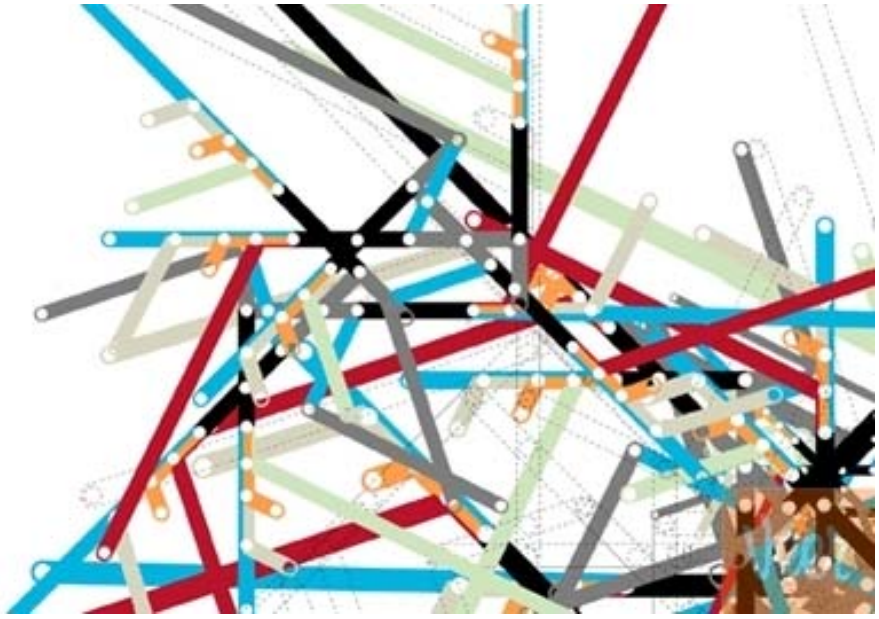
Database Processing

11th Edition

David M. Kroenke and David J. Auer

APPENDIX A

Getting Started with Microsoft Access 2007



Executive Editor: Bob Horan
Editorial Director: Sally Yagan
AVP/Editor-in-Chief: Eric Svendsen
Assistant Editor: Kelly Loftus
Director of Marketing: Patrice Lumumba Jones
Senior Marketing Manager: Anne Fahlgren
Marketing Assistant: Susan Osterlitz
Senior Managing Editor: Judy Leale
Production Project Manager: Kelly Warsak
Senior Operations Supervisor: Arnold Vila
Operations Specialist: Ilene Kahn
Senior Art Director: Janet Slowik
Design Director: Christy Mahon
Cover and Interior Design: Karen Quigley
Manager, Rights and Permissions: Charles Morris
Manager, Cover Visual Research & Permissions: Karen Sanatar
Cover Illustration: Stockbyte/Veer/Corbis
Media Director: Lisa Rinaldi
Lead Media Project Manager: Denise Vaughn
Full-Service Project Management: Jennifer Welsch/BookMasters, Inc.
Composition: Integra Software Services
Printer/Binder: Quebecor World-Versailles
Cover Printer: Lehigh-Phoenix Color/Hagerstown
Text Font: 10/12 Kepler MM

Credits and acknowledgments borrowed from other sources and reproduced, with permission, in this textbook appear on appropriate page within text.

Library of Congress Cataloging-in-Publication Data

Kroenke, David.
Database processing: fundamentals, design, and implementation/David M. Kroenke, David J. Auer.—11th ed.
p. cm.
Includes bibliographical references and index.
ISBN 978-0-13-230267-8 (casebound : alk. paper)
1. Database management. I. Auer, David J. II. Title.

QA76.9.D3K7365 2009
005.74—dc22

2009011144

10 9 8 7 6 5 4 3 2 1

Prentice Hall
is an imprint of

PEARSON

www.pearsonhighered.com

ISBN 10: 0-13-230267-5
ISBN 13: 978-0-13-230267-8

Appendix A

Getting Started with Microsoft Access 2007

Chapter Objectives

- To be able to create databases in Access 2007.
- To be able to create tables in Access 2007.
- To understand Access 2007 data types.
- To be able to insert data into tables in Access 2007.
- To be able to create relationships between tables in Access 2007.
- To be able to create Query-by-Example (QBE) queries in Access 2007.
- To understand the use of the Form Wizard in Access 2007.
- To understand the use of the Report Wizard in Access 2007.

As discussed in Chapter 1, Microsoft Access is a personal database that combines a DBMS with an application generator. The DBMS performs the standard DBMS functions of database creation, processing, and administration, whereas the application generator adds the abilities to create and store forms, reports, queries, and other application-related functions. In this appendix, we will create a new database, create tables and relationships, insert data into tables, and create queries, forms and reports.

We will begin by creating an Access database to store the database tables and the application forms, reports, and queries. We will build the example Student-Class-Grade database we used in Chapter 1 to illustrate basic relational database concepts. The tables and relationship links are shown in Figure A-1.

We can write the table structure of that database in database **schema** format as:

STUDENT (StudentNumber, LastName, FirstName, EmailAddress)

CLASS (ClassNumber, ClassName, Term, Section)

GRADE (*StudentNumber*, *ClassNumber*, Grade)

In the schema, table names are written in all uppercase letters (for example, STUDENT). Column names are written with the initial letter of each name capitalized (for example, Grade). If the column name is a compound name, then the initial letter of each word in the name is capitalized (for example, StudentNumber).

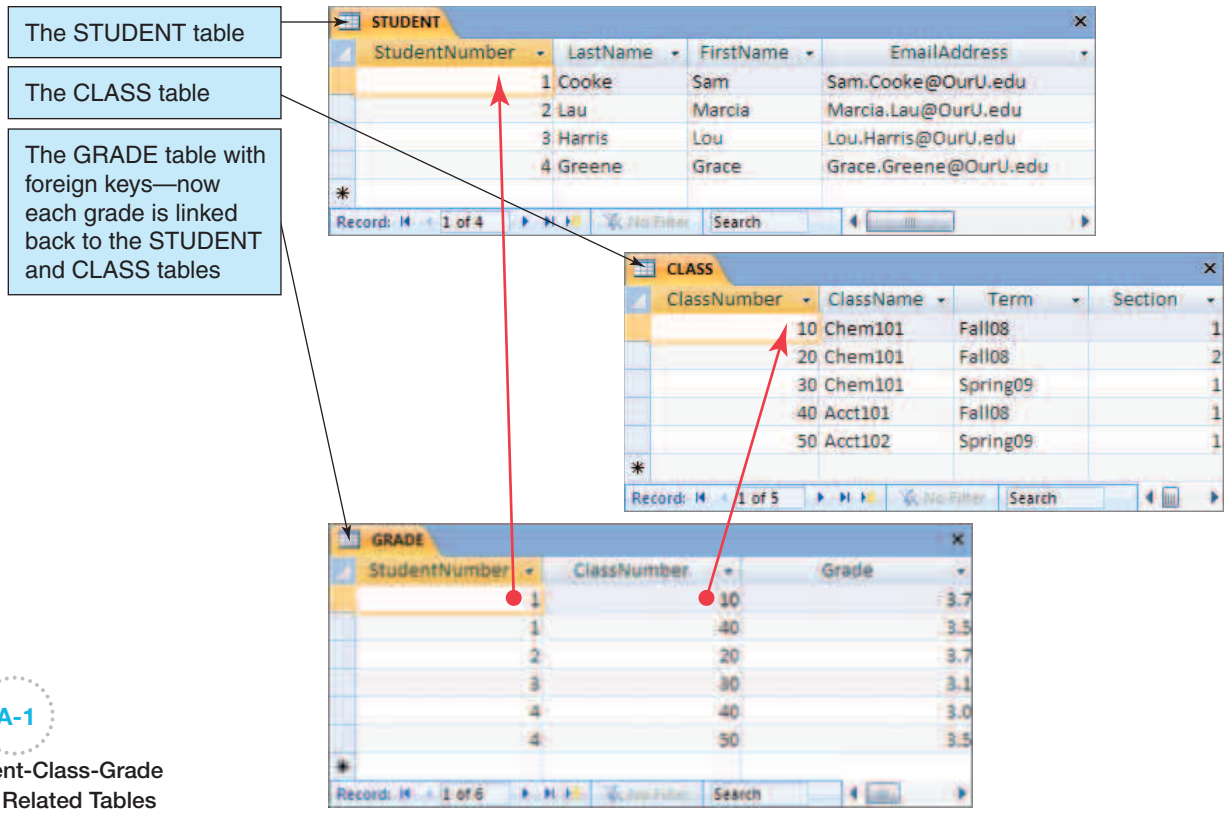


Figure A-1
The Student-Class-Grade Database Related Tables

Table Keys

Each table has a key, which is one or more columns that uniquely identify a row. You will learn the importance of keys, their properties, and their uses throughout this text. For now, just understand that the values of a **primary key** column identify a unique row in the table. In the schema, primary key columns in a table are underlined. Thus, the primary key of STUDENT is StudentNumber. This means that a particular value of StudentNumber, say 4, identifies one and only one row in STUDENT. Similarly, the primary key of CLASS is ClassNumber.

No single column can be a primary key for the GRADE table. A student may have several grades recorded, so StudentNumber, by itself, is not a primary key. Similarly, a class will have many students, so ClassNumber, by itself, cannot be the primary key either. However, if we assume that a student takes a class just once, then the combination (StudentNumber, ClassNumber) is a primary key.

Relationships Among Tables

As described in Chapter 1, one of the characteristics of a relational database is that the rows in tables can relate to one another. For the example in Figure A-1, rows in the STUDENT table are related to rows in the GRADE table by the column StudentNumber. As shown in that figure, the student with StudentNumber 4 has earned all of the grades in the GRADE table that have a matching value of 4 in the StudentNumber column. The StudentNumber column in the CLASS table, which creates the relationship link to the STUDENT table, is called a **foreign key**. Similarly, rows in the CLASS table are related to rows in the GRADE table by the column ClassNumber, and the ClassNumber column in GRADE is also a foreign key. In the database schema, foreign key columns in a table are shown in italic.

Creating an Access Database

We will name our Access database **Student-Grade-Class**. Our first step is to create a new Access database.

Creating the Access Database Student-Grade-ClassWMCRM

1. Select **Start | All Programs | Microsoft Office | Microsoft Office Access 2007**. The Microsoft Access 2007 window appears, as shown in Figure A-2. Note that the *Getting Started with Microsoft Office Access* page is displayed in the Microsoft Access window, along with a *Template Categories* pane.

   NOTE: The menu commands, icon location, and file locations used here are those found when using the Microsoft Windows Vista operating system. If you are using the Microsoft Windows XP operating system, the exact terminology will vary somewhat, but these variations will not change the required actions.

   NOTE: Microsoft Access 2007 is used in this book, and the wording of the steps and appearance of the screenshots reflect its use. If you have a different version of Access, there will be some differences in the step details and in what you see onscreen. However, the basic functionality is the same, and you can complete these steps using any version of Access.

2. Click the **Blank Database** button in the *New Blank Database* section of the *Getting Started with Microsoft Access* page. The Blank Database pane appears, as shown in Figure A-3.

   NOTE: By default, the database will be created in your *Documents* folder (in Windows XP, this is called the *My Documents* folder). If you want to create the database in a different folder, browse to the correct location by clicking the Open button.

3. Type the database name **Student-Class-Grade.accdb** into the File Name text box and then click the **Create** button.

   NOTE: If you clicked the Open button to browse to a different file location, use the File New Database dialog box to create the new database file. Once you have browsed to the

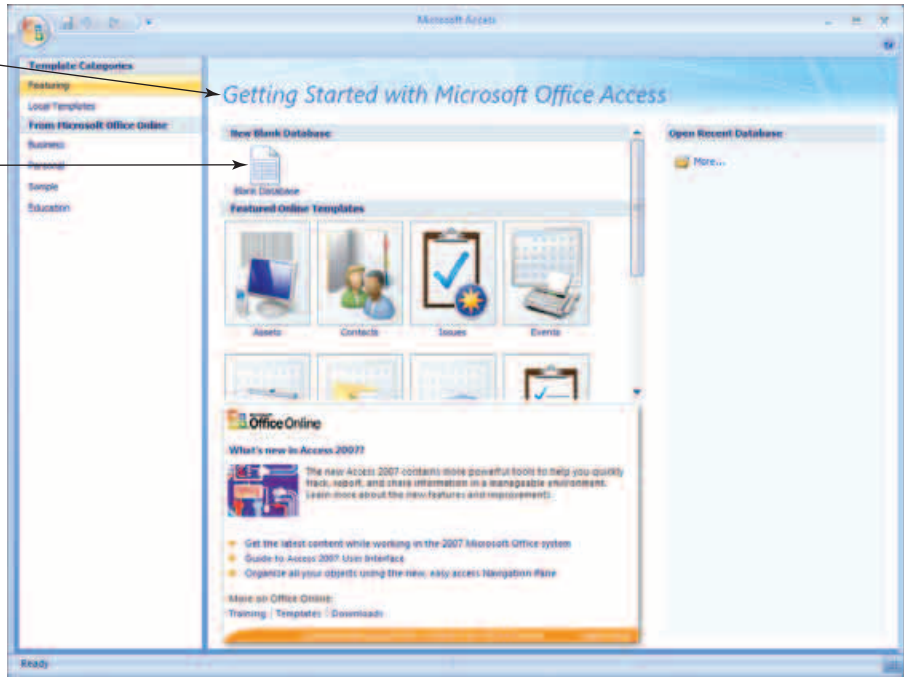
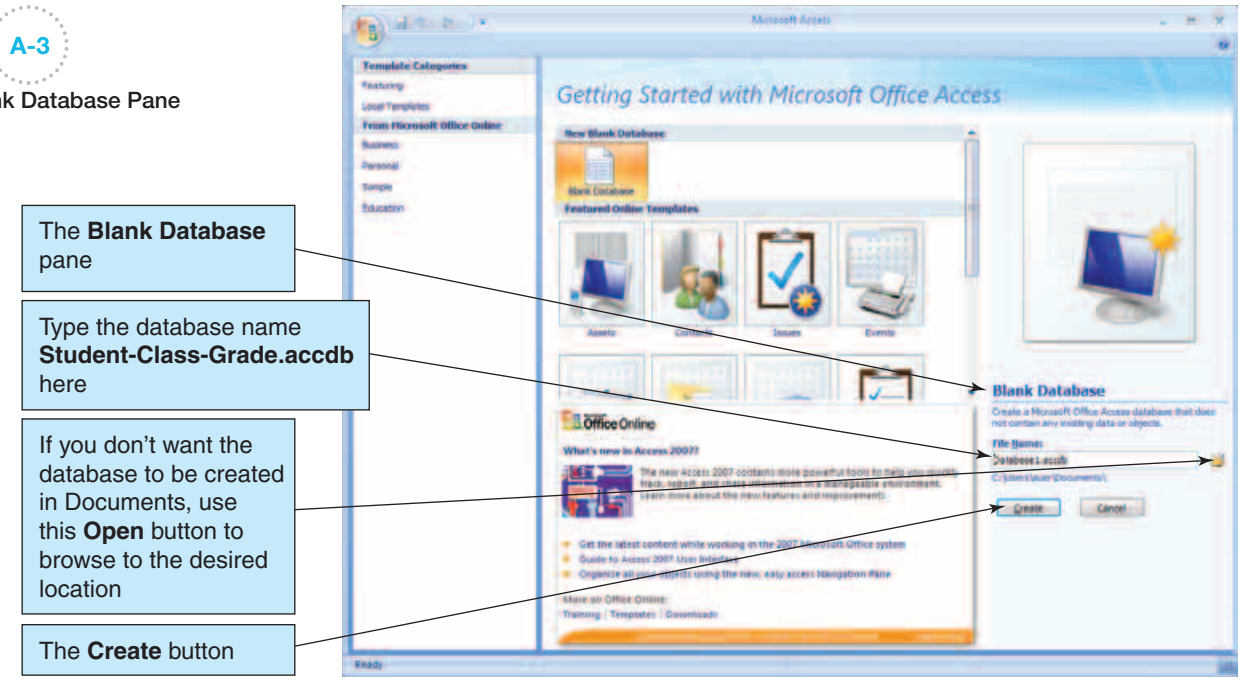


Figure A-2
The Microsoft Access 2007 Window

Figure A-3
The Blank Database Pane



correct folder, type the database name in the File Name text box of the File New Database dialog box, and then click the OK button to create the new database.

