
      cmdOpenFile.Enabled = False  ' Disable button
      cmdNext.Enabled = True
```

Fig. 15.6 Reading data sequentially from a random-access file (part 1 of 3).

```
      cmdDone.Enabled = True
   Else
      MsgBox ("You must specify a file name")
   End If
End Sub

Private Sub cmdNext_Click()
   ' Read record from file
   Do
      Get #1, , mUdtClient
   Loop Until EOF(1) Or mUdtClient.accountNumber <> 0

   If EOF(1) Then
      cmdNext.Enabled = False
      Exit Sub
   End If

   If mUdtClient.accountNumber <> 0 Then
      txtAccount.Text = Str$(mUdtClient.accountNumber)
      txtFirstName.Text = mUdtClient.firstName
      txtLastName.Text = mUdtClient.lastName
      txtBalance.Text = Str$(mUdtClient.balance)
   End If
End Sub

Sub cmdDone_Click()
   Close #1
   cmdOpenFile.Enabled = True
   cmdNext.Enabled = False
   cmdDone.Enabled = False
   txtAccount.Text = ""
   txtFirstName.Text = ""
   txtLastName.Text = ""
   txtBalance.Text = ""
End Sub

Private Sub Form_Terminate()
   Close #1
End Sub
```

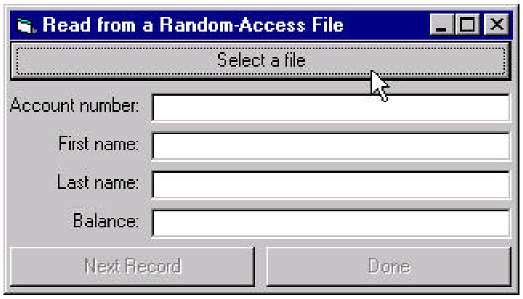
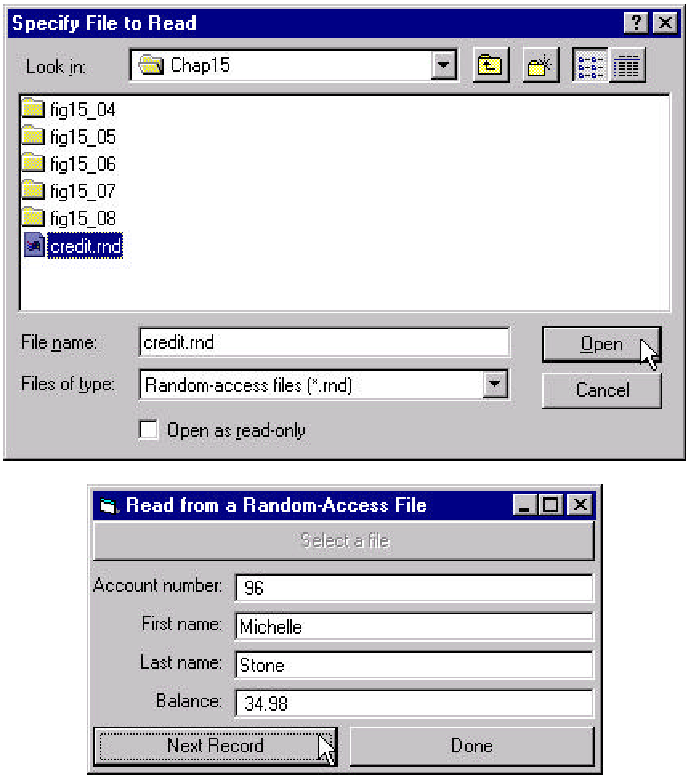


Fig. 15.6 Reading data sequentially from a random-access file (part 2 of 3).



Look in: Chap15

File name: credit.rnd

Files of type: Random-access files (*.rnd)

□ Open as read-only

Account number: 96

First name: Michelle

Last name: Stone

Balance: 34.98

Fig. 15.6 Reading data sequentially from a random-access file (part 3 of 3).

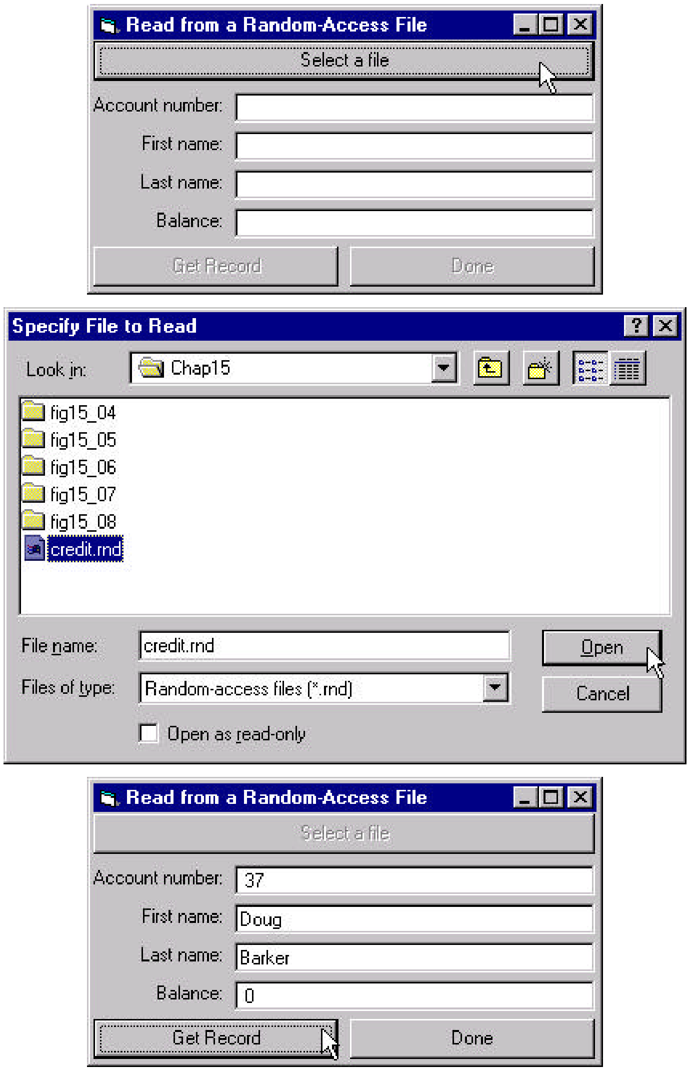
```
' Fig. 15.7
' Reading data randomly from a random-access file
Option Explicit

