Introductory Access

TeachUcomp, Inc.

...it's all about you

TeachUcomp, Inc.
Phone: (877) 925-8080
Web: http://www.teachucomp.com

Introduction and Overview

Welcome to TeachUcomp, Inc.'s Introductory Access course. This class is the first of the Microsoft Access classes. Access is one of the most popular database programs available. This class is designed to familiarize the student with little or no knowledge of Access or relational database design with the basic components of the program and some of the fundamental aspects of relational data modeling.

Access is an excellent program to learn, as the skills that you learn within Access can save valuable time and money for organizations by automating, organizing and structuring their data processing capabilities.

Access is a multi-featured database program in which one creates powerful relational desktop databases or web-based apps that store and manipulate data. It is a very useful program and has many features that can automate and simplify job tasks. Whether you want it to create charts, reports, data entry forms, or data sources; Access can assist you in accomplishing these tasks quickly and easily.

The introductory segment of this course will focus on giving the student who possesses little to no knowledge of Access the basic overview of how Access works, the components needed to create a simple desktop database or web-based app, and their basic functionality.

Table of Contents

CHAPTER 1-
GETTING ACQUAINTED WITH ACCESS

1.1- Creating a New Database- 2013:

When Access opens, it displays a window which allows you to create a new database file that will contain either a desktop database or web-based app. A new Access database file is a container that will hold all of the tables, view definitions, forms, reports, queries, macros, and modules required by the desktop database or web-based app. Within Access, a desktop database is simply a database that is intended to be used on a single computer or within a local network. A web-based app is a database application that is intended to be shared on the Internet using SharePoint or Office 365 services.

After starting Access you can choose to create a new database file from scratch using a blank desktop database or web-based app or you can create one that contains some basic database objects by selecting one of the templates shown.

To create a new database file in Access, simply click on the name of the type of template that you want to use as the basis for your database file within the listing shown in the startup screen. At that point a smaller window will appear onscreen where you can enter a name for the file into the "File Name" field. For locally saved files, the default location where Windows will save the file will be shown below the name. To change this location you can click the small folder button at the right end of the "File Name" field to open a "File New Database" window where you can change the directory into which to save the file.

If you are saving a web-based app, then enter a URL address of a site to save the file to by typing the URL into the "Web Location" field.

Once you have given your Access file a name, and decided where to save it, then click the "Create" button within the window to create the Access database file. Once you have created a new Access file, you will then see the main Access user interface where you will design the objects within the desktop database or web-based app.

1.2- Creating a New Database- 2010:2007:

A new database is a container that will hold all of the tables, form, reports, queries, macros, and modules that you create. In Access 2010, you can create a new, blank database by clicking the "File" tab in the Ribbon. Then click the "New" command. Then select the "Blank database" choice in the "Available Templates" section. In Access 2007, you can create a new blank database by simply clicking the "Blank Database" button in the "New Blank Database" section of the "Getting Started with Microsoft Access" welcome screen.

In the "Blank Database" pane that appears at the right side of the screen, you can enter a name for the database into the "File Name:" text box. If you want to change the default folder where the database file will be stored, you can click the small folder button at the right end of the "File Name:" text box in order to launch the "File New Database" dialog box. Use this dialog box to give the new database file a name and also select where you want to save the file. When you are ready, click the "OK" button to close the dialog box.

Then click the "Create" button to create the new database file. Once that is done, the new blank database will appear in the main Access interface.

1.3- Overview of a Database:

In Access, you are manipulating a contained collection of smaller objects within the database file. Although the terms “database” and “table” are often used interchangeably, in Access you should refer to the entire collection of tables, queries, forms, reports, macros, and modules as the “database” and only refer to tables as “tables” for clarity’s sake.

Access is also what is referred to as a “relational database” program. In a relational database, you store large amounts of data into the smallest possible increments within tables. You then relate these tables together by joining common fields between them. In this way, you store less redundant data and your database will operate more quickly and efficiently. When you relate tables, you are then able to access any data in the related tables.

A database file is designed to store information and retrieve it at a later point in time. The many types of objects within a database file work together to allow you to do this. However, in order to create an effective and useful database file, you must learn how to design and create many different types of objects. This is one of the primary reasons that learning database design is more difficult than learning many other types of applications. Now you will examine the various types of objects in a basic database file and what their purpose is within the overall scheme of database design.

The first and most fundamental object type within a database is the **table**. A table is a collection of data about a certain subject: like customers, vendors or suppliers. It consists of columns and rows into which you store data. The columns all contain only one type of data, and are called “fields.” For example, within a customer table you might have a “First_Name” field into which you place only your customer’s first names. The rows in a table all contain one set of related field information for a single entry, and are called “records.” For example, in your customer table you may have a customer record that contains all of the field information about that customer contained within one row.

Tables are the building blocks of almost all other types of database objects. Tables contain all of the information that is to be stored, manipulated and retrieved. Therefore, almost everything in a database is fundamentally dependent on the tables and their structure. So, while tables are often the database objects with which new users are most familiar, it is important not to approach table design haphazardly. Errors made during the creation and design of the tables will often cause problems in the functionality of the related objects, forcing you to go back and re-design or edit the tables as well as the other related objects if you proceed with your database design too quickly. Creating well-designed data tables and joining them appropriately is one of the most difficult aspects of database design. It is certainly the aspect which many new users have the most difficulty understanding. It is also the most important aspect of database design.

The next type of database object to discuss is the **query**. The purpose of a query is to extract only the data that you want or need to view from the tables. These objects are the “heart” of database design-and the whole point of using databases. The queries provide the data that is needed by the other database objects, often working in the background. So mastering queries will also be an important part of creating a functional database. While queries are mainly used to extract data for reporting, you will also learn how they can be used to modify data, as well.

The next type of database object to review is the **form**. Forms are often used as user interfaces for the associated, underlying tables. They are also used to control the flow of the database program for users. A form typically allows users of the database to edit data or click other buttons that may launch reports and perform other user-related tasks within the database. Forms are the “face” of your database, as they are often all that the user will see and interact with when using a finished database application. Within web-based applications in Access 2013, the **view** is also used to serve the purpose of forms within the app.

The next type of database object to discuss is the **report**. Reports are a commonly used way of showing data extracted from the tables by the queries in a more “printer-friendly” format than the query itself

1.3- Overview of a Database (cont'd.):

can provide. Reports can also perform secondary calculations and analysis on the query data, making them very powerful data analysis tools.

The next type of database object to examine is the **macro**. Macros are small bits of visually-created programming that help automate processes within a database. For example, if you wanted a user to click a button in a form to launch a report, you could create a macro that automatically runs a report. You could then attach the macro to the button's "OnClick" event, so that when a user clicks the button within the form it runs the macro- thus running the report.

The final type of database object to discuss is the **module**. Modules are similar in purpose to macros. However, they are created in a non-visual environment. When creating modules, you actually have to type code into a separate, "Microsoft Visual Basic" application window. It uses a sister language of the Visual Basic language, called Visual Basic for Applications (or VBA), to create programs that can be much more complex in nature than the ones created by macros. However, many improvements have been added to the functionality of the macros in Access so that the usage of modules will rarely be needed by the typical Access database designer. Many database designers will not make much use of modules, but they can be valuable for the professional database designer.

A database should be simple, logical, and straightforward in its design. In general, you use **forms** to enter information into **tables**. The data is then stored into these tables, which are *related* to each other as necessary. You can the use **queries** to pull specific information from the **tables** in the database. The **queries** often form the basis for **reports,** which will then allow you to view the information you requested.

Once this system is in place, you can automate it by using **macros** and **modules** to simplify and streamline the processes involved in entering, storing and retrieving data. *This is the main reason that you use databases:* ***to enter, store, and retrieve data.***

1.4- The Access Interface:

Within the Access interface, unlike many other Microsoft applications, you have three different areas in which you will perform tasks: the "Application Window," which is the outer frame of the program that contains the Ribbon, which is used to execute commands; the "Navigation Pane," which displays all of the various objects in the database; and the "Tabbed Documents" area where you create, display, and edit database objects in their own separate, tabbed windows.

Inside of the main application window, you will see the "Navigation Pane" at the left side of the application window. As you select objects from the Navigation Pane to open and edit them, they will appear in their own tabbed windows to the right of the Navigation Pane. The "Application Window" consists of the Quick Access toolbar, the Ribbon, and the Status Bar at the bottom of the application window.

The "Navigation Pane" organizes and displays the various objects that have been created within the database. If you created a blank database then you will not have any objects in your database, other than the new, blank table that is opened by default when you create a new, blank database in Access. You can change the way that the navigation pane organizes and displays the database objects, however, that will be covered in a separate lesson. For now, it is enough to understand what this object is called and what its purpose is within the program.

Each object that you open or edit from within the Navigation Pane will appear in its own tabbed window to the right of the Navigation Pane. In each tab you will see the object's name. The type of object shown is indicated by the small icon next to the name of the object in the tab. You can click on the tabs of the objects that you have opened to display their content. You can close any currently displayed tabbed window by clicking the small "x" at the right end of the list of opened tabbed documents.

1.5- Touch Mode- 2013:

Because of the increased use of tablets, Access 2013 has been redesigned with a new mode to allow for easier access to the buttons and other commands within the Ribbon and Quick Access toolbar. This mode is called touch mode. When you enter touch mode within the Access 2013 interface, the Ribbon and Quick Access toolbar are enlarged and extra space is added around the buttons and commands within the Ribbon and Quick Access toolbar so that you can more easily access them on your touch-based tablet.

To enable touch mode within Access 2013, click the small drop-down arrow at the right end of the Quick Access toolbar to display a listing of the most commonly used commands. Then click or tap the "Touch/Mouse Mode" command in the drop-down menu to add that button to the Quick Access toolbar.

You can then enable or disable touch mode in Access 2013 by clicking or tapping the "Touch/Mouse Mode" button within the Quick Access toolbar. From the drop-down menu that then appears, you can select the mode you prefer to use: "Mouse" or "Touch." When "Touch" mode is enabled the buttons within the Ribbon and Quick Access toolbar will appear larger and with more space surrounding them onscreen. You can select the "Mouse" choice to toggle touch mode off, restoring the default size of the buttons onscreen.

1.6- Viewing Database Objects in the Navigation Pane:

As mentioned earlier, a database is really the entire collection of tables, queries, forms, reports, macros, and modules. In Access, you can only work with one database file at a time. Every time you open a database file in Access, its contents will appear in its own Navigation Pane.

Each type of database object is represented within the Navigation Pane, however the view that is displayed by default within the Navigation Pane may not allow you to view all of the objects easily. To be able to show the objects within the database, click drop-down arrow in the small title bar at the top of the Navigation Pane and then choose the "Object Type" command. You can then click the same drop-down arrow again. Notice that this time you will see the various types of database objects that you can show in a listing at the bottom of the drop-down menu. There is also the "All Access Objects" choice shown at the bottom of the drop-down menu. Ensure that this command is selected in order to display all objects in your database, grouped by category in collapsible and expandable groupings within the Navigation Pane. If you do not have any objects in your database yet, then this panel will show no groupings until you have created objects of the various types.

Once you have objects created within your database, then you can click on the name of the object category shown in the Navigation Pane to show the names of the objects that you have created. You can right-click on any object shown in the Navigation Pane and then select either the "Open" or "Design View" command in the pop-up menu that appears to open the selected object in its own tabbed window using the view that you specified.

1.7- Opening and Closing Databases- 2013:

To re-open a database you have already created and saved, first launch Access 2013. In the listing at the side of the initial window you can simply click on the name of the recently opened database that you wish to reopen shown under the "Recent" section.

To open a database file that is not listed here in Access 2013, click the "Open Other Files" link shown within the startup screen, or click the "File" tab within the Ribbon when a database file is opened and then click the "Open" command. At the right side of the backstage view that is shown, select the general location of the saved file from the listing shown. Then click the "Browse" button that appears to the right to select the desired folder to view within the "Open" dialog box that appears. When you can see the database file you wish to open within the dialog box, click it to select it and then click the "Open" button within the dialog box to open the selected file.

To close a database file in Access 2013, click the "File" tab within the Ribbon and then click the "Close" command at the left side of the backstage view.

If you want to close Access entirely click the "X" in the upper right corner of the application window.

1.8- Opening and Closing Databases- 2010:2007:

To re-open a database you have already created and saved, first launch Access. Then click the "Recent" category at the left side of the application. In the listing at the side of the initial window you can simply click on the name of the recently opened database that you wish to reopen.

If you want to open a database that is not in this list, then you can instead click either the "File" tab in the Ribbon if using Access 2010, or click the Microsoft Office button if using Access 2007. You can then click the "Open" command that appears. Doing that will launch the "Open" dialog box. Here, you navigate to the folder that contains the database file that you wish to open. When you see the database file that you want to open listed in the main window, you can click it to select it. Then click the "Open" button in the lower right corner of the "Open" dialog box to open the database.

If you wish to close a database file you can simply click either the "File" tab in the Ribbon if using Access 2010, or click the Microsoft Office button if using Access 2007, and then select the "Close Database" command.

If you want to close Access entirely click the "X" in the upper right corner of the application window or click either the "File" tab in the Ribbon if using Access 2010, or click the Microsoft Office button if using Access 2007, and then select the "Exit" button.

ACTIONS-
Getting Acquainted with Access

CREATING A NEW DATABASE FILE- 2013:

1. To create a new database file in Access 2013, simply click on the name of the type of template that you want to use as the basis for your database file within the listing shown in the startup screen.
2. At that point a smaller window will appear onscreen where you can enter a name for the file into the "File Name" field.
3. For locally saved files, the default location where Windows will save the file will be shown below the name.
4. To change this location you can click the small folder button at the right end of the "File Name" field to open a "File New Database" window where you can change the directory into which to save the file.
5. If you are saving a web-based app, then enter a URL address of a site to save the file to by typing the URL into the "Web Location" field.
6. Once you have given your Access file a name, and decided where to save it, then click the "Create" button within the window to create the Access database file.

CREATING A NEW DATABASE FILE- 2010:2007:

1. In Access 2010, click the "File" tab in the Ribbon and then click the "New" command. Then select the "Blank database" choice in the "Available Templates" section.
2. In Access 2007, you can create a new blank database by simply clicking the "Blank Database" button in the "New Blank Database" section of the "Getting Started with Microsoft Access" welcome screen.
3. In the "Blank Database" pane that appears at the right side of the screen, you can enter a name for the database into the "File Name:" text box.
4. If you want to change the default folder where the database file will be stored, you can click the small folder button at the right end of the "File Name:" text box in order to launch the "File New Database" dialog box. Use this dialog box to give the new database file a name and also select where you want to save the file.
5. When you are ready, click the "OK" button to close the dialog box.
6. Then click the "Create" button to create the new database file.

OVERVIEW OF A DATABASE:

Name:	Description:
Table	Contains and displays information stored in its columns and rows. Each data type is stored in a column called a FIELD. A set of data is stored in a row called a RECORD. The table is the basic data storage object in a database.
Query	Used to extract particular records and data types from a table. A way of pulling only the records that you need/want to see.
Form	A screen that is used for providing user interfaces in Access. Much like a **view** in a web-based app.
Report	A way of displaying data from your tables for others to use. Reports can perform calculations on field information, and are generally used to display data from queries and then perform data analysis on these results.
Macro	A set of commands in Access that are stored together as a single unit. You can run macros to perform multiple actions in Access. It is basically a small program.
Module	A stored procedure that is written in Visual Basic. You can write and run modules to enhance the power of Access with additional programming.

ACTIONS-
Getting Acquainted with Access

THE ACCESS INTERFACE:

1. In Access, you have three main areas where you will perform tasks; the “Application Window,” the “Navigation Pane,” and the “Tabbed Documents” area.
2. When Access opens, it displays a window which allows you to create a new, blank (empty) database or create a database from one of the templates shown. Once you create a database, you will then see the main Access user interface where you will design the database.
3. The “Application Window” consists of the Quick Access toolbar, the Ribbon, and the Status Bar at the bottom of the application window.
4. The “Navigation Pane” is located at the left side of the application window. The “Navigation Pane” organizes and displays the various objects that have been created in the database.
5. Each object that you open or edit from within the Navigation Pane will appear in its own tabbed window to the right of the Navigation Pane. In each tab you will see the object’s name. The type of object shown is indicated by the small icon next to the name of the object in the tab. You can click on the tabs of the objects that you have opened to display their content.
6. You can close any currently displayed tabbed window by clicking the small “x” at the right end of the list of opened tabbed documents.

USING TOUCH MODE- 2013:

1. To enable touch mode within Access 2013, click the small drop-down arrow at the right end of the Quick Access toolbar to display a listing of the most commonly used commands.
2. Then click or tap the “Touch Mode” command in the drop-down menu to add that button to the Quick Access toolbar.
3. You can then enable or disable touch mode in Access 2013 by clicking or tapping the “Touch Mode” option button within the Quick Access toolbar.
4. When the option button appears selected, then touch mode is enabled and the Ribbon and Quick Access toolbar will appear larger onscreen.
5. You can click or tap the “Touch Mode” button again to deselect the button and toggle touch mode off, restoring the default size of the onscreen elements.

VIEWING DATABASE OBJECTS IN THE NAVIGATION PANE:

1. To be able to show the objects within the database, click drop-down arrow in the small title bar at the top of the Navigation Pane and then choose the “Object Type” command.
2. You can then click the same drop-down arrow again. Notice that this time you will see the various types of database objects that you can show in a listing at the bottom of the drop-down menu. There is also the “All Access Objects” choice shown at the bottom of the drop-down menu. Ensure that this command is selected in order to display all objects in your database, grouped by category in collapsible and expandable groupings within the Navigation Pane. If you do not have any objects in your database yet, then this panel will show no groupings until you have created objects of the various types.
3. Once you have objects created within your database, then you can click on the name of the object category shown in the Navigation Pane to show the names of the objects that you have created.
4. You can right-click on any object shown in the Navigation Pane and then select either the “Open” or “Design View” command in the pop-up menu that appears to open the selected object in its own tabbed window using the view that you specified.