4. The new database appears, as shown in Figure A-4. The Microsoft Access window itself is now named Student-Grade-Class: Database (Access 2007) – Microsoft Access to include the database name. Note that because this is a new database, Access 2007 has assumed that you will want to immediately create a new table. Therefore, a new table named Table1 is displayed in Datasheet view in the document window. We do not want this table open at this time, so click the Close document button shown in Figure A-4.
5. The Access 2007 window with the new database appears, as shown in Figure A-5. You can see most of the features of the Microsoft Office Fluent user interface in this window.

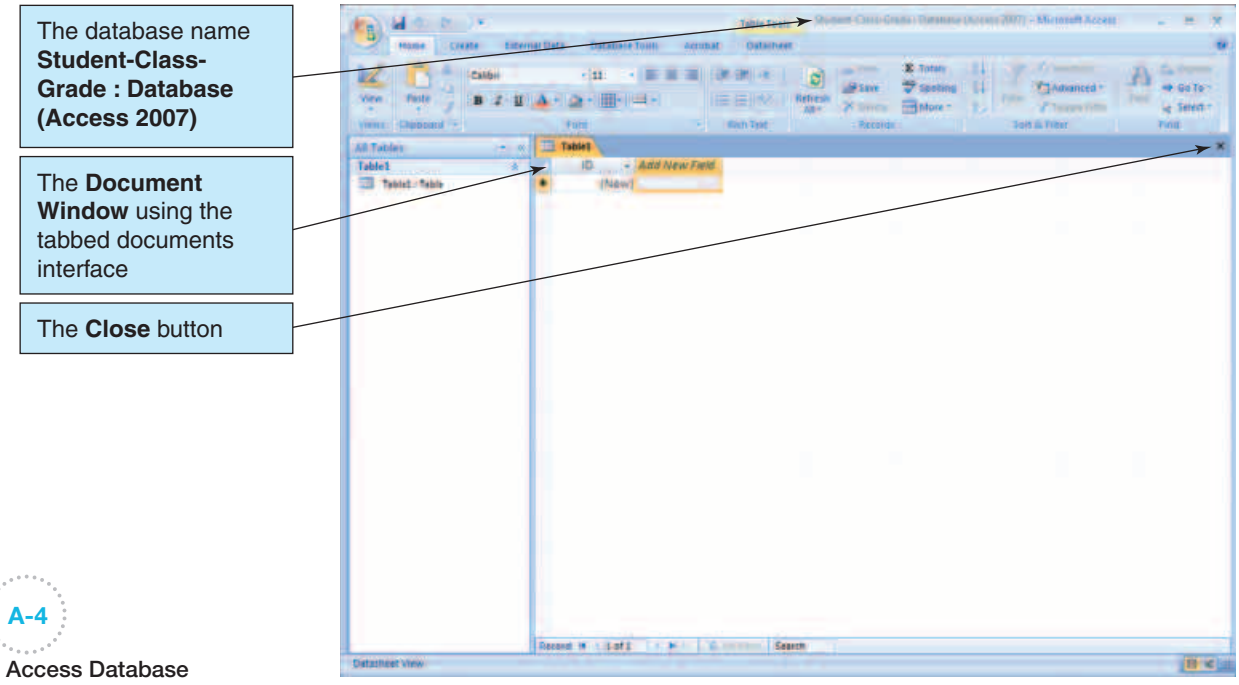


Figure A-4
The New Access Database

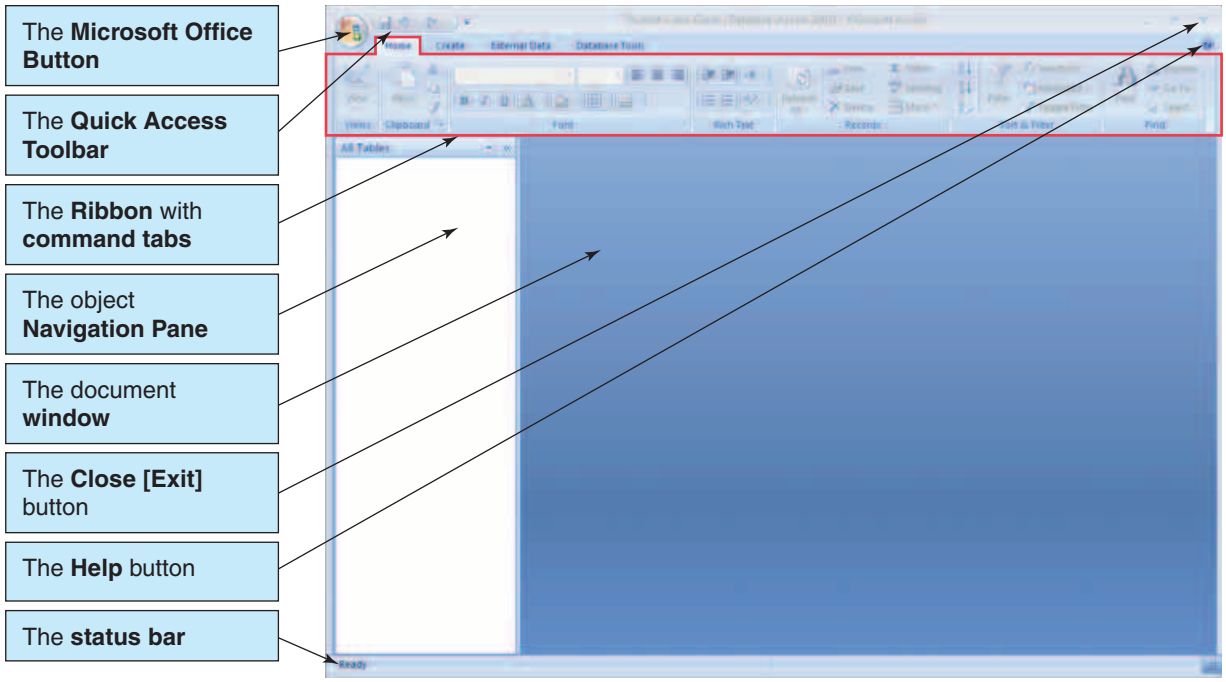


Figure A-5
The Microsoft Office Fluent User Interface

The Microsoft Office Fluent User Interface

Microsoft Access 2007 uses the **Microsoft Office Fluent user interface** found in most (but not all) of the Microsoft Office 2007 applications. The major features of the interface can be seen in Figure A-5.

The Ribbon and Command Tabs

The tabbed Ribbon, or just **Ribbon**, shown in Figure A-5 is the main Access 2007 command interface. The interface provides a set of **command tabs** that you use to access tools that are grouped into sets of related commands. Each Office 2007 application has a Home tab and a set of additional tabs specific to each application.

The default Access command tabs are the **Home**, **Create**, **External Data**, and **Database Tools** tabs. In each command tab, the currently available commands are shown in color, and the unavailable commands are shown in gray. You will learn about the various commands as they are needed, so for now you can just become familiar with each command tab and its command groups.

BY THE WAY Microsoft documentation varies about just what constitutes the Ribbon. The Microsoft Office Button is usually not considered part of the Ribbon, whereas the Quick Access Toolbar documentation is split and often includes the toolbar in the Ribbon. In this book, we will define the Ribbon to only include the command tabs and contextual command tabs.

Contextual Command Tabs

In addition to the basic command tabs, some Office 2007 applications, including Access, have additional **contextual command tabs** with associated command groups. These are displayed as needed, depending on the task. You can see an example in Figure A-4, where the **Table Tools**

contextual grouping of tabs adds the **Datasheet** contextual command tab into the set of command tabs available on the Ribbon. You will learn about the other contextual tabs as you encounter them.

Database Objects and the Navigation Pane

Microsoft uses the term **object** as a general name for the various parts of an Access database. Thus, a *table* is an object, a *report* is an object, a *form* is an object, and so on. Access objects are displayed in the Access **Navigation Pane**, as shown in Figure A-5. However, because you have not created any objects in the Student-Class-Grade database, the Navigation Pane is currently empty.

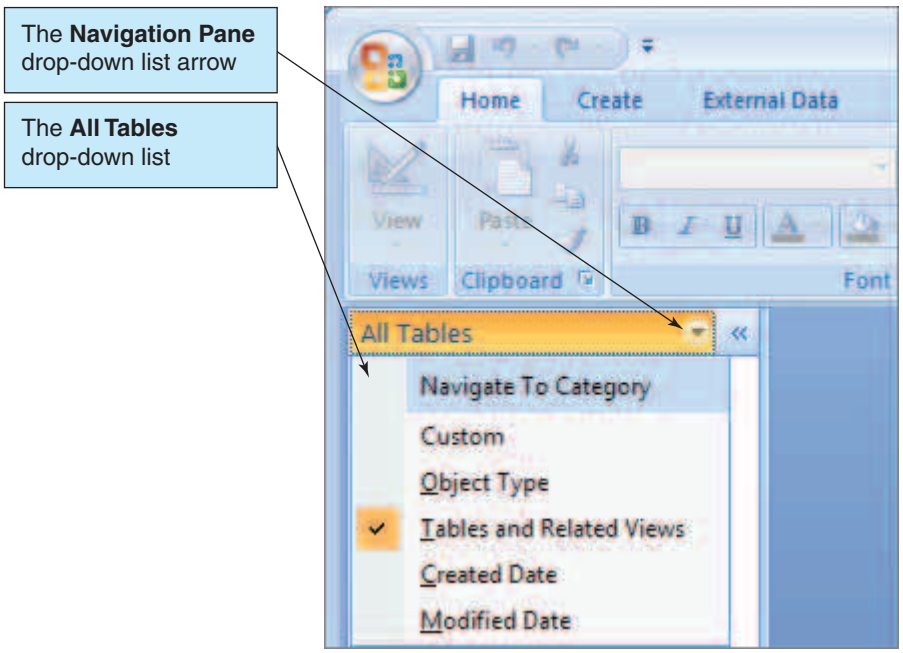
However, the Navigation Pane is currently labeled as *All Tables*, which means that *only* database tables and their related views are shown. Because we want to be able to see *all* Access objects (for example, tables, forms, reports) in the Navigation Pane as we create them, we will need to set it to show all Access objects

Selecting Objects in the Access 2007 Navigation Pane

1. Click the **Navigation Pane** drop-down arrow in the upper-right corner of the Navigation Pane. The **Navigation Pane** drop-down list appears, as shown in Figure A-6.
2. Click **Object Type** in the **Navigation Pane** drop-down list, and then click the Navigation Pane drop-down arrow again. The All Access Objects drop-down list now appears, as shown in Figure A-7.
3. Press the **Esc** key to close the Navigation Pane drop-down list.
4. Because we have not created any tables or other objects in the Student-Class-Grade database, the Navigation pane for All Access Objects is currently empty. This is shown in Figure A-8.

Note that we can hide the Navigation Pane if we want to by clicking the left-facing double-chevron button on the upper-right corner of the Navigation Pane shown in Figure A-8. If we click the button, the Navigation Pane shrinks to a small band labeled *Navigation Pane* on the right side of the Access 2007 window. The band will have a right-facing double-chevron button that you can click to restore the Navigation Pane when you want to use it again.

Figure A-6
The Navigation Pane Drop-Down List



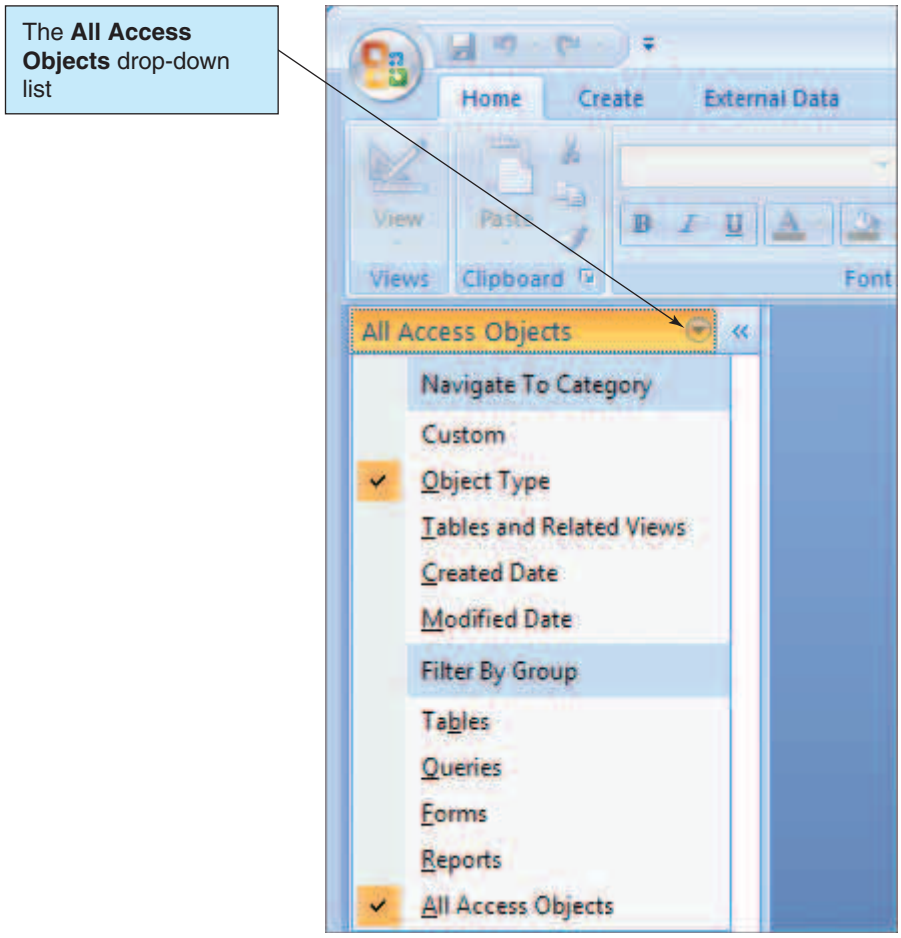


Figure A-7

The All Access Objects Drop-Down List

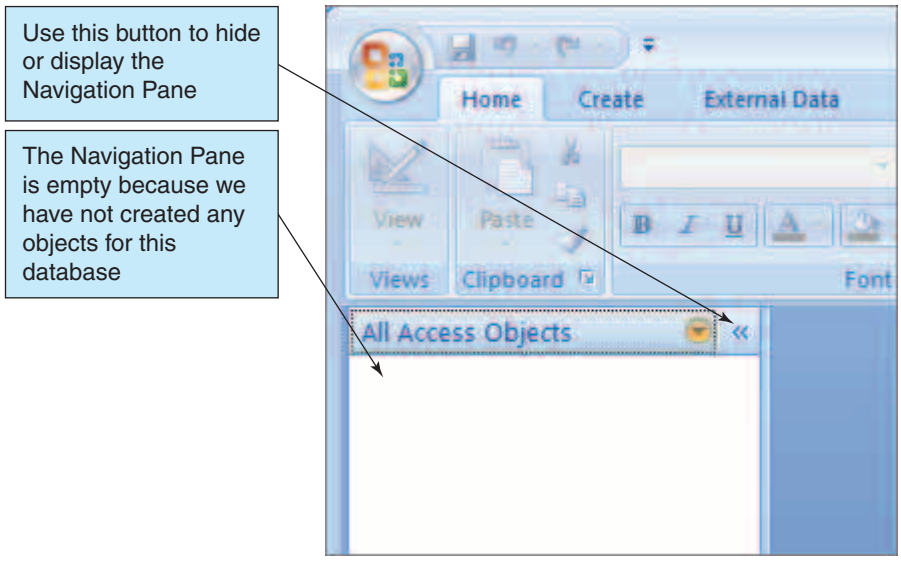


Figure A-8

The Empty Navigation Pane

Closing a Database and Exiting Microsoft Access

The **Close** button shown in Figure A-5 is actually a *close and exit button*. You can click it to close the active database and then exit the Access program. Note that Access actively saves most changes to a database, and it prompts you with *Save* command requests when they are needed. For example, when you close a table with modified column widths, Access asks if you want to save the changes in the table layout. Therefore, you do not need to save Access databases

the way you save Word documents and Excel workbooks. You can simply close a database, knowing that Access has already saved all critical changes since you opened it.

