

SOLUTIONS MANUAL



Building Accounting Systems Using Access 2003

James T. Perry • Gary P. Schneider



Chapter 2

Introduction to Microsoft Access

Notes to the Instructor

This chapter introduces the Microsoft Access 2002 database management software. Students will learn about using tables, displaying database information, finding answers to questions with database queries, using forms, and printing database reports. Some students already have an adequate working knowledge of Microsoft Access and therefore this chapter may be unnecessary review for them. We suggest that you poll your students to determine their experience levels before proceeding. If you determine that the class has sufficient experience, it may be best to proceed to Chapter 3, Databases and Accounting Systems.

The Companion CD contains complete files that aid students in their learning process. We refer to these files in the process of completing the textbook exercises.

The many exercises presented in Chapter 2 convey the chapter's content via a hands-on approach. It is important that the students read and carefully follow each exercise. The exercises sometimes build on one another. The exercises are an important part of the carefully crafted student learning experience.

Lecture Outline

The pages that follow provide an outline of the contents of Chapter 2.

Objectives

- Describe the Access database management system in detail
- As a backdrop example, use a small fictitious brokerage firm to illustrate how an organization uses a database
- Important topics include the following:
 - Starting and exiting Access.
 - Understanding the Access work surface icons.
 - Using the Access objects, including tables, queries, forms, and reports.
 - Opening and displaying database tables.
 - Retrieving information by writing database queries.
 - Creating and using forms to display and query tables and databases.
 - Designing and using database reports.

Introduction

What is Access?

- A relational database management system that supports small to medium-sized database applications. Microsoft Corporation produces the database system.

What is a Relational Database?

- Dr. E.F. Codd, working for IBM at the time, is the acknowledged father of modern relational database management systems.
- A relational database system normally contains tables, queries, forms, and reports
- A table is a two dimensional object having rows and columns and resembling a worksheet
- Each column of a table lists a different characteristic known as an *attribute*
- A collection of tables that are related to one another form a database
- One table may represent employees' personal information, another may represent employees' skills, and another may represent employees' work history
- When a database consists of a single file, it is known as a *flat file*. (In reality, this situation *rarely* occurs.)

Starting Access

Exercise 2.1: Starting Microsoft Access. This exercise shows the students how to begin an Access session; it also describes the Access startup window (see Figure 2.1).

Finding Help

Exercise 2.2: Obtaining Help. This exercise shows students how to locate help. You might want to ensure that they understand how to seek context-specific help so that they can be self-sufficient when expert human advice is unavailable. Search for a couple of different terms to illustrate how help works. Also illustrate that the Access toolbar has *ToolTips*, which are displayed when the mouse hovers over any of the toolbar icons.

Printing Help

- Locate the help screen you would like to print
- Choose the desired help topic
- Click the Help page Print button and then click OK

Exiting Access

- Exiting Access updates all changes to the database, closes all databases, and returns to the Desktop or another suspended program
- Students have an especially difficult time understanding that Access is continually updating database objects and that it is not normally necessary to “save” data before exiting Access. Reassure the students that if an altered object has not yet been saved, Access will display a dialog box asking you if you want to save the changed object.
- Choose Exit from the File menu to leave Access

Examining the Access Environment

Access Work Surface

- File commands in the Startup window include:
 - New Database, which allows the user to create a database
 - Open Database, which makes available existing database objects such as tables, queries, forms, and so forth
 - Exit Access, which halts the Access program
- The menu bar for an open database contains the menu items File, Edit, View, Insert, Tools, Window, and Help (see and discuss Figure 2.4)
- File menu commands for an open or newly created database include:
 - Get External Data, Close, Save, Save As, Export
 - Page Setup, Print Preview, Print, Send To, and Database Properties
- The View menu lets you display different object types in the database window: tables, queries, forms, reports, Pages, Macros, or Modules
- The Insert menu allows you to insert tables, queries, forms, and other objects into the current database from a variety of sources
- The Tools menu provides spelling correction, the Relationships command, Security, and Options to set database-wide default parameters
- The Window menu allows you to arrange the work surface windows. If you have two tables and one form open in a given database, then you can select Tile Vertical from the Window menu to make all open windows, including the Database window, visible.
- The Help command provides context-sensitive help

Access Objects

- Objects are the structures you create and methods you employ to maintain and display your data
- Objects described in this chapter include tables, queries, forms, and reports
- Tables (see Figures 2.5 and 2.6, for example):
 - Hold all data in the database. (Stress to the students that only *tables* hold the data—no other of the Access objects—including queries, forms, or reports—hold data.)
 - Are two dimensional and have rows and columns
 - Row ordering is unimportant, as the rows can be sorted and rearranged and not change the fundamental table information
 - Column ordering is unimportant, meaning that the particular ordering of the columns bears no significance, and *any* table column may be placed in any particular position
- Queries (Figure 2.7 shows an example of a query that selects rows in which an employee's salary is larger than \$50,000):
 - Access supports several types of queries classified as either *action* queries or *selection* queries. Select queries are the most common type. They pose questions of the database and return answers in a *dynaset*
 - Subsets of rows are returned when selection criteria are specified to filter the data