ACTIONS-
Getting Acquainted with Access

OPENING AND CLOSING DATABASES- 2013:

1. To re-open a database you have already created and saved, first launch Access 2013. In the listing at the side of the initial window you can simply click on the name of the recently opened database that you wish to reopen shown under the “Recent” section.
2. To open a database file that is not listed here in Access 2013, click the “Open Other Files” link shown within the startup screen, or click the “File” tab within the Ribbon when a database file is opened and then click the “Open” command.
3. At the right side of the backstage view that is shown, select the general location of the saved file from the listing shown.
4. Then click the “Browse” button that appears to the right to select the desired folder to view within the “Open” dialog box that appears.
5. When you can see the database file you wish to open within the dialog box, click it to select it and then click the “Open” button within the dialog box to open the selected file.
6. To close a database file in Access 2013, click the “File” tab within the Ribbon and then click the “Close” command at the left side of the backstage view.
7. If you want to close Access entirely click the “X” in the upper right corner of the application window.

OPENING AND CLOSING DATABASES- 2010:2007:

1. To re-open a database you have already created and saved, first launch Access.
2. Then click the “Recent” category at the left side of the application.
3. In the listing at the side of the initial window you can simply click on the name of the recently opened database that you wish to reopen.
4. If you want to open a database that is not in this list, then you can instead click either the “File” tab in the Ribbon if using Access 2010, or click the Microsoft Office button if using Access 2007.
5. You can then click the “Open” command that appears. Doing that will launch the “Open” dialog box.
6. Navigate to the folder that contains the database file that you wish to open.
7. When you see the database file that you want to open listed in the main window, you can click it to select it.
8. Then click the “Open” button in the lower right corner of the “Open” dialog box to open the database.
9. If you wish to close a database file you can simply click either the “File” tab in the Ribbon if using Access 2010, or click the Microsoft Office button if using Access 2007, and then select the “Close Database” command.
10. If you want to close Access entirely click the “X” in the upper right corner of the application window or click either the “File” tab in the Ribbon if using Access 2010, or click the Microsoft Office button if using Access 2007, and then select the “Exit” button.

EXERCISES-
Getting Acquainted with Access

Purpose:

1. To be able to find and use the basic objects in the Access application in Access 2013.

Exercises:

1. Open your Access application.
2. In the startup screen, click the “Blank desktop database” template to select it.
3. In the small window that then appears onscreen, type “test” into the “File Name:” field and then click the “Create” button.
4. Find the “Navigation Pane” at the left side of the Access window.
5. Click the small “x” button at the far right end of the “Table1” tab to close the tabbed window.
6. Click the “File” tab in the Ribbon.
7. Click the “Close” command at the left side of the Backstage View.
8. Under the “Recent” section at the left side of the startup screen, click the “test” entry in order to reopen the database that you just created.
9. Click the “X” button in the far upper-right corner of the Access application to exit the program.

EXERCISES-
GETTING ACQUAINTED WITH ACCESS

Purpose:

1. To be able to find and use the basic objects in the Access application in Access 2010 and 2007.

Exercises:

1. Open your Access application.
2. If using Access 2010, click the “File” tab in the Ribbon.
3. In the welcome screen, click the “Blank Database” option to select it.
4. At the right side of the window, enter “test” into the “File Name:” text box, and then click the “Create” button.
5. Find the “Navigation Pane” at the left side of the window.
6. Click the small “x” button at the far right end of the “Table1” tab listing to close the tabbed window.
7. If using Access 2010, click the “File” tab in the Ribbon, and then click the “Close Database” command at the left side of the Backstage View. If using Access 2007, click the Microsoft Office button and then select the “Close Database” command.
8. If using Access 2010, click the “Recent” command at the left side of the Backstage View shown when you click the “File” tab in the Ribbon. Then click the “test” entry shown in the section to the right in order to reopen the database that you just created. If using Access 2007, click the “test” entry in the “Open Recent Database” pane at the right side of the “Getting Started with Microsoft Office” window to reopen the database.
9. If using Access 2010, click the “File” tab in the Ribbon again to display the Backstage View of the database file, and then select the “Exit” command to close the application. If using Access 2007, click the Microsoft Office button and then select the “Exit Access” command to exit the program.

CHAPTER 2-
CREATING RELATIONAL DATABASE TABLES

2.1- THE 'FLAT-FILE' METHOD OF DATA STORAGE

2.2- THE RELATIONAL MODEL OF DATA STORAGE

2.3- TIPS FOR CREATING A RELATIONAL DATABASE

2.4- CREATING RELATIONAL DATABASE TABLES

2.5- ASSIGNING A PRIMARY KEY TO A TABLE

2.1- The 'Flat-File' Method of Data Storage:

Access is a *relational* database application. So what does the term *relational* mean, and how is this important? The term *relational* describes the method used for storing data within the database tables. However, it may be easier to understand the relational model of data storage by contrasting it with another method of storage that you may be more familiar with: the 'flat-file' method.

Information is frequently stored in large 'flat-files.' For example, assume that you want to create a database file that stores your company's customer information. You would begin by listing the different *attributes* of the customer that you wish to record. You may want to record customer information like the "first name," the "last name," the "company name," and other relevant pieces of information. Perhaps you could create a table in an application like Microsoft Excel where you can create columns for each piece of information that you wish to record. You can then list each customer's information in the rows underneath the columns, creating a basic table. Assume it looks like the following example.

	A	B	C	D	E	F	G
1	FirstName	LastName	Company	Address	City	State	Zip
2	Jon	Doe	Cost-Mor	1564 Crestview Ln.	Lansing	MI	48841
3	Henry	King	Shopalot	567 Elm St.	Detroit	MI	48543
4	Jenna	Smith	Shopalot	567 Elm St.	Detroit	MI	48543
5	Donetta	Smith	Smith Manufacturing	100 Main St.	Grand Rapids	MI	48867

For many types of databases, the structure shown would work well. This is a 'flat-file' list or table. What you are doing when using this type of database is recording a single piece of information, like the "FirstName," "LastName," or "Address," about a single entity- in this example, a customer. The reason that this type of data structure works well in the example given is because for each entity (the customer), you are only recording information that has a "1 to 1" relationship to the entity.

So, what does this "1 to 1" relationship between the entity (the customer) and the data you are recording ("FirstName," "LastName," etc.) mean? What this means is that for each entity or subject (in this case- the customer), you are only recording information about that entity for which there would only be one "answer." For example, each customer would only have one "first name" and one "last name." They would work for only one "company." So the term "1 to 1" refers to the relationship between the subject of the table (customers) and the data being collected about the entities. Because for each (one) customer, there is only one possible piece of data to record in the column, the relationship between the data and the entity is "1 to 1." If this is the type of database that you are trying to create, simple Microsoft Excel tables will work well.

The problem occurs when you try to use a 'flat-file' approach to model a more complex entity or subject, like "sales." For example, assume you wanted to expand the customer database from the last 'flat-file' database to include sales data. Now, in addition to the information already being collected, you also want to record each customer sale. First, you would start by listing what data about each sale that you want to record. Keeping the example simple, assume you decide to record the "sale date," the "items" purchased, the "quantity" of items purchased, and the "amount" paid for each item. You may decide to add the following columns to the 'flat-file' data structure.

	A	B	C	D	E	F	G	H	I	J	K
1	FirstName	LastName	Company	Address	City	State	Zip	SaleDate	Items	Quantity	Amount
2	Jon	Doe	Cost-Mor	1564 Crestview Ln.	Lansing	MI	48841	1/1/2013	Shoes	1	$ 50.00
3	Jon	Doe	Cost-Mor	1564 Crestview Ln.	Lansing	MI	48841	1/1/2013	Shoelaces	1	$ 2.50
4	Jon	Doe	Cost-Mor	1564 Crestview Ln.	Lansing	MI	48841	1/1/2013	Shoe Polish	1	$ 12.75
5	Henry	King	Shopalot	567 Elm St.	Detroit	MI	48543				
6	Jenna	Smith	Shopalot	567 Elm St.	Detroit	MI	48543				
7	Donetta	Smith	Smith Manufacturing	100 Main St.	Grand Rapids	MI	48867				

TeachUcomp Teacher:
Notice the redundant data that must be stored!

2.1- The 'Flat-File' Method of Data Storage (cont'd.):

This may appear to work, at first glance. However, you will immediately begin to encounter problems when you begin to enter records into the file. To begin with, each time a customer makes a purchase, you must re-enter all of the "FirstName," "LastName," etc. information again. This alone is irritating enough.

However, you will also soon run into another problem: What do you do when a customer purchases multiple items in an order? One solution often proposed at this point is to enter another row (with all of the redundant information) for each item purchased. However, you will find that this file will grow quite quickly down the table, and you will also have to enter a lot of redundant customer data for each item purchased. This is not an elegant solution and will inevitably waste data storage space as well as the time and effort of the person who performs data entry.

Another solution often proposed at this point is to create additional columns (like "Item1," "Item2," "Item3," "Quantity1," "Quantity2," Quantity3," etc.) instead of having to enter additional rows of information. While this may seem like a good alternate solution, what will you do when someone purchases 100 items? Will you really create a set of 3 columns ("Item," "Quantity," "Amount") for each item purchased, producing a table over 300 columns across? Would you simply leave them blank if the person orders only 1 item, wasting valuable storage space? In this solution, you are simply substituting columnar growth (across) for vertical growth (down). This is not an elegant solution either.

So why is there a problem now, when there wasn't one earlier? The answer is that now you are no longer trying to model a "1 to 1" data relationship in the table. Recording sales information is simply more complex than recording customer information. What you are trying to record now is what is referred to as a "1 to many" relationship. Basically, for each entity (the customer), you are now trying to record data in the columns which could occur more than once per customer (the "Items" ordered). You would be in a sorry state if each customer could only purchase a single item. You must allow for the fact that in a sale, each customer may order *many* items. The relationship between customers and items purchased is a "1 to many" relationship. When you find that you are trying to model a "1 to many" relationship, it is then that you must abandon the 'flat-file' method of data storage where you try to place all of the information that you want to record into a single table, and instead turn to the relational model of data storage for the solution.

2.2- The Relational Model of Data Storage:

The relational model of data storage allows you to more easily and effectively model a complex entity or subject, like sales. The relational model of data storage eliminates redundant data entry and also creates less data to store, making the relational database model smaller and faster than the 'flat-file.'

When you create a relational database, you will first need to perform some *data modeling*. Data modeling allows you to ensure that you are recording all of the information needed, and also helps you identify the entities involved and their relationships to each other.

When you create a relational database, you need to identify the unique entities involved in the process that you are modeling. These "entities" will often become the various tables in your database. So, for example, in the sales database example from the last lesson, the "Customer" is an entity. Within each table created for each entity, you must *only* list fields, or columns, of information which share a "1 to 1" relationship with the entity, or subject, of the table. So for example, in a "Customer" table, you would want to place the field "FirstName," assuming that each customer only has a single "FirstName" to record. You would **not** want to place "Item" in the "Customer" table, as the relationship between the customer and the items purchased is "1 to many." So what would one do with the column of "Item?" In the relational model, *each field (column) of information is an attribute of an entity*. For what entity is "Item" an attribute? In other words, with what "entity" does the "Item" (a description of the item purchased) have a "1 to 1" relationship?

2.2- The Relational Model of Data Storage (cont'd.):

Perhaps you may initially think that the “Item” is an attribute of the “Sale.” However, could you have a single “Sale” with multiple “Items” ordered? Probably so. In that case, it must be an attribute of something else. In this case “Item” is probably going to be an attribute of an “Item” entity; meaning that you will probably need to create an “Item” table.

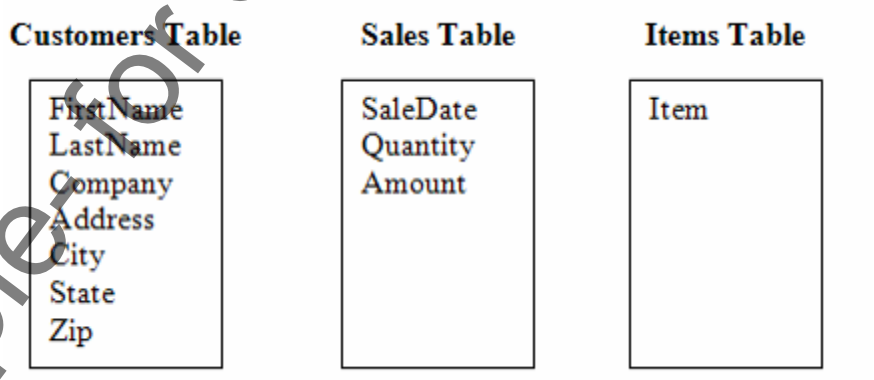
Many times, when initially approaching data modeling, it may be easier to list the various attributes that you wish to record, and then try to find what “entities” the attributes describe. The “entities” will become the various tables in your database. The “attributes” will become columns within the entity tables. Remember that each attribute (column) in your table must share a “1 to 1” relationship with the “subject” of the table (the entity).

In either case, you should probably keep your information written down on paper until you have a rough idea of what information it is that you want to record about the various entities involved with the process or system which you are trying to model. It is a rare feat to have your preliminary sketch of the relational database tables turn out to be the finished model that you will actually create in Access. Many times you will need to create a model, look for problems with the model you have created, and then edit the design until you are ready to attempt creating the tables.

Let’s take a look at a preliminary model of the “sales” database from the prior example. First, make a listing of the various pieces of information that you want to record. These become the attributes of the various entities. Next, try to find what entities these attributes describe and list those too.

Attribute:	Belongs to Entity:		Attribute:	Belongs to Entity:		Attribute:	Belongs to Entity:
FirstName	Customers		City	Customers		Items	Items
LastName	Customers		State	Customers		Quantity	Sales
Company	Customers		Zip	Customers		Amount	Sales
Address	Customers		SaleDate	Sales			

Next, make some sketches of the tables that show the fields of information within them. This can help you start to visualize what tables you will need to create, and will also allow you to see how the tables will eventually be *related* to each other in a larger, relational database structure.

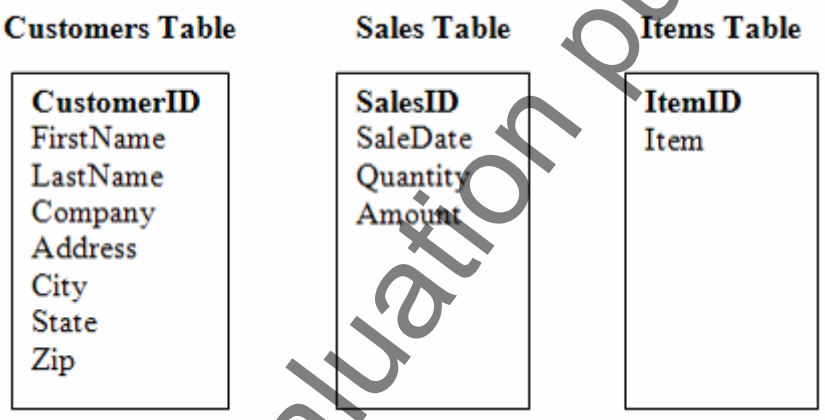


Once you have a rough idea of what you would like to record and what tables you will need in order to record the information you need, you must then ensure that each table has what is called a “primary key.” A primary key is a column, or combination of columns, that will produce a unique value for each row in a table. Many times, an additional column is added to the tables in order to provide this unique identification. You can assign each record a unique number in an “ID” column. For example, that is what your social

2.2- The Relational Model of Data Storage (cont'd.):

security number is used for by the government. You also have a unique driver's license number, as well. If you were recording any of these pieces of information, you could use those as the "primary key" in the table. If, however, you aren't recording any type of unique information, then often you must assign your own unique values. Many companies, for example, assign "Customer ID" numbers in order to uniquely identify each customer. Let's look at how your data model will change once you assign "primary keys" to your tables.

For example, you need a way to uniquely identify each customer. In the current data model, there isn't any kind of information that would enable you to uniquely identify each record (row) within the "Customers" table. So you could add an additional field (column) of information to this table: "CustomerID." Assume that you then add another column for "SalesID" to the "Sales" table, and an "ItemID" field to the "Items" table. In the sketch below, each "primary key" field is shown in bold within each table diagram. So, the data model would look something like this:



The "primary key" is a very important concept in a relational database, because it is through the primary key assignment that you create the necessary relationships between the data tables. For example, examine the relationship between the "Customers" table and the "Sales" table in terms of the "1 to 1" and "1 to many" relationship. In this case, for each (1) customer there can potentially be many sales. So, the tables will share a "1 to many" relationship. This is the most common type of relationship between tables, with extremely few exceptions. What you need to do next is find a way to join the "many" side of this relationship to the "one" side of the relationship. You need to relate each entry in the "Sales" table to a customer in the "Customers" table.

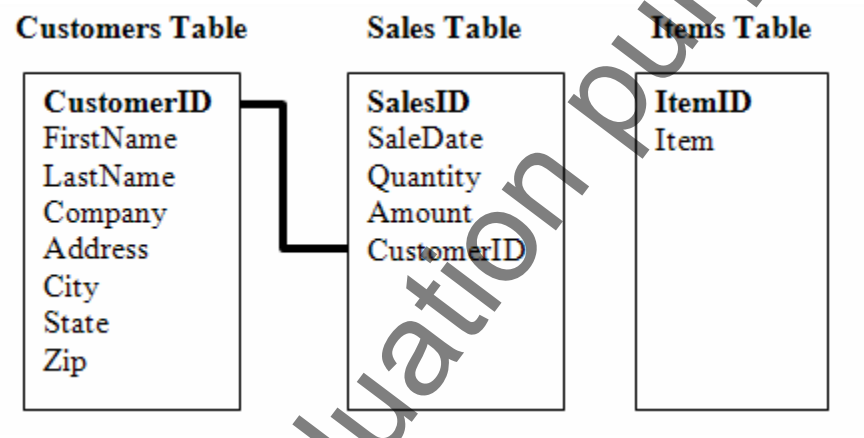
In order to join tables, they must have a shared, or common, field between them. This would be a field that contains the same kind of data in both tables. So, in this example, you are trying to assign each sale to a customer. To do this, you would want to add a field to the "Sales" table that corresponds to a field in the "Customers" table. Which field would you choose? The answer is: the "primary key" field!