Closing a Database and Exiting Access

1. Click the **Close** button. The database closes, and you exit the Access program.

> **BY THE WAY** Instead of clicking the Close button, you can simultaneously close the database and exit Access by double-clicking the **Microsoft Office Button**. To close just the database while leaving Access open, select **File | Close**.

Opening an Existing Access Database

When we open an existing database, Access 2007 gives us the option of using Access security options to shut down certain Access 2007 features in a database to protect ourselves against harm not only from viruses, but also from other possible problems. Unfortunately, the Access 2007 security options also shut down significant and needed operational features of Access. Therefore, we should normally enable the features that the Access 2007 security warning warns us about when we open an existing database.

Opening a Recently Opened Access Database

1. Open Access by selecting **Start | All Programs | Microsoft Office | Microsoft Access 2007**. Access is displayed, as shown in Figure A-9.
2. As shown in Figure A-9, the database file Student-Class-Grade.accdb is listed in the **Open Recent Database** pane.
3. Click the **Microsoft Office Button** to display the File menu, as shown in Figure A-10. Note that the database file Student-Class-Grade.accdb is listed in the **Recent Documents** pane.

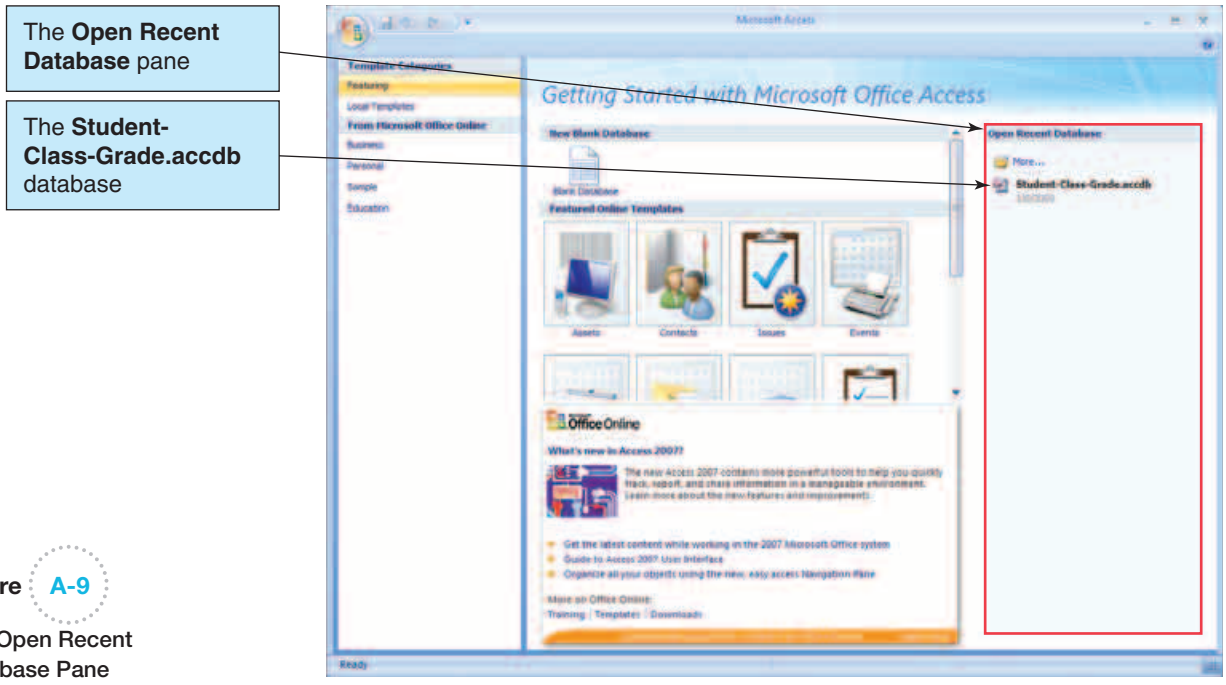


Figure A-9
The Open Recent Database Pane

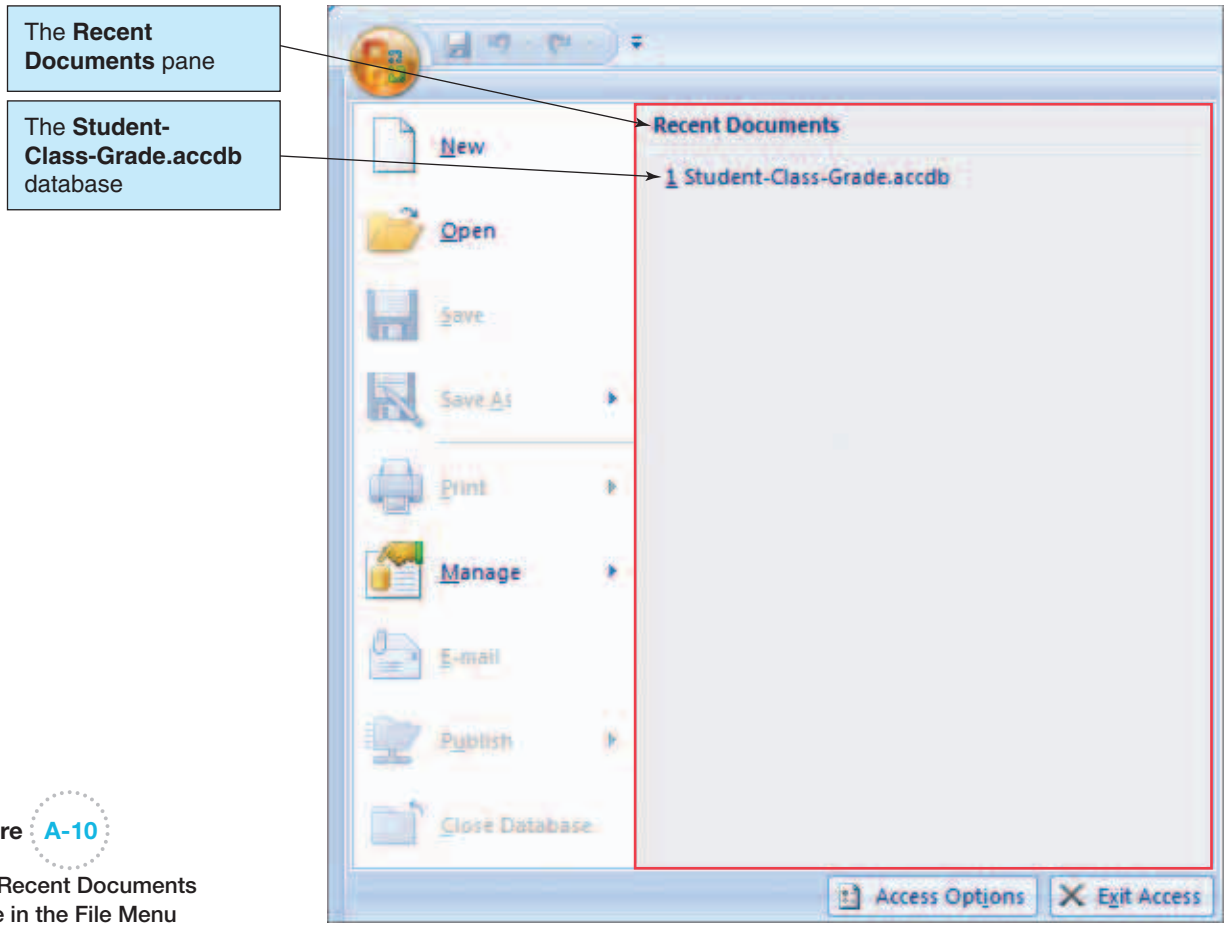


Figure A-10
The Recent Documents Pane in the File Menu

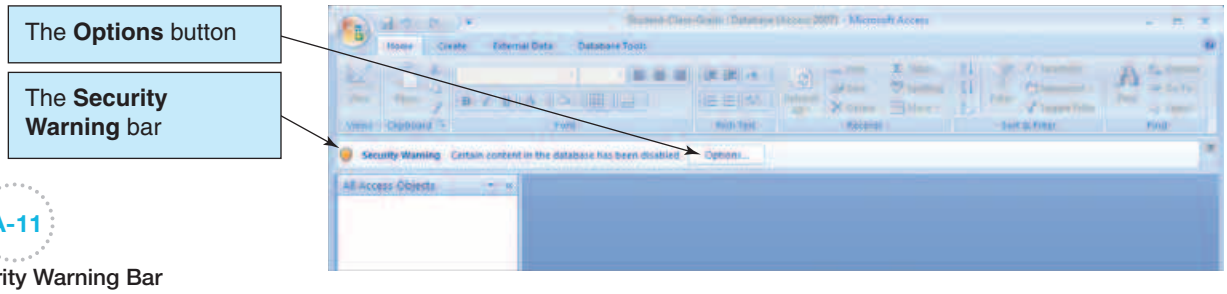


Figure A-11
The Security Warning Bar

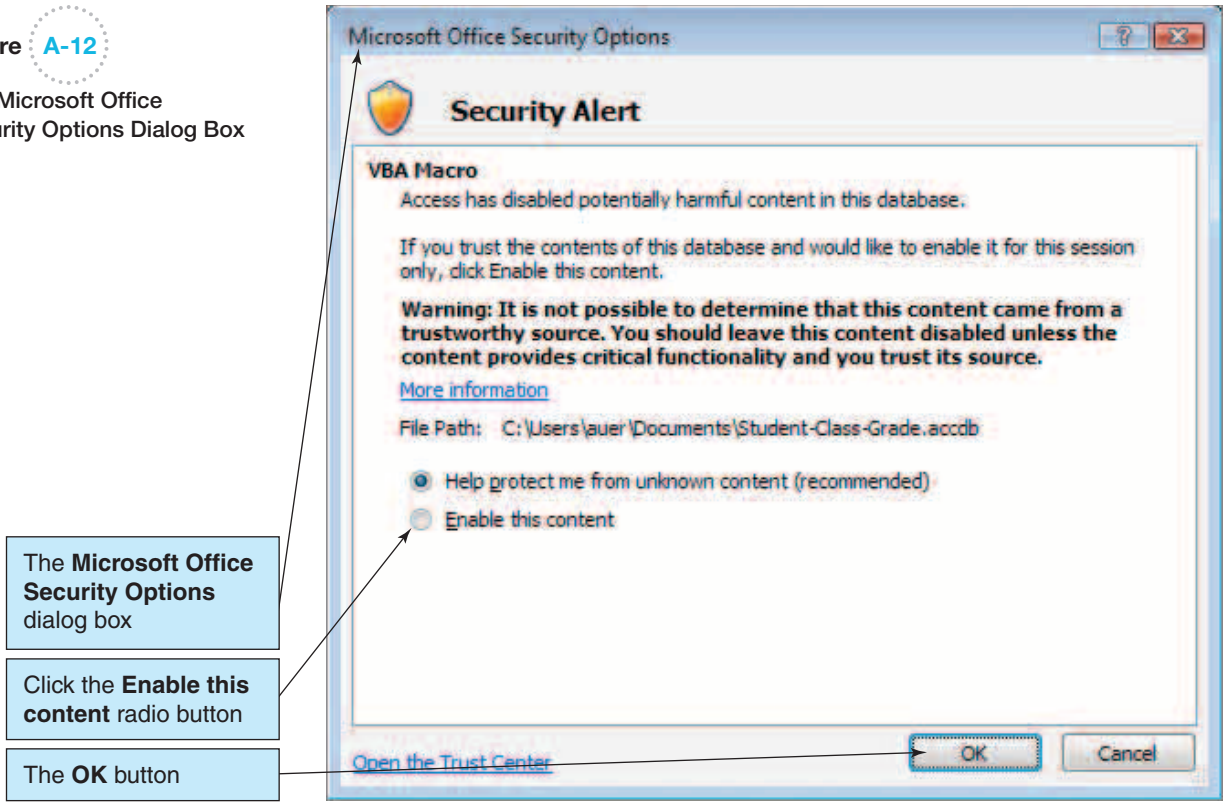
4. We can open the database by clicking on *either* entry. Because the **File** menu is open, click the Student-Class-Grade.accdb filename in the **Recent Documents** pane to open the database.
5. A **Security Warning** bar appears with the database, as shown in Figure A-11.
6. Click the Security Warning bar's **Options** button to display the **Microsoft Office Security Options** dialog box, shown in Figure A-12.
7. Click the **Enable this content** radio button to select this option and then click the **OK** button.

Creating Access Database Tables

Now we need to create the tables in the Student-Class-Grade database—STUDENT, CLASS, and GRADE. However, because foreign key columns need their values to already exist in the primary keys of the tables being linked to, we should create our tables (and add data to our tables) in a specific order: we must create and populate (fill with data) STUDENT and CLASS

Figure A-12

The Microsoft Office Security Options Dialog Box



first, and then create and populate GRADE (which has the foreign keys linking to STUDENT and CLASS). We will create the STUDENT table first.

The STUDENT table will contain the columns and characteristics shown in the table in Figure A-13. The column characteristics are type, key, required, and remarks.

Type refers to the kind of data the column will store. Some possible Access data types are shown in Figure A-14. For STUDENT, most data are being stored as **text** data (also commonly called **character** data), which means we can enter strings of letters, numbers, and symbols (a space is considered a symbol).

The number behind the word *Text* indicates how many characters can be stored in the column. For example, student last names may be up to 25 characters long. The only **number**, or **numeric**, data column in the STUDENT table is StudentNumber, which is listed as **AutoNumber**. This indicates that ACCESS will automatically provide a sequential number for this column for each new student that is added to the table.

Key refers to table identification functions assigned to a column. These are described in detail in Chapter 3. At this point, you simply need to know that a primary key is a column value used to identify each row, and, therefore, the values in this column must be unique. This is the reason for using the AutoNumber data type, which automatically assigns a unique number to each row in the table as it is created.