- Forms (Figure 2.8):
 - Provide a way to view table data one row at a time
 - Facilitate data entry for inexperienced users
 - Display data from tables or queries but do not actually *hold* data
 - Have navigation buttons that speed movement from record to record
- Reports (Figure 2.9):
 - Provide formatted, hard-copy output
 - Display database information that can be supplied by tables, queries, or both
 - Can be customized to produce typeset-level output

Working with Databases and Tables

Note: Be sure to re-emphasize the “standard” procedure for working with the Companion CD databases: Copy the database from the Companion CD to the hard disk. Then, remind them to alter the copied database’s read-only property by right clicking the database, selecting Properties, and clearing the Read-only check box. Once students have copied and unprotected a database, they can work with the hard disk *copy*. You may want to show students exactly how to use Windows Explorer to locate and copy the database **Ch02.mdb**. Chapter 1 exercises illustrate how to do this also. Showing students these simple procedures may reduce or eliminate questions later. Otherwise, students may learn the hard way that an elaborate query they have just created cannot be stored on the currently open Companion CD database file!

Opening a Database

- Copy the Companion CD database to the hard disk of your computer
- **Ch02.mdb** contains the database for this chapter
- Remind students to alter the copied database’s read only property (see Exercise 1.14) before opening the database.

Exercise 2.3: Opening a Database. This exercise shows the students how locate and open a database on the Companion CD (Figures 2.10 and 2.11). Note: Data for this exercise and other exercises throughout the text are stored on the Companion CD that comes with the textbook. Please remind the students to copy the database **Ch02.mdb** found in the folder Ch02 to the hard disk before using the database. Please refer to the textbook exercise.

Looking at Your Data through Different Windows

- Data can be viewed and inspected through a Table (Datasheet) window, Form window, or a Report window—individually or simultaneously (Figure 2.12 displays Table and Form windows together)
- Object naming conventions for the text and Companion Disk include typing:
 - *tbl* preceding an object name for a table (e.g., *tblBroker*)
 - *qry* preceding an object name for a query (e.g., *qryInvoice*)
 - *frm* preceding an object name for a form (e.g., *frmClient*)
 - *rpt* preceding an object name for a report (e.g., *rptCustomer*)
- Emphasize that it is important to learn—memorize at first—these rules to conform to standard conventions and make dealing with objects consistent and uniform

Opening a Table

Exercise 2.4: Opening a Table. This exercise shows the students how to open the **tblClient** table (Figure 2.13) and examines the table Datasheet toolbar (Figure 2.14). The table **tblClient** is one of tables that contains information about a fictitious stock brokerage firm.

Moving Around a Table

- When you display a table, there are several ways to move from one record to the next. You can:
 - Select Edit, Go To, and then click First, Last, Next, Previous, or New Record
 - Use the keyboard PgUp, PgDn, or arrow up and down arrow keys
 - Use the table navigation buttons that appear in the bottom edge of the Table window
- A *row pointer* rests in the currently-selected row
- The leftmost column of the Table window is the *record selector column* (this column is not stored in the table)

Searching for a Value in a Column

- Use the Find command to locate a specific field value in a large table
- Exercise 2.5 shows you how

Exercise 2.5: Searching for a Row Containing a Particular Value. This exercise shows the student how to quickly locate a client's name in the previously opened **tblClient** table using a substring search argument. Please refer to the textbook exercise for details of how to do this.

- Discuss with students what happens if the search value cannot be found so they understand the difference between “successful” and “unsuccessful” searches

Changing a Table's Display Characteristics

- You can change a table's *display* characteristics by choosing one or more columns and then clicking the right mouse button; the table's structure remains unchanged
- If you close a table after changing its display characteristics (but not its structure), Access prompts you as to whether or not you want to save the table's display characteristics
- Tell students that saving altered table display characteristics isn't particularly important because doing so merely saves the way the table is portrayed on screen

Exercise 2.6: Changing a Table's Display Characteristics. This exercise shows students how to change a column's displayed position and width in the **tblClient** table (Figure 2.15). Remind students that changing a table's display characteristics including column positions, widths, etc. is a cosmetic change that does not affect the underlying table's column order or other properties. Please refer to the textbook exercise.

Sorting Table Rows

- You can locate a particular record much more easily when the table's records are sorted
- You can use *Quick sort* to organize a table on a single column
- Applying an *Advanced Filter/Sort* allows you to create a far more complex, multicolumn sort procedure

Exercise 2.7: Sorting a Table. This exercise shows students ways to sort the **tblClient** table in ascending order by the Last Name field. Please refer to the textbook exercise.

