

# MultiUser Guided Tour

(30 points, estimated time: 90 minutes)


## Purpose

This Guided Tour will provide an opportunity to work in a multi-user application within a split-file architecture. You will work with locking options available on bound forms. You'll use page-level locking and will Refresh and Requery a form's recordset. I presume you've already read and studied the assigned readings.

---

## Part 1 Getting Ready

---

1. Use the link in the Blackboard assignment to download two files to your **CIS217 Data Files\Others** folder:
  - MultiUser Demos BackEnd.accdb
  - MultiUser Demos FrontEnd (Left).accdb
2. Open the back-end database file, which has four tables. Two tables contain test data for customers and accounts (*tblAccount* and *tblCustomer*). The other two (*ztblAccount* and *ztblCustomer*) can be copied and pasted to obtain fresh copies of each test table.
3. Close the back-end database.
4. Now open *MultiUser Demos FrontEnd (Left).accdb*. Before going further, check your workstation's settings for whether row-level or page-level locking is in use as a database opened. Use *Office Button*  > *Access Options* > *Advanced*. Now scroll to the bottom. In the Advanced section notice whether the option *Open databases by using record level locking* is checked. If necessary, remove the check so Access will use **page-level** locking instead of row-level locking. Also, notice the value for *Refresh interval (sec)*. It is likely at its default value of 60. This setting determines the number of seconds after which Access automatically refreshes records in Datasheet or Form views. We'll refer to this setting later in this guided tour.
5. Launch *Word* (or another word processor) and create a new document with your name and MultiUser Guided Tour Answers in the top line. Save the document in your **\CIS217 Data Files\Others** folder as **MultiUser Guided Tour Answers - yournamehere.doc**. Keep *Word* open so you can type answers to the tasks below.

---

## Part 2: Linking to the Back End

---

6. Notice this front-end database has no tables! As a result, forms have no source of data and will fail when you try to open them. Double-click any form now, read the error message, then click OK to dismiss it.
7. You need to establish *links* to the tables in the back-end database file. Select *External Data* > *Access* > *Browse...* > navigate to *MultiUser Demos BackEnd.accdb* > click *Open* > click *Link to the data source by creating a linked table* > click *OK*. A *Link Tables* dialog appears. Select both *tblAccount* and *tblCustomer* > click *OK*. After some disk activity, you'll see two linked tables in the *Navigation Pane*.
8. Double-click *tblCustomer* to ensure the link to the back-end table operates correctly. Can you see customer records in a datasheet? (If not, return to the previous step.) Close *tblCustomer*.
9. Double-click the same form you attempted to open in Task 6. Can it now display customer records? Close the form.
10. Compact your front-end database. Close your front-end database.

---

### Part 3: MultiUser Operations

---

You can simulate *multiuser* operations on a single workstation by having two different instances of *Access* with each running the front-end database, with each front-end linked to the same back-end tables. The next steps accomplish this.

11. Copy *MultiUser Demos FrontEnd (Left).accdb* and rename the copy *MultiUser Demos FrontEnd (Right).accdb*.
12. Double-click *MultiUser Demos FrontEnd (Left).accdb*. If necessary, maximize the *Access* window.
13. Double-click *MultiUser Demos FrontEnd (Right).accdb*. If necessary, maximize the *Access* window.

You now have two separate instances of *Access* with each running a copy of the front-end database, with each linked to the same back-end database. Of course in a production environment, we'd place the back-end database file in a shared folder on the network so users could share the data tables as they work at their individual workstations.

14. Look at the *Taskbar* at the bottom of the screen. The Start button is on the left. You'll see two buttons for *Access*, one for each instance you've started. The *Taskbar* button for *Word* is there as well.
15. Launch Windows Explorer and view the contents of your **\CIS217 Data Files\Others** folder. Notice there are two (2) record locking information (.LAACB) files, one for each instance of the front-end database now open. [Note: if you don't see file extensions, use *Windows Explorer's View > Folder Options... > View tab > deselect the check box for Hide file extensions for known file types > then click OK.*] Notice there is not currently a record locking information file for the back-end since neither of the front-ends has an active connection to any of the tables in the back-end. Keep Windows Explorer open so you can easily view the contents of your **\CIS217 Data Files\Others** folder.

---

### Part 4: Exploring Locking Options with Bound Forms

---

As you know from your reading, each form has a *Record Locks* property which determines how records are locked and what happens when two users try to edit the same record at the same time using a *bound form*. The property can be set to *All Records*, *Edited Record*, or *No Locks*.

