

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmAddLogoSubform"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmAddStudent"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Image29_Click()
13 On Error GoTo ErrHandler:
14 If Me.Dirty Then
15     Me.Dirty = False
16 End If
17 Me.Visible = False
18
19 DoCmd.SetWarnings False
20 DoCmd.OpenQuery "qryDeleteStudentRecord", acViewNormal, acEdit
21 DoCmd.SetWarnings True
22
23 ErrHandler:
24 If Err.Number = 3022 Then
25     MsgBox "There is already a record in the system for this Academic Year and UPN, please try
another UPN or Academic Year", vbInformation, fncAppName
26     Response = 0
27 End If
28 End Sub
29
30 Private Sub Image44_Click()
31 Me.Undo
32 Me.Tag = "Cancel"
33 Me.Visible = False
34 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmBackEndSubform"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Command2_Click()
13 Me.NewLocation = fncFileToLink
14 End Sub
15
16 Function PriMakeHiddenAttachedTable(strDatabaseName As String, _
17     strTableName As String, strAttachedTableName As String)
18
19     Dim db As DAO.Database
20     Dim td As DAO.TableDef
21
22     Set db = CurrentDb
23     Set td = db.CreateTableDef(strAttachedTableName)
24
25 If Mid$(strDatabaseName, 2, 1) = ":" Then
26     'connect mdb file
27     td.Connect = ";Database=" & strDatabaseName
28 Else
29     'connect odbc connection
30     td.Connect = "ODBC;" & strDatabaseName
31 End If
32
33     td.SourceTableName = strTableName
34     db.TableDefs.Append td
35     Application.SetHiddenAttribute acTable, strAttachedTableName, True
36
37
38
39     Set td = Nothing
40     Set db = Nothing
41
42 End Function
43
44 Private Sub Image25_Click()
45 'This keeps the same table name on the front end,
46 'even if it's different than the back end table name.
47 DoCmd.RunCommand acCmdSaveRecord
48 If IsNull(Me.NewLocation) Then
49     MsgBox "You must select the data file location before updating.", vbInformation,
49         fncAppName
```

```
50 Else
51   Dim rs As Recordset
52   Dim TblName As String
53   Dim ForeignTblName As String
54   Dim OldDbPath As String
55   Dim NewDbPath As String
56   Dim strSQL As String
57   Dim QueryExpression As String
58
59   ' Only check if new path is a file location (second character is a :)
60   If Mid$(NewLocation, 2, 1) <> ":" Then
61     MsgBox "Invalid file path", vbInformation, fncAppName
62     Exit Sub
63   End If
64
65   strSQL = "SELECT MSysObjects.Name, MSysObjects.ForeignName INTO TempLink " & _
66     "FROM MSysObjects " & _
67     "WHERE (((MSysObjects.Name) Like 'tbl*') AND ((MSysObjects.Database) Is Not
Null));"
68
69   DoCmd.SetWarnings False
70   DoCmd.RunSQL strSQL
71   DoCmd.SetWarnings True
72
73   Set rs = CurrentDb.OpenRecordset("TempLink")
74   NewDbPath = Me.NewLocation
75   'relink the tables
76   While Not rs.EOF
77     TblName = rs.Name
78     ForeignTblName = rs.ForeignName
79     'delete old link
80     DoCmd.DeleteObject acTable, TblName
81     'make new link, retaining the front end name, even if the back end name is different
82     Call PriMakeHiddenAttachedTable(NewDbPath, ForeignTblName, TblName)
83     rs.MoveNext
84   Wend
85
86   rs.Close
87
88   DoCmd.SetWarnings False
89   DoCmd.RunSQL strSQL
90   DoCmd.OpenQuery "qryUpdateRunOnce", acViewNormal, acEdit
91   DoCmd.SetWarnings True
92
93   Set rs = CurrentDb.OpenRecordset("TempLink")
94   rs.Close
95
96   MsgBox "Data tables linked.", vbInformation, fncAppName
97   'set icon location
98   Dim strIconPath As String
99   strIconPath = Left([NewLocation], InStrRev([NewLocation], "\")) & "\" &
fncShortAppName & ".ico"
100  Dim db As DAO.Database, prp As DAO.Property
101  On Error GoTo GotErr
102  Set db = CurrentDb
```

```
103 db.Properties("AppIcon") = strIconPath 'Attempt to assign
104 Application.RefreshTitleBar 'Update On Screen!
105 Me.NewLocation = Null
106 Me.CurrentLocation.Requery
107
108 GotErr:
109 If Err.Number = 3270 Then 'property doesn't exist!
110     Set prp = db.CreateProperty("AppIcon", dbText, strIconPath)
111     db.Properties.Append prp 'New property set!
112 Else
113     'MsgBox "Error: " & Err.Number & vbCrLf & Err.Description
114 End If
115 Exit Sub
116 End If
117 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmChart"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Form_Close()
13 Dim rs As DAO.Recordset
14 Dim db As Database
15 Dim strSQL As String
16 Dim iCounter As Integer
17
18 iCounter = Me.Graph0.Rotation
19 Set db = CurrentDb
20 strSQL = "SELECT tblSystem.ChartRotation FROM tblSystem;"
21 Set rs = db.OpenRecordset(strSQL)
22 rs.MoveFirst
23 rs.Edit
24 rs.Fields("ChartRotation").value = iCounter
25 rs.Update
26
27 Set rs = Nothing
28 Set db = Nothing
29 End Sub
30
31 Private Sub Form_Error(DataErr As Integer, Response As Integer)
32 Response = 0
33 End Sub
34
35 Private Sub Form_Open(Cancel As Integer)
36 Dim rs As DAO.Recordset
37 Dim db As Database
38 Dim strSQL As String
39 Dim iCounter As Integer
40
41 iCounter = Me.Graph0.Rotation
42 Set db = CurrentDb
43 strSQL = "SELECT tblSystem.ChartRotation FROM tblSystem;"
44 Set rs = db.OpenRecordset(strSQL)
45 rs.MoveFirst
46 iCounter = rs.Fields("ChartRotation").value
47
48 Set rs = Nothing
49 Set db = Nothing
50 Me.Graph0.Rotation = iCounter
```

```
51 End Sub
52
53 Private Sub Image86_Click()
54 DoCmd.OutputTo acOutputForm, "frmChart", "PDFFormat(*.pdf)", "", True, "", ,
  acExportQualityPrint
55 End Sub
56
57 Private Sub Image4_Click()
58 Dim iCounter As Integer
59 iCounter = Me.Graph0.Rotation + 10
60
61 If iCounter >= 360 Then
62 iCounter = 10
63 Me.Graph0.Rotation = iCounter
64 Else
65 Me.Graph0.Rotation = iCounter
66 End If
67 End Sub
68
69 Private Sub Image44_Click()
70 DoCmd.Close
71 End Sub
72
73 Private Sub Image5_Click()
74 Dim iCounter As Integer
75 iCounter = Me.Graph0.Rotation - 10
76
77 If iCounter <= 0 Then
78 iCounter = 350
79 Me.Graph0.Rotation = iCounter
80 Else
81 Me.Graph0.Rotation = iCounter
82 End If
83 End Sub
84
85 Private Sub Image79_Click()
86 On Error GoTo ErrHandler:
87 DoCmd.RunCommand acCmdPrint
88
89 ErrHandler:
90
91 Response = 0
92 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmDataMaintenanceSubform"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Image17_Click()
13 DoCmd.RunCommand acCmdSaveRecord
14 Dim strMessageText As String
15 If IsNull(Me.SelectAcYr) Then
16     MsgBox "Please select an academic year to delete.", vbInformation, fncAppName
17 Else
18     DoCmd.SetWarnings False
19     DoCmd.OpenQuery "qryDeleteDataset", acViewNormal, acEdit
20     DoCmd.SetWarnings True
21     strMessageText = "Dataset deleted for " & Me.SelectAcYr & "."
22     MsgBox strMessageText, vbInformation, fncAppName
23     Me.SelectAcYr.Requery
24     Me.SelectAcYr = Null
25 End If
26 End Sub
27
28 Private Sub Image19_Click()
29 DoCmd.RunCommand acCmdSaveRecord
30 Dim strMessageText As String
31 If IsNull(Me.SelectAcYr) Then
32     MsgBox "Please select an academic year to lock.", vbInformation, fncAppName
33 Else
34     DoCmd.SetWarnings False
35     DoCmd.OpenQuery "qryLockAll", acViewNormal, acEdit
36     DoCmd.SetWarnings True
37     strMessageText = "Dataset locked for " & Me.SelectAcYr & "."
38     MsgBox strMessageText, vbInformation, fncAppName
39     Me.SelectAcYr = Null
40 End If
41 End Sub
42
43 Private Sub Image20_Click()
44 DoCmd.RunCommand acCmdSaveRecord
45 If IsNull(Me.SelectAcYr) Then
46     MsgBox "Please select an academic year to unlock.", vbInformation, fncAppName
47 Else
48     DoCmd.SetWarnings False
49     DoCmd.OpenQuery "qryUnlockAll", acViewNormal, acEdit
50     DoCmd.SetWarnings True
```



```
51     MsgBox "Unlocked", vbInformation, fncAppName
52     Me.SelectAcYr = Null
53 End If
54 End Sub
55
56 Private Sub Image21_Click()
57 DoCmd.RunCommand acCmdSaveRecord
58 Dim strMessageText As String
59 If IsNull(Me.SelectAcYr) Then
60     MsgBox "Please select an academic year to delete.", vbInformation, fncAppName
61 Else
62     DoCmd.SetWarnings False
63     DoCmd.OpenQuery "qryDeleteLeavers", acViewNormal, acEdit
64     DoCmd.SetWarnings True
65     strMessageText = "Leavers deleted for " & Me.SelectAcYr & "."
66     MsgBox strMessageText, vbInformation, fncAppName
67     Me.SelectAcYr.Requery
68     Me.SelectAcYr = Null
69 End If
70 End Sub
71
72 Private Sub Image33_Click()
73 Call BackUpBE
74 End Sub
75
76 Private Sub SelectAcYr_GotFocus()
77 Me.SelectAcYr.Requery
78 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmDetail"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Form_Open(Cancel As Integer)
13 Me.frmDetailSubform.SourceObject = "frmDetailSubformShort"
14 End Sub
15
16 Private Sub Image21_Click()
17 DoCmd.OpenForm "frmEnterPasswordSummary", acNormal, "", "", , acDialog
18 End Sub
19
20 Private Sub Image23_Click()
21 DoCmd.Close
22 DoCmd.OpenForm "frmEnterDataSEN", acNormal, "", "", , acNormal
23 End Sub
24
25 Private Sub Image42_Click()
26
27 On Error GoTo ErrHandler:
28 DoCmd.OutputTo acTable, "tblDetailTemp", "MicrosoftExcelBiff8(*.xls)", "", True, "", 0
29
30 ErrHandler:
31 If Err.Number = 2501 Then
32 Response = 0
33 End If
34 End Sub
35
36 Private Sub Image44_Click()
37 DoCmd.Close
38 DoCmd.SetWarnings False
39 DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
40 DoCmd.OpenQuery "qryDetailAppendAll", acViewNormal, acEdit
41 DoCmd.SetWarnings True
42 DoCmd.OpenForm "frmStartup", acNormal, "", "", , acNormal
43 End Sub
44
45 Private Sub Image48_Click()
46 DoCmd.SetWarnings False
47 DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
48 DoCmd.OpenQuery "qryDetailAppendAll", acViewNormal, acEdit
49 DoCmd.SetWarnings True
50 Me.frmDetailSubform.Requery
```