Remember that each primary key field is designed to uniquely identify each record in the table, so you can add a field to the "Sales" table that will make a reference to the values in the "CustomerID" field of the "Customers" table. That way, when you enter a record into the "Sales" table in the future, you will only need to enter the "CustomerID" number of the customer to whom the sale was made, practically eliminating redundant data entry! So you can see one advantage of the relational model. In this model, you only have to enter the customer's data once in the "Customers" table, and then assign them a unique "CustomerID." When you then enter sales for that customer into the "Sales" table, you only need to make a reference to the appropriate "CustomerID" in the sales record to indicate who made the purchase! This allows you to

2.2- The Relational Model of Data Storage (cont'd.):

store much less redundant data, making them smaller and faster to use than the 'flat-file" table. It is also important to note that the "CustomerID" field which is added to the "Sales" table is **not** a primary key! That table already has a "primary key" field in the "SalesID" field, which uniquely identifies each sale- like a receipt number. Technically, the field in the "many" table which makes a reference back to the primary key in the "one" table is called a "foreign key." It's only purpose is to relate the two tables, and the values within a foreign key are almost always non-unique within the column.

Don't worry about the mechanics of the data entry, or how to create primary keys and table joins just yet. It will explained in later lessons. For now, just try to comprehend the concepts and reasoning behind the relational database design. Let's examine how the table diagram has changed to reflect the newly created relationship between "Customers" and "Sales."



Next, you will want to examine the other relationships between the tables. For example, what is the relationship between "Customers" and "Items?" Don't be hasty- not every table in the database has to be directly related to every other table. The only way that customers and items are related is that the customer purchases the items when making a sale. The "Customers" and "Items" do not have a direct connection. However, in a relational database, as long as every table is connected in an appropriate manner to the correct table, you can find out how they are related to each other through the tables by which they are connected. In summary, the "Customers" are connected to the "Items," but only through the "Sales."

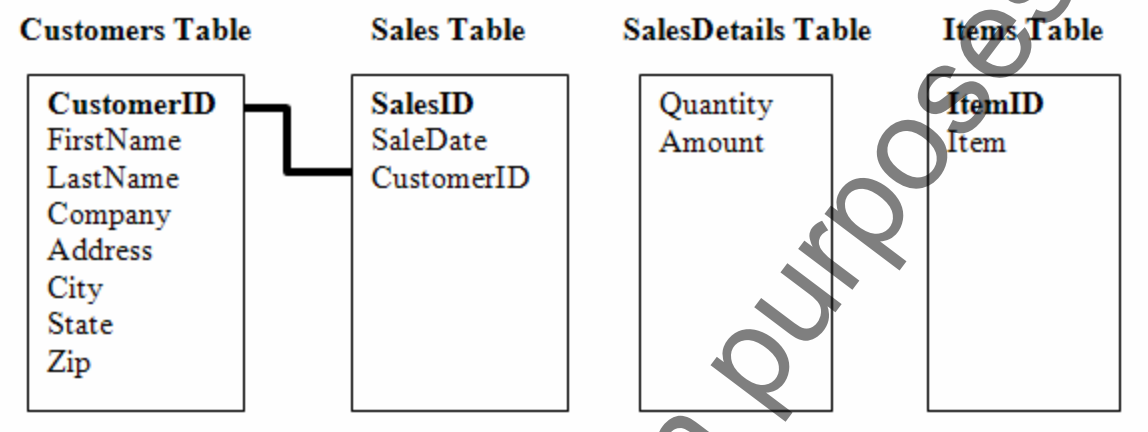
So, how are the "Sales" table and the "Items" table connected? Well, for each sale, there may be many items ordered. Also, each item may appear in more than one sale! In relational database design, you cannot (or **should** not) create a "many to many" relationship. That would make no sense from a strictly logical point of view. You need to be able to tell which items were ordered in which sale, while reducing the amount of data entry. Also, you may notice another problem with the current data model- the "Amount" field is attached to the "Sales" table. In this context- this field would be the "SalesTotal." If that is the case, then how can you record the price of each item at the time of sale? What if the price of each item changes in the future? Is the "Amount" also an attribute of the item?

What you are starting to see is that you need to be able to link the unique sales records to the unique items ordered on each sale. You need a "SalesDetails" table in order to do this. But what fields do you place into the new "SalesDetails" table? The answer is: anything that is an aspect of the "many" side of the "Sales" transaction. For example, the "SaleDate" field can stay in the "Sales" table because each sale only happens on a specific date. The "Quantity" of the items purchased at the time of sale is actually part of the "many" aspect of the sale and should be moved to the new "SalesDetails" table, along with the "Amount" field. The "CustomerID" will stay tied to the "Sales" table, as each purchase is made by a single customer.

2.2- The Relational Model of Data Storage (cont'd.):

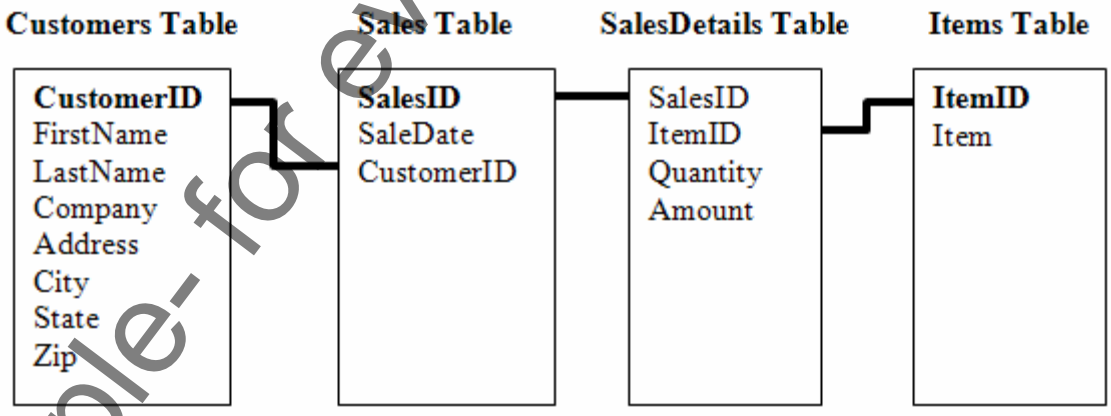
So now examine how this new “SalesDetails” table will affect the data model.

Below is a diagram of the new tables in the data model. You must also remember that the new table of “SalesDetails” will also need to contain a “primary key” field.



Before assigning the “primary key” field, look at how you will relate the “Sales” table to the “SalesDetails” table. The tables are related in that each sale may have one or more items purchased in each sale. So, you need to join each record in the “SalesDetails” table to the “Sale” record to which it corresponds. To do this, you will add a “foreign key” into the “SalesDetails” table that corresponds to the data in the “primary key” in the “Sales” table. So, you will add the “SalesID” field to the “SalesDetails” table.

Next, examine the relationship between the “Items” table and the “Sales Details” table. In this case, for each item ordered in a transaction shown in the “SalesDetails” table, it must make a reference to a unique item in the “Items” table. So, you will add the foreign key field of “ItemID” to the “SalesDetails” table. Then you can “join” the tables through their “shared” or common fields. Examine how the data table diagrams in the data model may look after performing these two tasks.

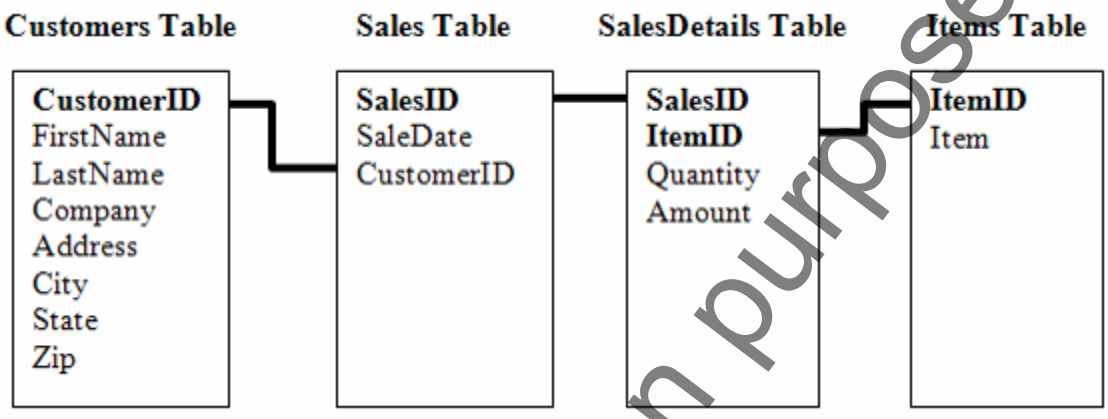


Now you have created all of the necessary relationships between the tables. However, the “SalesDetails” table is still missing a “primary key” field. You could add another field as the primary key, such as “SalesDetailID.” However, you could also see if there is a combination of fields that already exists that, when combined, produce a unique row value. In fact there is: the combination of “SalesID” plus “ItemID.” There should never be a repeating combined value in those two columns. If there were, it would mean that the same item was recorded in the same order twice. If that were the case, it should only be recorded once in the “SalesDetails” table with a “2” into the “Quantity” field. So, assuming that you make

2.2- The Relational Model of Data Storage (cont'd.):

this combination of fields the primary key for the table, let's examine the data diagram.

This is the final data diagram based on the information that was needed to record the sales. Obviously, there is more information that could be recorded, but this example is only supposed to illustrate some of the decisions that should go into table design before you begin to create tables in Access.



2.3- Tips for Creating a Relational Database:

While there are no "hard and fast" rules about creating relational database tables, there are a few tips that you should try to follow when beginning database design. First, examine all current documentation used to collect and store the information that you now want to store in the new database. This step ensures that when you are creating your data tables and performing your data modeling, you won't leave out a critical part of your database. Doing that often leads to frustrating periods of re-design. Also, consider what the database will need to contain in terms of the forms and reports that you need to design. Also consider the need of the users who will want to run these reports and perform data entry in the forms. You should gather information from those users who need to use the database that you create.

Next, use the entity/attribute and relationship modeling that was discussed in the previous lesson. This is a helpful first step in discovering how your tables should be structured. In it also important to note that the way that you store data in the tables in a database is independent of how the users will input the data into the tables. Once you learn more about form design, this will become readily apparent.

When performing data modeling, you may want to start by listing the entities, or "subjects" of the tables, in the database along with their properties or attributes that you want or need to record. You may also find it easier to begin by listing the attributes and then trying to discover to which entities the attributes refer. Once you have accomplished this part, sketch the entities as tables and find or create the primary keys needed for each table. Sketch relationships between the tables and list the type of relationship that the tables share. About 99% of the time, this will be a "1 to many" relationship.

After you have a preliminary table sketch, you can then turn to "normalization" guidelines to assist you in analyzing the database structure for its relational "soundness" of design. These guidelines were created to assist the relational database designer in creating sound relational structures that do not break any of the foundational tenets of relational database design. While these are not "rules" per se, you shouldn't violate one of the normalization guidelines without having a very good reason for doing so. If you decide to do so, document your reasoning for making such a break. When a relational database follows one of the normalization guidelines, it is said to follow the "form" of the guideline.

2.3- Tips for Creating a Relational Database (cont'd.):

While there have actually been many normalization guidelines proposed, many database designers find it is adequate to design their relational databases to satisfy the normalization guidelines through the third or fourth normal "forms."

The first normal form requires atomic, or unique, values at each column and row intersection in the entity table. There should be no repeating groups. Thus, no "Item1," "Item2" column design like you may see in a 'flat-file' table layout.

Second normal form requires that every "non-key" column in a table must depend on the primary key. A table must also not contain a "non-key" column that pertains to only part of a composite, or multi-column, primary key.

The third normal form requires that no "non-key" column should depend on another "non-key" column. This is very similar to the second normal form. You shouldn't have a field that is an attribute of a non-primary key column in a table.

Fourth normal form forbids independent "1-to-many" relationships between primary key columns and non-key columns.

As you begin your modeling your database tables, be sure to document your work as you create your initial designs. Correct violations of normal form that you see, or make conscious decisions to override them. Always document why you chose to make the changes that you do make. After you create your basic tables and relationships, review your design. Then create the database tables in Access and enter some preliminary or "test" data to see if your design works or how well it works. Reevaluate your design and fix flaws as required. Always document the reasons that you decide to change the table design.

2.4- Creating Relational Database Tables:

Tables are so commonly thought of when one speaks of a database that the terms are practically interchangeable. A table is an organized structure that holds information. It consists of "fields" of information into which you enter your "records." A field is a single column within a table, consisting of one ***category*** of information. A record is ***a collection of related data fields*** that describe a single item contained in a row within a table.

After creating the data model for your database tables, you will want create the table structures in Access. At least at this point, Access will make it easy for you to do so.

One way to start is by creating the tables in "Design View." To create a new database table in "Design View," click the "Table Design" button in the "Tables" group on the "Create" tab in the Ribbon. When you do this, a new table will display in the tabbed documents area.

In "Design View," you will not see the actual data that is stored in your table, you will only see a representation of the structure of the table. You will, however, have much greater control over the structure and properties of the fields within your table than if you begin by creating a table in "Datasheet View." Note that this window is divided into two panes: the "design grid" at the top where you enter the field names and data types; and the "properties" section beneath it. In the "design grid" at the top of the table design view there is a small box at the extreme left end of each field. This is the "row selector" button. You can click this small square to select the entire row. You will need to do this frequently in Access, so become used to where this object is.

In "Design View," you begin by entering the field names into the "Field Name" column. The "Field Name" column is where you type the names for the fields in your new table. Field names must be unique within a table, and should be brief, yet descriptive. You should also consider **not** placing spaces within the

2.4- Creating Relational Database Tables (cont'd.):

field names. If you want, you can adopt a convention such as capitalizing the first letter of each word in a field name, or using the underscore character instead of a literal space between words in a field name. Also, the order of fields entered into this column will be the order that they will be displayed from left to right in the "datasheet view," which is the view that allows you to see the actual data in the tables.

Next, for each field that you create, you must assign it a data type by using the drop-down that appears when you click into the "Data Type" column to the right of the field name. Each field that you create must have a data type assigned to it. This tells Access what kind of data you will be storing within the field. The default data type for new fields in Access 2013 is "Short Text." In Access 2010 and 2007, this default data type was simply called "Text." In relational databases, the more varied kinds of data types that exist within a table, the quicker it will be to index and query those tables, so feel free to change the type as needed. Review the various data types that you can assign to fields in tables.

Data Types in Access	
Name:	Description:
Short Text *("Text" in 2010 & 2007)*	Contains text, or a combination of text, numbers, and other information. Maximum length is 255 characters in length.
Long Text *("Memo" in 2010 & 2007)*	A longer version of a text field. Maximum length is 65, 535 characters.
Number	Can contain only numeric data that you want to perform calculations on. NOT phone numbers or zip codes as you do not perform calculations with these numbers, they are text fields.
Date/Time	Contains a date or time code. Useful for Date/Time calculations.
Currency	Similar to the Number data type in function, but formatted as currency. Uses fixed point calculation, which is faster than the floating point calculation used in Number fields.
AutoNumber	Assigns a unique numeric ID to all records entered in the table. Useful as a primary key field.
Yes/No	Stores Logical data types: "Yes/No," "True/False," "On/Off," "-1/0." Used when only two possible values in a field can exist.
OLE Object	Connects a field to another Windows application. You can use OLE Object data types for additional graphics, calendars, or video/audio files.
Hyperlink	Contains a hyperlink to an address on the World Wide Web.
Attachment	Allows you to attach any type of supported file, such as images, or spreadsheets, for example. Provides greater attachment flexibility than the OLE Object field and also uses storage space more efficiently than OLE fields do.
Calculated	Allows you to create a calculated field, which contains a value that is derived by performing a function on other table fields using an expression that you create.
Lookup Wizard…	Guides you through setting up a lookup field, which will then contain values from another table, query, or values that you enter by hand. Useful for combo boxes and list boxes in forms.

2.4- Creating Relational Database Tables (cont'd.):

Below the design grid is the “Properties” section where you can set the properties of the currently selected field within the table. In Access 2010, you will also see the properties of the table itself in the “Properties” pane at the right side of the table. Do not confuse the two panes. The properties that are shown at the bottom of the design grid are the properties of the currently selected field. In this area the properties of the currently selected field will be displayed on two tabs called “General” and “Lookup.” You can edit or set the field’s properties here by changing values shown, as needed. You will look at some of the properties that you can change for selected fields in a later lesson. For now, simply familiarize yourself with where the field properties are located.

2.5- Assigning a Primary Key to a Table:

In Access, you should assign a **primary key** to each table that you create. A primary key is simply a field or group of fields that acts as a *unique identifier* for each record in the table. So you should use a field or group of fields that will always contain a unique value as your primary key.

The advantages of assigning a primary key will become more evident as you learn about relational database structures. You will see that when you assign a primary key to a table, it automatically indexes, or sorts, the table by the primary key. Also, because a primary key assigns a unique identification to each record, it reduces the chance of duplicate information being entered into the table.

You can create a primary key on a single field by clicking the row selector button of the field which you want to assign as the primary key of the table within the table design view. Then just click the “Primary Key” button in the “Tools” group on the “Design” tab within the “Table Tools” contextual tab in the Ribbon in order to set the selected field as the primary key of the table. You will see a small picture of a key appear in the row selector box to indicate that the field is now assigned as the primary key. If you select the primary key field and click the “Primary Key” button again, it will remove that field from being the primary key within the table.