Printing a Table

- Print a table by selecting the table (open or not) in the Table window and then execute File, Print. You can also click the toolbar Print button (but this starts printing *immediately*)
- Choose any desired options in Print dialog box (Figure 2.16) and then click OK
- You can print detailed information about the definition of a table by executing a slightly more complicated sequence of steps. This sequence is: Tools, Analyze, Documentor; then check the name of the object whose structure you want to print and click OK. Finally, select Print from the File menu

Querying a Database

- Querying a database is posing a question which returns an answer in the form of a virtual table known by the name Microsoft Access-specific name of *dynaset* (see Figure 2.18)
- A query is a stored definition that specifies all tables involved in data retrieval, which columns are to be retrieved, which records are to be included in the result, and any calculations to be performed (see Figure 2.18)
- Most queries are called *selection* queries because they retrieve selected rows from tables
- Other queries do not return answers, but insert new records, delete existing records, update data, and create new table columns. These are known as *action queries*
- Access uses QBE (query by example) in which you give an example of the result you want
- Use *Hungarian notation* to name all queries: begin the name with *qry* as the query prefix followed by capitalized words representing the remainder of the name. For example, *qryFindHighSales* (without spaces)

Using a Query

Exercise 2.8: Running a Query. This exercise shows students how to open and run an existing query, *qryCaliforniaClients*. Remind students that they can run an existing query—one that is stored in the Queries area of the database—by selecting the query and clicking the Open button or by simply double-clicking the query name. (We prefer the latter because it involves fewer actions.) Please refer to the textbook exercise for further details.

Creating a One-Table Query

- Queries that restrict which rows are returned by using some criteria (e.g., customer invoices over 59 days old) are the basis of a fundamental relational database operation called *selection* (you specify which rows to select)
- You can form one-table queries by selecting a table and then placing column names in the QBE grid corresponding to the columns you wish to see. You can specify selection criteria to restrict the returned rows to those which satisfy the criteria

Exercise 2.9: Creating a One-Table Query. This exercise shows the students how to search for and select a specific customer from the *tblStock* table and return the information in a *dynaset* (Figures 2.19, 2.20, and 2.21).

Saving a Query

- Queries can be saved and re-executed periodically to produce current lists of clients, spare parts, invoiced over 60 days past due, etc., reflecting changes in the underlying table(s) over time
- With more than one query open, only the query in the *active* window is saved
- The dynaset *cannot* be saved directly, but you can turn the query generating the dynaset into a so-called *make table* query. Generally, however, you want to save only the query and not its results
- A Try It example encourages students to create and then save a query on their own. (You should encourage students to do all the “Try It” examples. Doing so will sharpen their skills)

Sorting the Results

- The normal, default order of dynaset rows is ascending order by the table’s primary key
- If a table has no primary key, then table rows are displayed in no particular order
- A dynaset can be sorted by any field or combination of fields by indicating which columns are the sort columns in the query’s definition
- Select and order columns according to sort criteria—the first column having sorting selected is primary sort column while subsequent columns similarly marked are secondary (and beyond) sort columns
- Select sort option in the sort row of the QBE grid from the drop-down list

Using More Complex Selection Criteria

- Queries that must satisfy more than one condition, *with all conditions being true*, use the AND operator (e.g., stock transactions AND how many occurred in January)
- A *projection* operation displays a subset of the columns of the table
- Figure 2.23 shows a list of comparison operators used by Access to compare two strings or numerical values
- Figure 2.24 displays the list of Access’ logical operators
- Comparison operators are <, >, =, <=, >=, and <>
- Logical operators are *And*, *Or*, and *Not*
- Queries can be created by joining any number of table pairs or queries

Exercise 2.10: Writing Selection Criteria Using an “AND” Operator. This exercise shows the students how to use the AND operator to query and project January’s stock transactions from the *tblStock* table (Figure 2.22). Be sure to point out to students that dates used in selection criteria must be enclosed in a pair of # signs.

Creating Selection Criteria Using the OR Operator

- When any one of several criteria is all that is required for a row to be displayed in a dynaset, then you use the OR operator. For example, you use OR to return rows from the *tblClient* table when any one stock of several possible stocks have been purchases (e.g., “which clients have purchased Intel or Microsoft stock recently?”).

Exercise 2.11: Forming an “OR” Query. This exercise shows students how to form a query involving two different fields using the *tblStock* table (Figure 2.25). Walk through this example in

class. It is a good and representative example of a relatively complex query involving only one table.

- Queries having two or more independent selection criteria (e.g., PurchPrice OR PurchStockName) have as many criteria rows in the query grid as there are individual criteria for which any one being true is sufficient to satisfy the condition
- Queries with alternate selection criteria on *one* field (e.g., “Biogen” Or “Microsoft” in the PurchStockName column): write both values (names in this example) in the same Criteria row, separating the two values with *Or*
- An asterisk following a criterion (e.g., “In*”) for character string criterion is called a *wild card character*, which allows extensions like *Inc.* and *International* both to be included in the answer
- Access ignores capitalization and locates matching rows based on spelling alone. This is a very important point that students often overlook. Emphasize that fact to them. It doesn’t matter, for instance, whether you specify the criterion *Microsoft*, *mICRosoFt*, or *microsoft*. All return a row in which the characters *microsoft* occur (spelled correctly), regardless of capitalization