```
51 Me.SelectGender.Requery
52 Me.SelectFSM.Requery
53 Me.SelectYearGroup.Requery
54 Me.SelectRegGroup.Requery
55 Me.SelectSEN.Requery
56 Me.SelectStudent.Requery
57 Me.SelectEthnicity.Requery
58 Me.SelectAcYr.Requery
59 Me.Select1stLanguage.Requery
60 Me.SelectEAL.Requery
61 Me.SelectNoD.Requery
62 Me.SelectKS.Requery
63 Me.SelectAcYr = "All"
64 Me.SelectEthnicity = "All"
65 Me.SelectFSM = "All"
66 Me.SelectGender = "All"
67 Me.SelectRegGroup = "All"
68 Me.SelectSEN = "All"
69 Me.SelectYearGroup = "All"
70 Me.Select1stLanguage = "All"
71 Me.SelectEAL = "All"
72 Me.SelectStudent = "All"
73 Me.SelectNoD = "All"
74 Me.SelectKS = "All"
75 End Sub
76
77 Private Sub Image49_Click()
78 DoCmd.OpenForm "frmReports", acNormal, "", "", , acNormal
79 End Sub
80
81 Private Sub Select1stLanguage_AfterUpdate()
82 DoCmd.SetWarnings False
83 DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
84 DoCmd.OpenQuery "qryDetailAppend", acViewNormal, acEdit
85 DoCmd.SetWarnings True
86 Me.SelectGender.Requery
87 Me.SelectFSM.Requery
88 Me.SelectYearGroup.Requery
89 Me.SelectRegGroup.Requery
90 Me.SelectSEN.Requery
91 Me.SelectStudent.Requery
92 Me.SelectEthnicity.Requery
93 Me.SelectAcYr.Requery
94 Me.SelectKS.Requery
95 Me.frmDetailSubform.Requery
96 Me.Select1stLanguage.Requery
97 End Sub
98
99 Private Sub SelectAcYr_AfterUpdate()
100 DoCmd.SetWarnings False
101 DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
102 DoCmd.OpenQuery "qryDetailAppend", acViewNormal, acEdit
103 DoCmd.SetWarnings True
104 Me.SelectGender.Requery
```

```
105 Me.SelectFSM.Requery
106 Me.SelectYearGroup.Requery
107 Me.SelectRegGroup.Requery
108 Me.SelectSEN.Requery
109 Me.SelectStudent.Requery
110 Me.SelectEthnicity.Requery
111 Me.Select1stLanguage.Requery
112 Me.SelectKS.Requery
113 Me.frmDetailSubform.Requery
114 Me.SelectAcYr.Requery
115 End Sub
116
117 Private Sub SelectEAL_AfterUpdate()
118     DoCmd.SetWarnings False
119     DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
120     DoCmd.OpenQuery "qryDetailAppend", acViewNormal, acEdit
121     DoCmd.SetWarnings True
122 Me.SelectGender.Requery
123 Me.SelectFSM.Requery
124 Me.SelectYearGroup.Requery
125 Me.SelectRegGroup.Requery
126 Me.SelectSEN.Requery
127 Me.SelectStudent.Requery
128 Me.SelectEthnicity.Requery
129 Me.Select1stLanguage.Requery
130 Me.SelectAcYr.Requery
131 Me.SelectKS.Requery
132 Me.frmDetailSubform.Requery
133 Me.SelectEAL.Requery
134 End Sub
135
136 Private Sub SelectEthnicity_AfterUpdate()
137     DoCmd.SetWarnings False
138     DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
139     DoCmd.OpenQuery "qryDetailAppend", acViewNormal, acEdit
140     DoCmd.SetWarnings True
141 Me.SelectAcYr.Requery
142 Me.SelectGender.Requery
143 Me.SelectFSM.Requery
144 Me.SelectYearGroup.Requery
145 Me.SelectRegGroup.Requery
146 Me.SelectSEN.Requery
147 Me.SelectStudent.Requery
148 Me.Select1stLanguage.Requery
149 Me.SelectKS.Requery
150 Me.frmDetailSubform.Requery
151 Me.SelectEthnicity.Requery
152 End Sub
153
154 Private Sub SelectFSM_AfterUpdate()
155     DoCmd.SetWarnings False
156     DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
157     DoCmd.OpenQuery "qryDetailAppend", acViewNormal, acEdit
158     DoCmd.SetWarnings True
```

```
159 Me.SelectAcYr.Requery
160 Me.SelectEthnicity.Requery
161 Me.SelectGender.Requery
162 Me.SelectYearGroup.Requery
163 Me.SelectRegGroup.Requery
164 Me.SelectSEN.Requery
165 Me.SelectStudent.Requery
166 Me.Select1stLanguage.Requery
167 Me.SelectKS.Requery
168 Me.frmDetailSubform.Requery
169 Me.SelectFSM.Requery
170 End Sub
171
172 Private Sub SelectGender_AfterUpdate()
173     DoCmd.SetWarnings False
174     DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
175     DoCmd.OpenQuery "qryDetailAppend", acViewNormal, acEdit
176     DoCmd.SetWarnings True
177 Me.SelectAcYr.Requery
178 Me.SelectEthnicity.Requery
179 Me.SelectFSM.Requery
180 Me.SelectYearGroup.Requery
181 Me.SelectRegGroup.Requery
182 Me.SelectSEN.Requery
183 Me.SelectStudent.Requery
184 Me.Select1stLanguage.Requery
185 Me.SelectKS.Requery
186 Me.frmDetailSubform.Requery
187 Me.SelectGender.Requery
188 End Sub
189
190 Private Sub SelectKS_AfterUpdate()
191     DoCmd.SetWarnings False
192     DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
193     DoCmd.OpenQuery "qryDetailAppend", acViewNormal, acEdit
194     DoCmd.SetWarnings True
195 Me.SelectAcYr.Requery
196 Me.SelectEthnicity.Requery
197 Me.SelectFSM.Requery
198 Me.SelectGender.Requery
199 Me.SelectRegGroup.Requery
200 Me.SelectSEN.Requery
201 Me.SelectStudent.Requery
202 Me.Select1stLanguage.Requery
203 Me.SelectYearGroup.Requery
204 Me.frmDetailSubform.Requery
205 Me.SelectKS.Requery
206 End Sub
207
208 Private Sub SelectNoD_AfterUpdate()
209 If Me.SelectNoD = "All" Then
210     DoCmd.SetWarnings False
211     DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
212     DoCmd.OpenQuery "qryDetailAppend", acViewNormal, acEdit
```

```
213 DoCmd.SetWarnings True
214 Else
215 DoCmd.SetWarnings False
216 DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
217 Call NatureOfDifficultyFilter
218 DoCmd.SetWarnings True
219 End If
220 Me.SelectAcYr.Requery
221 Me.SelectGender.Requery
222 Me.SelectFSM.Requery
223 Me.SelectYearGroup.Requery
224 Me.SelectRegGroup.Requery
225 Me.SelectSEN.Requery
226 Me.SelectStudent.Requery
227 Me.Select1stLanguage.Requery
228 Me.SelectKS.Requery
229 Me.frmDetailSubform.Requery
230 Me.SelectEthnicity.Requery
231 End Sub
232
233 Private Sub SelectRegGroup_AfterUpdate()
234 DoCmd.SetWarnings False
235 DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
236 DoCmd.OpenQuery "qryDetailAppend", acViewNormal, acEdit
237 DoCmd.SetWarnings True
238 Me.SelectAcYr.Requery
239 Me.SelectEthnicity.Requery
240 Me.SelectFSM.Requery
241 Me.SelectGender.Requery
242 Me.SelectStudent.Requery
243 Me.SelectSEN.Requery
244 Me.SelectYearGroup.Requery
245 Me.Select1stLanguage.Requery
246 Me.SelectKS.Requery
247 Me.frmDetailSubform.Requery
248 Me.SelectRegGroup.Requery
249 End Sub
250
251 Private Sub SelectSEN_AfterUpdate()
252 DoCmd.SetWarnings False
253 DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
254 DoCmd.OpenQuery "qryDetailAppend", acViewNormal, acEdit
255 DoCmd.SetWarnings True
256 Me.SelectAcYr.Requery
257 Me.SelectEthnicity.Requery
258 Me.SelectFSM.Requery
259 Me.SelectGender.Requery
260 Me.SelectRegGroup.Requery
261 Me.SelectStudent.Requery
262 Me.SelectYearGroup.Requery
263 Me.Select1stLanguage.Requery
264 Me.SelectKS.Requery
265 Me.frmDetailSubform.Requery
266 Me.SelectSEN.Requery
```