You can create a multiple-field primary key by holding down the “Ctrl” key on your keyboard and then clicking the row selector boxes of the fields in the design grid that you want to select as being members of the composite primary key. Release the “Ctrl” key when you have selected the necessary fields, and then click the “Primary Key” button in “Tools” group on the “Design” tab within the “Table Tools” contextual tab in the Ribbon. If you have trouble finding a natural primary key within your table, consider creating a new “AutoNumber” data type field as a unique identifier for the records in your table, which you may then assign as the table’s primary key.

ACTIONS-
Creating Relational Database Tables

CREATING A NEW TABLE IN DESIGN VIEW:

1. Click the “Table Design” button in the “Tables” group on the “Create” tab in the Ribbon. When you do this, a new table will display in the tabbed documents area.
2. Type the name of the new table field into the “Field Name” column.
3. Press “Tab” on your keyboard to move to the next column to the right.
4. Select the drop-down menu in the “Data Type” column and assign the field a data type.
5. Press “Tab” on your keyboard to move to the “Description” column.
6. You may type a description of the data that will be stored in this field, if needed.
7. Press “Tab” on your keyboard to move down to the next row.
8. Repeat steps 6 through 11 above until you have created all of the necessary fields.
9. Click the row selector at the left end of the row that contains the field which you want to set as the “primary key” for the table.
10. Click the “Primary Key” button in “Tools” group on the “Design” tab within the “Table Tools” contextual tab in the Ribbon .
11. Click “Save” button in the Quick Access toolbar.
12. Type a name for the new table into the dialog box that appears, and then click “OK.”

ASSIGNING A PRIMARY KEY TO A TABLE:

1. Select the field or fields which you want to assign as the primary key of the table within the table design view.
2. Then click the “Primary Key” button in “Tools” group on the “Design” tab within the “Table Tools” contextual tab in the Ribbon in order to set the selected field or field as the primary key of the table.
3. If you select the primary key field or fields and then click the “Primary Key” button again, it will remove the selected field or fields from being the primary key within the table.

EXERCISES-
CREATING RELATIONAL DATABASE TABLES

Purpose:

1. To be able to create basic tables in a relational database.

Exercises:

1. Open your Access application.
2. Open the "test" database that you created in the Exercise at the end of Chapter 1.
3. Click the "Table Design" button in the "Tables" group on the "Create" tab in the Ribbon.
4. Enter the following for the "Field Name" and "Data Type" columns within the new table. Note that the "Short Text" data type is called "Text" in Access 2010 and 2007.

Employees

Field Name	Data Type
EmployeeID	AutoNumber
FirstName	Short Text
LastName	Short Text

5. Select the row selector button at the left end of the "Employee ID" row and then click the "Primary Key" button in the "Tools" button group on the "Design" tab of the "Table Tools" contextual tab within the Ribbon to assign it as the primary key within the table.
6. Click "Save" in the Quick Access toolbar and type "Employees" into the "Table Name:" text box within the "Save As" dialog box.
7. Click "OK" in the "Save As" dialog box to save the table structure.
8. Close the table.
9. Using steps 3 through 8 above as a guide, create the following tables. Don't forget to assign the primary key(s) for each! The name for each table is shown in the title bar of each table below and the field(s) that make up the primary keys are shown for each table, as well.

Sales

Field Name	Data Type
SaleID	AutoNumber
EmployeeID	Number
CustomerID	Number
Saledate	Date/Time

SalesDetails

Field Name	Data Type
SaleID	Number
ItemID	Number
Price	Currency
Quantity	Number

Items

Field Name	Data Type
ItemID	AutoNumber
ItemName	Short Text

Customers

Field Name	Data Type
CustomerID	AutoNumber
CompanyName	Short Text
Address	Short Text
City	Short Text
State	Short Text
Zip	Short Text

CHAPTER 3-
USING TABLES

3.1- Using Datasheet View:

While you can create a table in "datasheet view," it is **not** recommended. It is a poor place to design tables due to its lack of control over the data types assigned to the fields, and its complete inability to change the properties of fields. If you do decide to create a table in datasheet view, you should certainly view the table in "design view" at some point to ensure that it is correctly constructed. Some people like to start in datasheet view because they feel more familiar with the table layout shown in the datasheet view. If you are migrating to Access from a spreadsheet program like Excel, this may be the case for you, as well.

Datasheet view is for adding, editing, and deleting data records in a table. To open a table that you have created in datasheet view, double-click the name of the table that you want to open from the table list shown in the Navigation Pane. Alternately, you can select the name of the table that you want to open and then press the "Enter" key on your keyboard to open it in datasheet view, as well.

When you open a table in datasheet view, the fields are the columns displayed across the top of the table from left to right. The records are contained in the rows from top to bottom. You can view multiple records at a time in datasheet view. The record and field headings for the active cell, where data entry will occur, are highlighted in orange for ease of visibility. The active record is highlighted blue. The active cell is colored gray with a red border.

3.2- Navigating in Datasheet View:

When you are in datasheet view, you can move from left to right through the rows by pressing either the "Tab" or "Enter" keys on your keyboard. You can move from right to left by pressing "Shift"+"Tab" on your keyboard. You can also use the arrow keys on your keyboard to traverse the records, if you like.

If you prefer to use your mouse, you may click into any data field currently viewable in the screen to place your "insertion marker," which is the thin vertical blinking line, into the desired data field. Once the insertion marker is in the field, you can then change and edit the information within the field.

You can navigate through the records in large tables by using the vertical and horizontal scroll bars at the right and bottom sides of the datasheet view. Just click the arrow buttons at the ends of the vertical or horizontal scroll bars to quickly move through the records in the table.

You can also move through the records in the table using the "First Record," "Previous Record," "Next Record," "Last Record," and "New (blank) Record" buttons located in the button group at the lower left corner of the datasheet view to go to that record in your table. There is a white box in the center of the button group which shows you the number of the current record along with the total number of table records. You can "jump" to a record if you know its number by clicking into the white text box in the middle of the button group, typing the number of the record to which you want to move, and then pressing "Enter" on your keyboard. The number of the row that you entered will then be the selected row in your table.

3.3- Adding Records in Datasheet View:

In datasheet view, you will see a blank row that shows an asterisk (✱) in the row selector box at its left end. That is the "New Record" row. When you enter a new record into the table, it is added to the bottom of the table in the "New Record" row.

To add new records into a table, simply click into this row and enter your record information. The asterisk will change to a picture of a "pencil" as you do this, which lets you know which record you are currently editing. Another new "New Record" row will then appear below the row into which you are entering information. After opening a table in datasheet view, you can quickly go to the "New Record" row by clicking

3.3- Adding Records in Datasheet View (cont'd.):

the "New Record" button at the right end of the "Record Navigation" button group in the lower left corner of the datasheet view. It's the button with the [►✱] face. Your cursor will then automatically enter into that row, where you can enter the new record's data.

3.4- Editing and Deleting Records in Datasheet View:

To edit a record in datasheet view, simply click into the desired field of the record that you want to edit to place the insertion point into the field. Once the insertion point is within the field, you can edit the field information just as you would in a text document. To save the changes, just exit the cell. Changes to the data in a table are automatically saved after you make them. You only need to click the "Save" button in the Quick Access toolbar after making a structural change to a table, such as widening a column's display or adding or deleting columns of information within the table in design view.

To delete an entire record in datasheet view, you first need to click the row selector at the far left end of the row to select the entire record. Then press the "Delete" key on your keyboard, or click the "Delete" button in "Records" group on the "Home" tab in the Ribbon. You will then be prompted to decide whether or not you really want to delete the record. Click "Yes" or "No" depending on the situation, but remember that once records are deleted from the table it is a **permanent** change and cannot be "Undone" or recovered! This can lead to some frustration with the program if you aren't careful when deleting database records.

3.5- Inserting New Fields:

Once you have created your tables, you may need to modify their structures at a later point in time. You should make the changes in the table's design view. Another way to open a table in design view is to simply select the name of the table into which you want to insert a new field within the Navigation Pane. Then hold down the "Ctrl" key on your keyboard. Then press the "Enter" key once to open the selected table in design view. Then release the "Ctrl" key.

In the table design view, click the row selector of the field before which you want to insert the new field. For example, if you wanted to insert a new field for "Hire Date" into a table and wanted it to be placed **before** the "Start Date" field, you would click the row selector of the "Start Date" field. Next, click the "Insert Rows" button in the "Tools" group on the "Design" tab in the "Table Tools" contextual tab in the Ribbon. A new blank row will then be inserted above the selected row. This is your new field. You simply need to enter a name for the field and then select a data type to finish creating the new field. Then you need to save the structural modifications that you have made. Simply click the "Save" button in the Quick Access toolbar to save the structural modifications that you have made.

3.6- Renaming Fields:

With Access, you do have the flexibility to rename fields that you have already created. You should be **extremely** careful when you do this, as any changes that you make to field names are **not** necessarily updated in all of the related reports, forms, or queries that were previously created and therefore referred to the "old" field names. So, if you rename a field that was used by any other database objects, you should double-check those objects and change the name of the field reference there, if necessary. Doing that ensures that all of the related database objects that were looking for the "old" field name will now look for

3.6- Renaming Fields (cont'd.):

the new field name to access the data within the field.

To rename a field, open the table within which you want to rename a field in the table design view. Then click into the "Field Name" column of the field whose name you wish to change. Type a new name for the field, and then click the "Save" button in the Quick Access toolbar to save your structural changes.

3.7- Deleting Fields:

You can also delete table fields that you do not use. Once again, just as when changing a field name, make sure that there aren't any queries, forms, reports or macros that make reference to the field or use data contained within the field before you delete it.

To delete a field from a table, first open the table in table design view. Next, click the row selector button at the far left end of the field that you want to delete. Click the "Delete Rows" button in the "Tools" group on the "Design" tab of the "Table Tools" contextual tab. Access will display a warning prompt asking you if you really want to delete the field and all of the data within the field. To finish deleting the field and its data, click the "Yes" button. You can click the "No" button to cancel deleting the field, if needed. Then click the "Save" button in the Quick Access toolbar to save your structural modifications.

ACTIONS-
Using Tables

OPENING A TABLE IN DATASHEET VIEW:

1. Double-click the name of the table that you want to open in datasheet view from the Navigation Pane.
2. Alternately, you can select the name of the table that you want to open and then press the "Enter" key on your keyboard to open it in datasheet view, as well.

NAVIGATING IN DATASHEET VIEW:

1. Open the desired table in datasheet view.
2. Use the arrow keys on your keyboard to move "up," "down," "left" or "right" through the data.
3. Use "Tab" or "Enter" keys on your keyboard to move to the right through the fields, or "Shift"+"Tab" to move to the left through the fields.
4. Use the horizontal and vertical scroll bars in the datasheet view to scroll quickly through the records.
5. Use the "First Record," "Previous Record," "Next Record," and "Last Record" buttons in the lower left corner of datasheet view to move to the designated record within the datasheet view.

ADDING RECORDS IN DATASHEET VIEW:

1. Open the desired table in datasheet view.
2. Click the "New Record" button at the right end of the record navigation button group located in the lower left corner of the datasheet view. It's the button with the arrow and asterisk [►✱] on its face.
3. Enter the information into the fields in the "New Record" row. It's the bottommost row in the datasheet view that displays the asterisk [✱] at the left end of the row.
4. When you have finished entering the new record, you can move down to enter the next new record into the new row that has appeared.
5. Close the table when you are finished adding records.

EDITING RECORDS IN DATASHEET VIEW:

1. Open the desired table in datasheet view.
2. Click into the field in the record that contains the information that you want to edit. The "insertion marker" should appear within the designated field.
3. Edit the record data as needed. Changes are saved as soon as you make them.

DELETING RECORDS IN DATASHEET VIEW:

1. Open the desired table in datasheet view.
2. Click the row selector at the left end of the row which you wish to delete.
3. Click the "Delete" button in the "Records" group on the "Home" tab in the Ribbon.
4. Click "Yes" at the prompt to permanently delete the selected record.

ACTIONS-
Using Tables

INSERTING FIELDS INTO A TABLE:

1. Open the table into which you want to insert the new fields in design view.
2. Click the row selector of the row above which you wish to insert the new field to select the row.
3. Click the “Insert Rows” button in the “Tools” group on the “Design” tab of the “Table Tools” contextual tab in the Ribbon. A new row will be inserted above the row you initially selected. This is your new field.
4. Click into the “Field Name” column, and type a name for your new field.
5. Select the type of data it will store from the “Data Type” drop-down.
6. Type a brief description of the field into the “Description” column, if desired.
7. Click the “Save” button in the Quick Access toolbar to save your structural modifications.

RENAMING FIELDS:

1. Open the table which contains the field you wish to rename in design view.
2. Click into the “Field Name” column of the field that you want to rename, and type a new name.
3. Click the “Save” button in the Quick Access toolbar to save your structural modifications.

DELETING FIELDS:

1. Open the table which contains the field you wish to delete in design view.
2. Click the row selector of the row which you wish to delete.
3. Click the “Delete Rows” button in the “Tools” group on the “Design” tab of the “Table Tools” contextual tab in the Ribbon.
4. A pop-up dialog box will appear, asking you if you really want to delete this field. Click “Yes” to delete the field and all of its data.
5. Click the “Save” button in the Quick Access toolbar to save your structural modifications.

EXERCISES-
USING TABLES

Purpose:

1. To be able to navigate tables in datasheet view and add records to tables.

Exercises:

1. Open your Access application.
2. Open the "test" database you completed from the Exercise at the end of the previous chapter.
3. Display the database tables in the Navigation Pane, if needed.
4. Double-click the "Employees" table in the Navigation Pane to open it in datasheet view.
5. Enter the following records. Don't worry about the "AutoNumber" field- it will assign the records a number automatically as you enter the adjacent data into the row:

Employees

EmployeeID	FirstName	LastName
1	Joe	Smith
2	Fred	Smith
3	Mary	Jones
4	Greg	King
5	Jack	Wells
(New)		

6. Close the table, and then enter the following records into the tables shown below and on the next page:

Customers

CustomerID	CompanyName	Address	City	State	Zip
1	Compco	100 Main St.	Lansing	MI	48912
2	Sal's Auto Shop	550 Elm St.	Holt	MI	48842
3	Capital Consulting	125 Crescent	Lansing	MI	48912
4	Fred's Food Store	650 Lincoln St.	Ionia	MI	48846
5	Flowers N' More	233 E. Grand River	East Lansing	MI	48823
6	Kandy Korner	180 Hagadorn	East Lansing	MI	48823
7	Rodgers Roofing	250 Pine St.	Lansing	MI	48821
(New)					

Exercises (cont'd.):

6. Continue entering the following records into the tables listed below. You will need to manually type the information into the "SaleID" field for the "SalesDetails" table. Close and save the tables when you are finished.

SalesDetails

SaleID	ItemID	Price	Quantity
1	1	$1.50	2
2	2	$2.00	5
3	3	$5.00	10
4	4	$20.00	3
5	5	$99.00	1
6	6	$0.50	10
6	7	$1.00	150
6	8	$125.00	1
6	9	$1.25	20
7	10	$2.00	2
8	8	$125.00	2
9	1	$1.50	3
9	2	$2.00	2
9	4	$20.00	6
10	1	$1.50	10
11	8	$125.00	2
12	9	$1.25	6
13	10	$2.00	5
14	4	$20.00	5
15	6	$0.50	11
16	5	$99.00	3
17	1	$1.50	4
18	5	$99.00	2

Sales

SaleID	EmployeeID	CustomerID	Saledate
1	1	2	1/1/2007
2	1	2	1/2/2007
3	1	2	1/3/2007
4	2	1	1/1/2007
5	2	3	1/2/2007
6	2	3	1/3/2007
7	3	4	1/1/2007
8	3	2	1/15/2007
9	4	4	1/10/2007
10	4	2	1/10/2007
11	5	6	1/21/2007
12	5	6	1/22/2007
13	4	7	1/25/2007
14	5	5	1/25/2007
15	4	5	1/15/2007
16	1	4	1/10/2007
17	1	3	1/24/2007
18	2	2	1/20/2007
(New)			

Items

ItemID	ItemName
1	Pens
2	Paperclips
3	Markers
4	Paper
5	Palm Pilots
6	Notebooks
7	Manilla File Folders
8	Filing Cabinets
9	Hanging File Folders
10	Staples
(New)	

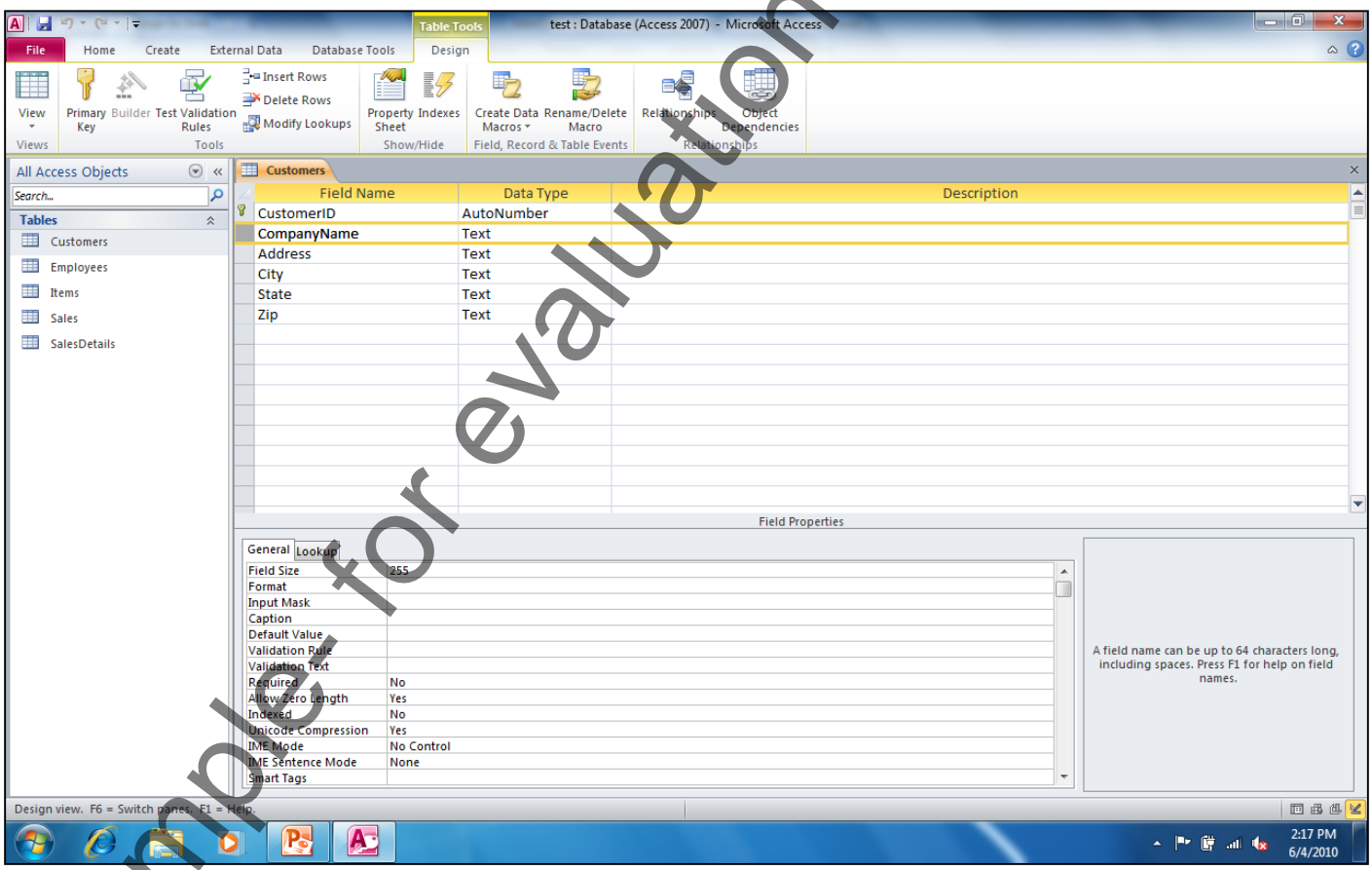
CHAPTER 4-
Field Properties

4.1- Setting Field Properties:

You can set the properties of the table fields that you create in the design view of the table. When you open tables in design view, you name the fields and assign them a data type using the top half of the screen which is called the table design grid. Below that, in the "Field Properties" section, you set the properties of the field that is currently selected in the table design grid on the two tabs labeled "General" and "Lookup."