Private Type ClientRecord
   accountNumber As Integer
   lastName As String * 15
   firstName As String * 15
   balance As Currency
End Type

Dim mUdtClient As ClientRecord   ' user-defined type

Private Sub Form_Load()
   cmdGet.Enabled = False
   cmdDone.Enabled = False
End Sub
```

Fig. 15.7 Reading randomly from a random-access file (part 1 of 4).

```

Sub cmdOpenFile_Click()
   Dim recordLength As Long
   Dim filename As String

   ' Determine number of bytes in a ClientRecord object
   recordLength = LenB(mUdtClient)

   dlgOpen.ShowOpen
   filename = dlgOpen.filename

   If dlgOpen.FileTitle <> "" Then
      ' Open file for writing
      Open filename For Random Access Read As #1 _
         Len = recordLength
      cmdOpenFile.Enabled = False  ' Disable button
      cmdGet.Enabled = True
      cmdDone.Enabled = True
   Else
      MsgBox ("You must specify a file name")
   End If
End Sub

Private Sub cmdGet_Click()
   On Error Resume Next
   ' Read record from file
   If txtAccount.Text <> "" Then
      Get #1, Val(txtAccount.Text), mUdtClient

      If mUdtClient.accountNumber <> 0 Then
         txtAccount.Text = Str$(mUdtClient.accountNumber)
         txtFirstName.Text = mUdtClient.firstName
         txtLastName.Text = mUdtClient.lastName
         txtBalance.Text = Str$(mUdtClient.balance)
      ElseIf mUdtClient.accountNumber = 0 Then
         txtFirstName.Text = "Record not found"
         txtLastName.Text = ""
         txtBalance.Text = ""
      End If
   Else
      MsgBox ("You must specify an Account Number")
   End If
End Sub

Sub cmdDone_Click()
   Close #1
   cmdOpenFile.Enabled = True
   cmdGet.Enabled = False
   cmdDone.Enabled = False
   txtAccount.Text = ""
   txtFirstName.Text = ""
   txtLastName.Text = ""
```

Fig. 15.7 Reading randomly from a random-access file (part 2 of 4).