```
267 End Sub
268
269 Private Sub SelectStudent_AfterUpdate()
270     DoCmd.SetWarnings False
271     DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
272     DoCmd.OpenQuery "qryDetailAppend", acViewNormal, acEdit
273     DoCmd.SetWarnings True
274     Me.SelectAcYr.Requery
275     Me.SelectEthnicity.Requery
276     Me.SelectFSM.Requery
277     Me.SelectGender.Requery
278     Me.SelectRegGroup.Requery
279     Me.SelectSEN.Requery
280     Me.SelectYearGroup.Requery
281     Me.Select1stLanguage.Requery
282     Me.SelectKS.Requery
283     Me.frmDetailSubform.Requery
284     Me.SelectStudent.Requery
285 End Sub
286
287 Private Sub SelectYearGroup_AfterUpdate()
288     DoCmd.SetWarnings False
289     DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
290     DoCmd.OpenQuery "qryDetailAppend", acViewNormal, acEdit
291     DoCmd.SetWarnings True
292     Me.SelectAcYr.Requery
293     Me.SelectEthnicity.Requery
294     Me.SelectFSM.Requery
295     Me.SelectGender.Requery
296     Me.SelectRegGroup.Requery
297     Me.SelectSEN.Requery
298     Me.SelectStudent.Requery
299     Me.Select1stLanguage.Requery
300     Me.SelectKS.Requery
301     Me.frmDetailSubform.Requery
302     Me.SelectYearGroup.Requery
303 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmDetailSubform"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Collapse_Click()
13 Call fncCollapse
14 End Sub
```


Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmDetailSubformShort"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Expand_Click()
13 Call fncExpand
14 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmEditStudent"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Image27_Click()
13 DoCmd.OpenForm "frmEnterPasswordDeleteSingle", acNormal, "", "", , acDialog
14 End Sub
15
16 Private Sub Image29_Click()
17 DoCmd.RunCommand acCmdSaveRecord
18 DoCmd.Close
19 End Sub
20
21 Private Sub Image44_Click()
22 Me.Undo
23 DoCmd.Close
24 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmEnterDataSEN"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Private Sub Form_Close()
11 Dim rs As DAO.Recordset
12 Dim db As Database
13 Dim strSQL As String
14
15 Set db = CurrentDb
16 strSQL = "SELECT tblStudent.NoSEN " & _
17         "FROM tblStudent " & _
18         "WHERE (((tblStudent.NoSEN)=False) AND ((tblStudent.PMLD)=False) AND
((tblStudent.SLD)=False) AND ((tblStudent.MLD)=False) AND ((tblStudent.SpLD)=False) AND
((tblStudent.SpLangComm)=False) AND ((tblStudent.ASD)=False) AND
((tblStudent.Visual)=False) AND ((tblStudent.Hearing)=False) AND ((tblStudent.MSI)=False)
AND ((tblStudent.EBSD)=False) AND ((tblStudent.Physical)=False) AND
((tblStudent.Other)=False));"
19 Set rs = db.OpenRecordset(strSQL)
20
21 If rs.RecordCount = 0 Then
22 Else
23   rs.MoveLast
24   If rs.RecordCount > 0 Then
25     MsgBox "There are " & rs.RecordCount & " record(s) with no data entered for them, these
will be changed to 'No SEN'.", vbInformation, fncAppName
26     DoCmd.SetWarnings False
27     DoCmd.OpenQuery "qryUpdateNoSEN", acViewNormal, acEdit
28     DoCmd.SetWarnings True
29     DoCmd.OpenForm "frmStartup", acNormal, "", "", , acNormal
30   Else
31     End If
32 End If
33 Set rs = Nothing
34 Set db = Nothing
35 DoCmd.SetWarnings False
36 DoCmd.OpenQuery "qryEnterDataSENTempDelete", acViewNormal, acEdit
37 DoCmd.OpenQuery "qryEnterDataSENTempAppendAll", acViewNormal, acEdit
38 DoCmd.SetWarnings True
39 End Sub
40
41 Private Sub Image21_Click()
42 DoCmd.OpenForm "frmEnterPasswordSEN", acNormal, "", "", , acDialog
43 End Sub
44
45 Private Sub Image25_Click()
```

```
46 DoCmd.Close
47 DoCmd.SetWarnings False
48 DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
49 DoCmd.OpenQuery "qryDetailAppendAll", acViewNormal, acEdit
50 DoCmd.SetWarnings True
51 DoCmd.OpenForm "frmDetail", acNormal, "", "", , acNormal
52 End Sub
53
54 Private Sub Image44_Click()
55 DoCmd.Close
56 Dim rs As DAO.Recordset
57 Dim db As Database
58 Dim strSQL As String
59
60 Set db = CurrentDb
61 strSQL = "SELECT tblStudent.NoSEN " & _
62     "FROM tblStudent " & _
63     "WHERE (((tblStudent.NoSEN)=False) AND ((tblStudent.PMLD)=False) AND
((tblStudent.SLD)=False) AND ((tblStudent.MLD)=False) AND ((tblStudent.SpLD)=False) AND
((tblStudent.SpLangComm)=False) AND ((tblStudent.ASD)=False) AND
((tblStudent.Visual)=False) AND ((tblStudent.Hearing)=False) AND ((tblStudent.MSI)=False)
AND ((tblStudent.EBSD)=False) AND ((tblStudent.Physical)=False) AND
((tblStudent.Other)=False));"
64 Set rs = db.OpenRecordset(strSQL)
65
66 If rs.RecordCount = 0 Then
67     DoCmd.OpenForm "frmStartup", acNormal, "", "", , acNormal
68 Else
69     rs.MoveLast
70     If rs.RecordCount > 0 Then
71         MsgBox "There are " & rs.RecordCount & " record(s) with no data entered for them, these
will be changed to 'No SEN'.", vbInformation, fncAppName
72         DoCmd.SetWarnings False
73         DoCmd.OpenQuery "qryUpdateNoSEN", acViewNormal, acEdit
74         DoCmd.SetWarnings True
75         DoCmd.OpenForm "frmStartup", acNormal, "", "", , acNormal
76     Else
77     End If
78 End If
79 Set rs = Nothing
80 Set db = Nothing
81 DoCmd.SetWarnings False
82 DoCmd.OpenQuery "qryEnterDataSENTempDelete", acViewNormal, acEdit
83 DoCmd.OpenQuery "qryEnterDataSENTempAppendAll", acViewNormal, acEdit
84 DoCmd.SetWarnings True
85 End Sub
86
87 Private Sub Image45_Click()
88     Dim strUPN As String
89     DoCmd.OpenForm "frmAddStudent", , , , acFormAdd, acDialog
90     If Forms!frmAddStudent.Tag <> "Cancel" Then
91         strUPN = Forms!frmAddStudent!UPN
92         Application.Echo False
93         Forms!frmEnterDataSEN.frmEnterDataSENSubform.Form.Requery
94         With Forms!frmEnterDataSEN.frmEnterDataSENSubform.Form.RecordsetClone
```

```
95     .FindFirst "[UPN]=" & strUPN & ""
96     If Not .NoMatch Then
97         If Forms!frmEnterDataSEN.frmEnterDataSENSubform.Form.Dirty Then
98             Forms!frmEnterDataSEN.frmEnterDataSENSubform.Form.Dirty = False
99         End If
100        Forms!frmEnterDataSEN.frmEnterDataSENSubform.Form.Bookmark = .Bookmark
101    End If
102    End With
103    Application.Echo True
104 End If
105 DoCmd.Close acForm, "frmAddStudent"
106 End Sub
107
108 Private Sub Image48_Click()
109 DoCmd.RunCommand acCmdSaveRecord
110 Me.SelectRegGroup = "All"
111 Me.SelectYearGroup = "All"
112 Me.SelectSENStatus = "All"
113 Me.SelectAcYr = "All"
114 DoCmd.SetWarnings False
115 DoCmd.OpenQuery "qryEnterDataSENTempDelete", acViewNormal, acEdit
116 DoCmd.OpenQuery "qryEnterDataSENTempAppend", acViewNormal, acEdit
117 DoCmd.SetWarnings True
118 Me.SelectAcYr.Requery
119 Me.SelectRegGroup.Requery
120 Me.SelectSENStatus.Requery
121 Me.SelectYearGroup.Requery
122 Me.frmEnterDataSENSubform.Requery
123 End Sub
124
125 Private Sub Lock_Click()
126 DoCmd.RunCommand acCmdSaveRecord
127 DoCmd.SetWarnings False
128 DoCmd.OpenQuery "qryLockEnterSENData", acViewNormal, acEdit
129 DoCmd.SetWarnings True
130 Me.frmEnterDataSENSubform.Requery
131 MsgBox "Locked", vbInformation, "SEN Register"
132 End Sub
133
134 Private Sub SelectAcYr_AfterUpdate()
135 DoCmd.SetWarnings False
136 DoCmd.OpenQuery "qryEnterDataSENTempDelete", acViewNormal, acEdit
137 DoCmd.OpenQuery "qryEnterDataSENTempAppend", acViewNormal, acEdit
138 DoCmd.SetWarnings True
139 Me.SelectSENStatus.Requery
140 Me.SelectRegGroup.Requery
141 Me.SelectYearGroup.Requery
142 Me.frmEnterDataSENSubform.Requery
143 End Sub
144
145 Private Sub SelectRegGroup_AfterUpdate()
146 DoCmd.SetWarnings False
147 DoCmd.OpenQuery "qryEnterDataSENTempDelete", acViewNormal, acEdit
148 DoCmd.OpenQuery "qryEnterDataSENTempAppend", acViewNormal, acEdit
```

```
149 DoCmd.SetWarnings True
150 Me.SelectAcYr.Requery
151 Me.SelectSENStatus.Requery
152 Me.SelectYearGroup.Requery
153 Me.frmEnterDataSENSubform.Requery
154 End Sub
155
156 Private Sub SelectSENStatus_AfterUpdate()
157 DoCmd.SetWarnings False
158 DoCmd.OpenQuery "qryEnterDataSENTempDelete", acViewNormal, acEdit
159 DoCmd.OpenQuery "qryEnterDataSENTempAppend", acViewNormal, acEdit
160 DoCmd.SetWarnings True
161 Me.SelectAcYr.Requery
162 Me.SelectRegGroup.Requery
163 Me.SelectYearGroup.Requery
164 Me.frmEnterDataSENSubform.Requery
165 End Sub
166
167 Private Sub SelectYearGroup_AfterUpdate()
168 DoCmd.SetWarnings False
169 DoCmd.OpenQuery "qryEnterDataSENTempDelete", acViewNormal, acEdit
170 DoCmd.OpenQuery "qryEnterDataSENTempAppend", acViewNormal, acEdit
171 DoCmd.SetWarnings True
172 Me.SelectAcYr.Requery
173 Me.SelectSENStatus.Requery
174 Me.SelectRegGroup.Requery
175 Me.frmEnterDataSENSubform.Requery
176 End Sub
177
178 Private Sub Unlock_Click()
179 DoCmd.RunCommand acCmdSaveRecord
180 DoCmd.OpenForm "frmEnterPasswordEnterSENData", acNormal, "", "", , acDialog
181 DoCmd.RunCommand acCmdSaveRecord
182 Me.frmEnterDataSENSubform.Requery
183 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmEnterDataSENSubform"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Private Sub ASD_AfterUpdate()
11   If Me.ASD = True Then
12     Me.NoSEN = False
13   Else
14     End If
15   End Sub
16
17 Private Sub EBSD_AfterUpdate()
18   If Me.EBSD = True Then
19     Me.NoSEN = False
20   Else
21     End If
22   End Sub
23
24 Private Sub Forename_DbClick(Cancel As Integer)
25   If Me.Locked = True Then
26     MsgBox "You must unlock the record for editing before continuing.", vbInformation,
27     fncAppName
27     Exit Sub
28   Else
29     If Me.Dirty Then Me.Dirty = False
30     DoCmd.OpenForm "frmEditStudent", acNormal, "", "", acEdit, acDialog
31   End If
32 End Sub
33
34 Private Sub Form_Current()
35   If Me.Locked = True Then
36     Me.PMLD.Enabled = False
37     Me.PMLD.Locked = True
38     Me.SLD.Enabled = False
39     Me.SLD.Locked = True
40     Me.MLD.Enabled = False
41     Me.MLD.Locked = True
42     Me.SpLD.Enabled = False
43     Me.SpLD.Locked = True
44     Me.SpLangComm.Enabled = False
45     Me.SpLangComm.Locked = True
46     Me.ASD.Enabled = False
47     Me.ASD.Locked = True
48     Me.Visual.Enabled = False
49     Me.Visual.Locked = True
```

```
50 Me.Hearing.Enabled = False
51 Me.Hearing.Locked = True
52 Me.MSI.Enabled = False
53 Me.MSI.Locked = True
54 Me.EBSD.Enabled = False
55 Me.EBSD.Locked = True
56 Me.Physical.Enabled = False
57 Me.Physical.Locked = True
58 Me.Other.Enabled = False
59 Me.Other.Locked = True
60 Else
61 Me.PMLD.Enabled = True
62 Me.PMLD.Locked = False
63 Me.SLD.Enabled = True
64 Me.SLD.Locked = False
65 Me.MLD.Enabled = True
66 Me.MLD.Locked = False
67 Me.SpLD.Enabled = True
68 Me.SpLD.Locked = False
69 Me.SpLangComm.Enabled = True
70 Me.SpLangComm.Locked = False
71 Me.ASD.Enabled = True
72 Me.ASD.Locked = False
73 Me.Visual.Enabled = True
74 Me.Visual.Locked = False
75 Me.Hearing.Enabled = True
76 Me.Hearing.Locked = False
77 Me.MSI.Enabled = True
78 Me.MSI.Locked = False
79 Me.EBSD.Enabled = True
80 Me.EBSD.Locked = False
81 Me.Physical.Enabled = True
82 Me.Physical.Locked = False
83 Me.Other.Enabled = True
84 Me.Other.Locked = False
85 End If
86 End Sub
87
88 Private Sub Hearing_AfterUpdate()
89 If Me.Hearing = True Then
90 Me.NoSEN = False
91 Else
92 End If
93 End Sub
94
95 Private Sub Lock_Click()
96 DoCmd.SetWarnings False
97 DoCmd.OpenQuery "qryLockAll", acViewNormal, acEdit
98 DoCmd.SetWarnings True
99 Me.Requery
100 End Sub
101
102 Private Sub Locked_AfterUpdate()
103 If Me.Locked = True Then
```



```
104 Me.PMLD.Enabled = False
105 Me.PMLD.Locked = True
106 Me.SLD.Enabled = False
107 Me.SLD.Locked = True
108 Me.MLD.Enabled = False
109 Me.MLD.Locked = True
110 Me.SpLD.Enabled = False
111 Me.SpLD.Locked = True
112 Me.SpLangComm.Enabled = False
113 Me.SpLangComm.Locked = True
114 Me.ASD.Enabled = False
115 Me.ASD.Locked = True
116 Me.Visual.Enabled = False
117 Me.Visual.Locked = True
118 Me.Hearing.Enabled = False
119 Me.Hearing.Locked = True
120 Me.MSI.Enabled = False
121 Me.MSI.Locked = True
122 Me.EBSD.Enabled = False
123 Me.EBSD.Locked = True
124 Me.Physical.Enabled = False
125 Me.Physical.Locked = True
126 Me.Other.Enabled = False
127 Me.Other.Locked = True
128 Else
129 If Me.Dirty Then Me.Dirty = False
130 DoCmd.OpenForm "frmEnterPasswordSingle", acNormal, "", "", acEdit, acDialog
131 End If
132 End Sub
133
134 Private Sub MLD_AfterUpdate()
135 If Me.MLD = True Then
136 Me.NoSEN = False
137 Else
138 End If
139 End Sub
140
141 Private Sub MSI_AfterUpdate()
142 If Me.MSI = True Then
143 Me.NoSEN = False
144 Else
145 End If
146 End Sub
147
148 Private Sub NoSEN_AfterUpdate()
149 If Me.NoSEN = True Then
150 Me.ASD = False
151 Me.EBSD = False
152 Me.Hearing = False
153 Me.MLD = False
154 Me.MSI = False
155 Me.Physical = False
156 Me.PMLD = False
157 Me.SLD = False
```

```
158 Me.SpLangComm = False
159 Me.SpLD = False
160 Me.Visual = False
161 Else
162 End If
163 End Sub
164
165 Private Sub Other_AfterUpdate()
166 If Me.Other = True Then
167 Me.NoSEN = False
168 Else
169 End If
170 End Sub
171
172 Private Sub Physical_AfterUpdate()
173 If Me.Physical = True Then
174 Me.NoSEN = False
175 Else
176 End If
177 End Sub
178
179 Private Sub PMLD_AfterUpdate()
180 If Me.PMLD = True Then
181 Me.NoSEN = False
182 Else
183 End If
184 End Sub
185
186 Private Sub SENStatus_DbClick(Cancel As Integer)
187 If Me.Locked = True Then
188 MsgBox "You must unlock the record for editing before continuing.", vbInformation,
fncAppName
189 Exit Sub
190 Else
191 If Me.Dirty Then Me.Dirty = False
192 DoCmd.OpenForm "frmEditStudent", acNormal, "", "", acEdit, acDialog
193 End If
194 End Sub
195
196 Private Sub SLD_AfterUpdate()
197 If Me.SLD = True Then
198 Me.NoSEN = False
199 Else
200 End If
201 End Sub
202
203 Private Sub SpLangComm_AfterUpdate()
204 If Me.SpLangComm = True Then
205 Me.NoSEN = False
206 Else
207 End If
208 End Sub
209
210 Private Sub SpLD_AfterUpdate()
```

```
211 If Me.SpLD = True Then
212     Me.NoSEN = False
213 Else
214 End If
215 End Sub
216
217 Private Sub Surname_DblClick(Cancel As Integer)
218 If Me.Locked = True Then
219     MsgBox "You must unlock the record for editing before continuing.", vbInformation,
fncAppName
220     Exit Sub
221 Else
222     If Me.Dirty Then Me.Dirty = False
223     DoCmd.OpenForm "frmEditStudent", acNormal, "", "", acEdit, acDialog
224 End If
225 End Sub
226
227 Private Sub Unlock_Click()
228 DoCmd.OpenForm "frmEnterPasswordAll", acNormal, "", "", , acDialog
229 Me.Requery
230 End Sub
231
232 Private Sub UPN_DblClick(Cancel As Integer)
233 If Me.Locked = True Then
234     MsgBox "You must unlock the record for editing before continuing.", vbInformation,
fncAppName
235     Exit Sub
236 Else
237     If Me.Dirty Then Me.Dirty = False
238     DoCmd.OpenForm "frmEditStudent", acNormal, "", "", acEdit, acDialog
239 End If
240 End Sub
241
242 Private Sub Visual_AfterUpdate()
243 If Me.Visual = True Then
244     Me.NoSEN = False
245 Else
246 End If
247 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmEnterPasswordDeleteSingle"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Form_Open(Cancel As Integer)
13 Me.Caption = fncAppName
14 End Sub
15
16 Private Sub Image4_Click()
17 DoCmd.RunCommand acCmdSaveRecord
18 Dim rs As DAO.Recordset
19 Dim db As Database
20 Dim strSQL As String
21 Dim strPassword As String
22
23 Set db = CurrentDb
24 strSQL = "SELECT tblSystem.Password FROM tblSystem;"
25 Set rs = db.OpenRecordset(strSQL)
26 rs.MoveFirst
27
28 strPassword = rs.Fields("Password").value
29
30 Set rs = Nothing
31 Set db = Nothing
32
33 If Me.Password = strPassword Then
34 DoCmd.SetWarnings False
35 DoCmd.OpenQuery "qryDeleteStudentSingle", acViewNormal, acEdit
36 DoCmd.SetWarnings True
37 strMessageText = "Record deleted"
38 MsgBox strMessageText, vbInformation, fncAppName
39 DoCmd.Close
40 DoCmd.Close acForm, "frmEditStudent"
41 Forms!frmEnterDataSEN.frmEnterDataSENSubform.Form.Requery
42 Else
43 MsgBox "Incorrect Password, please try again", vbInformation, fncAppName
44 Me.Password = Null
45 Me.Password.SetFocus
46 End If
47 End Sub
48
49 Private Sub Image5_Click()
50 DoCmd.Close
```

