

# Adobe® LiveCycle™ Designer 7.0

## Providing interactive database lookup from forms

Follow this tip to learn how to use LiveCycle Designer to write data to and retrieve information from databases. Using these features, form designers can develop forms that allow users to blend manually entered information with enterprise data and display database record details in form fields based on selection criteria such as a key field. For example, people can use printable forms to view supplier details based on part numbers, or use an order form that populates fields such as color and size selectors based on model type.

The following steps describe how to create a form that allows a user to look up merchandise information by selecting an ID and clicking a button to refresh the information displayed. They also show you how to define data connection and add code to populate the list with part number.

**Note:** This tip assumes that you have a Microsoft® Access database file called purchase.mdb on your local computer. The database should contain a table called OfficeSupplies with the same field names and content as those in Table 1. A sample purchase.mdb and purchase.pdf are attached to this tip.

ID	PART_NO	DESCRIPTION	UNITPRICE
1	OS12324	Laser printer paper, letter, box of 5001 sheets	34.99
2	OS23561	Note pads, 4inx6in, box of 250	9.95
3	OS93851	Clear tape, 12mmx33mm, box of 100	19.99
4	OS40681	Letter size file folders, beige, 100	85.74
5	OS83955	Highlighting pens, blue, box of 12	8.56
6	OS11939	Felt stamp pad, black, 12	18
7	OS78869	Letter size white paper, box of 5000 sheets	50.23
14	OS22222	Cabinet	22
15	OS33333	Sansonite	33
17	OS4444	Sony walkman	23
19	OS54321	Cellular phone	111
20	OS6666	Tennis shoes	33

**Table 1:** Field names and sample data in the table of the attached database

- Complete the following steps to set up a Data Source Name (DSN) to facilitate connecting to the database file:
  - From the Windows Control Panel, select Administrative Tools > Data Sources (ODBC). Click the User DSN tab and select Add.
 

The Create New Data Source dialog box appears, allowing you to select a database driver.
  - Select Microsoft Access Driver (\*.mdb). Select Finish.
 

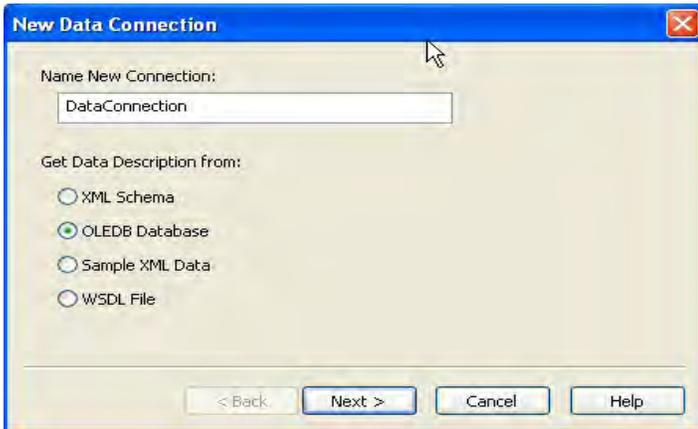
The ODBC Microsoft Access Setup dialog box appears.

- Enter a DSN and a Description.
  - Click Select and browse to the Access database that will be integrated with the form.
2. Launch LiveCycle Designer. Select File > New to create a new form design.
  3. Select File > New Data Connection to create a connection to the database.

The New Data Connection dialog box appears.

4. In the Name New Connection box, type `DataConnection`. Select OLEDB Database.

Your New Data Connection dialog box should look like this:



**Figure 1:** Specifying the new data connection's name and type

5. Select Next.

The OLEDB Connection dialog box appears.

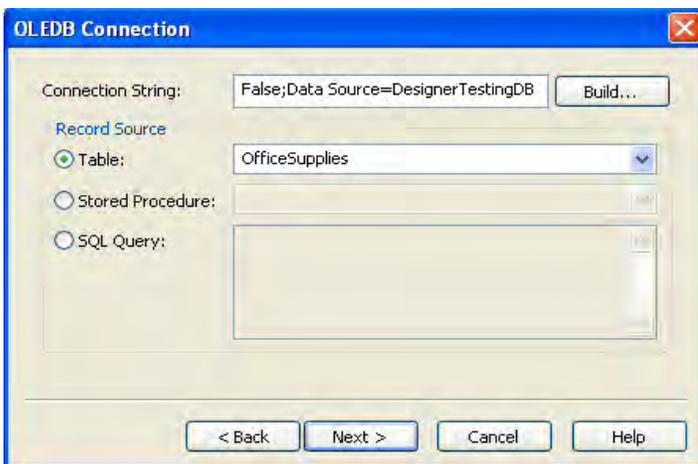
6. In the Connection String box, type the connection string, or select the Build button to build a connection to the database.

Assuming that `DesignerTestingDB` is the name of the DSN connection to the `Purchase.mdb` file, the connection string should be:

```
Provider=MSDASQL.1;Persist Security Info=False;Data Source=DesignerTestingDB
```

8. In the Record Source area, select Table. From the drop-down menu, select `OfficeSupplies`.

Your OLEDB Connection dialog box should look like this:



**Figure 2:** Specifying the connection string and record source

9. Select Next.

10. In the ADO Properties dialog box, accept the defaults and select Finish.

11. In the Library pallet of LiveCycle Designer 7.0, click the Standard tab. Add the following objects to the form design. On the Field tab and Binding tab of the Object pallet, customize each object as indicated in Table 2.