16. Switch into *Word* and **type Task 16 and the name and definition of each of these three property values** (*All Records*, *Edited Record*, or *No Locks*). Specifically, describe *what* gets locked and *when* each type of lock is attempted.
17. Switch into the Left instance. There are three forms and each form displays records from *tblCustomer*. As their names imply, one has its *Record Locks* set to *All Records*, another uses *Edited Record*, and the last uses *No Locks*. You'll start with the most restrictive, *All Records*.
18. Remain in the Left instance and open *frmCustomer-AllRecords* in *Design View* and check the value for the form's *Record Source* property (which is on the *Data* tab). The property confirms that this form displays records from *tblCustomer*. Now check that the *Record Locks* property is set to *All Records*. This property locks other users out of *tblCustomer* while this form is open. Switch to *Form View* and view the customer records.
19. Return to Windows Explorer and view your **\CIS217 Data Files\Others** folder. Notice there is now a record locking information file for the back-end since the Left front-end has an active connection to one of its tables.
20. Switch into the Right instance and open *frmCustomer-AllRecords*. You should receive an error message. Do you understand why the error occurred? Press [Alt] and [PrtScrn] keys simultaneously to copy the error message to the Clipboard, then switch into *Word* and **type Task 20 and the text of the error message and an explanation of why it occurred**. Then click *OK* to dismiss the error message.
21. Remain in the Right instance of *Access* and open *frmCustomer-Edited Record*. You should be able to open this form and view customer records. Try to edit the *City* field. You should hear a beep and see a message in the *Status Bar*. Return to *Word* and **type Task 21 and the text of the Status Bar and a detailed explanation of what the problem is**. Close *frmCustomer-Edited Record*.
22. Switch into the Left instance of *Access* and close *frmCustomer-AllRecords*.



23. Return to Windows Explorer and view your \CIS217 Data Files\Others folder. There is no longer a record locking information file for the back-end since the Left front-end no longer has an active connection to one of its tables.

---

## ***Part 5: Pessimistic Locking***

---

Next you'll use a form whose *Record Locks* property is set to *Edited Record*. This sets up *pessimistic* locking, which attempts a lock when a user begins to *edit* a record (or add a new record).

24. Remain in the Left instance and open *frmCustomer-EditedRecord*. The first record appears (CustomerID ALFKI). Once a user starts editing a record with this version of the form, the entire *4K page* containing the edited record will be locked. To demonstrate, change the first record's *City* to **Scottsdale**. Notice a  symbol appears in the form's *Record Selector*, indicating that the record is now dirty, locking the *4K page* containing this customer's record. Remain on this record *without* saving it.
25. Now switch into the Right instance and open *frmCustomer-EditedRecord*. Notice the *same* first record displays (CustomerID ALFKI). You'll see a  symbol in the form's *Record Selector*, indicating the *4K page* containing this record is locked. Also notice that the *City* field doesn't display Scottsdale (since the other user hasn't *saved* their changes yet). Try to edit the record by typing something in the *Company Name* field. You should hear a beep and will be unable to make changes to this record.
26. Remain in the Right instance and navigate to the second record. Try to edit this record. Do you hear another beep? Navigate to the third record. Can you edit it? How far down do you have to go before you can edit a record? Return to *Word* and **type Task 26 and the record number of the first record you're able to edit. Also type an explanation about why you weren't able to edit the previous several records.**
27. Return to the Right instance and use the Navigation Buttons to return to the first record. Try to edit it again. The page-level lock is still being held.
28. Switch to the Left instance. Save your changes (*City* is now Scottsdale) to the first record.
29. Switch to the Right instance. Press [F9] to Refresh the form's recordset. Does the form now display Scottsdale in the *City*? Try to change the *Country* to USA. Return to *Word* and **type Task 29 and answer the following questions: Can you now edit this record? Why can you edit the record now when you weren't able to before?**
30. Go to the Right instance and navigate to the last record. Remain on the last record.
31. Switch to the Left instance, and then navigate to the last record. Change the *Country* to USA. Remain on the last record without saving it.
32. Return to the Right instance and attempt to delete the last record. You'll get an error message indicating that you can't because the record is on a locked page. Remember that the Left instance is using a form whose *Record Locks* property set to *Edited Record*. Since that form is editing a record, the entire *4K page* containing the edited record was locked when that edit began, preventing your attempt to delete the record.
33. In the Right instance, click *OK* to dismiss the error message, and then close *frmCustomer - EditedRecord*.

---

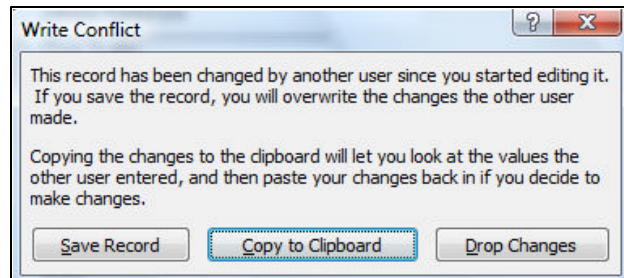
## ***Part 6: Optimistic Locking and Write Conflicts***

---

Now you'll use a form whose *Record Locks* property is set to *No Locks*. This employs *optimistic* locking, which attempts a lock when a user *saves* an edited a record.

34. Switch into the Left instance and close *frmCustomer - EditedRecord*. Then open *frmCustomer - NoLocks*. Since its *Record Locks* property has been set to *No Locks*, this form will attempt a lock when a user tries to save a dirty record.