51 End Sub

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmEnterPasswordEnterSENData"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Form_Open(Cancel As Integer)
13 Me.Caption = fncAppName
14 End Sub
15
16 Private Sub Image4_Click()
17 DoCmd.RunCommand acCmdSaveRecord
18 Dim rs As DAO.Recordset
19 Dim db As Database
20 Dim strSQL As String
21 Dim strPassword As String
22
23 Set db = CurrentDb
24 strSQL = "SELECT tblSystem.Password FROM tblSystem;"
25 Set rs = db.OpenRecordset(strSQL)
26 rs.MoveFirst
27
28 strPassword = rs.Fields("Password").value
29
30 Set rs = Nothing
31 Set db = Nothing
32
33 If Me.Password = strPassword Then
34 DoCmd.SetWarnings False
35 DoCmd.OpenQuery "qryUnlockEnterSENData", acViewNormal, acEdit
36 DoCmd.SetWarnings True
37 DoCmd.RunCommand acCmdClose
38 MsgBox "Unlocked", vbInformation, fncAppName
39 Else
40 MsgBox "Incorrect Password, please try again", vbInformation, fncAppName
41 Me.Password = Null
42 Me.Password.SetFocus
43 End If
44 End Sub
45
46 Private Sub Image5_Click()
47 DoCmd.RunCommand acCmdClose
48 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmEnterPasswordMaint"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Form_Open(Cancel As Integer)
13 Me.Caption = fncAppName
14 End Sub
15
16 Private Sub Image4_Click()
17
18 DoCmd.RunCommand acCmdSaveRecord
19 Dim rs As DAO.Recordset
20 Dim db As Database
21 Dim strSQL As String
22 Dim strPassword As String
23
24 Set db = CurrentDb
25 strSQL = "SELECT tblSystem.Password FROM tblSystem;"
26 Set rs = db.OpenRecordset(strSQL)
27 rs.MoveFirst
28
29 strPassword = rs.Fields("Password").value
30
31 Set rs = Nothing
32 Set db = Nothing
33
34 If Me.Password = strPassword Then
35 DoCmd.Close acForm, "frmStartup"
36 DoCmd.Close acForm, "frmEnterPasswordMaint"
37 DoCmd.OpenForm "frmMaintenance", acNormal, "", "", , acNormal
38 Else
39 MsgBox "Incorrect Password, please try again", vbInformation, fncAppName
40 Me.Password = Null
41 Me.Password.SetFocus
42 End If
43 End Sub
44
45 Private Sub Image5_Click()
46 DoCmd.RunCommand acCmdClose
47 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmEnterPasswordSEN"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Form_Open(Cancel As Integer)
13 Me.Caption = fncAppName
14 End Sub
15
16 Private Sub Image4_Click()
17
18 DoCmd.RunCommand acCmdSaveRecord
19 Dim rs As DAO.Recordset
20 Dim db As Database
21 Dim strSQL As String
22 Dim strPassword As String
23
24 Set db = CurrentDb
25 strSQL = "SELECT tblSystem.Password FROM tblSystem;"
26 Set rs = db.OpenRecordset(strSQL)
27 rs.MoveFirst
28
29 strPassword = rs.Fields("Password").value
30
31 Set rs = Nothing
32 Set db = Nothing
33
34 If Me.Password = strPassword Then
35 DoCmd.Close acForm, "frmEnterDataSEN"
36 DoCmd.Close acForm, "frmEnterPasswordSEN"
37 DoCmd.OpenForm "frmMaintenance", acNormal, "", "", , acNormal
38 Else
39 MsgBox "Incorrect Password, please try again", vbInformation, fncAppName
40 Me.Password = Null
41 Me.Password.SetFocus
42 End If
43 End Sub
44
45 Private Sub Image5_Click()
46 DoCmd.RunCommand acCmdClose
47 End Sub
```


Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmEnterPasswordSingle"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Form_Open(Cancel As Integer)
13 Me.Caption = fncAppName
14 End Sub
15
16 Private Sub Image4_Click()
17
18 DoCmd.RunCommand acCmdSaveRecord
19 Dim rs As DAO.Recordset
20 Dim db As Database
21 Dim strSQL As String
22 Dim strPassword As String
23
24 Set db = CurrentDb
25 strSQL = "SELECT tblSystem.Password FROM tblSystem;"
26 Set rs = db.OpenRecordset(strSQL)
27 rs.MoveFirst
28
29 strPassword = rs.Fields("Password").value
30
31 Set rs = Nothing
32 Set db = Nothing
33
34 If Me.Password = strPassword Then
35 DoCmd.SetWarnings False
36 DoCmd.OpenQuery "qryUnlockSingle", acViewNormal, acEdit
37 DoCmd.SetWarnings True
38 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!PMLD.Enabled = True
39 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!PMLD.Locked = False
40 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!SLD.Enabled = True
41 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!SLD.Locked = False
42 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!MLD.Enabled = True
43 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!MLD.Locked = False
44 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!SpLD.Enabled = True
45 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!SpLD.Locked = False
46 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!SpLangComm.Enabled =
    True
47 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!SpLangComm.Locked =
    False
48 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!ASD.Enabled = True
```

```
49 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!ASD.Locked = False
50 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!Visual.Enabled = True
51 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!Visual.Locked = False
52 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!Hearing.Enabled = True
53 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!Hearing.Locked = False
54 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!MSI.Enabled = True
55 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!MSI.Locked = False
56 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!EBSD.Enabled = True
57 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!EBSD.Locked = False
58 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!Physical.Enabled = True
59 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!Physical.Locked = False
60 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!Other.Enabled = True
61 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!Other.Locked = False
62 DoCmd.RunCommand acCmdClose
63 MsgBox "Unlocked", vbInformation, fncAppName
64 Else
65     MsgBox "Incorrect Password, please try again", vbInformation, fncAppName
66     Me.Password = Null
67     Me.Password.SetFocus
68 End If
69 End Sub
70
71 Private Sub Image5_Click()
72 [Forms]![frmEnterDataSEN]![frmEnterDataSENSubform].[Form]!Locked = True
73 DoCmd.RunCommand acCmdClose
74 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmEnterPasswordSummary"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Form_Open(Cancel As Integer)
13 Me.Caption = fncAppName
14 End Sub
15
16 Private Sub Image4_Click()
17
18 DoCmd.RunCommand acCmdSaveRecord
19 Dim rs As DAO.Recordset
20 Dim db As Database
21 Dim strSQL As String
22 Dim strPassword As String
23
24 Set db = CurrentDb
25 strSQL = "SELECT tblSystem.Password FROM tblSystem;"
26 Set rs = db.OpenRecordset(strSQL)
27 rs.MoveFirst
28
29 strPassword = rs.Fields("Password").value
30
31 Set rs = Nothing
32 Set db = Nothing
33
34 If Me.Password = strPassword Then
35 DoCmd.Close acForm, "frmDetail"
36 DoCmd.Close acForm, "frmEnterPasswordSummary"
37 DoCmd.OpenForm "frmMaintenance", acNormal, "", "", , acNormal
38 Else
39 MsgBox "Incorrect Password, please try again", vbInformation, fncAppName
40 Me.Password = Null
41 Me.Password.SetFocus
42 End If
43 End Sub
44
45 Private Sub Image5_Click()
46 DoCmd.RunCommand acCmdClose
47 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmEnterStarterLeaverDates"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Form_Open(Cancel As Integer)
13 Me.Caption = fncAppName
14 End Sub
15
16 Private Sub Image4_Click()
17 DoCmd.RunCommand acCmdSaveRecord
18 If IsNull(Me.FromDate) Or IsNull(Me.ToDate) Then
19     MsgBox "You must enter a start and end date", vbInformation, fncAppName
20     Me.FromDate.SetFocus
21 Else
22     Me.Visible = False
23     DoCmd.OpenReport "rptStartersLeavers", acViewReport, "", "", acNormal
24 End If
25 End Sub
26
27 Private Sub Image5_Click()
28 DoCmd.RunCommand acCmdClose
29 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmImported"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Form_Close()
13 DoCmd.SetWarnings False
14 DoCmd.OpenQuery "qryImportedDelete", acViewNormal, acEdit
15 DoCmd.SetWarnings True
16 End Sub
17
18 Private Sub Image44_Click()
19 DoCmd.Close
20 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmImportSubform"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Browse_Click()
13 Me.FileLocation = fncFileToImport
14 End Sub
15
16 Private Sub Image23_Click()
17 DoCmd.RunCommand acCmdSaveRecord
18 If IsNull(Me.AcademicYear) Or IsNull(Me.FileLocation) Then
19     MsgBox "You must select a file and input an academic year to import data (format:
20**/**)", vbInformation, fncAppName
20 Else
21     strFileLocation = Me.FileLocation
22     DoCmd.TransferText acImportDelim, "SENiBits Import Specification", "tblImport",
strFileLocation, False, ""
23     DoCmd.SetWarnings False
24     DoCmd.OpenQuery "qryImportAppend", acViewNormal, acEdit
25     DoCmd.OpenQuery "qryImportDelete", acViewNormal, acEdit
26     DoCmd.OpenQuery "qryUpdateNullEthnicity", acViewNormal, acEdit
27     DoCmd.OpenQuery "qryUpdateNullSENStatus", acViewNormal, acEdit
28     DoCmd.OpenQuery "qryUpdateNullFirstLanguage", acViewNormal, acEdit
29     DoCmd.OpenQuery "qryUpdateNullRegGroup", acViewNormal, acEdit
30     DoCmd.OpenQuery "qryUpdateNullYearGroup", acViewNormal, acEdit
31     If IsNull(Me.PreviousAcademicYear) = False Then
32         DoCmd.OpenQuery "qryStudentPreviousYear", acViewNormal, acEdit
33         DoCmd.OpenQuery "qryStudentUpdateFromPreviousYear", acViewNormal, acEdit
34         DoCmd.OpenQuery "qryStudentPreviousYearDelete", acViewNormal, acEdit
35     Else
36     End If
37     DoCmd.SetWarnings True
38     Me.AcademicYear = Null
39     Me.FileLocation = Null
40     DoCmd.OpenForm "frmImported", acNormal, "", "", , acDialog