Figure A-13

Database Column Characteristics for the STUDENT Table

Column Name	Type	Key	Required	Remarks
StudentNumber	AutoNumber	Primary Key	Yes	Surrogate Key
LastName	Text (25)	No	Yes	
FirstName	Text (25)	No	Yes	
EmailAddress	Text (100)	No	No	

Name	Type of Data	Size
Text	Characters and numbers	Maximum 255 characters
Memo	Large text	Maximum 65,535 characters
Number	Numeric data	Varies with number type
Date/Time	Dates and times from the year 100 to the year 9999	Stored as 8-byte double-precision integers
Currency	Numbers with decimal places	One to four decimal places
AutoNumber	A unique sequential number	Incremented by one each time
Yes/No	Fields that can contain only two values	Yes/No, On/Off, True/False, etc.
Hyperlink	A hyperlink address	Maximum 2,048 characters in each of three parts of the hyperlink address
OLE Object	An object embedded in or linked to an Access table	Maximum 1 GB
Attachment	Any supported file type can be attached to a record	Independent of Access

Figure A-14
Microsoft Access 2007 Data Types

Required refers to whether the column must have a data value. If it must, a value must be present in the column. If not, the column may be blank. Note that because StudentNumber is a primary key used to identify each row, it *must* have a value.

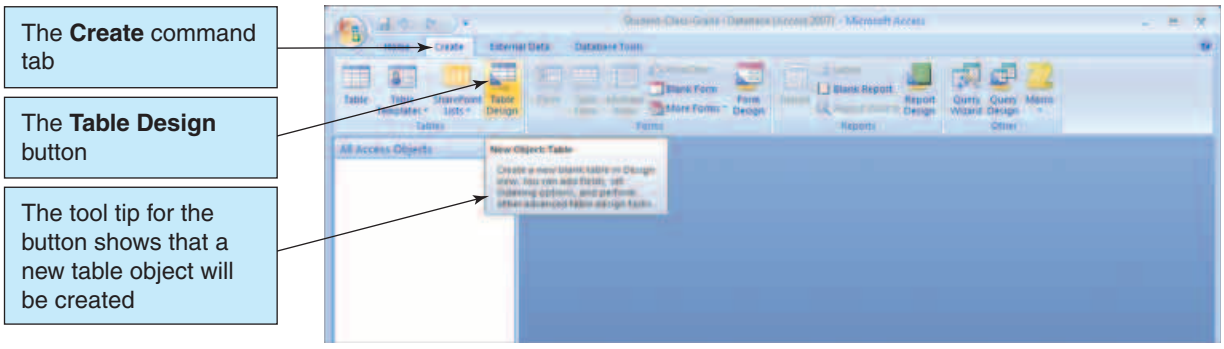
Remarks contains comments about the column or how it is used. For STUDENT, the only comment is that StudentNumber is a **surrogate key**. Surrogate keys are discussed in Chapter 3. At this point, you simply need to know that surrogate keys are usually computer-generated unique numbers used to identify rows in a table (that is, a primary key). This is done by using the Access AutoNumber data type.

Creating the STUDENT Table

1. Click the **Create** command tab to display the **Create** command groups.
2. Click the **Table Design** button, as shown in Figure A-15.
3. The **Table1** tabbed document window is displayed in **Design** view, as shown in Figure A-16. Note that along with the **Table1** window a contextual tab grouping named **Table Tools** is displayed and that this tab grouping adds a new command tab named **Design** to the set of command tabs displayed.

   NOTE: It seems like now would be a good time to name the new table STUDENT. With Access, however, you don't name a table until you save it the first time, and you can't save a table until you have at least one column defined. So, we will define the

Figure A-15
The Table Design Button



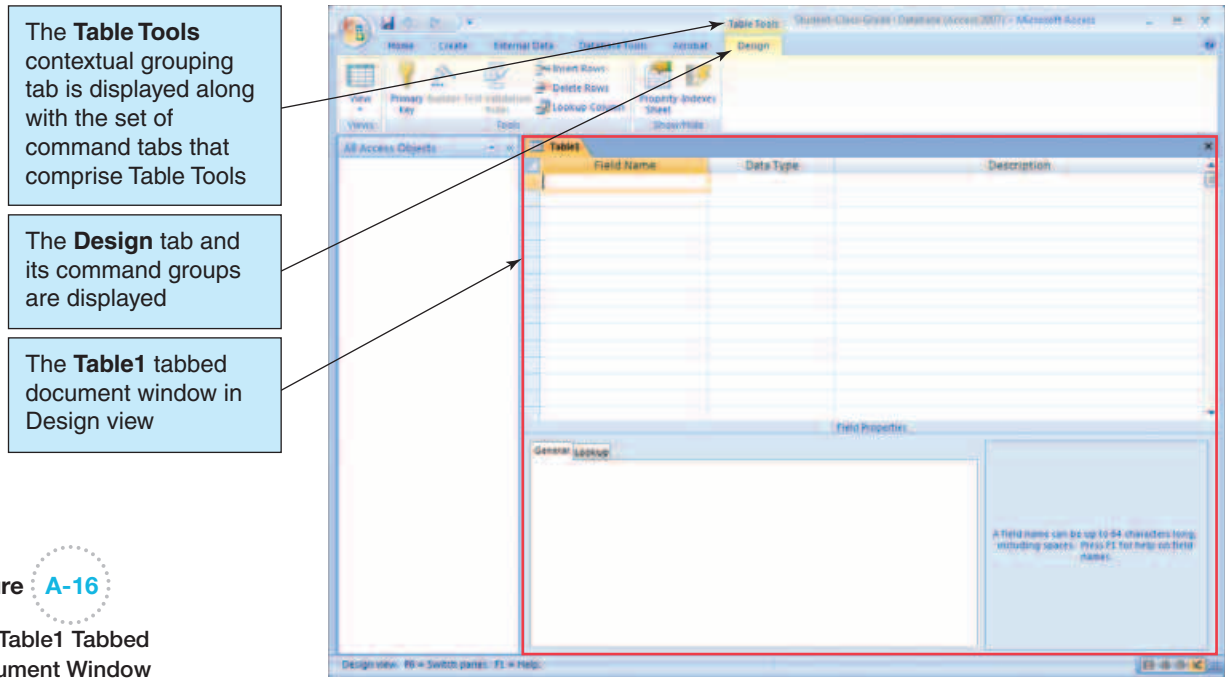


Figure A-16
The Table1 Tabbed Document Window

columns, and then we will save and name the table. If you want, save the table after you've defined just one column. This will close the table, so you'll have to reopen it to define the remaining columns.

4. In the **Field Name** column text box of the first line, type the column name **StudentNumber** and then press the **Tab** key to move to the **Data Type** column. (You can also click the Data Type column to select it.)

   NOTE: The terms *column* and *field* are considered synonyms in database work. The term *attribute* is also considered to be equivalent to these two words.

5. Select the **AutoNumber** data type for StudentNumber from the **Data Type** drop-down list, as shown in Figure A-17.
6. If you like, an optional comment may be stored in the Description column. To do so, move to the Description column by pressing the **Tab** key or clicking in the **Description** text box. Type the text **Surrogate key for STUDENT** and then press the **Tab** key to move to the next row. The **Table1** tabbed document window now looks as shown in Figure A-18.

   NOTE: The Remarks column in the set of database column characteristics shown in Figure A-13 is *not* the same as the table Description column shown in Figure A-18. Be careful not to confuse them. The Remarks column is used to record technical data, such as facts about table keys and data default values that are necessary for building the table structure. The Description column is used to describe to the user the data stored in that field so that the user understands the intended use of the field.

7. The other columns of the STUDENT table are created using the sequence described in steps 3 through 5.

   NOTE: See Figure A-21 for the Description entries.

8. To set the number of characters in text columns, edit the **Text Field Size** text box as shown in Figure A-19. The default value for Field Size is 50, and the maximum value is 255.
9. To make a column required, click anywhere in the column Data Type Required property text box to display the **Required** property drop-down list arrow button and then click the button to display the Required property drop-down list, as shown in Figure A-20.

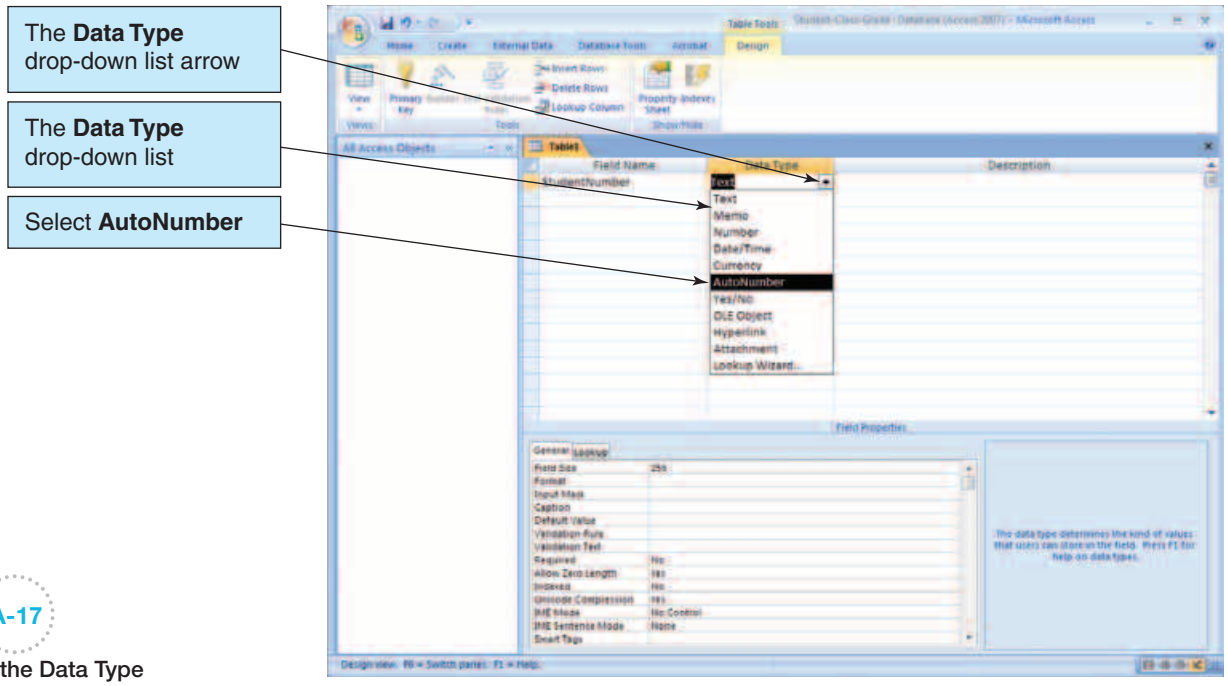


Figure A-17
Selecting the Data Type

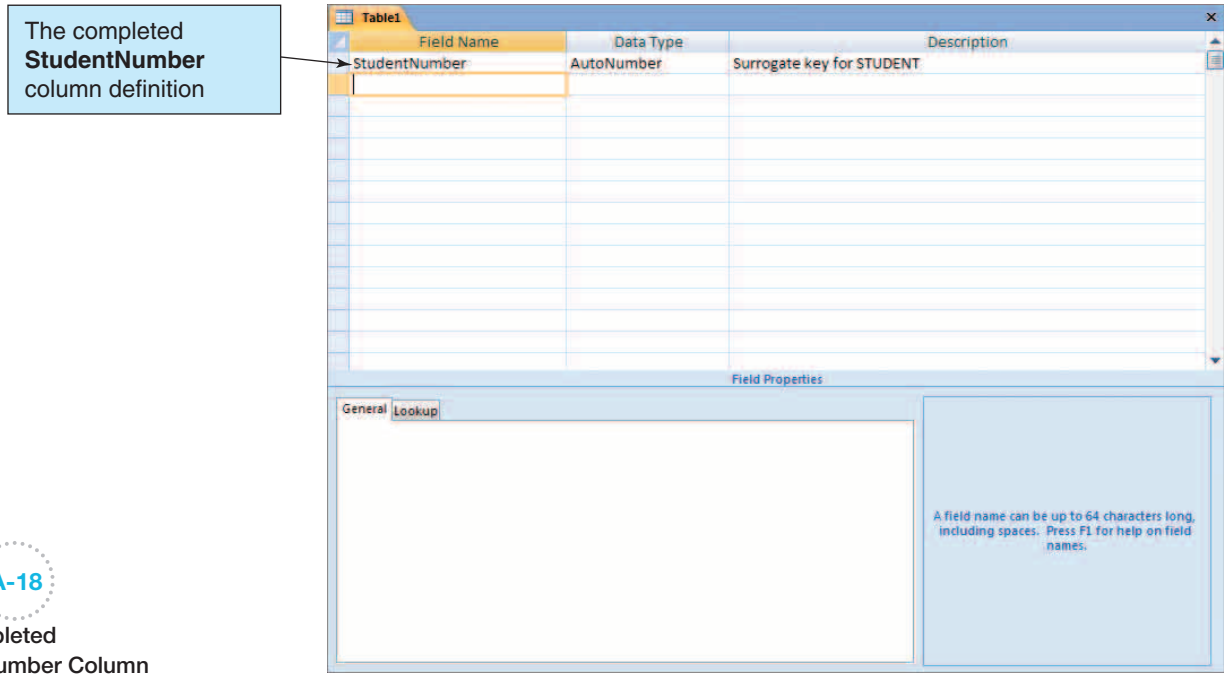


Figure A-18
The Completed StudentNumber Column

10. Select **Yes** from the Required property drop-down list. The default is No (not required), and Yes must be selected to make the column required.[1]

Now we need to set a primary key for the STUDENT table. According to Figure A-13, we need to use the CustomerID column as the primary key for this table.

[1] Microsoft Access has an additional Data Type property named Allow Zero Length. This property confounds the settings necessary to truly match the SQL constraint NOT NULL discussed in Chapter 7. However, the discussion of Allow Zero Length is beyond the scope of this book. See the Microsoft Access Help system for more information.