Object	Label	Name	Data format	Default binding
Drop-down List	Select ID	SelectField	Not Applicable	Normal
Button	Refresh	Not Applicable	Not Applicable	Not Applicable
Numeric Field	ID	ID	Float	<code>\$record.DataConnection.ID</code>
Text Field	Part Number	PART_NO	Not Applicable	<code>\$record.DataConnection.PART_NO</code>
Numeric Field	Unit Price	UNITPRICE	Float	<code>\$record.DataConnection.UNITPRICE</code>
Text Field	Description	DESCRIPTION	Not Applicable	<code>\$record.DataConnection.DESCRPTION</code>

Table 2: Object details in this example

Your form design should look like this:

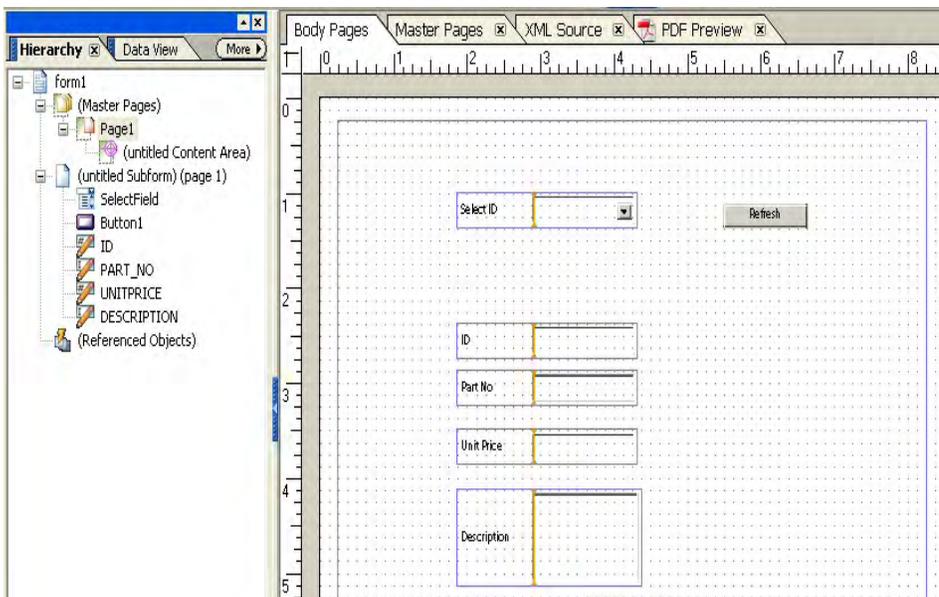


Figure 3: Objects in the form design

11. Select the Select ID object. In the Script Editor, choose the following:

- Show: initialize
- Language: JavaScript
- Run At: Client

12. In the Script editing field, type the following code:

```
/* This listbox object will populate two columns with data from a
data connection.

sDataConnectionName
- name of the data connection to get the data from.
- Note the data connection will appear in the Data View.

sColHiddenValue
- this is the hidden value column of the listbox.
- Specify the table column name used for populating.

sColDisplayText
- this is the display text column of the listbox.
- Specify the table column name used for populating.

These variables must be assigned for this script to run correctly.
*/
var sDataConnectionName = "DataConnection";
var sColHiddenValue = "ID";
var sColDisplayText = "PART_NO";

// Search for sourceSet node which matches the DataConnection name
var nIndex = 0;
while(xfa.sourceSet.nodes.item(nIndex).name != sDataConnectionName)
{
    nIndex++;
}
var oDB = xfa.sourceSet.nodes.item(nIndex);
oDB.open();
oDB.first();

// Search node with the class name "command"
nIndex = 0;
while(oDB.nodes.item(nIndex).className != "command")
{
    nIndex++;
}
// Need to set BOF and EOF to stay
oDB.nodes.item(nIndex).query.recordSet.setAttribute("stayBOF",
"bofAction");
oDB.nodes.item(nIndex).query.recordSet.setAttribute("stayEOF",
"eofAction");
```

```

// Search for the record node with the matching Data Connection name
nIndex = 0;
while(xfa.record.nodes.item(nIndex).name != sDataConnectionName)
{
    nIndex++;
}
var oRecord = xfa.record.nodes.item(nIndex);

// Find the value node
var oValueNode = null;
var oTextNode = null;
for(var nColIndex = 0; nColIndex < oRecord.nodes.length; nColIndex++)
{
    if(oRecord.nodes.item(nColIndex).name == sColHiddenValue)
    {
        oValueNode = oRecord.nodes.item(nColIndex);
    }
    else if(oRecord.nodes.item(nColIndex).name == sColDisplayText)
    {
        oTextNode = oRecord.nodes.item(nColIndex);
    }
}

while(!oDB.isEOF())
{
    this.addItem(oValueNode.value, oValueNode.value);
    //IDList.addItem(oValueNode.value, oTextNode.value);
    oDB.next();
}

// Close connection
oDB.close();

