

Example on Using Array



Fixed-Size One Dimensional Array

Option Explicit

Option Base 1

Dim aName(25) As String

Dim aSex(25) As String

Dim aAddress(25) As String

Dim strAction As String

Dim i As Integer, j As Integer, CIndex As Integer, NIndex As Integer

Private Sub LoadSexItem(ByVal s As String)

Dim i As Integer

For i = 0 To cboSex.ListCount - 1

 If cboSex.List(i) = s Then

 cboSex.ListIndex = i

 Exit For

 End If

Next i

End Sub

Private Sub ArrayToTextBox(ByVal AIndex As Integer)

txtName.Text = aName(AIndex)

LoadSexItem (aSex(AIndex))

txtAddress.Text = aAddress(AIndex)

End Sub

Private Sub TextBoxToArray(ByVal AIndex As Integer)

aName(AIndex) = txtName.Text

aSex(AIndex) = cboSex.List(cboSex.ListIndex)

aAddress(AIndex) = txtAddress.Text

End Sub

Private Sub LoadSex()

cboSex.AddItem "Male"

cboSex.AddItem "Female"

End Sub

Private Sub cmdCancel_Click()

EnableAtNew (True)

LockInput (True)

End Sub

Private Sub cmdDelete_Click()

For i = CIndex To NIndex

aName(i) = aName(i + 1)

aSex(i) = aSex(i + 1)

aAddress(i) = aAddress(i + 1)

Next i

NIndex = NIndex - 1

CIndex = CIndex + 1

If CIndex > NIndex Then

 CIndex = NIndex

End If

 ArrayToTextBox (CIndex)

End Sub

Private Sub cmdFirst_Click()

 CIndex = 1

 ArrayToTextBox (CIndex)

End Sub

Private Sub cmdLast_Click()

 CIndex = NIndex

 ArrayToTextBox (CIndex)

End Sub

Private Sub LockInput(LockState As Boolean)

 txtName.Locked = LockState

 cboSex.Locked = LockState

 txtAddress.Locked = LockState

End Sub

Private Sub cmdNew_Click()

 strAction = "New"

 LockInput (False)

 EnableAtNew (False)

End Sub

Private Sub EnableAtNew(Enab As Boolean)

 cmdNew.Enabled = Enab

 cmdUpdate.Enabled = Enab

 cmdDelete.Enabled = Enab

cmdSearch.Enabled = Enab

cmdFirst.Enabled = Enab

cmdPrevious.Enabled = Enab

cmdNext.Enabled = Enab

cmdLast.Enabled = Enab

cmdSave.Enabled = Not Enab

cmdCancel.Enabled = Not Enab

End Sub

Private Sub cmdNext_Click()

CIndex = CIndex + 1

If CIndex > NIndex Then

 MsgBox "Already at Last Record"

 CIndex = NIndex

 Exit Sub

End If

ArrayToTextBox (CIndex)

End Sub

Private Sub cmdPrevious_Click()

CIndex = CIndex - 1

If CIndex < 1 Then

 MsgBox "Already at First Record"

 CIndex = 1

 Exit Sub

End If

ArrayToTextBox (CIndex)

End Sub

Private Sub cmdSave_Click()

If strAction = "New" Then

 NIndex = NIndex + 1

TextBoxToArray (NIndex)

CIndex = NIndex

Else

TextBoxToArray (CIndex)

End If

EnableAtNew (True)

LockInput (True)

End Sub

Private Sub cmdSearch_Click()

Dim strName As String

Dim blnFound As Boolean

Dim i As Integer

blnFound = False

strName = InputBox("Please input a name : ")

For i = 1 To NIndex

 If aName(i) = strName Then

 blnFound = True

 CIndex = i

 Exit For

 End If

Next i

If blnFound = True Then

 MsgBox "Student Found:" & vbCrLf & "Name : " & aName(CIndex) & vbCrLf &

 "Sex : " & aSex(CIndex) & vbCrLf & "Address : " & aAddress(CIndex)

 ArrayToTextBox (CIndex)