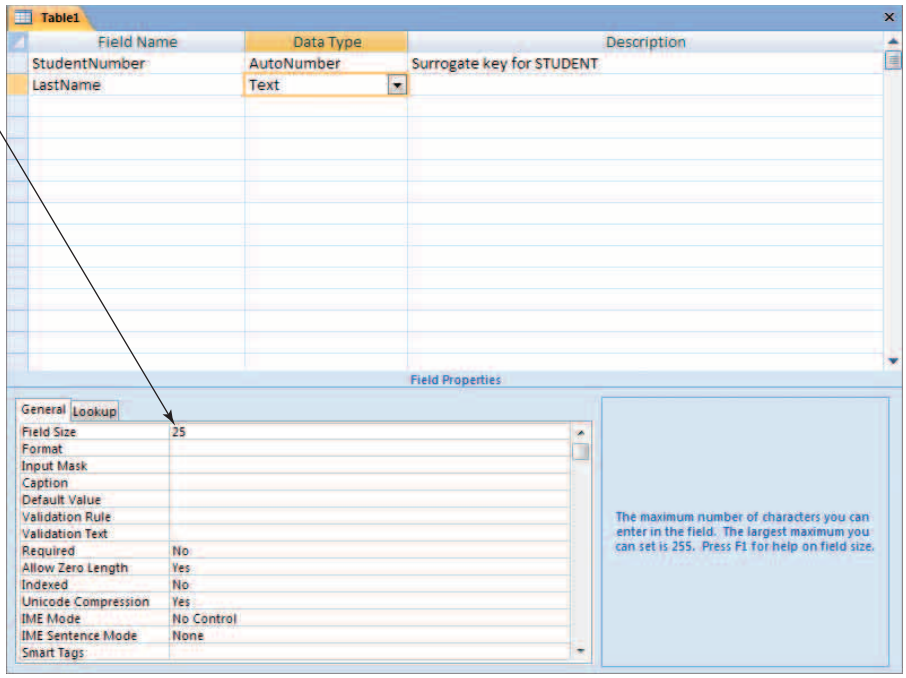


Figure A-19
Editing the Text Field Size

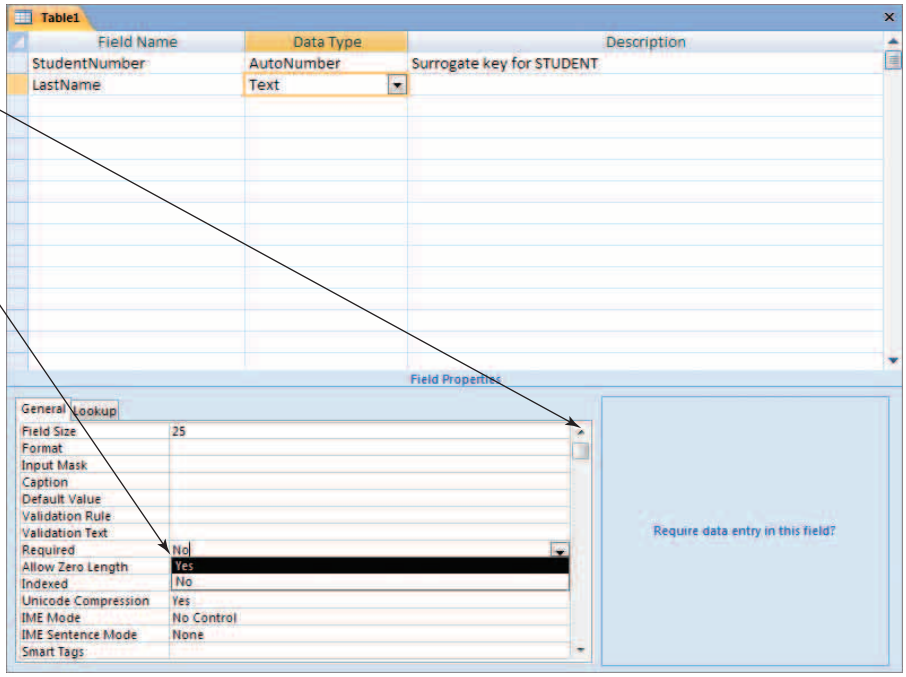


Figure A-20
Setting the Column Required Property Value

Setting the STUDENT Table Primary Key

1. Move the mouse pointer to the **row selector column** of the row containing the StudentNumber properties, as shown in Figure A-21, and click to select the row.
2. Click the **Primary Key** button in the Tools group of the Design tab, as shown in Figure A-22. StudentNumber is selected as the primary key for the STUDENT table.

We have finished building the STUDENT table. Now we need to name the table, save it, and close it.

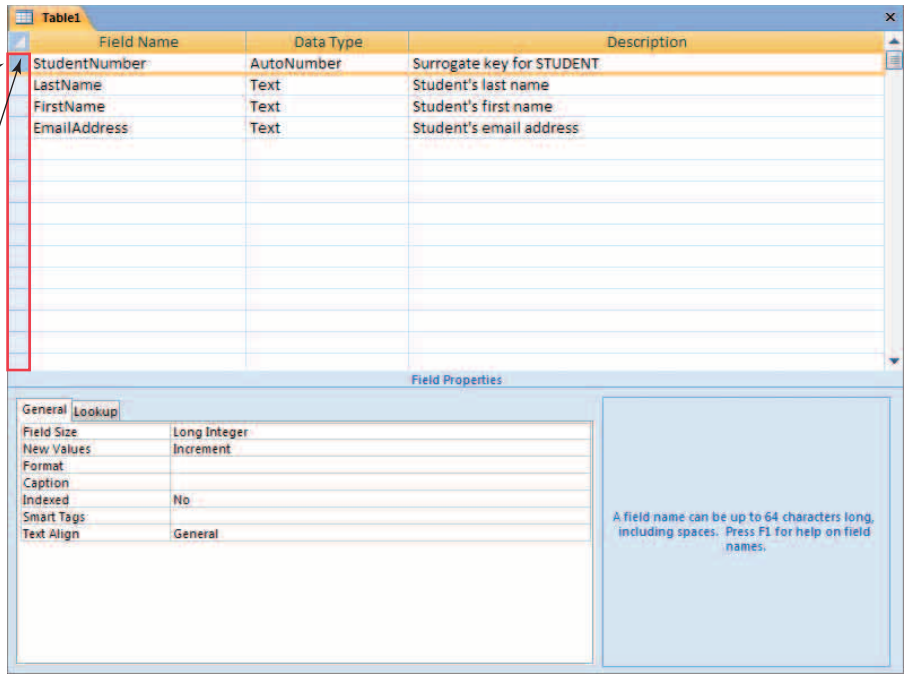


Figure A-21
Selecting the StudentNumber Row

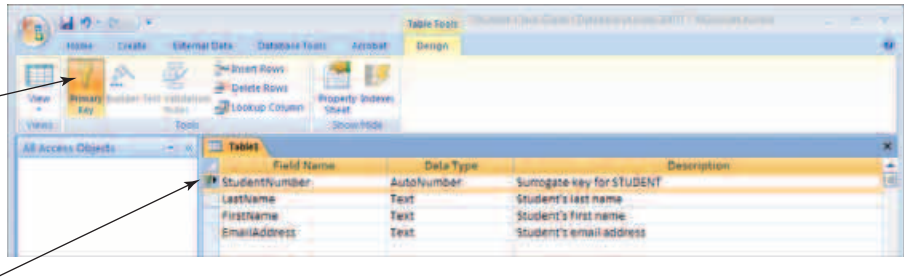


Figure A-22
Setting the Primary Key

Naming, Saving, and Closing the STUDENT Table

1. To name and save the STUDENT table, click the **Save** button in the Quick Access Toolbar. The **Save As** dialog box appears, as shown in Figure A-23.
2. Type the table name **STUDENT** into the **Save As** dialog box's Table Name text box and then click **OK**. The table is named and saved. The table name STUDENT now appears on the document tab, and the STUDENT table object is displayed in the Navigation Pane, as shown in Figure A-24.
3. To close the STUDENT table, click the **Close** button in the upper-right corner of the tabbed documents window. After the table is closed, the STUDENT table object remains displayed in the Navigation Pane, as shown in Figure A-25.

At this point, we need to build the CLASS and GRADE tables. The CLASS table will have the column characteristics shown in Figure A-26; the GRADE table will have the column characteristics shown in Figure A-27. Using steps similar to those you used for the STUDENT table, create the CLASS and GRADE tables. Note that GRADE has a **composite primary key**, and in order to designate this key you must select the row representing StudentNumber, as discussed earlier, and then CTRL-Click (hold down the CTRL key and then click) on the row representing ClassNumber so that both are highlighted when you click the Primary Key button.

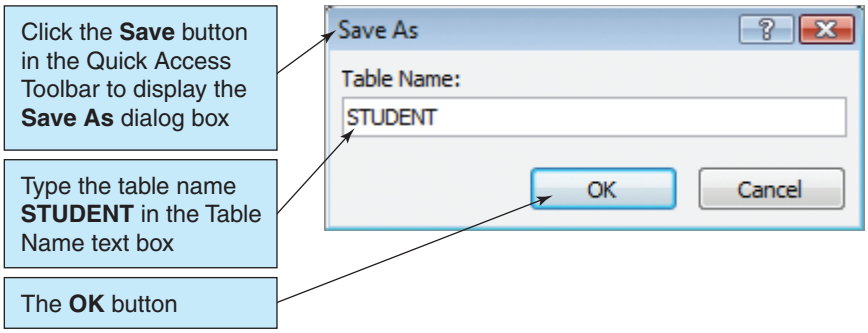


Figure A-23
Naming and Saving the STUDENT Table

Figure A-24
The Named STUDENT Table

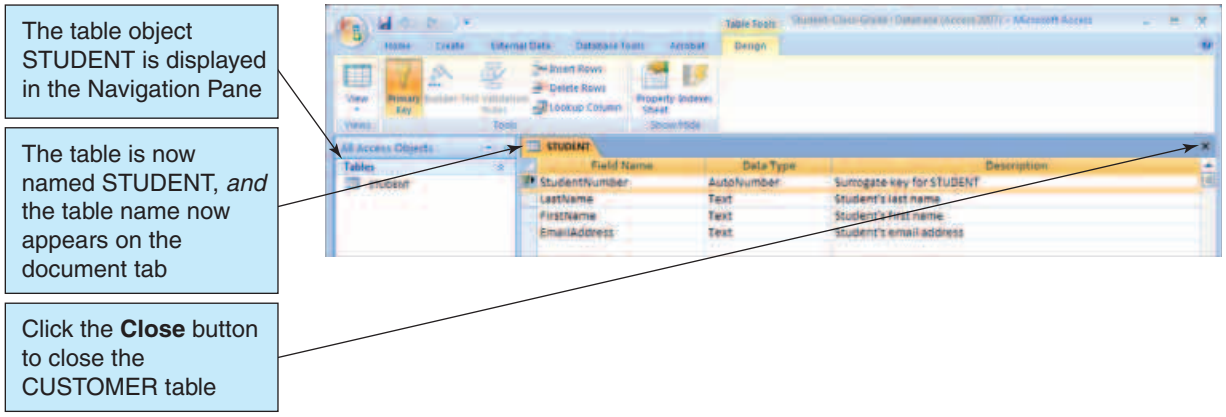


Figure A-25
The STUDENT Table Object

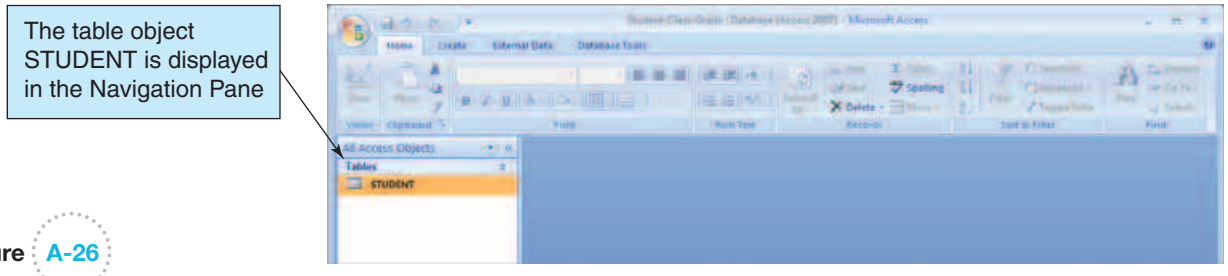


Figure A-26
Database Column Characteristics for the CLASS Table

Column Name	Type	Key	Required	Remarks
ClassNumber	Number	Primary Key	Yes	Long Integer
ClassName	Text (25)	No	Yes	
Term	Text (12)	No	Yes	
Section	Number	No	Yes	Integer

Column Name	Type	Key	Required	Remarks
StudentNumber	Number	Primary Key, Foreign Key	Yes	Long Integer
ClassNumber	Number	Primary Key, Foreign Key	Yes	Long Integer
Grade	Number	No	Yes	Decimal, Fixed, Scale = 2, Decimal Places = 1

Figure A-27
Database Column Characteristics for the GRADE Table

Inserting Data into Tables—The Datasheet View

There are three commonly used methods for adding data to a table. First, we can use a table as a **datasheet**, which is visually similar to and works like an Excel worksheet. When we do this, the table is in **Datasheet view**, and we enter the data cell by cell. Second, we can build a **data entry form** for the table and then use the form to add data. Third, we can use SQL to insert data. This section covers entering data into a datasheet. We will discuss forms later in this Appendix, and we will use SQL in Chapter 2.

In Access 2007, we can also use Datasheet view to create and modify table characteristics: When we open a table in Datasheet view, the Table Tools contextual tab includes a Datasheet command tab and ribbon with tools to do this. We do *not* recommend this and prefer to use Design view, as previously discussed in this section, for creating and modifying table structures.

Of course, before we can use either method for creating and modifying table structures, we need to put some data into the table. Figure A-28 shows some STUDENT data.

Adding Data to the STUDENT Table in Datasheet View

1. In the Navigation Pane, double-click the **STUDENT** table object. The STUDENT table window appears in a tabbed document window in Datasheet view, as shown in Figure A-29. Note that some columns on the right side of the datasheet do not appear in the window, but you can access them by scrolling or minimizing the Navigation Pane.

   NOTE: As in a worksheet, the intersection of a row and column is called a **cell** in a datasheet.

2. Click the **STUDENT** document tab to select the STUDENT table in Datasheet view.
3. Click the cell in the StudentNumber column with the phrase **(New)** in it to select that cell in the new row of the STUDENT datasheet.

Figure A-28
STUDENT Data

StudentNumber	LastName	FirstName	EmailAddress
1	Cooke	Sam	Sam.Cooke@OurU.edu
2	Lau	Marcia	Marcia.Lau@OurU.edu
3	Harris	Lou	Lou.Harris@OurU.edu
4	Greene	Grace	Grace.Greene@OurU.edu

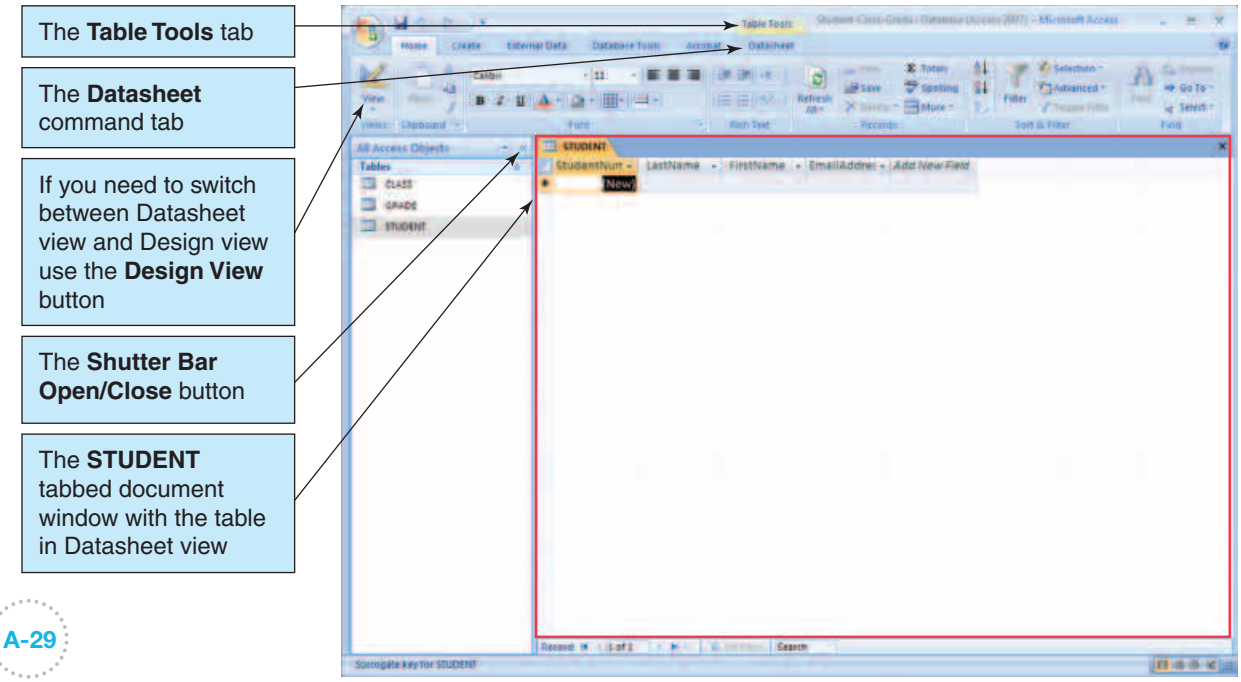


Figure A-29
The STUDENT Table in Datasheet View

4. Press the **Tab** key to move to the LastName cell in the new row of the STUDENT datasheet. For student Sam Cooke, type **Cooke** in the LastName cell. Note that as soon as you do this, the AutoNumber function puts the number 1 in the StudentNumber cell, and a new row is added to the datasheet, as shown in Figure A-30.
5. Using the **Tab** key to move from one column to another in the STUDENT datasheet, enter the rest of the data values for Sam Cooke.
6. The final result is shown in Figure A-31. Note that the width of the Email column was expanded using the mouse to move the border of the column—just as you do in an Excel worksheet.