To set a field property, you must first find the specific property of the field that you would like to set on the tabs. Most of the time, the property that you are trying to set is on the "General" tab. You only use the "Lookup" tab when you are manually setting properties of "lookup" fields which display values from another table or list.

The "General" tab contains field properties that allow you to control the size, display, default values, and many other features of the selected field. You can click into a property box and view information about its function or purpose in the right pane of the "Field Properties" section. If you need additional help in setting the value of a particular property, you can click into the property field about which you have questions, and then press the "F1" key for additional assistance. The help file will appear in a separate window, which you can read and print, if desired.



4.2- The Field Size Property:

You can use the "Field Size" property of a text field to set the number that you type as the maximum allowable number of characters in the selected field. This can be useful in restricting the amount of data that can be entered into the field. Access allows up to 255 characters in a text field, and also assigns that as the default field size.

Number fields are different from text fields when you set the "Field Size," in that you set the field size by assigning a type of number that can be stored in the field. The sizes available are: "Byte," "Integer," "Long Integer," "Single," "Double," "Replication ID," and "Decimal." Their various sizes and decimal restrictions are illustrated in that following table.

The default size for number fields is **Long Integer**. This is one of the largest field sizes. In Access, things will run more quickly if they are smaller, so you can use a smaller field size if you want. If not, don't worry, you probably won't notice any serious performance problems. Note that if you are joining the "Number" field to an "AutoNumber" field in a table relationship, then you should leave the setting at "Long Integer."

Name	Numeric Range:	Maximum Number of Decimals:
Byte	0 to 255 (no fractions)	0 (rounds numbers)
Integer	-32768 to 32767 (no fractions)	0 (round numbers)
Long Integer	-2,147,483,648 to 2,147,483,647 (no fractions)	0 (round numbers)
Single	-3.4×10^{38} to 3.4×10^{38}	7
Double	-1.797×10^{308} to 1.797×10^{308}	15
Replication ID	16 byte GUID	None
Decimal	-10^28-1 to 10^28-1	28

4.3- The Format Property for Date/Time Fields:

You can set the "Format" property for date/time fields to change the way that they will display dates and times in the table in datasheet view. The following settings are available for the "Format" property when you have a date/time field selected in Access.

Name	Description:	Example:
General Date	Default setting. If only a date is entered it will display the date. If only a time is entered it will only display the time.	8/2/2007 11:11:11 PM
Long Date	The day/month names are spelled out.	Thursday, August 2, 2007
Medium Date	The month is abbreviated.	02-Aug-07
Short Date	Date appears as numbers separated with forward slashes.	8/2/2007
Long Time	Time contains hours, minutes, and seconds. Colon separated and followed by AM/PM.	11:11:11 PM
Medium Time	Same as Long Time, but with no seconds.	11:11 PM
Short Time	Military time, no seconds.	23:11

4.4- The Format Property for Logical Fields:

You can set the "Format" property for logical fields to change the way that they will display in forms and reports. The following formats are available for logical fields in Access. To set this property, simply select the logical field in the table design grid. Then click into the "Format" property in the field properties section and select a choice from the drop-down menu available.

Name	Logical "True" Value:
True/False	True
Yes/ No	Yes
On/Off	On

4.5- Setting Default Values for Fields:

You can set the "Default Value" property to specify a value that the field should contain when it is created with new records. For example, you may have a "Yes/No" field for which you want to set a default value of "No." The value that you set can be a number, a text value, a date, or even some sort of calculated expression. You can apply this to almost any field type into which you wish to place a default value. For example, if you wanted to have a default value of "MI" appear in a "State" column within one of your database tables, you could select the "State" field in the table's design view and then type the text "MI" into the "Default Value:" field property. Then, when you enter new records in the future, they will appear with the "MI" value already contained in the "State" field. This can be a handy timesaver for fields where you want to record the data, but the data is almost always the same. This way, you'll only have to change the value in the field when it differs from the default value.

4.6- Setting Input Masks:

You can set up input masks to dictate a pattern used for data entry in selected fields. Access provides an easy step-by-step routine called the "Input Mask Wizard" that helps you to apply input masks to selected "text" and "date/time" fields.

To do this, first select the field to which you wish to apply an input mask. Click into the "Input Mask" property, and then click the "Expression Builder" button at the far right end of that property box (it has an ellipsis symbol on it […]) to launch the "Input Mask Wizard." You will need to save the table first. After saving the table, the "Input Mask Wizard" dialog box will appear. The wizard shows some of the most common input masks for a field. You can select whichever input mask you need. Answer each question posed in each screen, and click "Next" to continue. When you are done, you can click the "Finish" button.

Remember that the "Format" property will conflict with the "Input Mask" property if they are in opposition (which can happen at times with "date/time" fields), so you may want to edit or remove your "format" property first.

4.7- Setting Up Validation Rules and Responses:

You use the "Validation Rule" and "Validation Text" properties in tandem. Setting the "Validation Rule" property allows to use the "Expression Builder" dialog box to create a specific condition that will only allow data entry that meets the specified condition into the field. For example, you could specify that field entries for a "Birth Date" field cannot be a future date. Once the rule is in place, you type the error message that you want Access to display when a entry doesn't meet the specified criteria into the "Validation Text" property.

To set a validation rule on a selected field, you must enter an expression (which is much like an Excel formula) into the "Validation Rule" property. You can either type it into the box by hand or you can use the "Expression Builder" dialog box to assist you in creating the rule.

To use the "Expression Builder," click into the property field and then click the small ellipsis button that appears at the right end. This will launch the "Expression Builder," where you can use the buttons and functions listed in this dialog box to create your validation rule. Just click "OK" in the "Expression Builder" when you are finished to insert the rule you created into the property field.

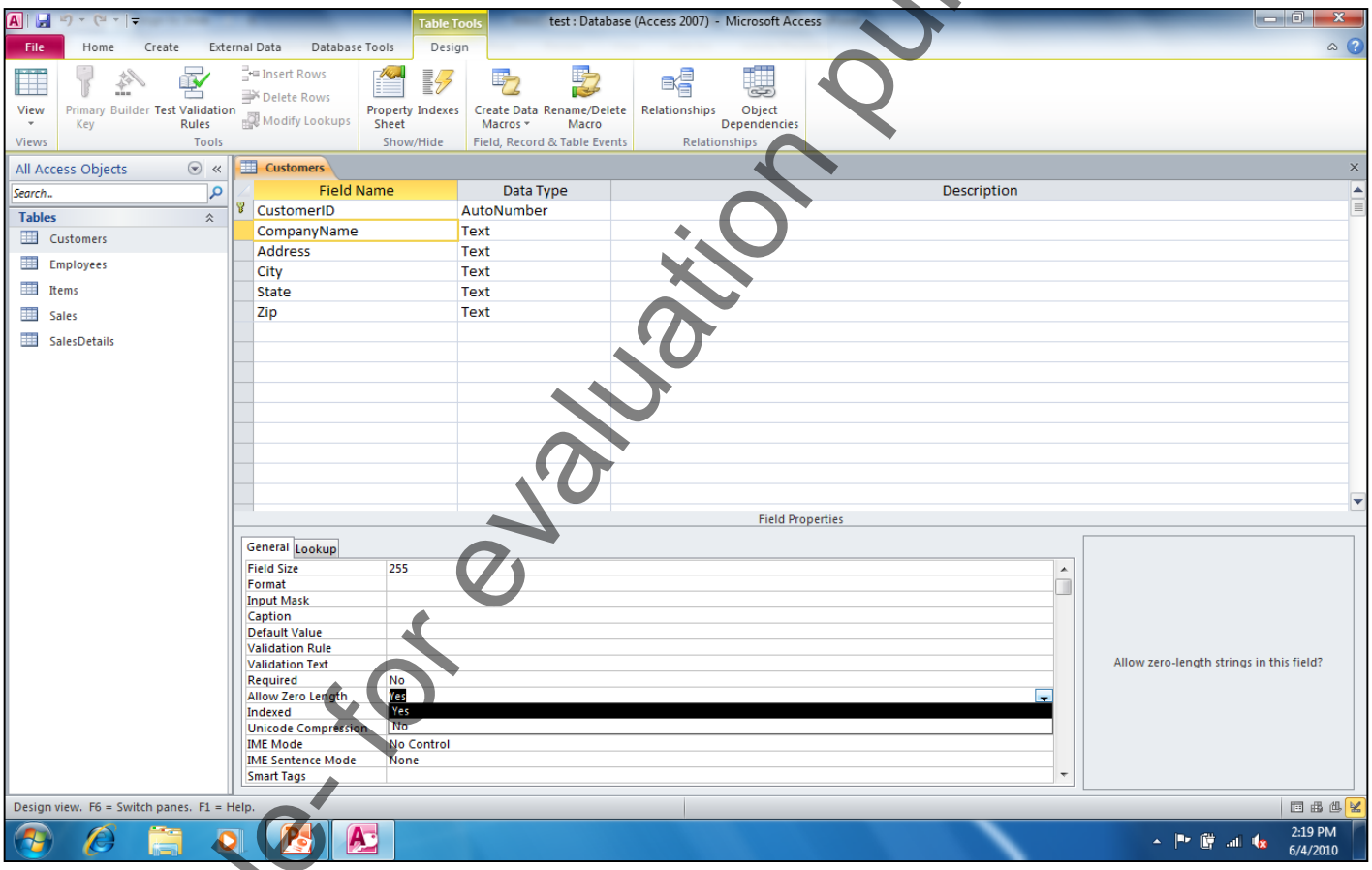
To then set what error message the user will see when they enter a value that violates the validation rule, click into the "Validation Text" property and type the text that you wish to display into that box.

4.8- Requiring Field Input:

You can also set the “Required” property for a selected field to either “Yes” or “No” to either require entry into the field, or not. This is handy for those fields where you want people to enter something into the field instead of skipping the data entry.

4.9- Allowing Zero Length Entries:

You can set the “Allow Zero Length” property for a selected field to either “Yes” or “No” to either require the data entry in the field to be of a length greater than zero (basically, no “Spacebar” values), or not. This is handy for those times when you want people to have to enter a value into the field, but you also don’t want the value entered to be the “empty” set, or invisible characters.



ACTIONS-
Field Properties

SETTING FIELD PROPERTIES IN A TABLE:

1. Select the table to open in "Design View" from the Navigation Pane.
2. Press "Ctrl" + "Enter" on your keyboard to open the table in design view.
3. Select the field in the design grid whose properties you want to set.
4. In the "Field Properties" section at the bottom of the window, select the "General" tab.
5. Find the property that you want to set, and modify the property's value as needed.
6. When you are done, close the table and save your changes.

SETTING FIELD SIZE:

1. Select the table to open in "Design View" from the Navigation Pane.
2. Press "Ctrl" + "Enter" on your keyboard to open the table in design view.
3. Select the field in the design grid whose properties you want to set.
4. In the "Field Properties" section at the bottom of the window, select the "General" tab.
5. Select the "Field Size" property.
6. For text fields, type a number that represents the maximum amount of characters allowed in the selected field. For number fields, select the type of number to use in the field.
7. When you are done, close the table and save your changes.

SETTING THE "FORMAT" PROPERTY FOR A DATE/TIME FIELD:

1. Select the table to open in "Design View" from the Navigation Pane.
2. Press "Ctrl" + "Enter" on your keyboard to open the table in design view.
3. Select the field in the design grid whose properties you want to set.
4. In the "Field Properties" section at the bottom of the window, select the "General" tab.
5. Click into the "Format" field property box and click the drop-down arrow that appears.
6. Select a date/time formatting for the field from the list of available formats.
7. When you are done, close the table and save your changes.

SETTING THE "FORMAT" PROPERTY FOR LOGICAL FIELDS:

1. Select the table to open in "Design View" from the Navigation Pane.
2. Press "Ctrl" + "Enter" on your keyboard to open the table in design view.
3. Select the field in the design grid whose properties you want to set.
4. In the "Field Properties" section at the bottom of the window, select the "General" tab.
5. Click into the "Format" field property box and click the drop-down arrow that appears.
6. Select one of the logical formatting options for the field from the list of available formats.
7. When you are done, close the table and save your changes.

ACTIONS-
Field Properties

SETTING DEFAULT VALUES FOR A FIELD:

1. Select the table to open in “Design View” from the Navigation Pane.
2. Press “Ctrl” + “Enter” on your keyboard to open the table in design view.
3. Select the field in the design grid whose properties you want to set.
4. In the “Field Properties” section at the bottom of the window, select the “General” tab.
5. Click into the “Default Value” field property and enter a default value for the selected field.
6. When you are done, close the table and save your changes.

SETTING AN INPUT MASK FOR A FIELD:

1. Select the table to open in “Design View” from the Navigation Pane.
2. Press “Ctrl” + “Enter” on your keyboard to open the table in design view.
3. Select the field in the design grid whose properties you want to set.
4. In the “Field Properties” section at the bottom of the window, select the “General” tab.
5. Click into the “Input Mask” field property.
6. Click the “Expression Builder” button […] at the far right end of the “Input Mask” text box.
7. Select the desired input mask from the list of masks that are displayed in the “Input Mask” dialog box, and then click “OK.”
8. When you are done, close the table and save your changes.

SETTING VALIDATION RULES AND VALIDATION TEXT RESPONSES FOR A FIELD:

1. Select the table to open in “Design View” from the Navigation Pane.
2. Press “Ctrl” + “Enter” on your keyboard to open the table in design view.
3. Select the field in the design grid whose properties you want to set.
4. In the “Field Properties” section at the bottom of the window, select the “General” tab.
5. Click into the “Validation Rule” field property.
6. You can enter your validation rule into the text box directly or you can click the “Expression Builder” button […] at the far right end of the “Validation Rule” text box and then create the validation rule in the expression builder dialog box. If you do this, click “OK” when you have finished entering the rule.
7. Click into the “Validation Text” field property.
8. Type the text response that you want Access to display when someone enters a value into the selected field that violates the validation rule that you created.
9. When you are done, close the table and save your changes.

ACTIONS-
FIELD PROPERTIES

REQUIRING FIELD INPUT:

1. Select the table to open in “Design View” from the Navigation Pane.
2. Press “Ctrl” + “Enter” on your keyboard to open the table in design view.
3. Select the field in the design grid whose properties you want to set.
4. In the “Field Properties” section at the bottom of the window, select the “General” tab.
5. Select the “Required” field property.
6. Select “Yes” from the drop-down menu of choices.
7. When you are done, close the table and save your changes.

ALLOWING ZERO LENGTH FIELD ENTRIES:

1. Select the table to open in “Design View” from the Navigation Pane.
2. Press “Ctrl” + “Enter” on your keyboard to open the table in design view.
3. Select the field in the design grid whose properties you want to set.
4. In the “Field Properties” section at the bottom of the window, select the “General” tab.
5. Select the “Required” field property, and click the drop-down arrow that appears at the right end.
6. Select “No” from the drop-down menu of choices.
7. Select the “Allow Zero Length” field property, and click the drop-down arrow that appears at the right end.
8. Select “Yes” from the drop-down.
9. When you are done, close the table and save your changes.