```
      txtBalance.Text = ""
   End Sub

   Private Sub Form_Terminate()
      Close #1
   End Sub
```



Read from a Random-Access File

Account number:

First name:

Last name:

Balance:

Specify File to Read

Look in: Chap15

fig15_04
fig15_05
fig15_06
fig15_07
fig15_08
credit.rnd

File name: credit.rnd

Files of type: Random-access files (*.rnd)

□ Open as read-only

Read from a Random-Access File

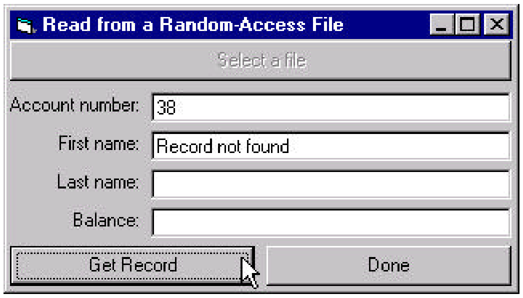
Account number: 37

First name: Doug

Last name: Barker

Balance: 0

Fig. 15.7 Reading randomly from a random-access file (part 3 of 4).



Account number: 38

First name: Record not found

Last name:

Balance:

Fig. 15.7 Reading randomly from a random-access file (part 4 of 4).

```
' Fig. 15.8
' Transaction processing program with random-access files
Option Explicit

Private Type ClientRecord
   accountNumber As Integer
   lastName As String * 15
   firstName As String * 15
   balance As Currency
End Type

Dim mUdtClient As ClientRecord   ' user-defined type

Private Sub Form_Load()
   tabOperations.Enabled = False
End Sub
```

Fig. 15.8 A transaction processing program (part 1 of 4).

```

Sub cmdOpenFile_Click()
   Dim recordLength As Long
   Dim filename As String

   ' Determine number of bytes in a ClientRecord object
   recordLength = LenB(mUdtClient)

   dlgOpen.ShowOpen
   filename = dlgOpen.filename

   If dlgOpen.FileTitle <> "" Then
      ' Open file for writing
      Open filename For Random Access Read Write As #1 _
         Len = recordLength
      cmdOpenFile.Enabled = False   ' Disable button
      cmdCloseFile.Enabled = True
      tabOperations.Enabled = True
   Else
      MsgBox ("You must specify a file name")
   End If
End Sub

' Create a text file representation of the random-access file
Private Sub cmdTextFile_Click()
   Dim filename As String, balanceString As String

   On Error Resume Next
   dlgTextFile.ShowOpen
   filename = dlgTextFile.filename

   If dlgTextFile.FileTitle <> "" Then
      ' Open file for writing
      Open filename For Output Access Write As #2
      Print #2, "Account";
      Print #2, Tab(10); "First Name";
      Print #2, Tab(28); "Last Name";
      Print #2, Tab(46); Format("Balance", "@@@@@@@@@@")

      Seek #1, 1            ' reposition to start of file
      Get #1, , mUdtClient  ' read first record

      While Not EOF(1)
         If mUdtClient.accountNumber <> 0 Then
            Print #2, mUdtClient.accountNumber;
            Print #2, Tab(10); mUdtClient.firstName;
            Print #2, Tab(28); mUdtClient.lastName;
            balanceString = _
               Format(mUdtClient.balance, "0.00")
            Print #2, Tab(46);
            Print #2, Format(balanceString, "@@@@@@@@@@")

         End If
```

Fig. 15.8 A transaction processing program (part 2 of 4).

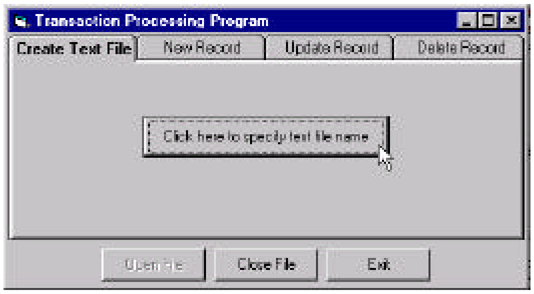
```

         Get #1, , mUdtClient  'read next record
      Wend

      Close #2
   Else
      MsgBox ("You must specify a file name")
   End If
End Sub

' Add a new record to the file
Private Sub cmdAddNew_Click()
   If txtNewAccount.Text <> "" Then
      Get #1, Val(txtNewAccount), mUdtClient  'read record

      If mUdtClient.accountNumber = 0 Then
         mUdtClient.accountNumber = Val(txtNewAccount)
         mUdtClient.firstName = txtNewFirstName.Text
         mUdtClient.lastName = txtNewLastName.Text
         mUdtClient.balance = txtNewBalance.Text
         Put #1, mUdtClient.accountNumber, mUdtClient
         MsgBox ("Account " & mUdtClient.accountNumber & _
                 " has been added to the file")
      Else
         MsgBox ("Account already exists")
      End If
   Else
      MsgBox ("You must enter an account number")
   End If
End Sub