Figure A-30
Entering Data Values for Sam Cooke

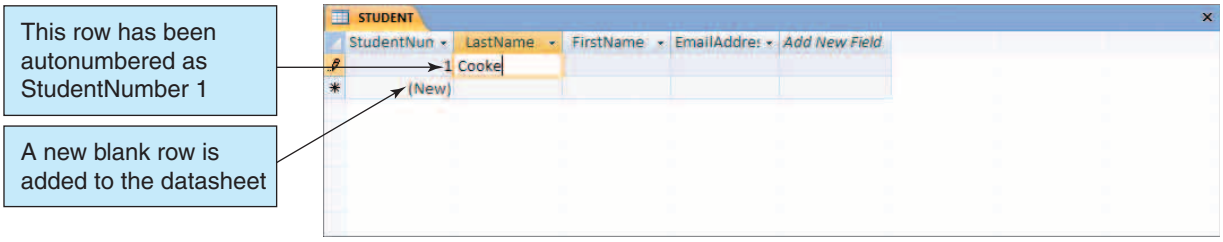
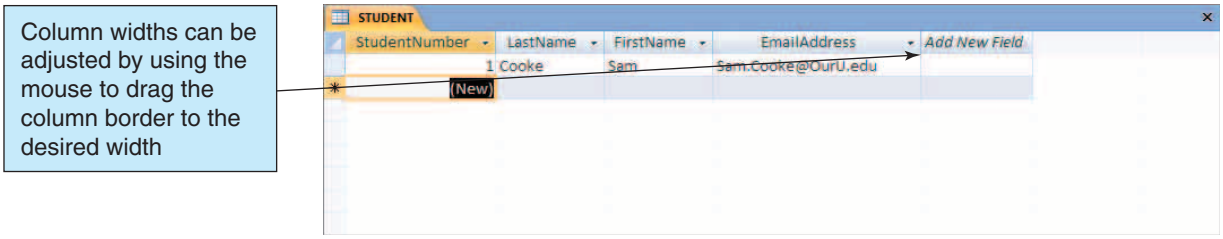


Figure A-31
The Completed Row of Data Values



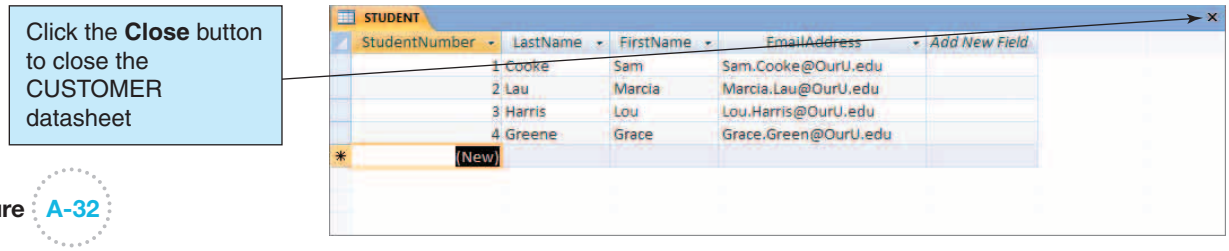


Figure A-32

The Completed STUDENT Datasheet

ClassNumber	ClassName	Term	Section
10	Chem101	Fall08	1
20	Chem 101	Fall08	2
30	Chem101	Spring09	1
40	Acct101	Fall08	1
50	Acct101	Spring09	1

Figure A-33

CLASS Data

NOTE: If you make a mistake and need to return to a cell, click the cell to select it, and you go into Edit mode. Alternatively, you can use **Shift-Tab** to move to the right in the datasheet and then press **F2** to edit the contents of the cell.

NOTE: Remember that LastName and FirstName *require* a data value. You will not be able to move to another row or close the table window until you have some value in each of these cells.

NOTE: Figure A-29 shows a column labeled Add New Field to the right of the Email column. This is a table tool in Datasheet view that you can use to create or modify table structures. We do not recommend using these tools; we prefer to use Design view instead!

7. Use the **Tab** key to move to the next row of the STUDENT datasheet and enter the data for Marcia Lau.
8. Enter the data for Lou Harris.
9. Enter the data for Grace Greene.
10. If necessary, adjust the datasheet column widths so that you can see the contents of the datasheet in one screen. The final result is shown in Figure A-32.
11. Click the **Close** button in the upper-right corner of the document window to close the STUDENT datasheet. If a dialog box appears asking if you want to save the changes you made to the layout (column widths), click the **Yes** button.

At this point, we need to enter data into the CLASS table, but *not* into the GRADE table. The data in the GRADE table will be entered only after we have created the relationships between the tables. The data for the CLASS table is shown in Figure A-33. Enter this data using the CLASS table in Datasheet view.

Modifying and Deleting Data in Tables in the Datasheet View

After data is entered into a table, you can modify or change it by editing the data values in the Datasheet view. You can also delete rows of data in the Datasheet view. For information on how to do this, see the Access 2007 documentation in the Help system.

Creating Relationships Between Tables

In Access, you build relationships between tables by using the **Relationships window**, which you access by using the **Tools | Relationships command**.

Creating the Relationship Between the STUDENT and GRADE Tables

1. Click the **Database Tools** tab to display the Database Tools command groups, as shown in Figure A-34.
2. Click the **Relationships** button in the Show/Hide group. As shown in Figure A-35, the Relationships tabbed document window appears, together with the Show Table dialog box. Note that along with the Relationships window, a contextual tab named Relationship Tools is displayed and that this tab adds a new command tab named Design to the set of command tabs displayed.
3. In the Show Table dialog box, click the **STUDENT** table to select it. Click the **Add** button to add STUDENT to the Relationships window.
4. In the Show Table dialog box, click the **GRADE** table to select it. Click the **Add** button to add GRADE to the Relationships window.
5. In the Show Table dialog box, click the **Close** button to close the dialog box.

Figure A-34

The Database Tools Ribbon

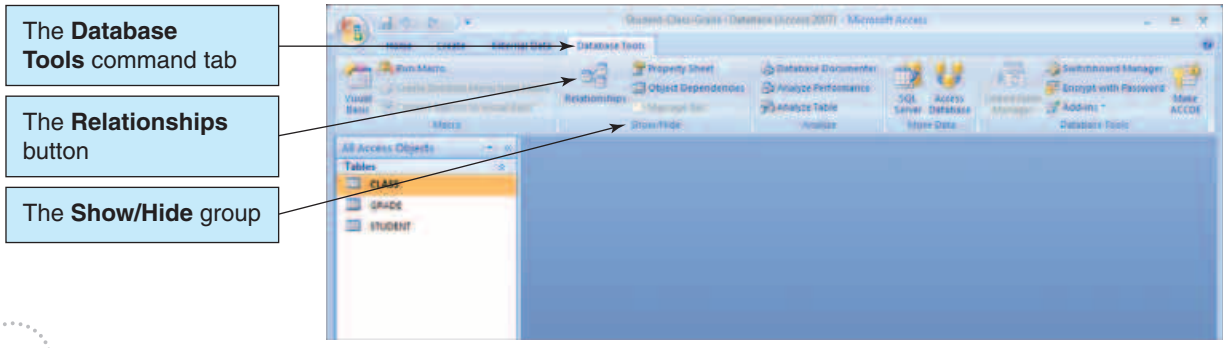
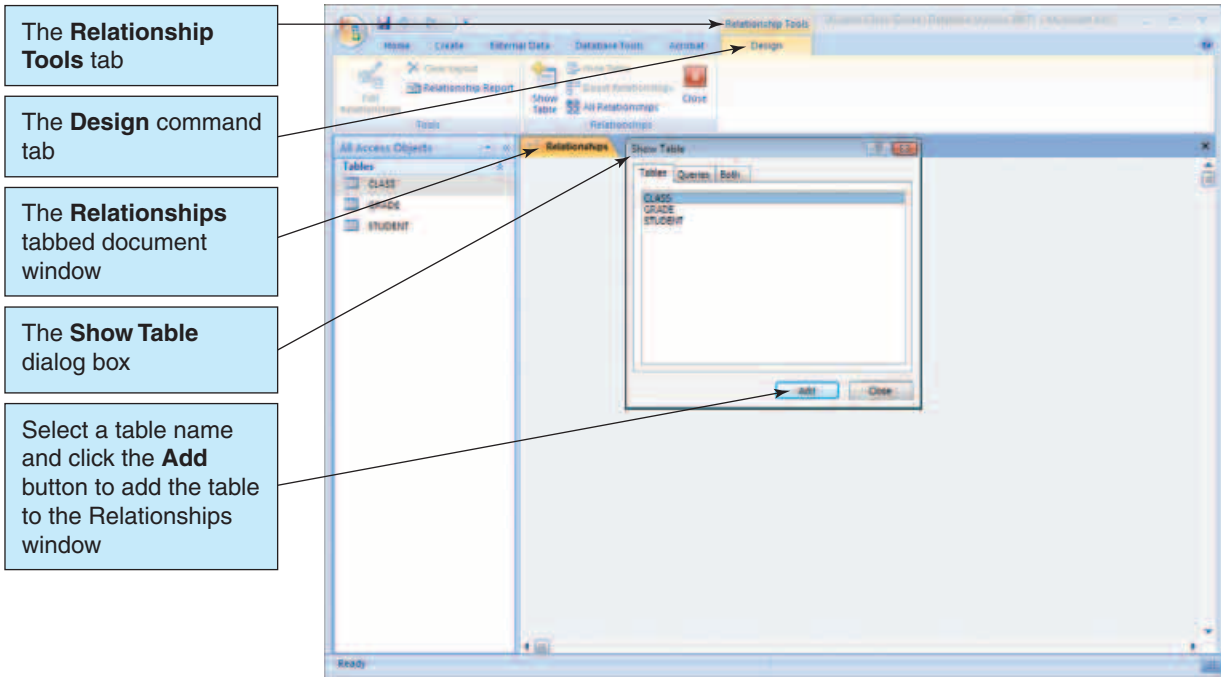


Figure A-35

The Relationships Window



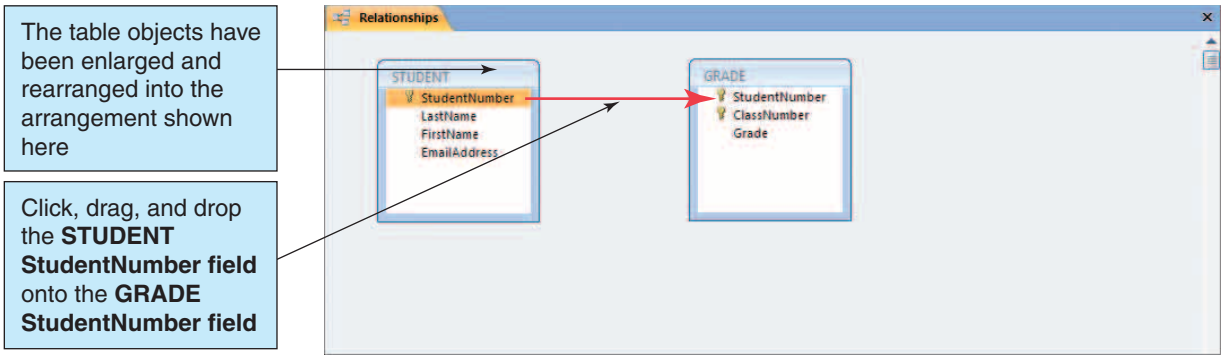


Figure A-36
The Table Objects in the Relationships Window

6. Rearrange and resize the table objects in the Relationships window using standard Windows drag-and-drop techniques. Rearrange the STUDENT and GRADE table objects until they appear as shown in Figure A-36. Now we are ready to create the relationship between the tables.

   NOTE: A formal description of how to create a relationship between two tables is "In the Relationships window, drag a primary key column and drop it on top of the corresponding foreign key column." It is easier to understand this after you have actually done it.

7. Click and hold the **column name StudentNumber in the STUDENT table** and then drag it over the **column name StudentNumber in the GRADE table**. The Edit Relationships dialog box appears, as shown in Figure A-37.

   NOTE: In CUSTOMER, CustomerID is the primary key, and in CONTACT, CustomerID is the foreign key.

8. Click the **Enforce Referential Integrity check box**.
9. Click the **Create** button to create the relationship between CUSTOMER and CONTACT. The relationship between the tables now appears in the Relationships window, as shown in Figure A-38.
10. To close the Relationships window, click the **Close** button in the upper-right corner of the document window. An Access dialog box appears, asking whether you want to save changes to the layout of relationships. Click the **Yes** button to save the changes and close the window.