35. Return to the Right instance and open *frmCustomer - NoLocks* there too. Remain on the first record, whose CustomerID is ALFKI.
36. Return to the Left instance. You are on the first record, whose CustomerID is ALFKI. Notice the contents of the *Contact Name* field. Type **your own name** in the *Contact Name* field and change the *Contact Title* to **Honcho**. A pencil symbol appears, indicating the record has been edited but not yet saved to disk. Since we're using *optimistic* locking, no lock is attempted until you try to *save* the record. Remain on this record without saving it.
37. Switch to the Right instance and change the *Contact Name* field to **Bob Along** and change the *Contact Title* to **Drone**. No conflict occurs because neither instance has tried to obtain a lock... yet.
38. Stay in the Right instance and save the record. Doing so locks the 4K page, saves your changed record (*Contact Name* now Bob Along and *Contact Title* to Drone) and releases the lock. This instance was able to successfully save its changes. The pencil symbol vanishes and is replaced by an arrowhead.
39. Switch into the Left instance. Notice your name is still displayed in the *Contact Name* field even though another user has changed and saved this record's data. To try to save your changes. You'll see a *Write Conflict* dialog. Drag it right so you can see the *Contact Name* field in your form. Read the message carefully, and then click *Copy to Clipboard*. This copies your entire record to the *Clipboard*. Return to Word, then **type Task 39 and answer the following question: What values are displayed in the *Contact Name* and *Contact Title* text boxes after you clicked *Copy to Clipboard*?**



40. Stay in the Left instance. You're still looking at the record the Right instance saved (*Contact Name* now Bob Along and *Contact Title* now Drone). Now you'll paste *your* changes back into the form. Right-click the arrowhead to select the entire record, then click *Paste* to retrieve your edited record from the *Clipboard*. Do you now see your name as the *Contact Name* and Honcho as the *Contact Title*? The pencil symbol reappears... since you just made the record dirty.
41. Stay in the Left instance and press [Esc] or Undo to abandon your pasted record. We'll let the Right instance's changes win this *Write Conflict* and allow Bob Along & Honcho to be stored. Close the form.

## Part 7: Refresh vs. Requery

The Access Help topic *Refresh a datasheet* states:

*If you are sharing a database over a network, other users could be changing data while you are viewing the same data in a datasheet. Microsoft Office Access updates the data that you see at specific intervals. However, you can immediately display the most current data by refreshing the records.*

**Refreshing** records only updates the data that is already in your datasheet. It does not reorder records, display new records, or remove deleted records and records that no longer meet specified criteria. To perform those tasks, you can **requery** the records.

Refresh and Requery are each available via the Ribbon's *Refresh* button, which contains choices *Refresh All* and *Refresh*. *Refresh All* actually performs a *Requery*. If you prefer to use keyboard shortcuts, press [F9] for Refresh and [Shift]+[F9] for Requery.



42. In the Left instance, open *frmCustomer - NoLocks*. You are on the first record (CustomerID ALFKI).
43. In the Right instance, open *frmCustomer - NoLocks*. You are on the first record (CustomerID ALFKI). Change the *Phone* field to your own phone number, then save the record.

44. Return to the Left instance and notice the *Phone* field's value. It likely remains unchanged. Earlier we saw the *Refresh interval* setting that controls how often Access *automatically* refreshes a form's recordset with the most current saved values. If this record's *Phone* field does not yet show your phone number (as entered and saved in the Right instance), wait up to 60 seconds and you should see it automatically refresh. We don't know how long ago Access performed the *previous* automatic refresh, so we may need to wait up to a minute for the *next* automatic refresh.
45. Remain in the Left session and type some other *Phone* number, then save the record.
46. Switch to the Right session. Rather than waiting for the next automatic refresh, press [F9] to immediately refresh. Do you now see the new phone number?
47. Still in the Right instance, navigate to the last record but don't make any changes. Remain on this last record. Switch into *Word* and **type Task 47 and type the number indicating how many records are currently in the form's recordset.**
48. Switch into the Left instance, navigate to the last record and delete it. You'll see the standard warning about deleting a record. Go ahead and delete the record. Leave the form open.
49. Return to the Right instance. Since the record this form was displaying has been deleted, you'll probably see something weird displayed in each bound control. If not, press [F9] now to *Refresh* the form's record source. Return to *Word* and **type Task 49 and the weird value displayed in each bound control.** Remain on this record.
50. Stay in the Right instance. Press [Shift]+[F9] to *Requery* the form's recordset. Return to *Word* and **type Task 50 and answer the following questions: how many total records are there and which record number are you now on?**

---

## ***Part 8: Wrapping Up***

---

51. Close each instance of *Access*.
52. Return to *Word*. You may want to refine your answers to Task 16 now that you've used each possible RecordLocks setting. Save your final answers. Exit *Word*.
53. Use *Windows Explorer* to backup your MultiUser Guided Tour Answers.doc file.