' Update an existing record
Private Sub cmdUpdate_Click()
```

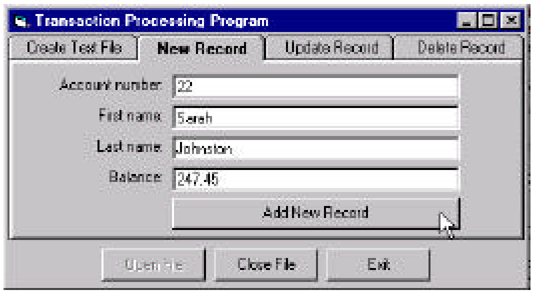
```
   Dim account As Integer, transactionAmount As Double
   On Error Resume Next

   account = Val(InputBox("Enter account number"))
   Get #1, account, mUdtClient  'read record

   If mUdtClient.accountNumber <> 0 Then
      txtUpdateAccount.Text = Str$(mUdtClient.accountNumber)
      txtUpdateFirstName.Text = mUdtClient.firstName
      txtUpdateLastName.Text = mUdtClient.lastName
      txtUpdateBalance.Text = Str$(mUdtClient.balance)
      transactionAmount = Val(InputBox( _
         "Enter transaction amount. Positive for charge. " & _
         "Negative for payment."))
      mUdtClient.balance = _
         mUdtClient.balance + transactionAmount
      txtUpdateBalance.Text = Str$(mUdtClient.balance)
      Put #1, mUdtClient.accountNumber, mUdtClient
```

Fig. 15.8 A transaction processing program (part 3 of 4).

```
   Else
      MsgBox ("Record " & account & " does not exist")
   End If

End Sub

' Delete the specified record
Private Sub cmdDelete_Click()
   Dim blankClient As ClientRecord
   On Error Resume Next

   Get #1, Val(txtDelete.Text), mUdtClient  'read record

   If mUdtClient.accountNumber <> 0 Then
      Put #1, mUdtClient.accountNumber, blankClient
      MsgBox ("Account # " & mUdtClient.accountNumber & _
         " has been deleted")
   Else
      MsgBox ("Record does not exist")
   End If
End Sub

Sub cmdCloseFile_Click()
   Close #1
   cmdOpenFile.Enabled = True
   cmdCloseFile.Enabled = False
End Sub

Private Sub Form_Terminate()
   Close
End Sub

Private Sub cmdExit_Click()
   Close
   End
End Sub
```

Fig. 15.8 A transaction processing program (part 4 of 4).



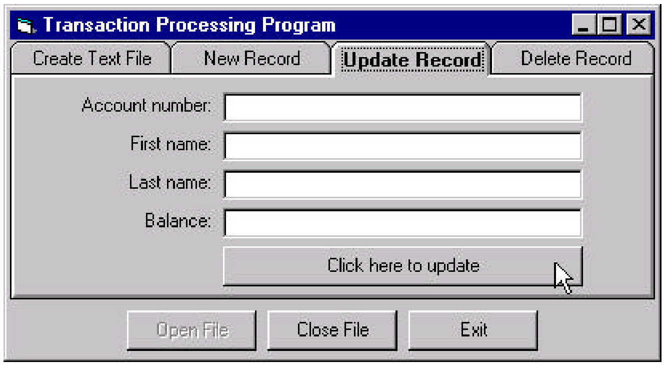


Account number: 22

First name: Sarah

Last name: Johnston

Balance: 247.45

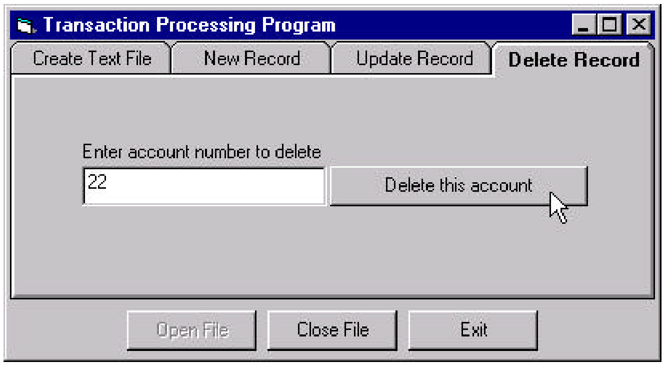


Account number:

First name:

Last name:

Balance:



Enter account number to delete

22