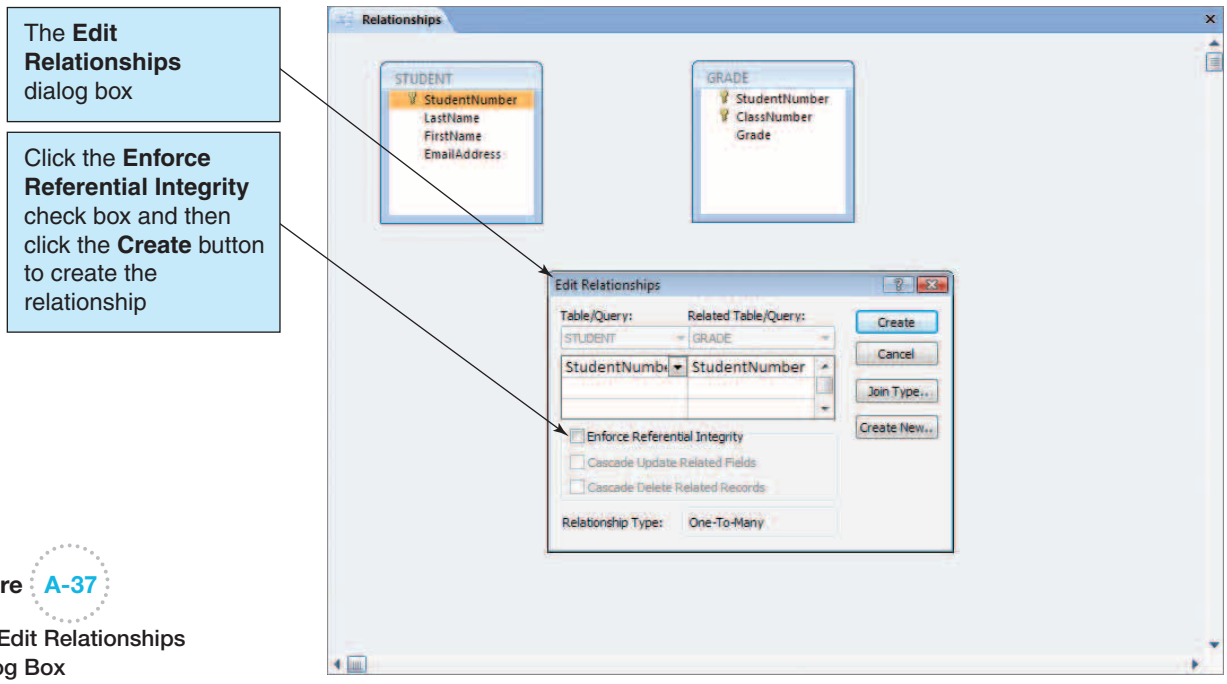


Figure A-37
The Edit Relationships Dialog Box

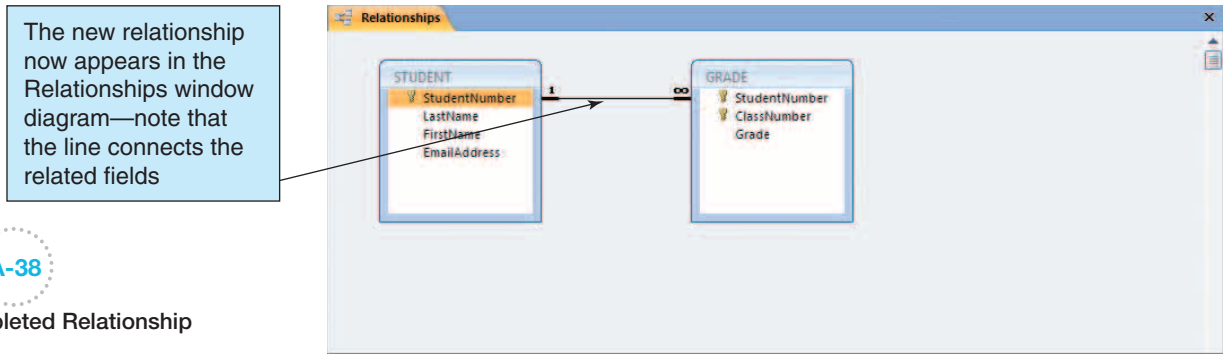


Figure A-38
The Completed Relationship

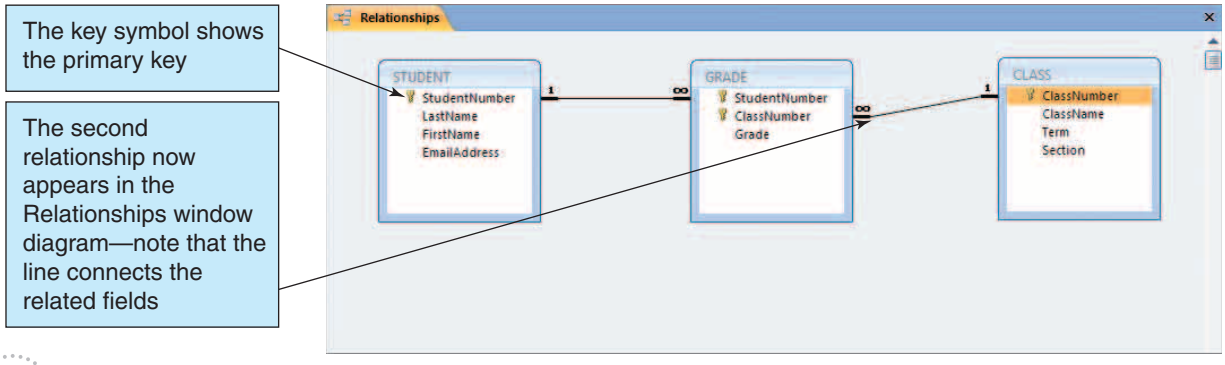


Figure A-39
The Completed Relationships for the Student-Class-Grade Database

Now we need to repeat essentially the same steps and create the relationship between CLASS and GRADE. When we are done, the relationships appear, as shown in Figure A-39.

At this point, referential integrity has been established between the foreign keys in GRADE and the corresponding primary keys in STUDENT and CLASS. This means that we cannot add a data value to StudentNumber or ClassNumber in CLASS unless the same value already exists in StudentNumber in STUDENT or ClassNumber in CLASS. This prevents us from inserting erroneous data into CLASS.

Now we can enter the CLASS data shown in Figure A-40 into the CLASS table. After this is done, we have completed creating and populating the Student-Class-Grade database.

Figure A-40
GRADE Data

StudentNumber	ClassNumber	Grade
1	10	3.7
1	40	3.5
2	20	3.7
3	30	3.1
4	40	3.0
4	50	3.5

Working with Microsoft Access Queries

We can create a Microsoft Access **query**, which is a question we ask the database, using two methods. The first is SQL, and the second is the Access version of **Query by Example (QBE)**, which uses the Access GUI to build queries. QBE is the Access default, and we will discuss it here. We will examine how to use SQL in Access in Chapter 2. It is important to note that Microsoft Access, unlike enterprise-class DBMS products, will allow you to save queries. This is a function of the applications development component of Microsoft Access, just as are forms and reports (which we will discuss later in this Appendix).

Queries are based on one or more tables (and can also be based on other existing saved queries). To understand how Microsoft Access QBE works, we will use QBE to create a multi-table query. The query will provide student enrollment and final grade information for each class and is the same query that was used as the basis for the Class Grade Report in Figure 1-11.

Creating and Running an Access QBE Query

1. Click the **Create** command tab to display the Create command groups, as shown in Figure A-41.
2. Click the **Query Design** button in the Other command group on the Create command tab.
3. The Query1 tabbed document window is displayed in Design view, along with the Show Table dialog box, as shown in Figure A-42.

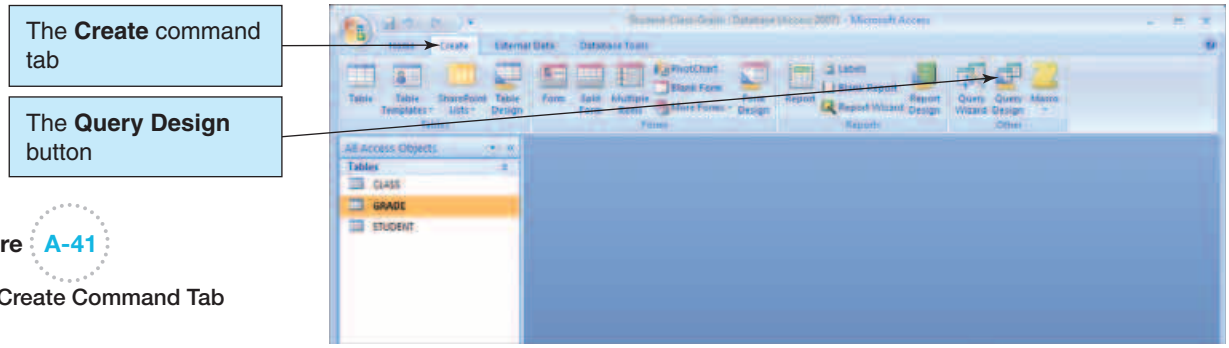


Figure A-41
The Create Command Tab

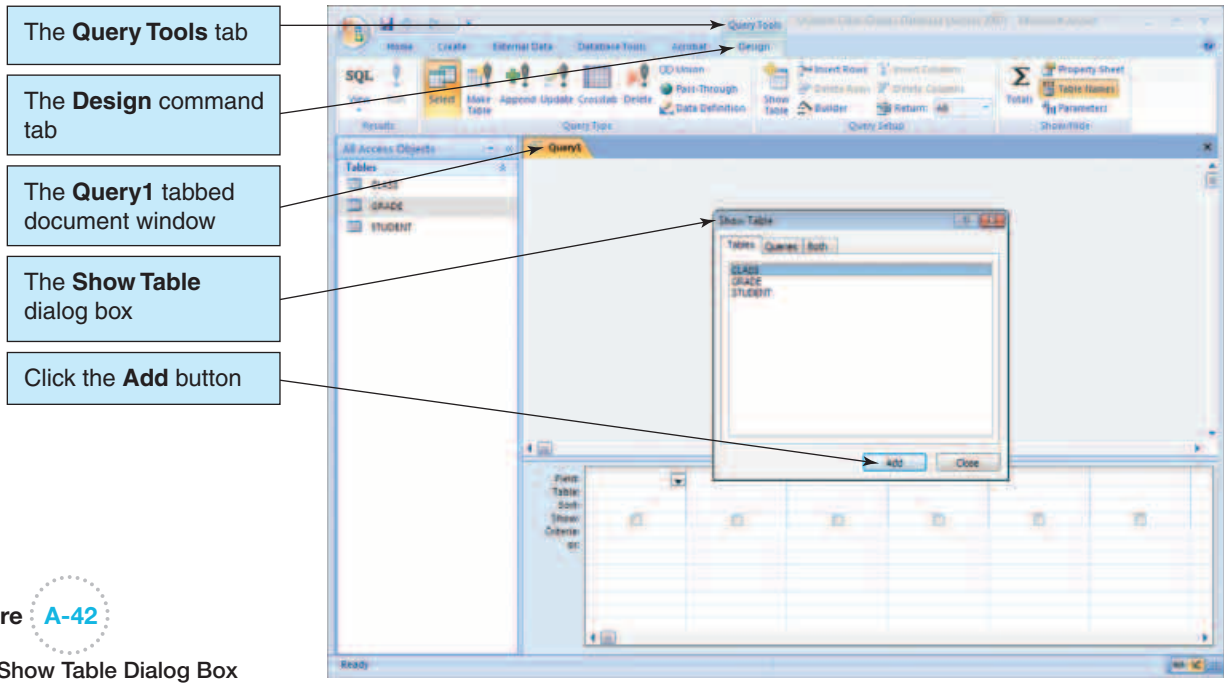


Figure A-42
The Show Table Dialog Box

4. In the Show Table dialog box, click **CLASS** to select the CLASS table. Click the **Add** button to add the CLASS table to the query.
5. In the Show Table dialog box, click **GRADE** to select the GRADE table. Click the **Add** button to add the GRADE table to the query.
6. In the Show Table dialog box, click **STUDENT** to select the STUDENT table. Click the **Add** button to add the STUDENT table to the query.
7. Click the **Close** button to close the Show Table dialog box.
8. Rearrange and resize the query window objects in the Query1 query document window, using standard Windows drag-and-drop techniques, until they appear as shown in Figure A-43.
9. Note the elements of the Query1 window shown in Figure A-43: Tables and their associated set of columns—called a field list—that are included in the query are shown in the upper pane, and the columns (fields) actually included in the query are shown in the lower pane. For each included column (field), you can set whether this column's data appear in the results, how the data are sorted, and the criteria for selecting which rows of data will be shown. Note that the first entry in the table's field list is the asterisk (*), which has its standard SQL meaning of "all columns in the table."
10. Click the **Shutter Bar Open/Close** button to collapse the Navigation Pane—we will need the extra space to build the query.
11. We include columns in the query by dragging them from the table's field list to a field column in the lower pane. Click and drag **ClassNumber in CLASS** to the first field column, as shown in Figure A-44. Note that the column is entered as ClassNumber from the table CUSTOMER.
12. Similarly, add the following columns to the query:
    a. CLASS.ClassName
    b. CLASS.Term
    c. CLASS.Section
    d. STUDENT.LastName
    e. STUDENT.FirstName
    f. GRADE.Grade
13. In the Sort: row of the Field properties, set the sort order for ClassNumber, Section, LastName, and FirstName to **Ascending**.

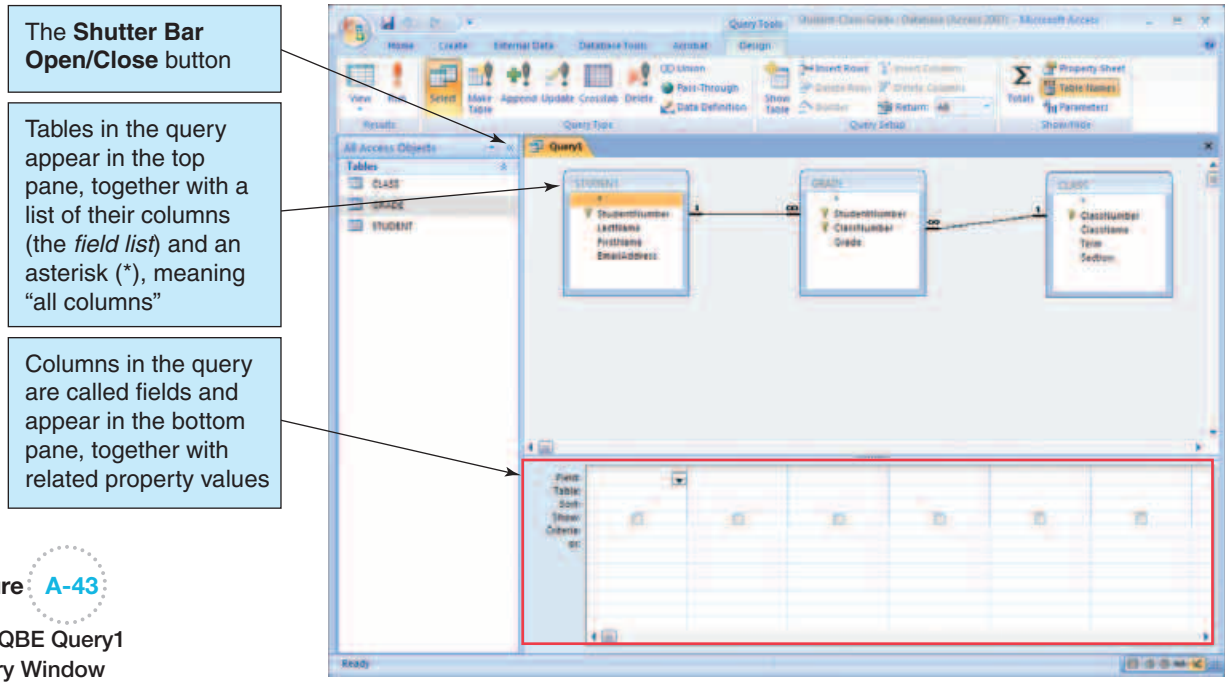


Figure A-43
The QBE Query1 Query Window

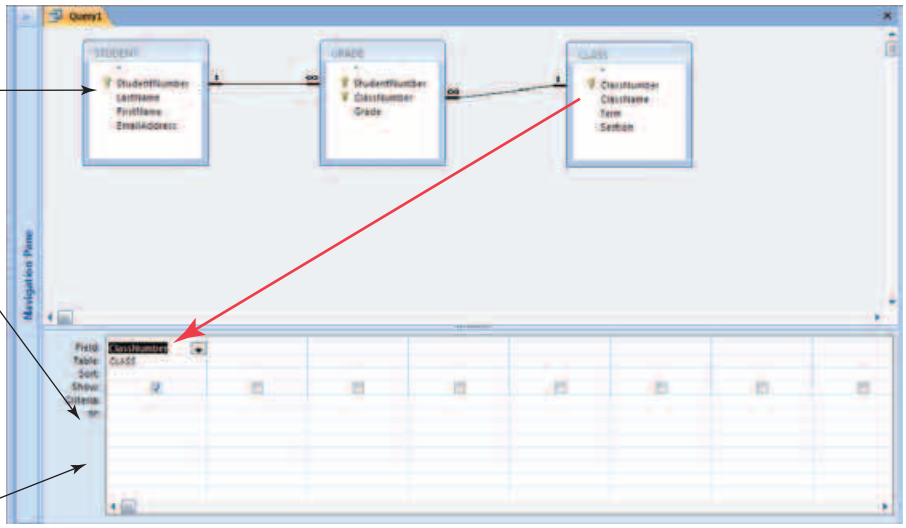
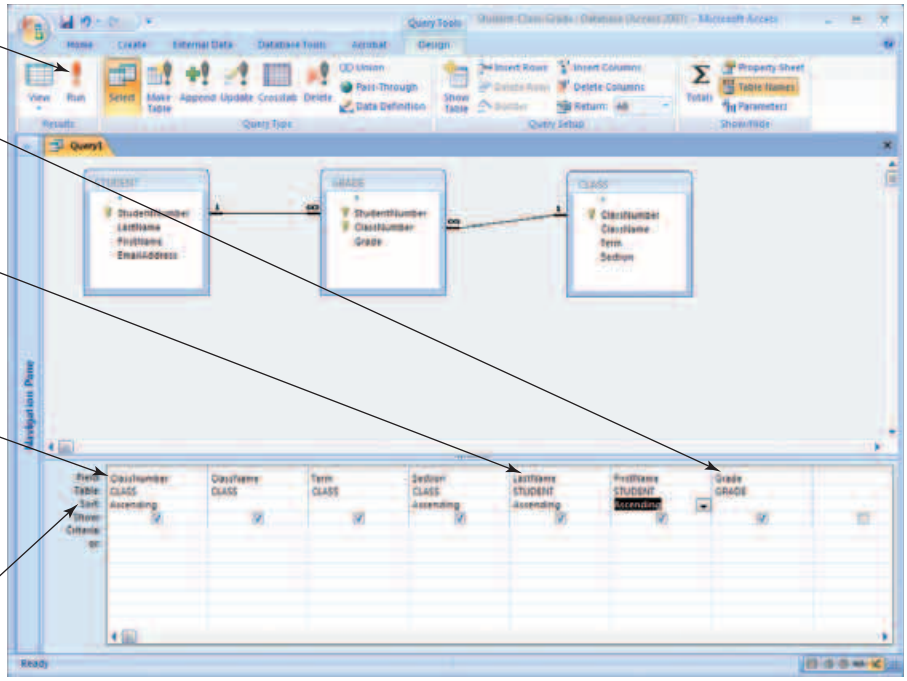