41 End If
42 End Sub
43
44 Private Sub Image25_Click()
45 DoCmd.RunCommand acCmdSaveRecord
46 If IsNull(Me.AcademicYear) Or IsNull(Me.FileLocation) Then
47     MsgBox "You must select a file and input an academic year to update data (format:
20**/**)", vbInformation, fncAppName
```

```
48 Else
49     strFileLocation = Me.FileLocation
50     DoCmd.TransferText acImportDelim, "SENiBits Import Specification", "tblImport",
strFileLocation, False, ""
51     DoCmd.SetWarnings False
52     DoCmd.OpenQuery "qryImportUpdate", acViewNormal, acEdit
53     DoCmd.OpenQuery "qryImportDelete", acViewNormal, acEdit
54     DoCmd.SetWarnings True
55     Me.AcademicYear = Null
56     Me.FileLocation = Null
57     MsgBox "Data updated", vbInformation, fncAppName
58 End If
59 End Sub
60
61 Private Sub Image33_Click()
62     Call BackUpBE
63 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmMaintenance"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Image17_Click()
13 Me.Subform.Visible = True
14 Me.Subform.SourceObject = "frmImportSubform"
15 Me.Subform.SetFocus
16 Me![Subform].Form![FileLocation].SetFocus
17 End Sub
18
19 Private Sub Image19_Click()
20 Me.Subform.Visible = True
21 Me.Subform.SourceObject = "frmAddLogoSubform"
22 Me.Subform.SetFocus
23 Me![Subform].Form![Logo].SetFocus
24 End Sub
25
26 Private Sub Image21_Click()
27 Me.Subform.Visible = True
28 Me.Subform.SourceObject = "frmBackEndSubform"
29 Me.Subform.SetFocus
30 Me![Subform].Form![NewLocation].SetFocus
31 End Sub
32
33 Private Sub Image23_Click()
34 Me.Subform.Visible = True
35 Me.Subform.SourceObject = "frmSystemSubform"
36 Me.Subform.SetFocus
37 Me![Subform].Form![Organisation].SetFocus
38 End Sub
39
40 Private Sub Image25_Click()
41 Me.Subform.Visible = True
42 Me.Subform.SourceObject = "frmDataMaintenanceSubform"
43 Me.Subform.SetFocus
44 Me![Subform].Form![SelectAcYr].SetFocus
45 End Sub
46
47 Private Sub Image28_Click()
48 DoCmd.Close
49 DoCmd.OpenForm "frmEnterDataSEN", acNormal, "", "", , acNormal
50 End Sub
```

```
51
52 Private Sub Image30_Click()
53 DoCmd.Close
54 DoCmd.SetWarnings False
55 DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
56 DoCmd.OpenQuery "qryDetailAppendAll", acViewNormal, acEdit
57 DoCmd.SetWarnings True
58 DoCmd.OpenForm "frmDetail", acNormal, "", "", , acNormal
59 End Sub
60
61 Private Sub Image44_Click()
62 DoCmd.Close
63 DoCmd.OpenForm "frmStartup", acNormal, "", "", , acNormal
64 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmPassword"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Command0_Click()
13 DoCmd.RunCommand acCmdClose
14 End Sub
15
16 Function PriMakeHiddenAttachedTable(strDatabaseName As String, _
17     strTableName As String, strAttachedTableName As String)
18
19     Dim db As DAO.Database
20     Dim td As DAO.TableDef
21
22     Set db = CurrentDb
23     Set td = db.CreateTableDef(strAttachedTableName)
24 If Mid$(strDatabaseName, 2, 1) = ":" Then
25     'connect mdb file
26     td.Connect = ";Database=" & strDatabaseName
27 Else
28     'connect odbc connection
29     td.Connect = "ODBC;" & strDatabaseName
30 End If
31
32     td.SourceTableName = strTableName
33     db.TableDefs.Append td
34
35
36     Set td = Nothing
37     Set db = Nothing
38
39 End Function
40
41 Private Sub Image4_Click()
42 DoCmd.RunCommand acCmdSaveRecord
43
44 DoCmd.SetWarnings False
45 DoCmd.OpenQuery "qryUpdateOrganisationStartup", acViewNormal, acEdit
46 DoCmd.SetWarnings True
47
48
49     'password
50 If IsNull(Me.Password) Or IsNull(Me.RePassword) Then
```

```
51   MsgBox "You must fill in all of the password fields.", vbInformation, fncAppName
52   Exit Sub
53 Else
54   If Me.Password <> Me.RePassword Then
55     MsgBox "The passwords do not match.", vbInformation, fncAppName
56     Exit Sub
57   Else
58     DoCmd.SetWarnings False
59     DoCmd.OpenQuery "qryUpdatePasswordStartup", acViewNormal, acEdit
60     DoCmd.SetWarnings True
61   End If
62 End If
63
64   DoCmd.RunCommand acCmdClose
65
66 End Sub
67
68 Private Sub Image5_Click()
69   MsgBox "You must create a system password before using the " & fncAppName,
vbInformation, fncAppName
70 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmReports"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Image44_Click()
13 DoCmd.Close
14 End Sub
15
16 Private Sub Image4_Click()
17 DoCmd.Close
18 DoCmd.OpenForm "frmEnterStarterLeaverDates", acNormal, "", "", , acNormal
19 End Sub
20
21 Private Sub Image46_Click()
22 DoCmd.Close
23 DoCmd.OpenForm "frmChart", acNormal, "", "", , acNormal
24 End Sub
25
26 Private Sub Image49_Click()
27 DoCmd.Close
28 DoCmd.OpenReport "rptStats", acViewReport, "", "", acNormal
29 End Sub
30
31 Private Sub Image50_Click()
32 DoCmd.Close
33 DoCmd.OpenReport "rptStatsYearGroup", acViewReport, "", "", acNormal
34 End Sub
35
36 Private Sub Image51_Click()
37 DoCmd.Close
38 DoCmd.OpenReport "rptStatsRegGroup", acViewReport, "", "", acNormal
39 End Sub
40
41 Private Sub Image6_Click()
42 DoCmd.Close
43 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmSetupLink"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Command0_Click()
13 DoCmd.RunCommand acCmdClose
14 End Sub
15
16 Function PriMakeHiddenAttachedTable(strDatabaseName As String, _
17     strTableName As String, strAttachedTableName As String)
18
19     Dim db As DAO.Database
20     Dim td As DAO.TableDef
21
22     Set db = CurrentDb
23     Set td = db.CreateTableDef(strAttachedTableName)
24 If Mid$(strDatabaseName, 2, 1) = ":" Then
25     'connect mdb file
26     td.Connect = ";Database=" & strDatabaseName
27 Else
28     'connect odbc connection
29     td.Connect = "ODBC;" & strDatabaseName
30 End If
31
32     td.SourceTableName = strTableName
33     db.TableDefs.Append td
34     Application.SetHiddenAttribute acTable, strAttachedTableName, True
35
36     Set td = Nothing
37     Set db = Nothing
38
39 End Function
40
41 Private Sub Command2_Click()
42 Me.NewLocation = fncFileToLink
43 End Sub
44
45 Private Sub Image4_Click()
46 'This keeps the same table name on the front end,
47 'even if it's different than the back end table name.
48 DoCmd.RunCommand acCmdSaveRecord
49 If IsNull(Me.NewLocation) Then
```

```
50  MsgBox "You must select the data file location before using the " & fncAppName & ".",  
    vbInformation, fncAppName  
51  Else  
52  Dim rs As Recordset  
53  Dim TblName As String  
54  Dim ForeignTblName As String  
55  Dim OldDbPath As String  
56  Dim NewDbPath As String  
57  Dim strSQL As String  
58  Dim QueryExpression As String  
59  
60  ' Only check if new path is a file location (second character is a :)  
61  If Mid$(NewLocation, 2, 1) <> ":" Then  
62      MsgBox "Invalid file path", vbInformation, fncAppName  
63      Exit Sub  
64  End If  
65  
66  strSQL = "SELECT MSysObjects.Name, MSysObjects.ForeignName INTO TempLink " & _  
67      "FROM MSysObjects " & _  
68      "WHERE (((MSysObjects.Name) Like 'tbl*') AND ((MSysObjects.Database) Is Not  
Null));"  
69  
70  
71  DoCmd.SetWarnings False  
72  DoCmd.RunSQL strSQL  
73  DoCmd.SetWarnings True  
74  
75  Set rs = CurrentDb.OpenRecordset("TempLink")  
76  NewDbPath = Me.NewLocation  
77  'relink the tables  
78  While Not rs.EOF  
79      TblName = rs!Name  
80      ForeignTblName = rs!ForeignName  
81      'delete old link  
82      DoCmd.DeleteObject acTable, TblName  
83      'make new link, retaining the front end name, even if the back end name is different  
84      Call PriMakeHiddenAttachedTable(NewDbPath, ForeignTblName, TblName)  
85      rs.MoveNext  
86  Wend  
87  
88  rs.Close  
89  
90  DoCmd.SetWarnings False  
91  DoCmd.RunSQL strSQL  
92  DoCmd.OpenQuery "qryUpdateRunOnce", acViewNormal, acEdit  
93  DoCmd.SetWarnings True  
94  
95  Set rs = CurrentDb.OpenRecordset("TempLink")  
96  rs.Close  
97  
98  
99  'set icon location  
100 Dim strIconPath As String  
101 strIconPath = CurrentProject.path & "\" & fncShortAppName & ".ico"  
102 Dim db As DAO.Database, prp As DAO.Property
```

```
103 On Error GoTo GotErr
104 Set db = CurrentDb
105 db.Properties("AppIcon") = strIconPath 'Attempt to assign
106 Application.RefreshTitleBar 'Update On Screen!
107 GotErr:
108 If Err.Number = 3270 Then 'property doesn't exist!
109 Set prp = db.CreateProperty("AppIcon", dbText, strIconPath)
110 db.Properties.Append prp 'New property set!
111 Else
112 'MsgBox "Error: " & Err.Number & vbCrLf & Err.Description
113 End If
114 Resume Next
115
116 DoCmd.RunCommand acCmdClose
117 DoCmd.OpenForm "frmStartup", acNormal, "", "", , acNormal
118 End If
119 End Sub
120
121 Private Sub Image5_Click()
122 MsgBox "You must select the data file location before using the " & "SEN Register",
vbInformation, "SEN Register"
123 DoCmd.Quit acExit
124 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmSplash"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Form_Open(Cancel As Integer)
13 DoCmd.ShowToolbar "Ribbon", acToolbarNo
14 End Sub
15
16 Private Sub Form_Timer()
17
18 Dim rs As DAO.Recordset
19 Dim db As Database
20 Dim strSQL As String
21
22 Set db = CurrentDb
23 strSQL = "SELECT tblRunOnce.Run, tblRunOnce.MachineName FROM tblRunOnce;"
24 Set rs = db.OpenRecordset(strSQL)
25 rs.MoveFirst
26
27 Call fncCompName
28
29 If rs.Fields("Run").value = False Or IsNull(rs.Fields("MachineName").value) Or
rs.Fields("MachineName").value <> fncCompName() Then
30 DoCmd.Close
31 DoCmd.OpenForm "frmSetupLink", acNormal, "", "", , acNormal
32 Else
33 DoCmd.Close
34 DoCmd.OpenForm "frmStartup", acNormal, "", "", , acNormal
35 End If
36
37 Set rs = Nothing
38 Set db = Nothing
39 End Sub
```


Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmStartup"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Form_Open(Cancel As Integer)
13   If IsNull(fncPassword) Then
14     DoCmd.OpenForm "frmPassword", acNormal, "", "", , acNormal
15   Else
16     End If
17   End Sub
18
19 Private Sub Image21_Click()
20   DoCmd.OpenForm "frmEnterPasswordMaint", acNormal, "", "", , acDialog
21   End Sub
22
23 Private Sub Image23_Click()
24   DoCmd.Close acForm, "frmStartup"
25   DoCmd.OpenForm "frmEnterDataSEN", acNormal, "", "", , acNormal
26   End Sub
27
28 Private Sub Image25_Click()
29   DoCmd.Close acForm, "frmStartup"
30   DoCmd.SetWarnings False
31   DoCmd.OpenQuery "qryDetailTempDelete", acViewNormal, acEdit
32   DoCmd.OpenQuery "qryDetailAppendAll", acViewNormal, acEdit
33   DoCmd.SetWarnings True
34   DoCmd.OpenForm "frmDetail", acNormal, "", "", , acNormal
35   End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Form_frmSystemSubform"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub ChangePassword_Click()
13 DoCmd.RunCommand acCmdSaveRecord
14 If IsNull(Me.OldPassword) Or IsNull(Me.NewPassword) Or IsNull(Me.ReNewPassword) Then
15     MsgBox "You must fill in all of the password fields.", vbInformation, fncAppName
16     Me.OldPassword.SetFocus
17     Exit Sub
18 Else
19     Dim rs As DAO.Recordset
20     Dim db As Database
21     Dim strSQL As String
22     Dim strPassword As String
23
24     Set db = CurrentDb
25     strSQL = "SELECT tblSystem.Password " & _
26         "FROM tblSystem;"
27     Set rs = db.OpenRecordset(strSQL)
28     rs.MoveFirst
29     strPassword = rs("Password").value
30     Set rs = Nothing
31     Set db = Nothing
32     If Me.OldPassword <> strPassword Then
33         MsgBox "The old password does not match the password stored in the database.",
34             vbInformation, fncAppName
34         Me.OldPassword.SetFocus
35         Exit Sub
36     Else
37         If Me.NewPassword <> Me.ReNewPassword Then
38             MsgBox "The new passwords do not match.", vbInformation, fncAppName
39             Me.NewPassword.SetFocus
40             Exit Sub
41         Else
42             DoCmd.SetWarnings False
43             DoCmd.OpenQuery "qryUpdatePassword", acViewNormal, acEdit
44             DoCmd.SetWarnings True
45             MsgBox "The password has been changed.", vbInformation, fncAppName
46             Me.OldPassword = Null
47             Me.NewPassword = Null
48             Me.ReNewPassword = Null
49             Me.NewHint = Null
```

```
50      Me.OldHint.Requery
51      End If
52      End If
53      End If
54      End Sub
55
56      Private Sub UpdateOrganisation_Click()
57      DoCmd.RunCommand acCmdSaveRecord
58      DoCmd.SetWarnings False
59      DoCmd.OpenQuery "qryUpdateOrganisation", acViewNormal, acEdit
60      DoCmd.SetWarnings True
61      MsgBox "The organisation name has been updated.", vbInformation, fncAppName
62      Me.Organisation = Null
63      Me.CurrentOrganisation.Requery
64      End Sub
```


Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Report_rptStartersLeavers"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Image44_Click()
13 DoCmd.Close acForm, "frmEnterStarterLeaverDates"
14 DoCmd.Close
15 End Sub
16
17 Private Sub Image79_Click()
18 On Error GoTo ErrHandler:
19 RunCommand acCmdPrint
20
21 ErrHandler:
22
23 Response = 0
24
25 End Sub
26
27 Private Sub Image86_Click()
28 On Error GoTo ErrHandler:
29 DoCmd.OutputTo acOutputReport, "rptStartersLeavers", "PDFFormat(*.pdf)", "", False, "", ,
acExportQualityPrint
30
31 ErrHandler:
32 If Err.Number = 2501 Or 2585 Then
33 Response = 0
34 End If
35 End Sub
36
37 Private Sub Report_Error(DataErr As Integer, Response As Integer)
38 Response = 0
39 End Sub
```


Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Report_rptStats"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Image44_Click()
13 DoCmd.Close
14 End Sub
15
16 Private Sub Image79_Click()
17 On Error GoTo ErrHandler:
18 RunCommand acCmdPrint
19
20 ErrHandler:
21
22 Response = 0
23
24 End Sub
25
26 Private Sub Image86_Click()
27 On Error GoTo ErrHandler:
28 DoCmd.OutputTo acOutputReport, "rptStats", "PDFFormat(*.pdf)", "", False, "", ,
acExportQualityPrint
29
30 ErrHandler:
31 If Err.Number = 2501 Then
32 Response = 0
33 End If
34 End Sub
35
36 Private Sub Report_Error(DataErr As Integer, Response As Integer)
37 Response = 0
38 End Sub
```

Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Report_rptStatsRegGroup"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Image44_Click()
13 DoCmd.Close
14 End Sub
15
16 Private Sub Image79_Click()
17 On Error GoTo ErrHandler:
18 DoCmd.RunCommand acCmdPrint
19
20 ErrHandler:
21
22 Response = 0
23 End Sub
24
25 Private Sub Image86_Click()
26
27 On Error GoTo ErrHandler:
28 DoCmd.OutputTo acOutputReport, "rptStatsRegGroup", "PDFFormat(*.pdf)", "", False, "", ,
acExportQualityPrint
29
30 ErrHandler:
31 If Err.Number = 2501 Then
32 Response = 0
33 End If
34 End Sub
35
36 Private Sub Report_Error(DataErr As Integer, Response As Integer)
37 Response = 0
38 End Sub
```