```

Your form design should look like this:

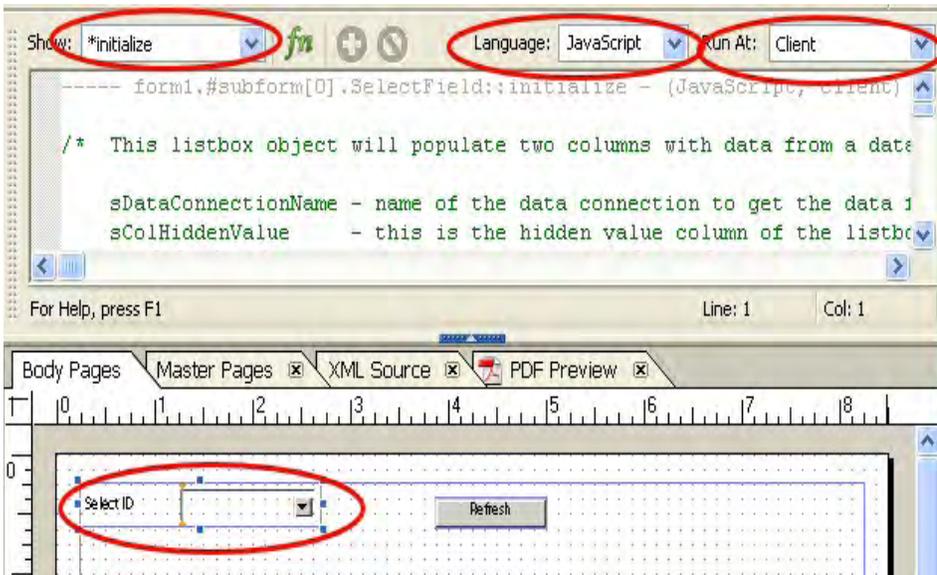


Figure 4: Adding JavaScript to populate the SelectField object

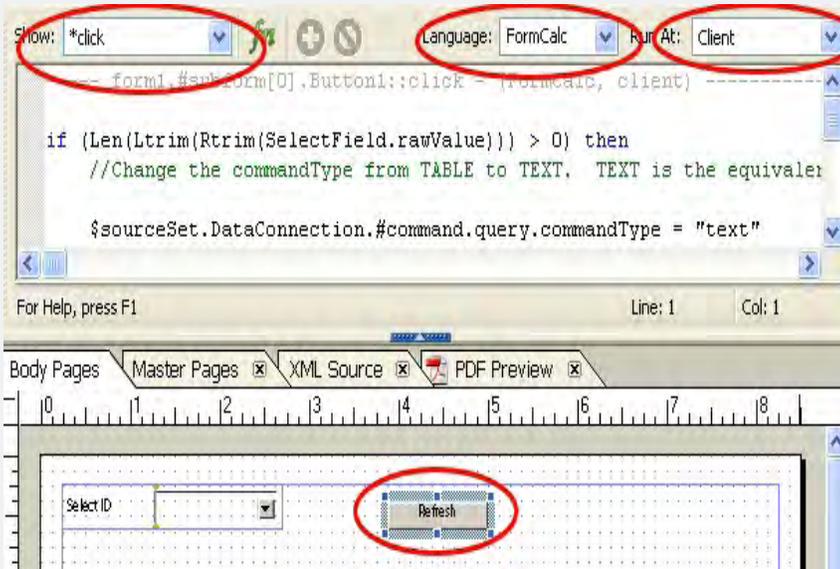
13. Select the Refresh object. In the Script Editor, choose the following:

- Show: click
- Language: FormCalc
- Run At: Client

14. In the Script editing area, type the following code:

```
if (Len(Ltrim(Rtrim(SelectField.rawValue))) > 0) then
    $sourceSet.DataConnection.#command.query.commandType = "text"
    $sourceSet.DataConnection.#command.query.select.nodes.
item(0).value = Concat("Select * from OfficeSupplies Where ID = ",
Ltrim(Rtrim(SelectField.rawValue)) ,",")
    //Reopen the Dataconnection
    $sourceSet.DataConnection.open()
endif
```

Your form design should look like this:



**Figure 5:** Adding FormCalc code to the Refresh button. This code repopulates each record field with new information from the database.

15. Click the PDF Preview tab in the Layout Editor.

You should be able to select an ID from the Select ID drop-down list and update the ID, Part Number, Unit Price, and Description fields to display appropriate details when you click Refresh.

16. Save the form design when you are satisfied.

## Better by Adobe.™

**Adobe Systems Incorporated**  
345 Park Avenue, San Jose, CA 95110-2704 USA  
[www.adobe.com](http://www.adobe.com)

Adobe, the Adobe logo, Adobe LiveCycle, and "Better by Adobe" are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries. Microsoft and Microsoft Office Access are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. All other trademarks are the property of their respective owners.

© 2005 Adobe Systems Incorporated. All rights reserved.  
Printed in the USA.

11/05