Figure A-44
Adding Columns to the QBE Query

14. The complete table now appears as shown in Figure A-45.
15. Click the **Shutter Bar Open/Close button** to expand the Navigation Pane.
16. Click the **Run** button on the Query Design in the Result command group of the Design command tab. The query results appear, as shown in Figure A-46.
17. To save the query, click the **Save** button on the Quick Access Toolbar to display the **Save As** dialog box. Type in the query name *QBEQuery-A-01* and then click the **OK** button. The query is saved, and the window is renamed with the new query name.
18. The query document window is now named *QBEQuery-A-01*, and a newly created QBEQuery-A-01 query object appears in a Queries section of the Navigation Pane.
19. Close the QBEQuery-A-01 query.
20. If Access displays a dialog box asking whether you want to save changes to the design of the query QBEQuery-A-01, click the Yes button.
21. The Student-Class-Database now appears as shown in Figure A-47.

Figure A-45
The Completed Three-Table QBE Query



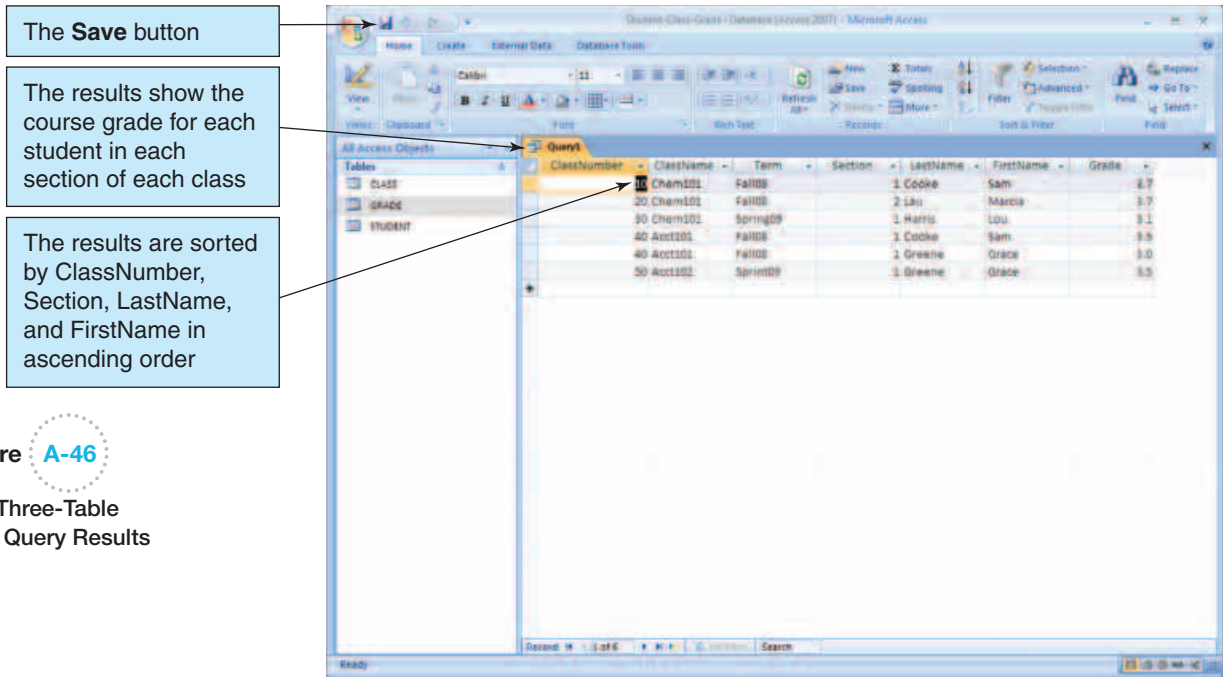


Figure A-46
The Three-Table QBE Query Results

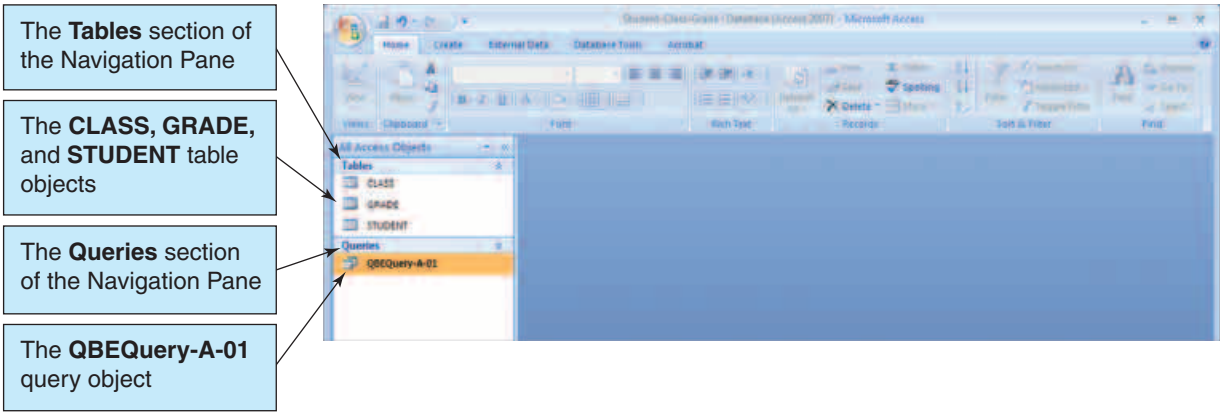


Figure A-47
The Database with the QBEQuery-A-01 Query Object

This is a fairly complex query, but it certainly illustrates how to use QBE for more complicated queries. And if you can run this query, you will be able to run queries using just one or two tables.

Microsoft Access Forms and Reports

Microsoft Access 2007 has the ability to create and store **forms** and **reports** as part of its application development tools. We will not discuss these features in detail, but we will note that Access has both a **Form Wizard** and a **Report Wizard** that will step you through the creation of basic forms and reports. These provide a good starting place for exploring forms and reports in Access 2007.

Creating Access 2007 Forms

The Form Wizard is somewhat hidden. The Form Wizard button is located in the More Forms gallery in the Forms group on the Create ribbon, as shown in Figure A-48. Once started, the Form Wizard takes us through a step-by-step process to create the form we want.

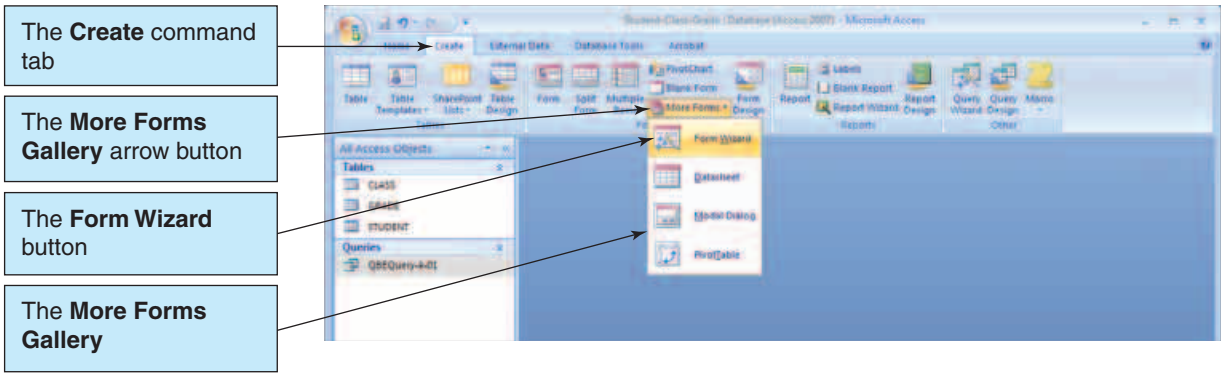


Figure A-48
The Form Wizard

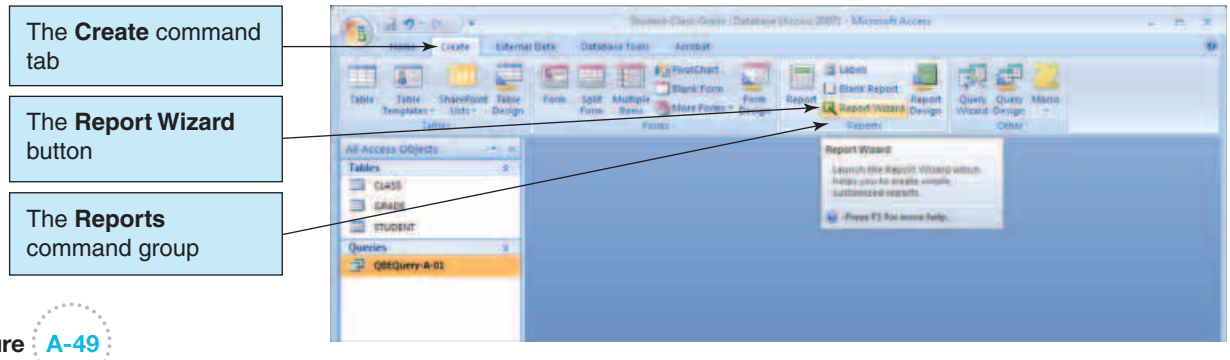


Figure A-49
The Report Wizard

Creating Access 2007 Reports

The Report Wizard is easier to find. The Report Wizard button is located in the Reports group on the Create ribbon, as shown in Figure A-49. Once started, the Form Wizard takes us through a step-by-step process similar to the one use by the Report Wizard, but this time to create the form we want.

Closing a Database and Exiting Access 2007

We have finished all the work we need to do in this appendix on getting started with Microsoft Access 2007. We have demonstrated how to create a database, build database tables, populate a table with data by using Datasheet view, create relationships between tables, and query a database using Access QBE. We have briefly introduced forms and reports.

At this point, you know enough to create and use basic Access databases. In particular, you know enough to create the Cape Codd database that will be used in Chapter 2 in our discussion of SQL queries. In fact, creating that database is part of the exercises in this appendix.

Having achieved our goal of getting you started using Microsoft Access 2007, we finish by closing the Student-Class-Grade database and Microsoft Access 2007.

Closing the WMCRM Database and Exiting Access 2007

1. To close the Student-Class-Grade database and exit Access 2007, click the **Close** button in the upper-right corner of the Access 2007 window.

Key Terms

AutoNumber
character
command tab
composite primary key
contextual command tabs
data entry form
datasheet
Datasheet view
Enforce Referential Integrity check box
foreign key
form
Form Wizard
key
Microsoft Office Fluent user interface
Navigation Pane
number
numeric
object
primary key
query
Query by Example (QBE)
Relationships window
remarks
report
Report Wizard
required
Ribbon
schema
surrogate key
text
Tools | Relationships command
type

Review Questions

A.1 Using the Student-Class-Grade database that you created in this Appendix:

A. Create and run an Access QBE query to duplicate the results in Figure 1-10. Save the query as QBEQuery-A-02.

B. Use the Form Wizard to create a data input form for the STUDENT table. Name the form *Student Data Input Form*. Create the data for two new students and add them to the STUDENT table.

C. Use the Form Wizard to create a data input form for the CLASS table. Name the form *Class Data Input Form*. Create the data for two new classes and add them to the CLASS table.

D. Use the Form Wizard to duplicate the CLASS form in Figure 1-9. Note that this form uses more than one table.

E. Use the Report Wizard to create a report of the data in the STUDENT table. Name the report *Student Data Report*.

F. Use the Report Wizard and the QBEQuery-A-01 query to duplicate the Class Grade Report shown in Figure 1-11.

A.2 In this exercise, you will build the Cape Codd database used for the SQL examples in Chapter 2. The Access 2007 tables and relationships are shown in Figure 2-2.

A. Create a new Access 2007 database named *Cape-Codd.accdb*.

B. The column characteristics for the RETAIL_ORDER table are shown in Figure A-50. Create the RETAIL_ORDER table.

C. The column characteristics for the SKU_DATA table are shown in Figure A-51. Create the SKU_DATA table.

D. The column characteristics for the ORDER_ITEM table are shown in Figure A-52. Create the ORDER_ITEM table.

Column Name	Type	Key	Required	Remarks
OrderNumber	Number	Primary Key	Yes	Long Integer
StoreNumber	Number	No	No	Long Integer
StoreZip	Text (9)	No	No	
OrderMonth	Text (12)	No	Yes	
OrderYear	Number	No	Yes	Integer
OrderTotal	Currency	No	No	

Figure A-50
Column Characteristics for the RETAIL_ORDER Table

E. The data for the RETAIL_ORDER table are shown in Figure 2-4. Populate the RETAIL_ORDER table.

F. The data for the SKU_DATA table are shown in Figure 2-4. Populate the RETAIL_ORDER table.

G. Create the relationship between the RETAIL_ORDER and ORDER_ITEM tables. Be sure to enforce referential integrity.

H. Create the relationship between the SKU_DATA and ORDER_ITEM tables. Be sure to enforce referential integrity.

Figure A-51
Column Characteristics for the SKU_DATA Table

Column Name	Type	Key	Required	Remarks
SKU	Number	Primary Key	Yes	Long Integer
SKU_Description	Text (35)	No	Yes	
Department	Text (30)	No	Yes	
Buyer	Text (30)	No	No	

Figure A-52
Column Characteristics for the ORDER_ITEM Table

Column Name	Type	Key	Required	Remarks
OrderNumber	Number	Primary Key, Foreign Key	Yes	Long Integer
SKU	Number	Primary Key, Foreign Key	Yes	Long Integer
Quantity	Number	No	Yes	Integer
Price	Currency	No	Yes	
ExtendedPrice	Currency	No	Yes	

I. The data for the ORDER_ITEM table are shown in Figure 2-4. Populate the RETAIL_ORDER table.

J. Why did you enter the ORDER_ITEM data only after creating the relationships between the tables?

K. Create a QBE to display Department and Buyer in the SKU_DATA table. Save the query as *QBEQuery-EA2-01*.