Code

```
1  VERSION 1.0 CLASS
2  BEGIN
3    MultiUse = -1 'True
4  END
5  Attribute VB_Name = "Report_rptStatsYearGroup"
6  Attribute VB_GlobalNameSpace = False
7  Attribute VB_Creatable = True
8  Attribute VB_PredeclaredId = True
9  Attribute VB_Exposed = False
10 Option Compare Database
11
12 Private Sub Image44_Click()
13 DoCmd.Close
14 End Sub
15
16 Private Sub Image79_Click()
17 On Error GoTo ErrHandler:
18 DoCmd.RunCommand acCmdPrint
19
20 ErrHandler:
21
22 Response = 0
23 End Sub
24
25 Private Sub Image86_Click()
26 On Error GoTo ErrHandler:
27 DoCmd.OutputTo acOutputReport, "rptStatsYearGroup", "PDFFormat(*.pdf)", "", False, "", ,
acExportQualityPrint
28
29 ErrHandler:
30 If Err.Number = 2501 Then
31 Response = 0
32 End If
33 End Sub
34
35 Private Sub Report_Error(DataErr As Integer, Response As Integer)
36 Response = 0
37 End Sub
```

Code

```
1 Attribute VB_Name = "modApplicationName"  
2 Option Compare Database  
3  
4 Public Function fncAppName() As String  
5 fncAppName = DLookup("ApplicationName", "tblSystem", "ID=1")  
6 End Function  
7  
8 Public Function fncShortAppName() As String  
9 fncShortAppName = Replace(DLookup("ApplicationName", "tblSystem", "ID=1"), " ", "")  
10 End Function
```

Code

```
1  Attribute VB_Name = "modBackEndLocation"
2  Option Compare Database
3
4  Public Function fncCurrentLocation()
5      Dim rs As DAO.Recordset
6      Dim db As Database
7      Dim strSQL As String
8
9      Set db = CurrentDb
10     strSQL = "SELECT MSysObjects.Database " & _
11             "FROM MSysObjects " & _
12             "GROUP BY MSysObjects.Database " & _
13             "HAVING (((MSysObjects.Database) Is Not Null));"
14     Set rs = db.OpenRecordset(strSQL)
15     rs.MoveFirst
16     fncCurrentLocation = rs("Database").value
17
18     Set rs = Nothing
19     Set db = Nothing
20
21 End Function
```

Code

```
1 Attribute VB_Name = "modBackup"
2 Option Explicit
3
4 Declare Function apiCopyFile Lib "kernel32" Alias "CopyFileA" _
5 (ByVal lpExistingFileName As String, _
6  ByVal lpNewFileName As String, _
7  ByVal bFailIfExists As Long) As Long
8
9 Sub CopyFile(SourceFile As String, DestFile As String)
10 '-----
11 ' PURPOSE: Copy a file on disk from one location to another.
12 ' ACCEPTS: The name of the source file and destination file.
13 ' RETURNS: Nothing
14 '-----
15 Dim Result As Long
16 If Dir(SourceFile) = "" Then
17     MsgBox Chr(34) & SourceFile & Chr(34) & _
18         " is not valid file name."
19 Else
20     Result = apiCopyFile(SourceFile, DestFile, False)
21 End If
22 End Sub
23
24 Function BackUpBE()
25
26 On Error GoTo Err_backup
27
28 Dim strNewBName As String
29 Dim strOldBName As String
30 Dim strDateStamp As String
31 Dim rs As DAO.Recordset
32 Dim db As Database
33 Dim strSQL As String
34
35 strSQL = "SELECT MSysObjects.Database " & _
36         "FROM MSysObjects " & _
37         "GROUP BY MSysObjects.Database " & _
38         "HAVING (((MSysObjects.Database) Is Not Null));"
39
40 Set db = CurrentDb
41 Set rs = db.OpenRecordset(strSQL)
42 rs.MoveFirst
43
44 strOldBName = rs.Fields("Database").value
45 strDateStamp = Format(Now, "yyyy_mm_dd_hh_mm_ss")
46 strNewBName = Left([strOldBName], InStr(1, [strOldBName], ".") - 1) & strDateStamp &
47 ".accdb"
48 'copy database
49 CopyFile strOldBName, strNewBName
```

50

51 **MsgBox "The back-end database has been backed up to " & strNewBName, vbInformation,**
"SEN Register"

52

53

54 **Set rs = Nothing**

55 **Set db = Nothing**

56

57 **Exit_Backup:**

58 **Exit Function**

59

60 **Err_backup:**

61 **MsgBox Err.Number & Err.Description**

62

63 **Resume Exit_Backup**

64

65 **End Function**

Code

```
1 Attribute VB_Name = "modExpandCollapse"  
2 Option Compare Database  
3  
4 Public Function fncExpand()  
5 Forms!frmDetail!frmDetailSubform.SourceObject = "frmDetailSubform"  
6 End Function  
7  
8 Public Function fncCollapse()  
9 Forms!frmDetail!frmDetailSubform.SourceObject = "frmDetailSubformShort"  
10 End Function
```

Code

```
1 Attribute VB_Name = "modExportToExcel"  
2 Option Compare Database  
3  
4 Public Function fncExportToExcel()  
5 DoCmd.OutputTo acTable, "tblDetailTemp", "MicrosoftExcelBiff8(*.xls)", "", True, "", 0  
6 End Function
```