EXERCISES-
FIELD PROPERTIES

Purpose:

1. To be able to set the properties of the database table fields.

Exercises:

1. Open your Access application.
2. Open the "test" database you completed from the Exercise at the end of the previous chapter.
3. Display the database tables in the Navigation Pane, if needed.
4. Select the "Employees" table from the Navigation Pane, and then press "Ctrl" + "Enter" on your keyboard to open it in design view.
5. Click the row selector for the "LastName" field.
6. Click the "General" tab at the bottom of the screen.
7. Click into the "Field Size" property and set the value to "75."
8. Click "Save" in the Quick Access toolbar. Click "Yes" in the pop-up message box that appears.
9. Close the table.
10. Select the "Sales" table from the Navigation Pane, and then press "Ctrl" + "Enter" on your keyboard to open it in design view.
11. Click the row selector for the "Saledate" field.
12. Click the "General" tab.
13. Click into the "Format" property, and use the drop-down at the right end of that field to select "Short Date."
14. Click "Save" in the Quick Access toolbar.
15. Close the table.
16. Select the "Customers" table from the Navigation Pane, and then press "Ctrl" + "Enter" on your keyboard to open it in design view.
17. Click the row selector for the "State" field.
18. Click the "General" tab at the bottom of the screen.
19. Click into the "Default Value" property and type "MI" in the field.
20. Select the row selector for the "Zip" field in the field grid at the top of the Design View.
21. Click into the "Input Mask" property, and then click the ellipses button (…) at the right end of the field.
22. Click "Yes" if Access tells you that it must save the table first.
23. Select "Zip Code" from the list of choices, and click "Next>"
24. Click the drop-down for "Placeholder character:," and scroll to the top of the list to select the " " character.
25. Click "Next> ."
26. Select "Without the symbols in the mask, like this;" and click "Next>" to continue.
27. Click "Finish."
28. Click "Save" in the Quick Access toolbar.
29. Close the table.

CHAPTER 5-
Joining Tables

5.1- The Relationships Window

5.2- Enforcing Referential Integrity

5.3- Creating Lookup Fields

5.1- The Relationships Window:

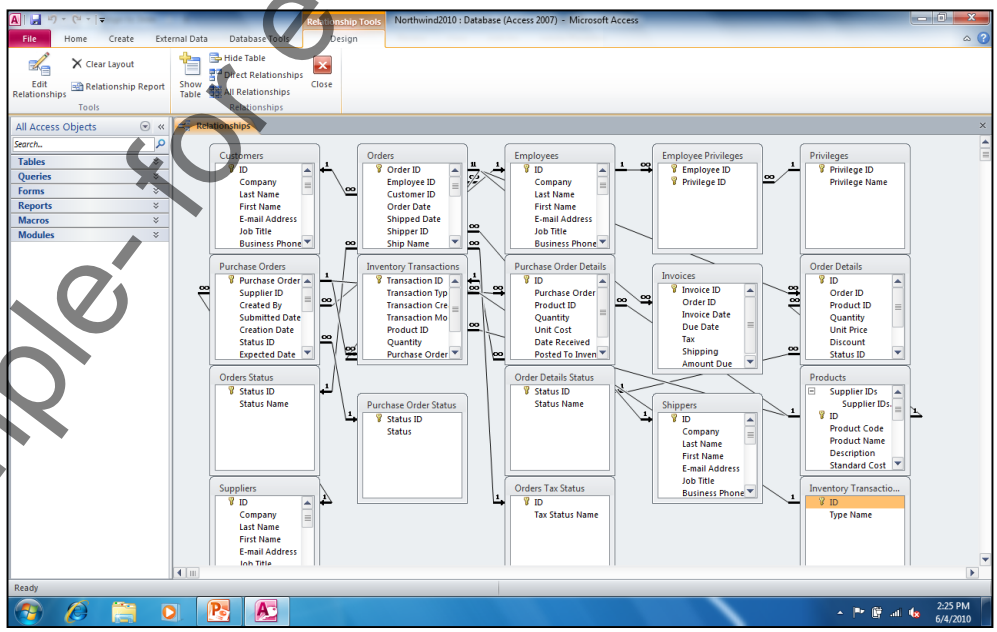
As you create tables in Access, you will want to be able to relate the tables so that you will be able to access information from them through their "shared" or "common" fields by which they are joined. In Access, you create relationships between tables in the "Relationships" window. You can access this window by clicking the "Relationships" button in the "Relationships" group ("Show/Hide" group in 2007) on the "Database Tools" tab in the Ribbon.

In the "Relationships" window, you add the tables from your database that you want to relate to each other. Remember that to join two tables together, they **must** have shared fields. To assist you in adding tables, the "Show Table" dialog box will appear. You can select the names of the tables to add and then click the "Add" button to add the tables to the Relationships window. When you are done, close the "Show Table" dialog box. You can access it again in the future by clicking the "Show Table" button in the "Relationships" group on the "Design" tab in the "Relationship Tools" contextual tab. Once you have added the necessary tables to the relationships window, you can then create joins between the common fields, as needed.

To relate two common table fields, simply click and drag the shared field from one table, and drop it on top of the common field in the related table to establish a join between them based on the values in the common field. You can actually click and drag from either one of the fields onto the other. Make sure, however, that you do pay attention to which two fields you are joining. You can join any two fields that have the same data type, so make sure you don't accidentally join two fields that you didn't mean to join. If you do this, you can delete the join between the table fields in this window, as well.

When you create a join, Access will then assign it one of the relationship types: A "one-to-many" relationship, in which one record from the "parent" table will have multiple related records in the "child" table; the "one-to-one" relationship, in which every record in the "parent" table has a single related entry in the "child" table; or the "Indeterminate" join, which usually is indicative of an error and is very rarely used. Access will determine the relationship type automatically when you create the join.

In the relationship window, primary key fields will appear with a "key" icon next to their names. Joined fields will also share a little black join line or black join arrow between the common fields in the two tables. This indicates the join type that is shared by the related fields.



5.2- Enforcing Referential Integrity:

As you create the appropriate relationships between the tables in your database, you will need to set the properties of the table joins to ensure that they are set up as you would like. The main join property that you will need to set is the "Referential Integrity" of the join.

Referential integrity refers to the built-in set of rules that Access uses to ensure that the relationship between the data within the joined fields of the tables is valid. It ensures that every unique value within the related field in the "child" table (the "many" table in a "1-to-many" join) has a corresponding value to which it is related in the "parent" table (the "1" table in a "1-to-many" join). This ensures that when you delete a record in the "parent" table (the "1" in a "1-to-many" join) that all of the related records in the "child" table (the "many" side of the "1-to-many" join) aren't "orphaned," or without a reference to a record in the "parent" table. This also prevents you from having meaningless data within your database tables, like a "sale" record without a "customer" reference.

In order to set up referential integrity, the relationship between the fields within the joined tables must fall within these rules:

1. The related field in the "parent" table is the primary key within the table.
2. The related fields in both tables must share the same (or compatible) data types.
3. Both tables must belong to the same database.

As long as you've got a relationship that falls in line with these rules, you can enforce referential integrity upon it. To do this, just check the checkbox for "Enforce Referential Integrity" in the "Edit Relationships" dialog box that appears when you create a table relationship. Below that, there are two options that can be set once you have selected to apply referential integrity: "Cascade Update Related Fields" and "Cascade Delete Related Records." Checking the "Cascade Update Related Records" checkbox will ensure that if a data value is changed in the joined field within the "parent" table, that the change will be cascaded to the related data in the joined field of the "child" table, updating them as well. Checking the "Cascade Delete Related Records" will ensure that if you delete a record in the "parent" table, all of the related records in the "child" table will be deleted as well.

Note that there is a "Join Type..." button in the "Edit Relationships" dialog box. If you click it, you will be presented with the three types of possible ways that the data between the tables could be related in the "Join Properties" dialog box. Often you will not need to change these settings, as you normally want join type "1" in a relational database. However, if you do change the join type settings in this window as part of the join between the tables, note that this permanently alters the way that data between the two tables is related. You can select either option 1, 2, or 3 in the "Join Properties" dialog box, and then click the "OK" button to return to the "Edit Relationships" dialog box. Note that when you choose a join type other than the default of "1," the join lines shown in the "Relationships" window change to join arrows. These arrows reflect the direction of the join between the two tables, per your selection in the "Join Properties" dialog box.

When you have set any of the options you wish to apply to the relationship in the "Edit Relationships" dialog box, click either the "Create" or "OK" buttons to set the join relationship. To edit the join in the future to change any settings, you must double-click the join line itself in the "Relationships" window to view its properties in the "Edit Relationship" dialog box again.

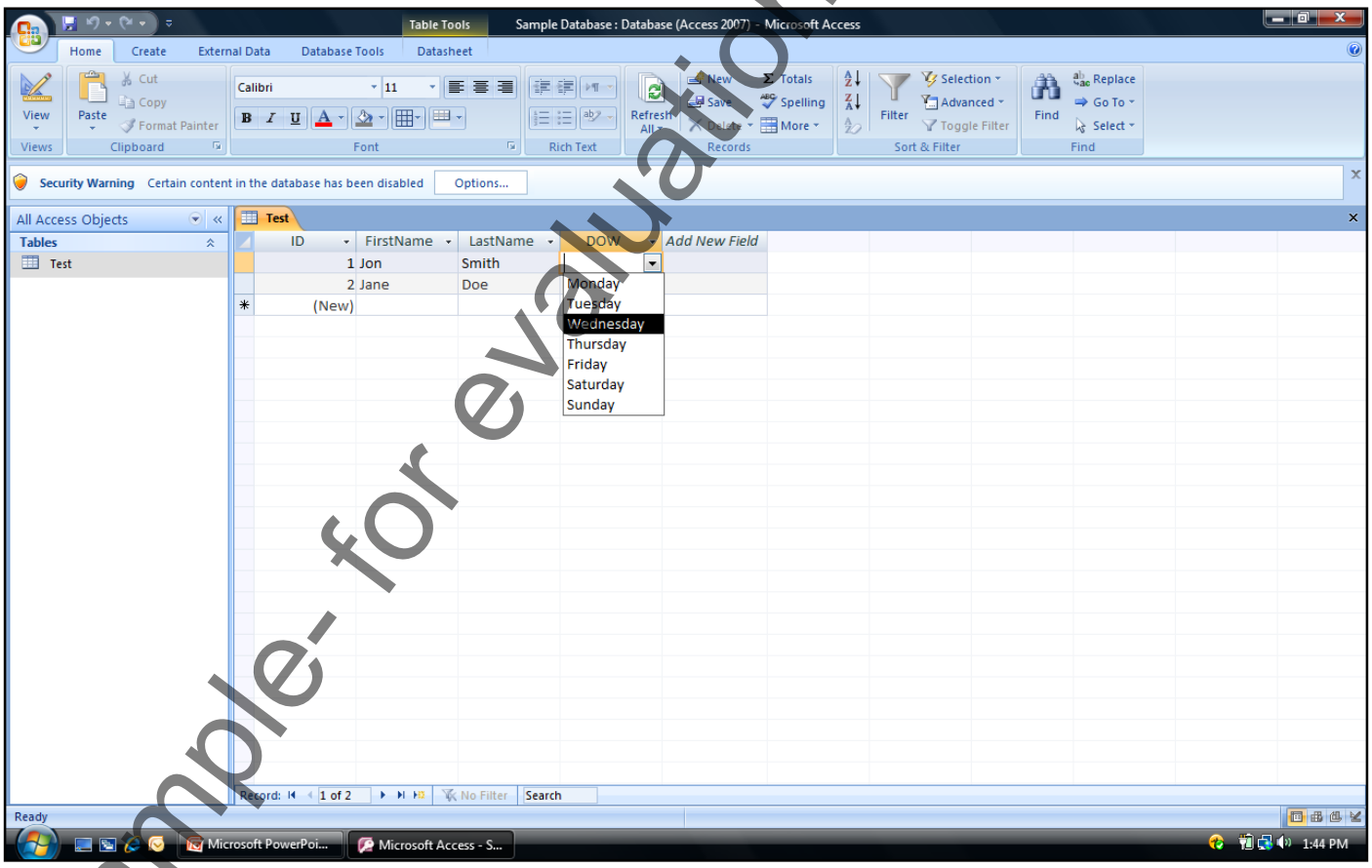
To delete a table join, right-click on the table join that you want to delete and then select the "Delete" command from the pop-up menu that appears.

5.3- Creating Lookup Fields:

Access can also create “lookup” fields within a table that can lookup the values in another table, query, or hand-typed list from which it will draw its values. If the field is looking up data from another table (versus a query or list), it will automatically create an additional join between the two tables which you will see in the Relationships window. Don’t panic if you see these appearing in the relationships window. These types of joins are simply needed for the purpose of the lookup field.

You create the lookup fields in the table design view when you are creating your tables. You can pick the “Lookup Wizard…” field type from the “Data Type” column drop-down while using table design view. This then launches the “Lookup Wizard” to help you create the lookup field. The wizard will ask you a series of questions regarding the lookup field, which you answer in each screen. You then click the “Next” button to continue from screen to screen. When you are done answering questions, click the “Finish” button inside the “Lookup Wizard” to create the lookup field. In the table design view, the “data type” of the lookup field will show the type of data that it is looking up. For example, if the values being looked up by the lookup field are numbers, then the “Data Type:” column of the lookup field changes to “Number.”

To then use the lookup field, you can switch to the table’s datasheet view. When you click into the lookup field, you will be presented with a drop-down menu of choices which it looks up from the table, query, or list from which you can select.



ACTIONS-
Joining Tables

RELATING TABLES (CREATING JOINS) IN THE RELATIONSHIPS WINDOW:

1. Open the database that contains the tables between which you wish to create a relationship (join).
2. Click the "Relationships" button in the "Relationships" group ("Show/Hide" group in 2007) on the "Database Tools" tab in the Ribbon to launch the "Relationships" window.
3. If you have no tables in the "Relationships" window yet, the "Show Table" dialog box will appear automatically. If you have previously added tables to this window, then you may need to click the "Show Table" button in the "Relationships" group on the "Design" tab in the "Relationship Tools" contextual tab to show the "Show Table" dialog box again.
4. Click the "Tables" tab in the "Show Table" dialog box.
5. Select the first table to add to the "Relationships" window from the "Tables" tab, and then click the "Add" button at the right of the "Show Table" dialog box to add the selected table to the "Relationships" window.
6. Repeat step 5 until all of the tables between which you want to establish joins are shown in the "Relationships" window.
7. Click the "Close" button in the "Show Tables" dialog box to close it.
8. Click and drag the field which you wish to join to another field from one table, and drop it on top of the field to which you wish to join it in the related table.
9. In the "Edit Relationships" dialog box, check the check box for "Enforce Referential Integrity," if needed.
10. Check the "Cascade Update Related Fields" and/or the "Cascade Delete Related Records" checkboxes, if you selected to enforce referential integrity and also wish to enforce cascading updates or deletion of records between the tables.
11. If desired, you can click the "Join Type…" button to open the "Join Properties" window. Here you can select the type of join to enforce between the two tables. When finished, you can click the "OK" button to apply the selected type of join and return to the "Edit Relationships" window.
12. Click either the "Create" or "OK" button to create the join between the two tables.
13. Repeat steps 8 through 12 until you have set up all of the necessary joins between your tables.
14. Click "Save" in the Quick Access toolbar to save your changes.
15. Click the "Close" button in the "Relationships" group on the "Design" tab of the "Relationship Tools" contextual tab to close the "Relationships" window when you are finished.

EDITING A JOIN BETWEEN TABLES IN THE RELATIONSHIPS WINDOW:

1. Open the database that contains the table join whose properties you wish to edit.
2. Click the "Relationships" button in the "Relationships" group ("Show/Hide" group in 2007) on the "Database Tools" tab in the Ribbon to launch the "Relationships" window.
3. Double-click the join that you want to edit in the Relationships window.
4. In the "Edit Relationships" dialog box, make whatever editing changes to the properties of the relationship that are needed.
5. Click "OK" to close the "Edit Relationships" dialog box.
6. Click "Save" in the Quick Access toolbar to save any changes that you made.
7. Click the "Close" button in the "Relationships" group on the "Design" tab of the "Relationship Tools" contextual tab to close the "Relationships" window when you are finished.

ACTIONS-
Joining Tables

DELETING A JOIN BETWEEN TABLES IN THE RELATIONSHIPS WINDOW:

1. Open the database that contains the table join that you wish to delete.
2. Click the "Relationships" button in the "Relationships" group ("Show/Hide" group in 2007) on the "Database Tools" tab in the Ribbon to launch the "Relationships" window.
3. Click the join line in the Relationships window that you want to delete to select it. The line should appear slightly thicker than when it is not selected.
4. Press the "Delete" key on your keyboard.
5. Click "Yes" to permanently delete the relationship from the table.
6. Click "Save" in the Quick Access toolbar to save any changes that you made.
7. Click the "Close" button in the "Relationships" group on the "Design" tab of the "Relationship Tools" contextual tab to close the "Relationships" window when you are finished.

CREATING A LOOKUP FIELD IN A TABLE:

1. Open the table in which you want to create a lookup field in the table design view.
2. Click into the "Data Type" column of the field which you want to make a lookup field, and select the "Lookup Wizard…" choice from the drop-down list that appears. This will launch the "Lookup Wizard."
3. In the "Lookup Wizard," select the radio button that corresponds to how you would like the column to get its values: from a table or query, or from a list that you type by hand. Then click the "Next>" button.
4. If you selected to **type your own values** in the last step: you will need to select the number of columns that you want the lookup field to display, and then enter the information to display into each column below that. When you are done, click the "Next>" button to move to the final screen where you can set the name of your field, if needed, and then click the "Finish" button to create the lookup field.
5. If you selected to **draw the lookup field's information from a table or query**: you will need to select the table or query from which you want to lookup values to use, and then click "Next>" to continue.
6. You will then need to select the fields from the selected table or query which you want to use as the choices from the lookup column's values. Click the fields that you want to use from the window on the left, and then click the ">" button to move them to the right window. Click "Next>" when you are done.
7. Here you can adjust the width of the columns that will be displayed, if you want to. Click "Next>" when you are done.
8. Type a name for your field, if needed, and then click the "Finish" button to insert the lookup field into your table.
9. Click "Save" in the Quick Access toolbar to save any changes that you made.