Including Expressions in a Query

- Expressions are special “fields” that perform and display the results of calculations in a dynaset. Of course, expressions *cannot* be stored in a table, because a table stores only values (invariants), not expressions (unlike a spreadsheet)
- Query expressions are based on existing table or query fields
- Query expressions are also known as *virtual fields*
- A Try It example shows students a query, *qryStockValue* which calculates totals and displays a new, virtual column in the dynaset (Figure 2.26)

Printing Dynasets

Printing a dynaset is straightforward:

- Display the dynaset
- Check the dynaset to ensure it is what you expect
- Choose File, Print (or click the Print button on the toolbar)
- Make necessary selections in the print dialog box (Figure 2.16), then click OK

Note: students sometimes become frustrated when they cannot format numeric fields the way they want. For example, if a dynaset returns a sales amount, students frequently try (and fail) to find a way to left or right align the sales amount and format it to display dollar signs. Tell them that queries merely deliver data to reports and forms. They really are not meant to produce fancy, formatted results. Queries permit a limited amount of formatting, but it is a waste of time to do so.

Printing Query Definitions

- A query’s definition is a list of its design characteristics
- Select the Queries in the Objects bar, select Tools menu
- Choose Analyze
- Choose Documentor

- Select the object type whose definition is to be printed (click the Queries tab in this case)
- Click the particular object name's check box (you can select several objects if you want) whose definition you want to print
- Click Ok, and then select File, Print

Creating Action Queries

- Action queries are another group of queries
- Action queries create tables, remove records, update fields, or add new records to an existing table
- In short, action queries alter information as opposed to selection queries, which passively display information without altering anything

Make Table Query

- Create a table from a dynaset (a select query will do) by defining and executing a Make Table query
- A Try It exercise illustrates how to create a table from a query that extracts stock transaction records for a particular time period
- The Make table query creates a new table only when you click the Run button on the toolbar or select Run from the Query menu

Update Query

- Update queries allow you to make changes to one or more fields in one or more records in a table
- An update query's criteria determine which rows receive the changes, and the Update To QBE grid row indicates the exact change(s) to be made
- A Try It exercise alters (increases by 5%) the salaries of brokers who are listed in the table *tblBroker* (see Figure 2.28)
- Emphasize to students that most updates cannot be reversed (undone) with a simple command. Only another update query *may* be able to undo any mistakes. If a student places the constant "500" in the Update To cell of a column, all rows fitting the criteria are updated to 500. If that is a mistake, then there is no way to restore the changed column values.
- On the other hand, if an Update To expression such as [BrokerSalary]*1.2 changes the salary values incorrectly, that type of mistake can be reversed

Delete Query

- Delete queries delete individual rows from a table
- Delete queries do not delete a table (i.e., its structure is not removed from existence)
- Figure 2.29 shows an example of a delete query to remove all transactions occurring prior to a particular date

Append Query

- An append query *adds* rows from one table to the end of another table
- Figure 2.30 illustrates an append query
- Ensure that both the source and destination tables in an append query have the same type and length fields. Otherwise, problems occur

Using Forms

- Forms provide convenient, less cluttered work surfaces through which you can enter or alter information in tables and queries
- Forms can be created that facilitate data entry and mimic already familiar paper forms
- Forms facilitate validation checks on entered data

Viewing a Table through a Form

Exercise 2.12: Opening and Using a Form. This exercise shows students how to open and move around the *frmClient* form found in the database **Ch02.mdb** (Figure 2.31).

- Form navigation buttons allow movement through rows of table
- A form presents an intuitive and pleasant looking interface that is often familiar to a user

Viewing a Query through a Form

- A Form can be created for a saved query—*qryStockValue* in this case (Figure 2.32)
- The total number of records displayed in a form—one after another—depends on how many rows are produced by query

Creating a Form Quickly

- A Try It exercise shows students how to create a quick form using AutoForm (Access' default-style form) using the *tblBroker* table found in the **Ch02.mdb** database, which is stored on the Companion CD (Figure 2.33)
- You can alter the form created by AutoForm after it is generated. Chapter 5 presents those details. Students should simply know that the generated form could be redesigned. An AutoForm is a convenient way to create a baseline form design

Saving a Form

- You save a newly created form design, not the data displayed within it, by selecting Save from the File menu. Note that any altered data is preserved automatically by Access when you either close the form or move to another record.
- A Try It exercise shows students how to save the AutoForm created previously (based on the table *tblBroker*) as the form *frmBrokerAutoForm*

Editing Data with a Form

- It is often easier to alter data in a table by using a form view of the table data
- A form displays one record at a time
- In Form view, click the field you want to change, make any changes, and move to another record to post the changes. Alternatively, you cause the changes to be posted when you close the form

Exercise 2.13: Editing Data with a Form. This exercise shows students how to edit data using the *frmBroker* form. You can change a field by moving to it and then entering a new value.

- Remind students that changes made to a record are not posted to the database until after you move to another record. In the case of a form, simply press PgDn (or PgUp) or click Tab a sufficient number of times to move to the next record.