Code

```
1 Attribute VB_Name = "modFileDialogs"
2 Option Compare Database
3
4 ***** Code Start *****
5 ' This code was originally written by Ken Getz.
6 ' It is not to be altered or distributed, 'except as part of an application.
7 ' You are free to use it in any application,
8 ' provided the copyright notice is left unchanged.
9 '
10 ' Code originally courtesy of:
11 ' Microsoft Access 95 How-To
12 ' Ken Getz and Paul Litwin
13 ' Waite Group Press, 1996
14 ' Revised to support multiple files:
15 ' 28 December 2007
16
17 Type tagOPENFILENAME
18     IStructSize As Long
19     hwndOwner As Long
20     hInstance As Long
21     strFilter As String
22     strCustomFilter As String
23     nMaxCustFilter As Long
24     nFilterIndex As Long
25     strFile As String
26     nMaxFile As Long
27     strFileTitle As String
28     nMaxFileTitle As Long
29     strInitialDir As String
30     strTitle As String
31     Flags As Long
32     nFileOffset As Integer
33     nFileExtension As Integer
34     strDefExt As String
35     ICustData As Long
36     lpfnHook As Long
37     lpTemplateName As String
38 End Type
39
40 Declare Function aht_apiGetOpenFileName Lib "comdlg32.dll" _
41     Alias "GetOpenFileNameA" (OFN As tagOPENFILENAME) As Boolean
42
43 Declare Function aht_apiGetSaveFileName Lib "comdlg32.dll" _
44     Alias "GetSaveFileNameA" (OFN As tagOPENFILENAME) As Boolean
45 Declare Function CommDlgExtendedError Lib "comdlg32.dll" () As Long
46
47 Global Const ahtOFN_READONLY = &H1
48 Global Const ahtOFN_OVERWRITEPROMPT = &H2
49 Global Const ahtOFN_HIDEREADONLY = &H4
50 Global Const ahtOFN_NOCHANGEDIR = &H8
```



```
51 Global Const ahtOFN_SHOWHELP = &H10
52 ' You won't use these.
53 'Global Const ahtOFN_ENABLEHOOK = &H20
54 'Global Const ahtOFN_ENABLETEMPLATE = &H40
55 'Global Const ahtOFN_ENABLETEMPLATEHANDLE = &H80
56 Global Const ahtOFN_NOVALIDATE = &H100
57 Global Const ahtOFN_ALLOWMULTISELECT = &H200
58 Global Const ahtOFN_EXTENSIONDIFFERENT = &H400
59 Global Const ahtOFN_PATHMUSTEXIST = &H800
60 Global Const ahtOFN_FILEMUSTEXIST = &H1000
61 Global Const ahtOFN_CREATEPROMPT = &H2000
62 Global Const ahtOFN_SHAREAWARE = &H4000
63 Global Const ahtOFN_NOREADONLYRETURN = &H8000
64 Global Const ahtOFN_NOTESTFILECREATE = &H10000
65 Global Const ahtOFN_NONETWORKBUTTON = &H20000
66 Global Const ahtOFN_NOLONGNAMES = &H40000
67 ' New for Windows 95
68 Global Const ahtOFN_EXPLORER = &H80000
69 Global Const ahtOFN_NODEREFERENCELINKS = &H100000
70 Global Const ahtOFN_LONGNAMES = &H200000
71
72 Function fncFileToLink()
73     Dim strFilter As String
74     Dim lngFlags As Long
75     strFilter = ahtAddFilterItem(strFilter, "SEN Register Data File (*.accdb)",
"SENRegisterDAT.accdb")
76
77     ' Uncomment this line to try the example
78     ' allowing multiple file names:
79     ' lngFlags = ahtOFN_ALLOWMULTISELECT Or ahtOFN_EXPLORER
80
81     Dim Result As Variant
82
83     Result = ahtCommonFileOpenSave(InitialDir:="C:\", _
84         Filter:=strFilter, FilterIndex:=3, Flags:=lngFlags, _
85         DialogTitle:="SEN Register")
86
87     If lngFlags And ahtOFN_ALLOWMULTISELECT Then
88         If IsArray(Result) Then
89             Dim i As Integer
90             For i = 0 To UBound(Result)
91                 MsgBox Result(i)
92             Next i
93         Else
94             Exit Function
95         End If
96     Else
97         fncFileToLink = Result
98     End If
99
100     ' Since you passed in a variable for lngFlags,
101     ' the function places the output flags value in the variable.
102     Debug.Print Hex(lngFlags)
103 End Function
```

```
104 Function fncFileToImport()
105     Dim strFilter As String
106     Dim lngFlags As Long
107     strFilter = ahtAddFilterItem(strFilter, "Text Files (*.txt)", "*.txt")
108
109     ' Uncomment this line to try the example
110     ' allowing multiple file names:
111     ' lngFlags = ahtOFN_ALLOWMULTISELECT Or ahtOFN_EXPLORER
112
113     Dim Result As Variant
114
115     Result = ahtCommonFileOpenSave(InitialDir:="C:\", _
116         Filter:=strFilter, FilterIndex:=3, Flags:=lngFlags, _
117         DialogTitle:=fncAppName)
118
119     If lngFlags And ahtOFN_ALLOWMULTISELECT Then
120         If IsArray(Result) Then
121             Dim i As Integer
122             For i = 0 To UBound(Result)
123                 MsgBox Result(i)
124             Next i
125         Else
126             Exit Function
127         End If
128     Else
129         fncFileToImport = Result
130     End If
131
132     ' Since you passed in a variable for lngFlags,
133     ' the function places the output flags value in the variable.
134     Debug.Print Hex(lngFlags)
135 End Function
136
137 Function GetOpenFile(Optional varDirectory As Variant, _
138     Optional varTitleForDialog As Variant) As Variant
139
140     ' Here's an example that gets an Access database name.
141     Dim strFilter As String
142     Dim lngFlags As Long
143     Dim varFileName As Variant
144
145     ' Specify that the chosen file must already exist,
146     ' don't change directories when you're done
147     ' Also, don't bother displaying
148     ' the read-only box. It'll only confuse people.
149     lngFlags = ahtOFN_FILEMUSTEXIST Or _
150         ahtOFN_HIDEREADONLY Or ahtOFN_NOCHANGEDIR
151     If IsMissing(varDirectory) Then
152         varDirectory = ""
153     End If
154     If IsMissing(varTitleForDialog) Then
155         varTitleForDialog = ""
156     End If
157
```

```
158 ' Define the filter string and allocate space in the "c"
159 ' string Duplicate this line with changes as necessary for
160 ' more file templates.
161 strFilter = ahtAddFilterItem(strFilter, _
162     "Access (*.mdb)", "*.MDB;*.MDA")
163
164 ' Now actually call to get the file name.
165 varFileName = ahtCommonFileOpenSave( _
166     OpenFile:=True, _
167     InitialDir:=varDirectory, _
168     Filter:=strFilter, _
169     Flags:=lngFlags, _
170     DialogTitle:=varTitleForDialog)
171 If Not IsNull(varFileName) Then
172     varFileName = TrimNull(varFileName)
173 End If
174 GetOpenFile = varFileName
175 End Function
176
177 Function ahtCommonFileOpenSave( _
178     Optional ByRef Flags As Variant, _
179     Optional ByVal InitialDir As Variant, _
180     Optional ByVal Filter As Variant, _
181     Optional ByVal FilterIndex As Variant, _
182     Optional ByVal DefaultExt As Variant, _
183     Optional ByVal FileName As Variant, _
184     Optional ByVal DialogTitle As Variant, _
185     Optional ByVal hwnd As Variant, _
186     Optional ByVal OpenFile As Variant) As Variant
187
188 ' This is the entry point you'll use to call the common
189 ' file open/save dialog. The parameters are listed
190 ' below, and all are optional.
191 '
192 ' In:
193 ' Flags: one or more of the ahtOFN_* constants, OR'd together.
194 ' InitialDir: the directory in which to first look
195 ' Filter: a set of file filters, set up by calling
196 ' AddFilterItem. See examples.
197 ' FilterIndex: 1-based integer indicating which filter
198 ' set to use, by default (1 if unspecified)
199 ' DefaultExt: Extension to use if the user doesn't enter one.
200 ' Only useful on file saves.
201 ' FileName: Default value for the file name text box.
202 ' DialogTitle: Title for the dialog.
203 ' hwnd: parent window handle
204 ' OpenFile: Boolean(True=Open File/False=Save As)
205 ' Out:
206 ' Return Value: Either Null or the selected filename
207 Dim OFN As tagOPENFILENAME
208 Dim strFileName As String
209 Dim strDialogTitle As String
210 Dim fResult As Boolean
211
```

```
212 ' Give the dialog a caption title.
213 If IsMissing(InitialDir) Then InitialDir = CurDir
214 If IsMissing(Filter) Then Filter = ""
215 If IsMissing(FilterIndex) Then FilterIndex = 1
216 If IsMissing(Flags) Then Flags = 0&
217 If IsMissing(DefaultExt) Then DefaultExt = ""
218 If IsMissing(FileName) Then FileName = ""
219 If IsMissing(DialogTitle) Then DialogTitle = ""
220 If IsMissing(hwnd) Then hwnd = Application.hWndAccessApp
221 If IsMissing(OpenFile) Then OpenFile = True
222 ' Allocate string space for the returned strings.
223 strFileName = Left(FileName & String(256, 0), 256)
224 strFileTitle = String(256, 0)
225 ' Set up the data structure before you call the function
226 With OFN
227     .lStructSize = Len(OFN)
228     .hwndOwner = hwnd
229     .strFilter = Filter
230     .nFilterIndex = FilterIndex
231     .strFile = strFileName
232     .nMaxFile = Len(strFileName)
233     .strFileTitle = strFileTitle
234     .nMaxFileTitle = Len(strFileTitle)
235     .strTitle = DialogTitle
236     .Flags = Flags
237     .strDefExt = DefaultExt
238     .strInitialDir = InitialDir
239     ' Didn't think most people would want to deal with
240     ' these options.
241     .hInstance = 0
242     .strCustomFilter = ""
243     .nMaxCustFilter = 0
244     .lpfnHook = 0
245     'New for NT 4.0
246     .strCustomFilter = String(255, 0)
247     .nMaxCustFilter = 255
248 End With
249 ' This will pass the desired data structure to the
250 ' Windows API, which will in turn it uses to display
251 ' the Open/Save As Dialog.
252 If OpenFile Then
253     fResult = aht_apiGetOpenFileName(OFN)
254 Else
255     fResult = aht_apiGetSaveFileName(OFN)
256 End If
257
258 ' The function call filled in the strFileTitle member
259 ' of the structure. You'll have to write special code
260 ' to retrieve that if you're interested.
261 If fResult Then
262     ' You might care to check the Flags member of the
263     ' structure to get information about the chosen file.
264     ' In this example, if you bothered to pass in a
265     ' value for Flags, we'll fill it in with the outgoing
```

```
266     ' Flags value.
267     If Not IsMissing(Flags) Then Flags = OFN.Flags
268     If Flags And ahtOFN_ALLOWMULTISELECT Then
269         ' Return the full array.
270         Dim items As Variant
271         Dim value As String
272         value = OFN.strFile
273         ' Get rid of empty items:
274         Dim i As Integer
275         For i = Len(value) To 1 Step -1
276             If Mid$(value, i, 1) <> Chr$(0) Then
277                 Exit For
278             End If
279         Next i
280         value = Mid(value, 1, i)
281
282         ' Break the list up at null characters:
283         items = Split(value, Chr(0))
284
285         ' Loop through the items in the "array",
286         ' and build full file names:
287         Dim numItems As Integer
288         Dim Result() As String
289
290         numItems = UBound(items) + 1
291         If numItems > 1 Then
292             ReDim Result(0 To numItems - 2)
293             For i = 1 To numItems - 1
294                 Result(i - 1) = FixPath(items(0)) & items(i)
295             Next i
296             ahtCommonFileOpenSave = Result
297         Else
298             ' If you only select a single item,
299             ' Windows just places it in item 0.
300             ahtCommonFileOpenSave = items(0)
301         End If
302     Else
303         ahtCommonFileOpenSave = TrimNull(OFN.strFile)
304     End If
305 Else
306     ahtCommonFileOpenSave = vbNullString
307 End If
308 End Function
309
310 Function ahtAddFilterItem(strFilter As String, _
311     strDescription As String, Optional varItem As Variant) As String
312
313     ' Tack a new chunk onto the file filter.
314     ' That is, take the old value, stick onto it the description,
315     ' (like "Databases"), a null character, the skeleton
316     ' (like "*.mdb;*.mda") and a final null character.
317
318     If IsMissing(varItem) Then varItem = "*.*"
319     ahtAddFilterItem = strFilter & _
```

```
320         strDescription & vbNullChar & _  
321         varItem & vbNullChar  
322 End Function  
323  
324 Private Function TrimNull(ByVal strItem As String) As String  
325     Dim intPos As Integer  
326  
327     intPos = InStr(strItem, vbNullChar)  
328     If intPos > 0 Then  
329         TrimNull = Left(strItem, intPos - 1)  
330     Else  
331         TrimNull = strItem  
332     End If  
333 End Function  
334  
335 Private Function FixPath(ByVal path As String) As String  
336     If Right$(path, 1) <> "\" Then  
337         FixPath = path & "\"  
338     Else  
339         FixPath = path  
340     End If  
341 End Function  
342  
343 ***** Code End *****  
344
```

Code

```
1  Attribute VB_Name = "modFilterNoD"
2  Option Compare Database
3
4  Public Sub NatureOfDifficultyFilter()
5  Dim strSQL As String
6  Dim strNoD As String
7  Dim strStudent As String
8
9  strNoD = Forms!frmDetail.SelectNoD
10 strAllNoD = Forms!frmDetail.SelectNoD
11
12 If strNoD = "Sp/Lang/Comm" Then
13 strNoD = "SpLangComm"
14 Else
15 End If
16
17 strSQL = "INSERT INTO tblDetailTemp ( ACYr, FirstName, LastName, FirstLanguage, Gender,
YearGroup, RegGroup, SENStatus, Ethnicity, FSM, UPN, PMLD, SLD, MLD, SpLD, SpLangComm,
ASD, Visual, Hearing, MSI, EBSD, Physical, Other, DOA, [Leaving Date], NoSEN ) " & _
18 "SELECT qryDetail4.ACYr, qryDetail4.FirstName, qryDetail4.LastName,
qryDetail4.FirstLanguage, qryDetail4.Gender, qryDetail4.YearGroup, qryDetail4.RegGroup,
qryDetail4.SENStatus, qryDetail4.Ethnicity, qryDetail4.FSM, qryDetail4.UPN, qryDetail4.PMLD,
qryDetail4.SLD, qryDetail4.MLD, qryDetail4.SpLD, qryDetail4.SpLangComm, qryDetail4.ASD,
qryDetail4.Visual, qryDetail4.Hearing, qryDetail4.MSI, qryDetail4.EBSD, qryDetail4.Physical,
qryDetail4.Other, qryDetail4.DOA, qryDetail4.[Leaving Date], qryDetail4.NoSEN " & _
19 "FROM qryDetail4 " & _
20 "WHERE (((qryDetail4." & strNoD & ") = True));"
21
22
23 DoCmd.RunSQL strSQL
24
25 End Sub
```

Code

```
1 Attribute VB_Name = "modGetCompName"  
2 Option Compare Database  
3  
4 Public Function fncCompName()  
5 fncCompName = Environ("computername")  
6 End Function
```


Code

```
1 Attribute VB_Name = "modPassword"  
2 Option Compare Database  
3  
4 Public Function fncPassword()  
5 fncPassword = DLookup("Password", "tblSystem", "ID=1")  
6 End Function
```

Code

```
1 Attribute VB_Name = "modShift"
2 Option Compare Database
3
4 Function ap_DisableShift()
5 'This function disable the shift at startup. This action causes
6 'the Autoexec macro and Startup properties to always be executed.
7
8 On Error GoTo errDisableShift
9
10 Dim db As DAO.Database
11 Dim prop As DAO.Property
12 Const conPropNotFound = 3270
13
14 Set db = CurrentDb()
15
16 'This next line disables the shift key on startup.
17 db.Properties("AllowByPassKey") = False
18
19 'The function is successful.
20 Exit Function
21
22 errDisableShift:
23 'The first part of this error routine creates the "AllowByPassKey
24 'property if it does not exist.
25 If Err = conPropNotFound Then
26 Set prop = db.CreateProperty("AllowByPassKey", _
27 dbBoolean, False)
28 db.Properties.Append prop
29 Resume Next
30 Else
31 MsgBox "Function 'ap_DisableShift' did not complete successfully."
32 Exit Function
33 End If
34
35 End Function
36
37 Function ap_EnableShift()
38 'This function enables the SHIFT key at startup. This action causes
39 'the Autoexec macro and the Startup properties to be bypassed
40 'if the user holds down the SHIFT key when the user opens the database.
41
42 On Error GoTo errEnableShift
43
44 Dim db As DAO.Database
45 Dim prop As DAO.Property
46 Const conPropNotFound = 3270
47
48 Set db = CurrentDb()
49
50 'This next line of code disables the SHIFT key on startup.
```

```
51 db.Properties("AllowByPassKey") = True
52
53 'function successful
54 Exit Function
55
56 errEnableShift:
57 'The first part of this error routine creates the "AllowByPassKey
58 'property if it does not exist.
59 If Err = conPropNotFound Then
60 Set prop = db.CreateProperty("AllowByPassKey", _
61 dbBoolean, True)
62 db.Properties.Append prop
63 Resume Next
64 Else
65 MsgBox "Function 'ap_DisableShift' did not complete successfully."
66 Exit Function
67 End If
68
69 End Function
70
```