EXERCISES-
JOINING TABLES

Purpose:

1. To be able to join your tables in a relational database structure.

Exercises:

1. Open your Access application.
2. Open the "test" database you completed from the Exercise at the end of the previous chapter.
3. Click the "Relationships" button in the "Relationships" group ("Show/Hide" group in 2007) on the "Database Tools" tab in the Ribbon to launch the "Relationships" window.
4. In the "Show Table" dialog box that appears, click and drag over all of the tables in the list of tables that appears to select them.
5. Click "Add" in the "Show Table" dialog box to add all of the selected tables into the Relationships window.
6. Click "Close" in the "Show Table" dialog box.
7. Click and drag the "EmployeeID" field from the "Employees" table, and drop it on top of the "EmployeeID" field in the "Sales" table (or vice-versa).
8. Click the "Enforce Referential Integrity" checkbox in the "Edit Relationships" window that appears.
9. Click "Create."
10. Click and drag the "CustomerID" field from the "Customers" table, and drop it on top of the "CustomerID" field in the "Sales" table (or vice-versa).
11. Click the "Enforce Referential Integrity" checkbox in the "Edit Relationships" window that appears.
12. Click "Create."
13. Click and drag the "ItemID" field from the "Items" table, and drop it on top of the "ItemID" field in the "SalesDetails" table (or vice-versa).
14. Click the "Enforce Referential Integrity" checkbox in the "Edit Relationships" window that appears.
15. Click "Create."
16. Click and drag the "SaleID" field from the "Sales" table, and drop it on top of the "SaleID" field in the "SalesDetails" table (or vice-versa).
17. Click the "Enforce Referential Integrity" checkbox in the "Edit Relationships" window that appears.
18. Click the "Cascade Delete Related Records" checkbox.
19. Click "Create."
20. Click "Save" in the Quick Access toolbar.
21. Click the "Close" button in the "Relationships" group on the "Design" tab of the "Relationship Tools" contextual tab to close the "Relationships" window when you are finished.
22. Close the database.

CHAPTER 6-
INDEXING TABLES

6.1- INDEXES

6.2- CREATING INDEXES

6.3- DELETING INDEXES

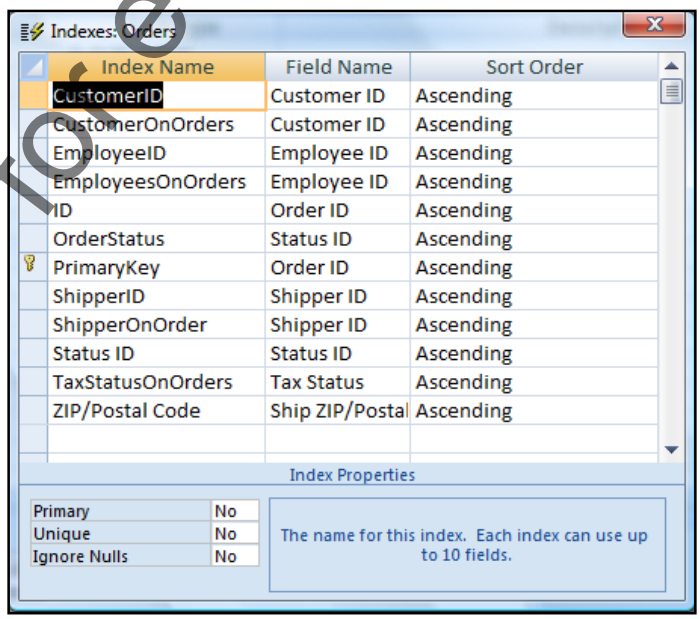
6.1- Indexes:

When you create an index for a table, you define a way that the data in the table may be sorted, using the fields that are available. Indexing a table is simply a way of organizing the data in the table to allow Access to complete query searches and sorting more rapidly. Indexing can help speed up the time that it takes to complete queries in Access, given a few criteria are met first. First of all, you should only index tables that have a variety of different data types within their fields. Second, indexing is more efficient if the data in your indexed fields give each record a more unique identification (like a primary key field). Indexing is not usually necessary on fields that have multiple repeating values. Third, you really only need to index fields which are used for criteria in queries. For example, if you are creating many queries that find records based on phone numbers, you may want to create an index on the field which contains the phone numbers. Assuming your table fields have met these criteria, it can be useful to apply an index to the desired fields to increase the sorting and processing capabilities of data in queries.

Unfortunately, if you apply an index to a table that contains multiple similar data types, or has multiple duplicate information in each indexed field, you may actually slow down the speed at which Access updates table information. In these cases, it is actually preferable to **not** index these fields, as indexing these types of fields will rob you of more time than the time saved by using the indexed fields in the query.

Also, Access decides when and if it will choose to use the indexes that you create. All that you can do as the user is simply create the indexes for the table fields. Access will decide when and if to use the indexes when performing a query. Many times, when a query is performed on a table which has very little information, Access will perform a “table scan” on the table records, looking at all of the data in all of the fields and then extracting the requested results. However, as you add more and more data to the tables, Access may find that it is easier to index (sort) the table by one of the available indexes first, and then extract the requested records. However, it cannot do this if there are no indexes for it to take advantage of.

To view the indexes that have been created for a table, simply open up the table in design view. Next, click the “Indexes” button in the “Show/Hide” group on the “Design” tab of the “Table Tools” contextual tab to view the indexes for your selected table in a separate “Indexes” dialog box. Almost every table will have at least one index: the primary key. The primary key is a type of index. Notice that when you open a table in datasheet view, it sorts the view by the values in the primary key column, by default.



6.2- Creating Indexes:

When you are creating indexes, you want to try and use field values that will identify each record in your table as uniquely as possible. If you are a good database designer, there will already be a single field in your table that already does this: your primary key field. However, you can create additional indexes on other fields to use in queries for faster query processing.

To create an index, open up the table which you would like to index in table design view. Then click the "Indexes" button in the "Show/Hide" group on the "Design" tab of the "Table Tools" contextual tab to show the "Indexes" dialog box. Click into the next available row under the "Index Name" column, and type a name for your new index. To the right, click into the "Field Name" column, and select the name of the field within the table which you wish to index. To the right of that, select whether that field should be sorted in "Ascending" (A-Z, 1-9) or "Descending" (Z-A, 9-1) order.

In the "Index Properties" section at the bottom of the dialog box, you have three drop-down text boxes into which you can set the properties of the index. The first property is "Primary," and can accept either a "Yes" or "No" value. Whatever index is the primary key of the table will show "Yes" in this property, and all others will show "No." There can only be one primary key field in a table.

The next property, "Unique," asks if the values within the field will always be unique (like the values in a primary key field are). Once again, you can select either "Yes" or "No," as appropriate.

You can then also set the "Ignore Nulls" property to "Yes" or "No" to either include or exclude "Null" (empty) values from the sorting. Nulls occur when there has been no data entry in the field for a record. For example, if you skipped entering an address into a customer record, the address field would contain a "null" value. It is not equivalent to zero, as zero is still a value. "Null" is simply "unknown."

When you have finished creating the index, you can close the "Indexes" dialog box. Then click the "Save" button in the Quick Access toolbar to save your structural modifications to the table.

6.3- Deleting Indexes:

If you have indexes in a table that you wish to delete, you can easily do so. Open up the table that contains the indexes that you would like to delete in table design view. Next, click the "Indexes" button in the "Show/Hide" group on the "Design" tab of the "Table Tools" contextual tab to view the "Indexes" dialog box.

In this box, click the row selector at the left end of the index or indexes that you would like to delete. Then just press the "Delete" button on your keyboard to delete those indexes. You can then close the "Indexes" dialog box.

Remember to click the "Save" button in the Quick Access toolbar to save the structural change to your table. If you forget and close the table, Access will prompt you to save your changes at that time. You can then just click "Yes" to the prompt to save the changes at that point in time, as well.

ACTIONS-
INDEXING TABLES

VIEWING TABLE INDEXES:

1. Open the table within which you want to view the table indexes in table design view.
2. Click the “Indexes” button in the “Show/Hide” group on the “Design” tab of the “Table Tools” contextual tab to open the “Indexes” dialog box.

CREATING AN INDEX:

1. Open the table within which you want to create an index in table design view.
2. Click the “Indexes” button in the “Show/Hide” group on the “Design” tab of the “Table Tools” contextual tab to open the “Indexes” dialog box.
3. Click into the next available row under the “Index Name” column, and type a name for your new index.
4. Click into the “Field Name” column, and select the name of the field within the table to index.
5. Click into the “Sort Order” column and select whether that field should be sorted in “Ascending” (A-Z, 1-9) or “Descending” (Z-A, 9-1) order.
6. In the “Index Properties” section, click into the “Primary” text box and select either “Yes” or “No.”
7. Click into the “Unique” text box and select either “Yes” or “No,” as appropriate.
8. Click into the “Ignore Nulls” text box and select either “Yes” or “No,” as appropriate.
9. Click the “X” in the upper-right corner of the “Indexes” dialog box to close it.
10. Click the “Save” button in the Quick Access toolbar to save your structural modifications to the table.

DELETING AN INDEX:

1. Open the table within which you want to delete an index in table design view.
2. Click the “Indexes” button in the “Show/Hide” group on the “Design” tab of the “Table Tools” contextual tab to open the “Indexes” dialog box.
3. Click the row selector button to the left of the index that you want to delete.
4. Press “Delete” on your keyboard.
5. Close the “Indexes” dialog box, and save the structural modifications to the table.

EXERCISES-
INDEXING TABLES

Purpose:

1. To be able to create indexes for your tables.

Exercises:

1. Open your Access application.
2. Open the "test" database you completed from the Exercise at the end of the previous chapter.
3. Display the database tables in the Navigation Pane, if needed.
4. Select the "Employees" table, and press "Ctrl" + "Enter" on your keyboard to open it in design view.
5. Click the "Indexes" button in the "Show/Hide" group on the "Design" tab of the "Table Tools" contextual tab to open the "Indexes" dialog box.
6. Click into the next available blank row, and type "LastName" as the "Index Name."
7. Click into the "FieldName" and select "LastName" from the drop-down.
8. Select "Ascending" from the "Sort Order" drop-down, if needed.
9. Ensure that the "Primary," Unique," and "Ignore Nulls" text boxes are all set to "No."
10. Close the "Indexes" dialog box.
11. Click "Save" in the Quick Access toolbar.
12. Close the table.
13. Close the database.

CHAPTER 7- QUERIES

7.1- Using the Simple Query Wizard:

You use a query to answer a question that you have about the information stored in the database tables. You can then further analyze the results that the queries pull to produce even more information than the query itself displays. Reports are often based on query results, upon which they then can perform additional mathematical and statistical calculations. Queries are also an excellent way to show information from related tables in a single result set, as the results that you pull from queries aren't limited to a single table. The power and flexibility of query design is the entire reason you use database programs. They can quickly access, calculate, and summarize the records that they pull from the various tables.

Access provides you with a simple query wizard which you can use to initially create queries. However, do not rely too heavily on this tool, as at some point you will inevitably have to create a query that is more complex than this tool allows. For advanced queries, you must learn how to create a query in query design view, versus using the wizard to create them.

However, to create a simple query using the wizard, click the "Query Wizard" button in the "Queries" group ("Other" group in 2007) on the "Create" tab in the Ribbon. In the "New Query" dialog box that appears, you can see the ways in which you can create queries. Select the "Simple Query Wizard" choice, and then click "OK" to begin.

In the first screen of the wizard, you must select the first table from which you will pull data by using the "Tables/Queries" drop-down. Once you have selected a table, the fields from that table will display in the "Available fields:" list. To add a field from the table into the query, select its name from the "Available fields:" list and click the ">" button to move it into the "Selected fields:" list. Repeat this as needed, selecting the fields that you will want to see in the query. When you are finished, click "Next >" to continue.

If you only selected fields from a single table in the first screen, then when you click the "Next >" button to continue, you will only need to provide the query with a name and then click the "Finish" button to finish creating the query. If, however, you picked data fields from two or more related tables, then when you click the "Next >" button, you will instead view a second screen which asks if you would like a "Detail" or "Summary" query. You can select the option button for the type of query that you wish to create. If you select "Summary," then you will be able to click the "Summary Options…" button to open the "Summary Options" dialog box. In this dialog box, you can select what type of summary to perform over a selected field. Make your selections, and then click the "OK" button to return to the "Simple Query Wizard."

Click "Next >" to continue. In the next screen, if you selected "Detail" on the previous screen, you will only need to name your query and then click "Finish" to create the query. If you selected "Summary" and your summary includes dates, you may be presented with additional date grouping options. Continue to answer any questions, as appropriate for your query and click "Next >" to continue until you reach the final screen where you must type a name for your query, and then click "Finish" to create the query.

7.2- Designing Queries:

To make a query in design view, click the "Query Design" button in the "Queries" group ("Other" group in 2007) on the "Create" tab in the Ribbon to create a new query in the query design view. The first thing you will see is the "Show Table" dialog box appear over the query design view. Just as with the "Relationships" window which you used earlier, here you will have to add the table or tables that you need for the query into the query design view. You simply select the names of the tables that you wish to add, and then click the "Add" button in the "Show Table" dialog box to add the necessary tables into the query.

The query design view gives you power and flexibility in designing queries. Although it isn't the only way to make them initially, you will have to learn how to use query design view at some point as you grow in your Access skill set. In query design view, the tables from which you extract data are placed into the top

7.2- Designing Queries (cont'd.):

section of the design view. You then add the fields from these tables that you want to view in your query results into the bottom grid section, which is called the QBE, or "Query By Example," grid.

If you want to add all of the fields of a query table into your query's result set, you can click and drag the first field in the table, which shows an asterisk, down into the QBE grid. That will then show all of the fields in that table in the result set of the query. Once the fields are in place, you also add any "Criteria" and "Sorting" options, as needed, to the QBE grid to filter and sort just the data that you wish to see.

Ensure that you only select the tables that you absolutely need in order to run the query. Adding additional tables which you will not use forces the query to access these tables whenever it is run, slowing it down pointlessly. It can also produce unexpected and sometimes erroneous results. As you add the necessary tables to the query, the joins which you created between the tables will also be displayed at the top of the query.

Make sure that you have added all of the necessary tables for your query. For example, assume that you have two tables from which you wish to extract data: the "Customers" table and the "Employees" table. However, also assume that those two tables do not share a direct join between them. In order for the query results to make any sense whatsoever, you would also have to add the table that is used to associate those two tables, as well. Assume that the "Employees" table is related to the "Customers" table through the "Sales" table. In this case, you would also have to add the "Sales" table to the query, even if you had no intention of displaying any data from that particular table. It is needed in order to relate the two tables from which you do want to extract data. If you add two tables that are not joined to each other in any way, the query result will often produce a Cartesian product, where every value in every row of one table is multiplied by the value in every row of the second table. You will usually notice when this happens, as you will probably have several hundred, if not thousand, more records in your query result set than you do data records in either table.

Once you have added the necessary tables to the query, click the "Close" button in the "Show Table" dialog box to close it and display the query design view beneath it. You should see the tables that you added shown as small table diagrams at the top of the query design view. If you forgot a table and need to add it into the query, you can click the "Show Table" button in the "Query Setup" group on the "Design" tab of the "Query Tools" contextual tab within the Ribbon to bring up the "Show Table" dialog box again. If you accidentally added a table which you do not need, you may right-click on the table diagram of the table that you do not want in the query design view, and then select the "Remove Table" choice from the pop-up menu that appears to remove the table from the query.

Next, you will need to add the fields that you want to show in the query result set from the tables into the grid at the bottom of the query design view. To do this, you can click and drag the name of the field that you want to display from the tables, and drop them into the columns at the bottom of the design grid. You can also double-click on the name of a field shown in the tables to add it to the design grid, as well. There are actually quite a few ways that you can add the fields from the tables into the grid area below. Note that the order in which the fields are listed in the grid is the order in which those fields will be displayed in the query result set.

Before you can remove a field which you accidentally added to the grid, or reorganize the order of the fields in the grid, you must first select the column to delete or move in the result set. To do this, place your mouse pointer slightly above the column in the grid area that you want to select, until you see a downward-pointing black arrow. Then click once to select the field. To delete it at that point, you may simply press "Delete" on your keyboard. To move it, place your white mouse arrow into the very top of the selected column, and then click and drag the selected column to the left or right. As you drag it, you will see a thick, black line appear between the columns over which you drag your mouse. This line represents where the

7.2- Designing Queries (cont'd.):

column will be inserted when you release your mouse.

Most often, after you have added the fields that you want to view into the query grid, you then add sorting and filtering criteria to the query. However, if you do not wish to restrict the data that is displayed, then you can simply run the query at this point. To run a query and view the result set, you can click the "Run" button in the "Results" group on the "Design" tab in the "Query Tools" contextual tab within the Ribbon to view the query's result set. The result set looks like a table does when viewed in datasheet view. However, a query result set is not, by default, a "base" table in the same way that your other database tables are. The "table" that is produced when you run a query disappears as soon as you close the query. A query is really a definition of what data should be retrieved and displayed from the tables. Therefore, a query always shows the most "up-to-date" data every time that you run it.

You can switch the query back to the query design view after you have run the query by clicking the "View" drop-down button in the "View" group on the "Home" tab in the Ribbon. If you click the "View" drop-down arrow, then just select the "Design View" choice from the drop-down menu. Either way, once you are ready to save your query, click the "Save" button in the Quick Access toolbar. You can then type a name for your query into the dialog box which appears, and click "OK" to save the query. You can then close the query without losing all of your query design work.

7.3- Joining Tables in a Query:

When you add multiple tables to a query in the query design view, the joins that you have established between tables within the "Relationships" window appear in the query, allowing you to access information from any related tables. However, if you are attempting to create a query using unrelated tables, you can create a temporary join between tables in the upper section of the query design view. For example, you could need to do this if you were trying to query two linked Excel tables using Access.