Else

 MsgBox "Student not Found"

End If

End Sub

```
Private Sub cmdUpdate_Click()  
    TextBoxToArray (CIndex)  
    strAction = "Update"  
    LockInput (False)  
    EnableAtNew (False)  
End Sub
```

```
Private Sub Form_Load()  
    Call LoadSex  
End Sub
```

Dynamic-Size One Dimensional Array

```
Option Explicit
```

```
Option Base 1
```

```
Dim aName() As String
```

```
Dim aSex() As String
```

```
Dim aAddress() As String
```

```
Dim strAction As String
```

```
Dim i As Integer, j As Integer, CIndex As Integer, NIndex As Integer
```

```
Private Sub cmdSave_Click()
```

```
    If strAction = "New" Then
```

```
        NIndex = NIndex + 1
```

```
        ReDim Preserve aName(NIndex)
```

```
        ReDim Preserve aSex(NIndex)
```

```
        ReDim Preserve aAddress(NIndex)
```

```
        TextBoxToArray (NIndex)
```

```
        CIndex = NIndex
```

```
    Else
```

```
        TextBoxToArray (CIndex)
```

```
End If
EnableAtNew (True)
LockInput (True)
End Sub
```

Fixed-Size Multi-Dimensional Array

```
Option Explicit
```

```
Option Base 1
```

```
Dim aStudent(3, 25) As String
```

```
Dim strAction As String
```

```
Dim i As Integer, j As Integer, CIndex As Integer, NIndex As Integer
```

```
Private Sub ArrayToTextBox(ByVal AIndex As Integer)
```

```
txtName.Text = aStudent(1, AIndex)
```

```
LoadSexItem (aStudent(2, AIndex))
```

```
txtAddress.Text = aStudent(3, AIndex)
```

```
End Sub
```

```
Private Sub TextBoxToArray(ByVal AIndex As Integer)
```

```
aStudent(1, AIndex) = txtName.Text
```

```
aStudent(2, AIndex) = cboSex.List(cboSex.ListIndex)
```

```
aStudent(3, AIndex) = txtAddress.Text
```

```
End Sub
```

```
Private Sub cmdDelete_Click()
```

```
For i = CIndex To NIndex
```

```
aStudent(1, i) = aStudent(1, i + 1)
```

```
aStudent(2, i) = aStudent(2, i + 1)
```

```
aStudent(3, i) = aStudent(3, i + 1)
```

```
Next i
```

```
NIndex = NIndex - 1
```

```
CIndex = CIndex + 1
If CIndex > NIndex Then
    CIndex = NIndex
End If
ArrayToTextBox (CIndex)
End Sub
```

```
Private Sub cmdSearch_Click()
```

```
    Dim strName As String
```

```
    Dim blnFound As Boolean
```

```
    Dim i As Integer
```

```
    blnFound = False
```

```
    strName = InputBox("Please input a name : ")
```

```
    For i = 1 To NIndex
```

```
        If aName(i) = strName Then
```

```
            blnFound = True
```

```
            CIndex = i
```

```
            Exit For
```

```
        End If
```

```
    Next i
```

```
    If blnFound = True Then
```

```
        MsgBox "Student Found:" & vbCrLf & "Name : " & aStudent(1, CIndex) & vbCrLf &
            "Sex : " & aStudent(2, CIndex) & vbCrLf & "Address : " & aStudent(3, CIndex)
```

```
        ArrayToTextBox (CIndex)
```

```
    Else
```

```
        MsgBox "Student not Found"
```

```
    End If
```

```
End Sub
```


Dynamic-Size Multi-Dimensional Array

Option Explicit

Option Base 1

Dim aStudent() As String

Dim strAction As String

Dim i As Integer, j As Integer, CIndex As Integer, NIndex As Integer

If strAction = "New" Then

 NIndex = NIndex + 1

ReDim Preserve aStudent(3, NIndex)

 TextBoxToArray (NIndex)

 CIndex = NIndex

Else

 TextBoxToArray (CIndex)

End If

EnableAtNew (True)

LockInput (True)

End Sub