Querying a Database with a Form

- You can query a database through a form
- Such a query method is called *Filter By Form* or *Filter By Selection*

Exercise 2.14: Filtering Data through a Form. This exercise shows students how to open a form and then select and display only form data that matches criteria supplied by what you type in one of the form's fields. The form *frmStockValue* illustrates the technique (see Figure 2.34). Clicking the ClientID field, you can select a particular client identification number and then use *Filter By Form* to display only the selected client's records.

Printing a Form

- Print one record by locating it within the form (use navigation buttons) and selecting File, Print, choosing "Selected Record(s)" option button, and clicking OK in Print dialog box
- Print a range of records by clicking the Pages option button and then entering the beginning and ending page numbers to print

A frequent problem that arises when students want to print a form is that several forms usually appear on a page. The trick to printing only a single form is to place a page break just below the last field of the form (in design view). However, the students have not learned about the toolbox yet, so you can merely mention it at this point. Alternatively, selecting the form and choosing the "Selected Record(s)" option is still the best way to print a single form on a page.

Designing Reports

- You cannot enter or edit data in a report
- Reports are a way to produce hard copy output—to display database information—in an orderly manner
- Reports can be created ranging from simple utilitarian to professional designs with multiple type faces, drop shadows, graphics, and colors

Previewing a Report

Exercise 2.15: Loading and Previewing a Report. This exercise shows students how to load the existing report *rptClientStocks*, preview it, and close it (Figure 2.35).

Creating a Report Quickly

- An AutoReport, like an AutoForm, is Access' default (and simplest) report
- AutoReports can be based on a table or query.

Exercise 2.16: Creating a Report Quickly. This exercise shows students how to create an AutoReport using the table *tblClient* as the basis of the report (Figure 2.36), and modify it using the Design window (Figure 2.37).

Saving a Report

- From the Design Window, select File, Save or select File, Save As
- Select and enter the report's name, remembering to precede the name with the standard report prefix, *rpt* (e.g., *rptClientTest*)

Exercise 2.17: Saving a Report. This exercise shows students how to save a newly created report, *rptClientTest*. Simply select Save As from the File menu, type the report's name, and click OK. The report's design is saved among the Report objects.

Answers to Review Exercises

Multiple Choice Questions

- | | | | |
|----|---|-----|---|
| 1. | d | 6. | d |
| 2. | b | 7. | d |
| 3. | c | 8. | c |
| 4. | c | 9. | a |
| 5. | b | 10. | a |

Discussion Questions

The solutions presented here come from the textbook discussion. Your students may include other insight that is relevant, but not presented in our solutions.

1. Those icons represent shortcuts to commands accessed from the menu bar, they do not replace the menu commands. The Toolbar icons comprising both buttons and tools differ from one window to another. The toolbar buttons that are unavailable in a particular situation are dimmed. Buttons that are not dimmed may be pressed to rapidly accomplish various tasks. Those shown in Figure 2.1 are the startup Desktop buttons. If you were working with tables, then a different set of buttons associated with table operations would be found on the Toolbar. The icons shown in Figure 2.4 are Database window icons. They are quite different from the startup Desktop's buttons. You will use both the buttons and menus to create and modify various database objects as you read through this text. (For more information on menus and buttons, please see the discussion under *Access Work Surface*.)
2. Occasionally, it is better to work with data in your tables one row at a time. Tables are not a very intuitive interface for some people, especially those who are not computer literate. Access forms solve this problem. Forms let you see the data from a table in a format that is easier to understand. You can see one row or many rows of a table. Figure 2.8 shows an example of a form showing the employee data in a more attractive and intuitive environment. (For more information on forms, please see the discussion about forms.)
3. The Steps are as follows:
 - Execute Access by double-clicking the Access icon found in one of the Windows groups
 - Choose File, Open Database. You will see the Open Database dialog box. Select the location of your files, then the appropriate file, and choose OK
 - In the Database window select Tables in the Objects bar, select the appropriate table name, then choose Open (or, double-click the table name to open it in Datasheet view)

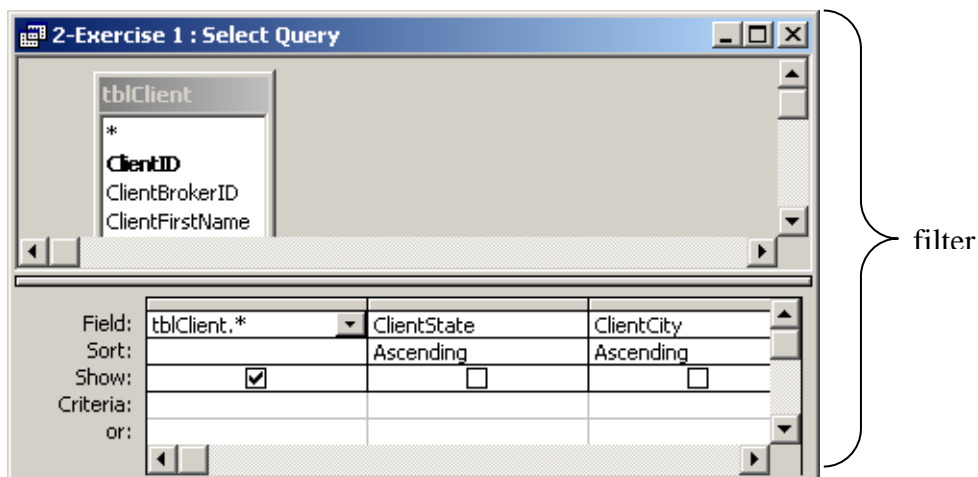
Note: Access provides several ways to view your data. You can inspect your data in a Table window (Datasheet view), which displays data in columns and rows—just like a spreadsheet's data. Or, you can use a Form window (choose AutoForm or create a New Form) to display one or more rows in a non-tabular format. Forms provide a attractive way to view and change data, because they can be designed to look like paper forms with which you are already familiar. Alternately, you can view your data in a report format (choose AutoReport or create a New Report) with the Report Window. The Report window provides a preview of a printed report so that you can review it like you would a hard copy report.

