

Editing Data in Access via Code

There are many ways to edit data using VBA code. Here are 4 examples of the different methods. All of these examples add a single row of data to a table. They are very simple examples to illustrate the syntax.

Objective

Add a row to the table "tbl_Items" with the following values :

ItemCode	A001
ItemDescription	Widget
ItemCost	10.23

Use a Query

Construct an append query that will add the row, e.g. "qry_AddRow"

Use the following within your code

```
Sub AddRow_Query ()
    'Opening an append query will execute the query
    DoCmd.OpenQuery QueryName:="qry_AddRow"
End Sub
```

Use SQL

Use the following within your code

```
Sub AddRow_SQL ()
    Dim stSQL As String
    'SQL string, only split over multiple line for layout purposes
    stSQL = "INSERT INTO tbl_Items " & _
           "( ItemCode, ItemDescription, ItemPrice ) " & _
           "SELECT 'A001', 'Widget', 10.23;"
    'Execute the SQL
    DoCmd.RunSQL stSQL
End Sub
```

If you need help to construct the SQL use the query designer to build the query. When the query does what you need look at the SQL view in design mode.

Note

Access will give warnings before you execute a query or run SQL. To turn off these warnings you need to add the following code before and after your code.

```
Sub AddRow_Query_NoWarnings ()
    DoCmd.SetWarnings WarningsOn:=False
    'Opening an append query will execute the query
    DoCmd.OpenQuery QueryName:="qry_AddRow"
    DoCmd.SetWarnings WarningsOn:=True
End Sub
```

Use DAO (Direct Access Objects)

This method allows you to manipulate each field in turn and although longer allows much more flexibility. It has been replaced in many places by ADODB.

```
Sub AddRow_DAO()
    Dim DB As DAO.Database
    Dim tblItems As DAO.Recordset
    'get the current database and recordset to work with
    Set DB = CurrentDb
    Set tblItems = DB.OpenRecordset("tbl_Items")
    'Add a new record and populate
    With tblItems
        .AddNew
        .Fields("ItemCode") = "A001"
        .Fields("ItemDescription") = "Widget"
        .Fields("ItemPrice") = 10.23
        .Update
    End With
    'close the recordset and database
    tblItems.Close
    DB.Close
End Sub
```

Use ADODB

This method is very similar to DAO, but can retrieve data from a wide variety of data sources.

```
Sub AddRow_ADODB()
    Dim cnn As ADODB.Connection
    Dim rsItems As New ADODB.Recordset
    'connect to current database and open recordset
    Set cnn = CurrentProject.Connection
    rsItems.Open Source:="tbl_Items", ActiveConnection:=cnn, _
        CursorType:=adOpenDynamic, LockType:=adLockOptimistic
    'Add a new record and populate
    With rsItems
        .AddNew
        .Fields("ItemCode") = "A001"
        .Fields("ItemDescription") = "Widget"
        .Fields("ItemPrice") = 10.23
        .Update
    End With
    'close the recordset and database
    rsItems.Close
    cnn.Close
End Sub
```

Note

For the DAO and ADODB to work you must have the correct libraries loaded. Your current libraries can be seen by going to References on the Tools menu.

For DAO you will need a reference named something similar to "DAO" or "Access Database Object".

For ADODB you will need a reference named "ActiveX Data Objects X.X Library"

Different versions of access load different libraries as standard.