4. One of the real power capabilities of relational databases is the ability to ask questions that return interesting and meaningful answers derived from a database. Relational database systems make asking questions particularly easy, and Access is no exception. A *query*, the usual name for a question, can be simple or complex and can involve only one table, dozens of tables, or even hundreds of tables. In a query, you specify which tables are involved in the data retrieval operation, which columns are to be retrieved, which records are to be returned, and any calculations to be performed. Most queries are called *selection queries*, because they retrieve rows from tables. There are other types of queries that do not return answers. Those queries are used to insert new records into a table, delete existing records from a table, update data in one or more columns of a table, or create new table columns.
5. Producing reports could not be much easier than the click of a button or two. Frequently, you will want to either preview a report on screen or produce a printed report that you can pass around at a meeting or keep as a permanent record. Access reports are just that—reports. You can neither enter nor edit data in a report. You can create reports ranging from simple, utilitarian designs to professional looking reports replete with attractive typefaces, drop shadows, and graphics. Reports typically display information from a table, a collection of related tables, or a query. Though a lot can be gained from looking at the query's results on screen, it is even more useful to have a printed report. (For more information on reports, please review the *Designing Reports* section.)

Problems

Note: Ensure that students have copied the database **Ch02.mdb** to the hard disk *before* doing any or all of the following exercises. (Remind them that they cannot alter any data on a CD-ROM, because it is a read only medium.)

1. Open table *tblClient* and then select Filter, Advanced Filter/Sort from the Records menu. Place *ClientState* and *ClientCity* on the Field row of the QBE grid (in that order); then select “Ascending” for both fields in the Sort row. Finally, click the Apply Filter button, found on the toolbar, to view the resulting sorted table. The next figure illustrates the Filter and dynaset:



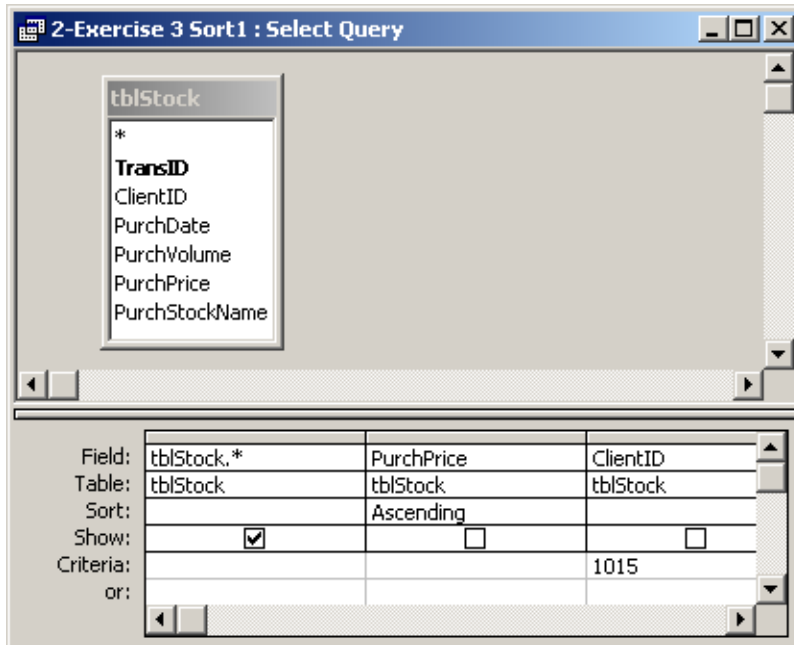
Client ID	Broker ID	First Name	Last Name	Street	City	State	ZIP Code
1010	702	Mark	Remes	456 Southern Accent Way	Little Rock	AR	88767
1013	702	John	Laskowski	9208 Goldcoast Drive	Alpine	CA	92014
1007	701	Marti	Mohler	431 University Street	Beverly Hills	CA	90210
1003	703	Joseph	Lateer	4908 Morena Blvd	Carlsbad	CA	92008
1015	701	Joseph	Nasbeth	1840 Orange Grove	El Cajon	CA	92020
1009	702	Joan	Kroc	1249 Via Sante Fe	La Jolla	CA	92022
1005	702	Betty	Karanski	8851 Deansworthy Street	La Mesa	CA	91941
1006	703	George	Hedrih	2083 Avenida Picante	Mt. Helix	CA	92020
1008	704	Edward	Jones	3821 Investments Heights Dr.	Newport Beach	CA	92221
1001	702	Paul	Liu	1234 Anywhere Street	San Diego	CA	92020
1012	704	Nancy	Truesdale	700 Baseline Street	Denver	CO	80302
1004	703	Gordon	Springer	4342 Tiger Paw Lane	Columbia	MO	65205
1002	701	Nancy	Carroll	6360 Eastshore Drive	Lincoln	NE	68508
1011	703	Paul	Mitchell	5101 Madison Avenue	Cincinnati	OH	45227
1014	704	Melinda	French	4101 Avocado Blvd	Seattle	WA	98101

- The exercise has been saved on the Instructor’s Companion CD (but *not* on the student’s version, of course) in the database **Ch02.mdb**. Click *Queries* on the Objects bar, select the query *2-Exercise 2*, and finally select File, Print to print the dynaset. The following figure shows the dynaset generated by the query.

First Name	Last Name	Client ID	Broker ID
Betty	Karanski	1005	702
Joan	Kroc	1009	702
John	Laskowski	1013	702
Paul	Liu	1001	702
Mark	Remes	1010	702

Field:	ClientFirstName	ClientLastName	ClientID	ClientBrokerID	ClientFirstName
Table:	tblClient	tblClient	tblClient	tblClient	tblClient
Sort:		Ascending			Ascending
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Criteria:				702	
or:					

3. You will find the answer to the exercise on the Instructor's Companion CD (but *not* on the student's version, of course) in the database **Ch02.mdb**. Click *Queries* on the Objects bar, select the query *2-Exercise 3 Sort1*, and finally select File, Print to print the dynaset. The second and third solutions are queries *2-Exercise 3 Sort2* and *2-Exercise 3 Sort3*, respectively. Open and print both of these queries. The six figures on the following pages show the Design view and Datasheet view for the preceding three sort assignments in this question.

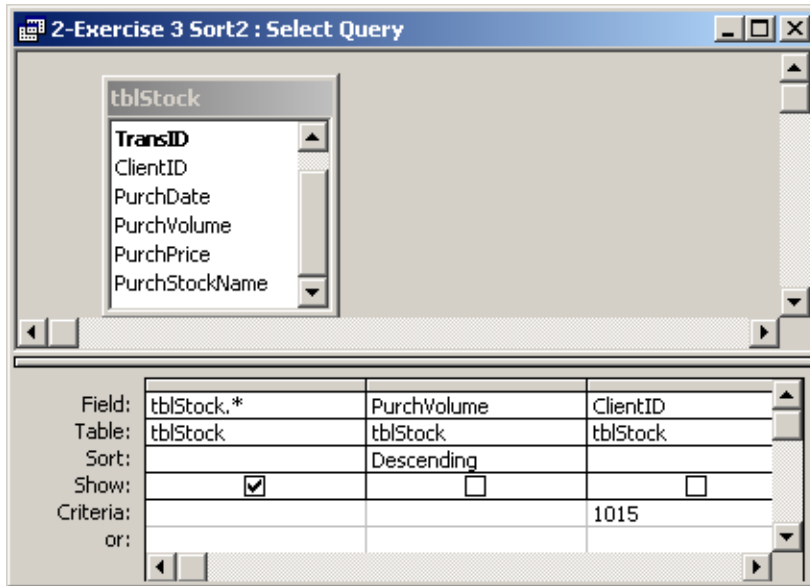


Design View

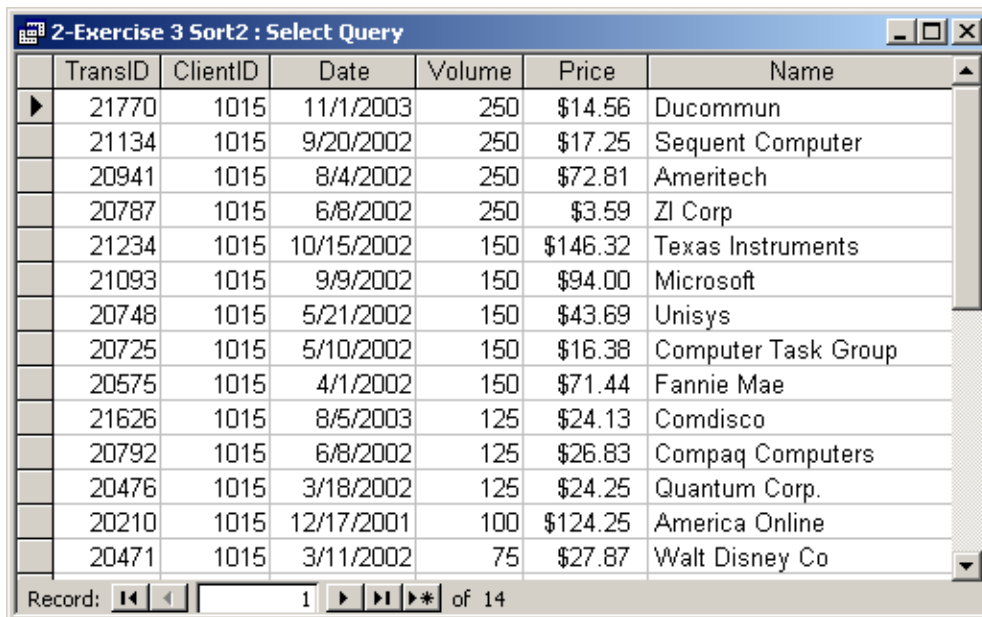
TransID	ClientID	Date	Volume	Price	Name
20787	1015	6/8/2002	250	\$3.59	Zi Corp
21770	1015	11/1/2003	250	\$14.56	Ducommun
20725	1015	5/10/2002	150	\$16.38	Computer Task
21134	1015	9/20/2002	250	\$17.25	Sequent Compu
21626	1015	8/5/2003	125	\$24.13	Comdisco
20476	1015	3/18/2002	125	\$24.25	Quantum Corp.
20792	1015	6/8/2002	125	\$26.83	Compaq Compu
20471	1015	3/11/2002	75	\$27.87	Walt Disney Co
20748	1015	5/21/2002	150	\$43.69	Unisys
20575	1015	4/1/2002	150	\$71.44	Fannie Mae
20941	1015	8/4/2002	250	\$72.81	Ameritech
21093	1015	9/9/2002	150	\$94.00	Microsoft
20210	1015	12/17/2001	100	\$124.25	America Online
21234	1015	10/15/2002	150	\$146.32	Texas Instrume

Record: 1 of 14

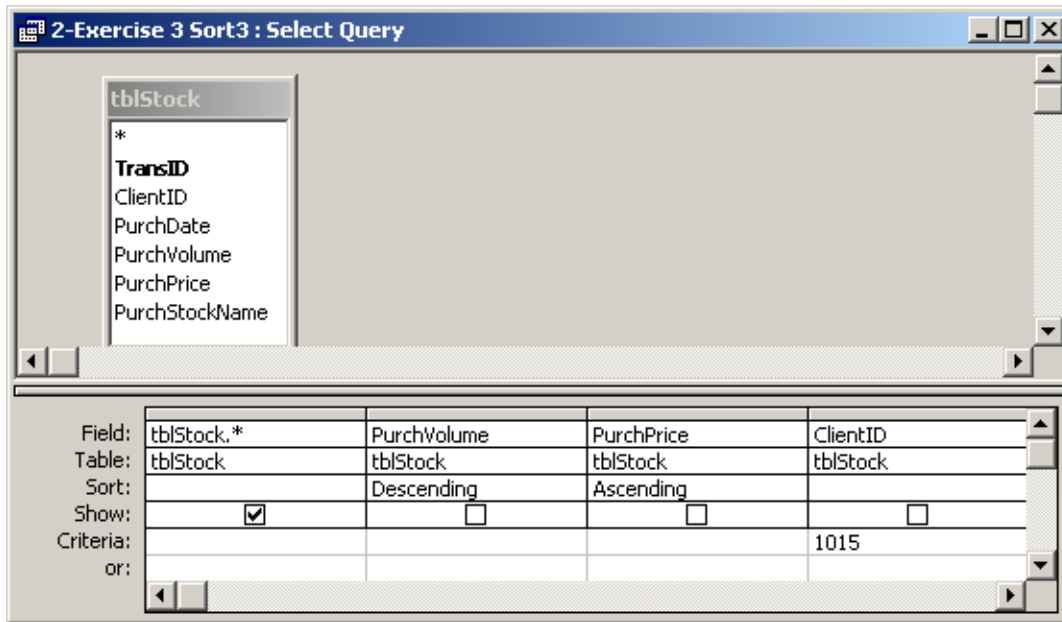
Datasheet View



Design View



Datasheet View



Design View



Datasheet View

4. The exercise has been saved on the Instructor's Companion CD (but *not* on the student's version, of course) in the database **Ch02.mdb**. Click *Forms* on the Options bar, select the form called *2-Exercise 4*, and print it. The completed form with an example record displayed in it is shown below.

Students should write their names on the form. Consider giving extra credit, perhaps, if they include their name like this message.

ClientID	1001
Name	Bank America
Volume	200
Price	\$74.44
Total	\$14,888.00

Record: 1 of 173

5. The exercise has been saved on the Instructor's Companion CD (but *not* on the student's version, of course) in the database **Ch02.mdb**. Click *Reports* on the Objects bar, select the report object called *2-Exercise 5*, and print or preview it. An example of the first page (in print preview) is shown below.

TransID	20057
ClientID	1004
Date	11/4/2001
Volume	200
Price	\$6.50
Name	Alpha Microsystems
TransID	20067
ClientID	1006
Date	11/8/2001
Volume	150
Price	\$40.00
Name	Stamps.com
TransID	20071
ClientID	1007
Date	11/8/2001
Volume	150
Price	\$6.50
Name	Alpha Microsystems