You would do this in the same way that you created permanent joins between the tables in the "Relationships" window. You simply click and drag the common field from one table and drop it on top of the related field in another table. You can then edit the relationship between them to assist you in producing the desired results from your query. To do this, double-click on the join line. The "Join Properties" dialog box will appear, showing the names of the linked fields and the tables. Below that, select either option 1, 2, or 3 to determine how you want the result set to be displayed. When you have the desired settings, click "OK." Query joins are particularly important, as the "type" of join that you select can drastically affect the result set.

7.4- Adding Criteria to the QBE Grid:

In Access, when you want to display records from a table based on the values within a selected query field, you need to enter a "criteria" for record selection. For example, say that you wanted to create a query that would show the names of all of your customers located in specific city, like "East Lansing." In the query design view, you would need to place your customer table into the query and then select the customer fields that you want to display (like "FirstName," "LastName," "City") in the design grid below. To then filter the records to show only customers where the "City" field is equal to "East Lansing" you would type "East Lansing" into the "Criteria:" row in the design grid, under the "City" field column. Then when you ran the query, it would display only the customers in the table that are located in East Lansing.

7.5- Running a Query:

When you are in query design view, you can run the query to view the result set by simply clicking the "Run" button in the "Results" group on the "Design" tab of the "Query Tools" contextual tab within the Ribbon. If the results aren't what you expected, you may need to re-design the query structure. From the displayed result set, you can easily return to the query design view by clicking the "View" button in the "Views" group on the "Home" tab in the Ribbon while viewing the displayed result set from the query.

You can also view the result set of a saved query by selecting its name from the list of queries shown in the Navigation Pane, and then pressing the "Enter" key on your keyboard. Alternately, you could simply double-click on the name of the query that you want to run within the listing shown in the Navigation Pane.

7.6- How is Using the QBE Grid Writing SQL Code?:

In Access, when you are visually creating the query in the query design view, what you are *really* doing is visually constructing SQL code. SQL stands for "Structured Query Language," and it is a multi-platform language used to access and retrieve data within many different database programs.

If you want to learn SQL, a good way to start is by viewing the SQL code that your Access queries produce. You can view the SQL code for any query by opening the query in design view, and then clicking the "View" drop-down button in the "Results" group on the "Design" tab of the "Query Tools" contextual tab within the Ribbon. From the drop-down menu of choices that appears, select the "SQL View" command. Access will then display your query as SQL code, so that you can examine how it works.

7.7- Sorting Query Results:

You can sort the results of a query by any field displayed within the QBE grid when the query is viewed in design view. To set the sorting in design view, just select the field in the QBE Grid by which you would like to sort the result set, and click into that field's "Sort:" row. You can use the drop-down to select either "Ascending" or "Descending" order. If you sort by multiple fields in the design view, the order of the sorting is applied by field from left to right when viewing the result set. When sorting in this manner, you are including the sorting criteria in the SQL code of the query itself. This then produces the same sort to the query result set each time you run the query. To remove sorting applied in this manner, select "(not sorted)" from the "Sort" row within the field from which you want to remove the sorting.

Alternately, you can apply sorting to the query result set that does not affect the SQL code of the query. It simply changes the way that the data from the query is displayed in the result set. You can apply more advanced sorting in this view, just as you can with table data when viewed in datasheet view.

When viewing the result set of the query, you can apply sorting to selected fields by clicking the small drop-down arrows at the top of the column and then selecting "Sort A to Z" or "Sort Z to A." You can also select a column and click either the "Ascending" or "Descending" buttons in the "Sort & Filter" group on the "Home" tab in the Ribbon. To remove sorting that you have applied, click the "Remove Sort" button ("Clear All Sorts" button in Access 2007) in the "Sort & Filter" group to remove all sorting from the query result set.

7.8- Hiding Fields in a Result Set:

Sometimes when you are creating queries, you need to add a field to the QBE grid for criteria purposes only, and don't particularly want the field itself to be displayed in the result set. Having additional fields to display in the result set can slow down query performance. However, if the field is required for sorting or criteria purposes it must be included in the QBE grid.

To keep a field that is required in the QBE Grid from being displayed in the result set, de-select the check box in the "Show:" row of the QBE Grid for the field that you wish to hide in the result set. It will still function as a query criteria field, and you can still sort by its values in the QBE Grid, but its data will not display in the result set.

7.9- Using Comparison Operators:

You can use comparison criteria in the QBE section of the query design view in order to search for criteria values that are not necessarily "equal to" a value. By using comparison operators, Access can expand its repertoire of query criteria to pull records that are "greater than" or "less than" a specified criteria value, for example. You still enter these criteria into the "Criteria:" row in the QBE grid, under the column by which you wish to filter. You can use the following comparison symbols for query criteria:

>	greater than
>=	greater than or equal to
<	less than
<=	less than or equal to
=	equals
<>	does not equal

To use a comparison operator with the query criteria, open the query up in design view. In the "Criteria:" row, type the comparison operator and the value to compare underneath the field by which you wish to compare the value.

When you are entering query criteria, if making a reference to a text value note that it should be enclosed in quotes " " to work. Date values must be enclosed in pound signs # # to work. References to other table fields must be enclosed in brackets [] to work.

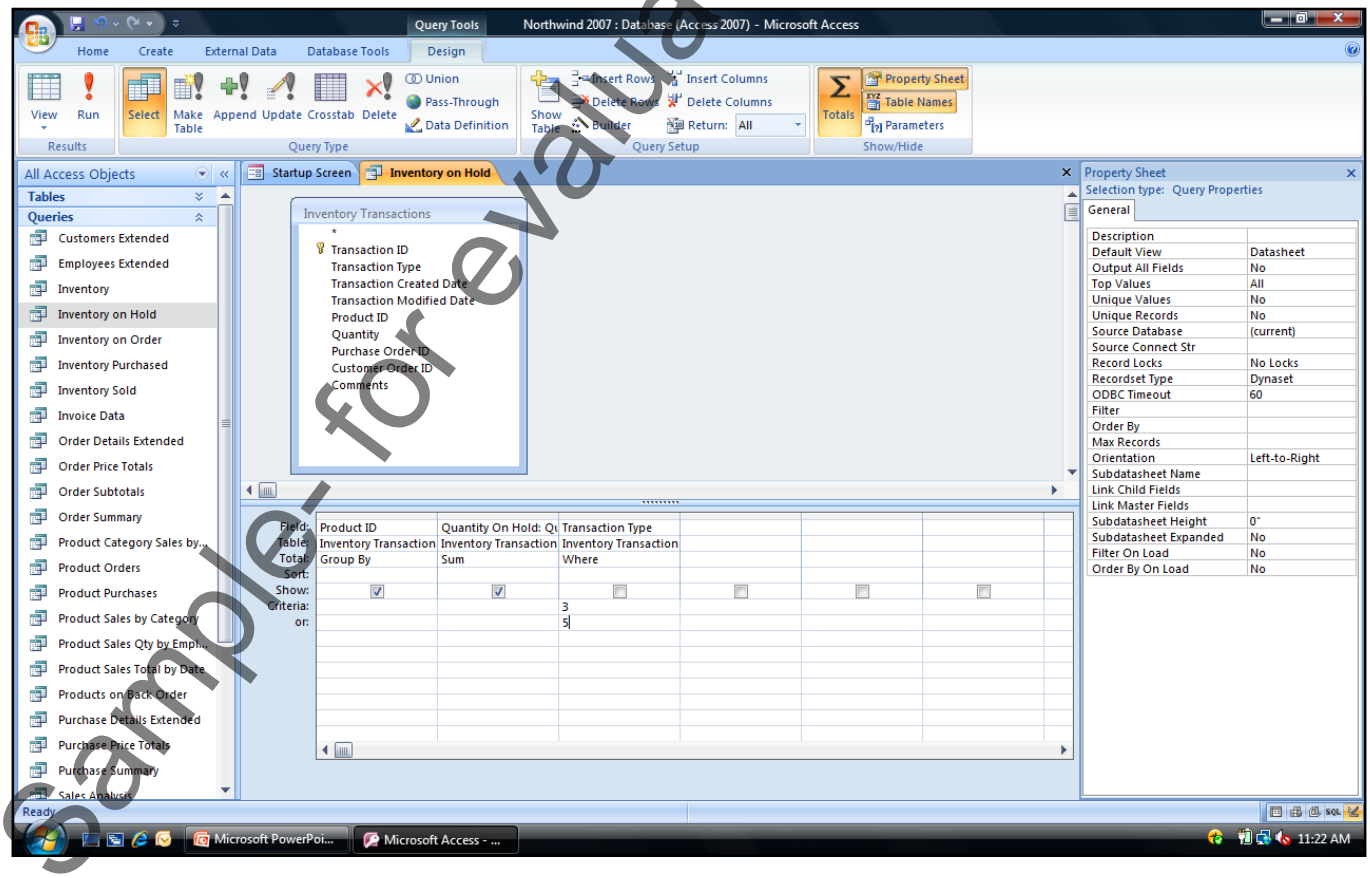
For example, assume that you had a "SaleDate" date/time data type field in the QBE grid of a query. Also assume that you wanted to find any sale that was on or after January 1st, 2007. To write the comparison criteria in this case, under the "SaleDate" column in the "Criteria:" row, you could enter **>= #01/01/2007#** as the comparison criteria. That would match the date values in the "SaleDate" column against the date value of 01/01/2007, and display any records where that value was greater than or equal to 01/01/2007.

7.10- Using "AND" and "OR" Conditions:

Next you will look at filtering the result set of a query by using multiple field criteria. Most often when you have multiple criteria in a query, you will either want the query to show records that match both "value X" AND "value Y" in different fields, or show records that contain "value X" OR "value Y" within the same or different fields. It's unusual to use an AND condition within a single field, but it isn't unheard of either. This is because when you use the AND condition, a record will only be returned in the result set if it meets all criteria conditions specified by the AND condition. When using the "OR" condition, a record will be displayed in the result set if it matches any of the criteria conditions specified.

When you use the QBE grid to set up AND and OR conditions, you must make use of the multiple fields in the "Criteria:" and "Or:" rows. To establish an AND condition between query criteria, the criteria must be placed into the *same criteria row*. This is typically how you will create an AND condition between query criteria. If you had to use an AND condition in a single field, however, it would have to be in this syntax: >10 AND <100, all in the same criteria cell. In this case you would find any amounts over 10 and under 100 within the field under which you place this criteria.

You may also join multiple query criteria together with the OR condition in a query. To establish an OR condition between query criteria, you place the query criteria *into separate criteria rows* under the same column. This is usually how you will use an OR condition. It also works the same way for OR conditions on multiple fields. Simply place the additional query criteria under the other field in the next "Or:" criteria row. It is useful to remember that any criteria placed into the *same* criteria row will be considered to be joined with the AND condition and any criteria placed into *different* criteria rows are considered to be joined with the OR condition.



ACTIONS- QUERIES

USING THE SIMPLE QUERY WIZARD:

1. Click the “Query Wizard” button in the “Queries” group (“Other” group in 2007) on the “Create” tab in the Ribbon.
2. In the “New Query” dialog box, select the “Simple Query Wizard” choice, and then click “OK” to begin.
3. In the first screen of the wizard, you must select the first table from which you will pull data by using the “Tables/Queries” drop-down. The fields from the table will display in the “Available fields:” list.
4. To add a field from the table into the query, select its name from the “Available fields:” list and click the “>” button to move it into the “Selected fields:” list.
5. Repeat this as needed, selecting the fields that you will want to see in the query.
6. When you are finished, click “Next >” to continue.
7. If you only selected fields from a single table in the first screen, then when you click the “Next >” button to continue, you will only need to provide the query with a name and then click the “Finish” button to finish creating the query.
8. If you picked data fields from two or more related tables, then when you click the “Next >” button, you will instead view a second screen which asks if you would like a “Detail” or “Summary” query. You can select the option button for the type of query that you wish to create.
9. If you select “Summary,” then you will be able to click the “Summary Options…” button to open the “Summary Options” dialog box. In this dialog box, you can select what type of summary to perform over a selected field. Make your selections, then click the “OK” button to return to the “Simple Query Wizard.”
10. Click “Next >” to continue. In the next screen, if you selected “Detail” on the previous screen, you will only need to name your query and then click “Finish” to create the query.
11. If you selected “Summary” and your summary includes dates, you may be presented with additional date grouping options. Continue to answer any questions, as appropriate for your query and click “Next >” to continue until you reach the final screen where you must type a name for your query, and then click “Finish” to create the query.

CREATING QUERIES IN DESIGN VIEW:

1. Click the “Query Design” button in the “Queries” group (“Other” group in 2007) on the “Create” tab in the Ribbon to create a new query in the query design view.
2. The “Add Table” dialog box will appear. Click the “Table” tab and select the name of a table to query for information. Click “Add” to add it to the query. Continue adding tables to your query by selecting the tables to add to the query, and clicking the “Add” button. Any tables that you related in the “Relationships” window will already appear with the appropriate table joins.
3. Click “Close” in the “Add Table” dialog box.
4. Select the first field that you want to add to your query by clicking and dragging the field from the table shown in the design view, and dropping it into the QBE grid section. You can also double-click the name of the field shown in the table to add it to the QBE grid, as well.
5. Continue adding fields to the query until you have all necessary fields added.
6. Click into the “Criteria:” row underneath any field for which you want to create query criteria and enter the desired query criteria into the “Criteria:” row.
7. Click the “Save” button in the Quick Access toolbar.
8. Type a name for your query into the dialog box that appears, and click “OK” to save it.

ACTIONS-
QUERIES

VIEWING QUERY RESULTS IN DATASHEET VIEW:

1. Select the name of the query whose result set you wish to view from the queries listed in the Navigation Pane, and then press the "Enter" key on your keyboard.
2. Alternately, you can double-click on the name of the query in the Navigation Pane whose result set you want to display.

RUNNING QUERIES FROM QUERY DESIGN VIEW:

1. Open the query whose result set you wish to view in query design view.
2. Click the "Run" button in the "Results" group on the "Design" tab in the "Query Tools" contextual tab within the Ribbon.

VIEWING A QUERY'S SQL CODE:

1. Open the query whose SQL code you wish to view in query design view.
2. Click the "View" drop-down button in the "Results" group on the "Design" tab in the "Query Tools" contextual tab within the Ribbon.
3. Select the "SQL View" command from the button's drop-down menu.

SORTING QUERY RESULTS:

1. To set the sorting in design view, select the field in the QBE Grid by which you would like to sort the result set, and click into that field's "Sort:" row.
2. You can use the drop-down to select either "Ascending" or "Descending" order. If you sort by multiple fields in the design view, the order of the sorting is applied by field from left to right when viewing the result set.
3. To remove sorting applied in this manner, select "(not sorted)" from the "Sort" row within the field from which you want to remove the sorting.
4. When viewing the result set of the query, you can apply sorting to selected fields by clicking the small drop-down arrows at the top of the column and then selecting "Sort A to Z" or "Sort Z to A."
5. You can also select a column and click either the "Ascending" or "Descending" buttons in the "Sort & Filter" group on the "Home" tab in the Ribbon.
6. To remove sorting that you have applied, click the "Remove Sort" button ("Clear All Sorts" button in Access 2007) in the "Sort & Filter" group to remove all sorting from the query result set.

HIDING FIELDS IN THE RESULT SET:

1. Open the query that contains fields that you wish to hide from the result set in query design view.
2. De-select the check box in the "Show:" row underneath the field that you need for query criteria, but do not want to display in the result set.
3. Click "Save" in the Quick Access toolbar to save your changes.

ACTIONS- Queries

COMPARISON OPERATORS LIST:

Operator:	Function:
>	greater than
>=	greater than or equal to
<	less than
<=	less than or equal to
=	equals
<>	does not equal

USING COMPARISON OPERATORS IN A QUERY:

1. Open the query in query design view to which you wish to apply comparison operators in the query criteria.
2. Click into the “Criteria:” row underneath the field for which you wish to apply a comparison query criteria.
3. Type the desired comparison operator, followed by the desired criteria values.
4. Click “Save” in the Quick Access toolbar to save your changes.

USING THE “AND” CONDITION TO JOIN MULTIPLE QUERY CRITERIA:

1. Open the query into which you wish to apply multiple query criteria joined with the “AND” condition in query design view.
2. Click into the “Criteria:” row underneath the first field into which you wish to place your criteria.
3. If you want to use the “AND” condition within a single field, type the first criteria followed by the word “and,” followed by the second criteria all under the same field.
4. If you want to use the “AND” condition in multiple fields, type the first criteria and then in the **same row** underneath the other fields, enter the other criteria.
5. Click “Save” in the Quick Access toolbar to save your changes.

USING THE “OR” CONDITION TO JOIN MULTIPLE QUERY CRITERIA:

1. Open the query into which you wish to apply multiple query criteria joined with the “OR” condition in query design view.
2. Click into the “Criteria:” row underneath the first field into which you wish to place your criteria.
3. Type the desired criteria.
4. Click into the “Or:” row beneath the “Criteria:” row, either in the same field or in another field, as needed.
5. Type the additional criteria.
6. Repeat steps 4 and 5 until you have created all of the necessary criteria
7. Click “Save” in the Quick Access toolbar to save your changes.

EXERCISES- QUERIES

Purpose:

1. To be able to create basic queries on your data.

Exercises:

1. Open your Access application.
2. Open the "test" database you completed from the Exercise at the end of the previous chapter.
3. Click the "Query Design" button in the "Queries" group ("Other" group in 2007) on the "Create" tab in the Ribbon.
4. In the "Show Table" dialog box, select "Customers" and click "Add."
5. Click "Close" in the "Show Table" dialog box.
6. Double-click on the "CompanyName" field in the table diagram to place it in the QBE grid below.
7. Double-click on the "City" field in the table diagram to add it to the QBE grid below.
8. Click into the "Sort:" row underneath the "CompanyName" field in the QBE Grid, and select "Ascending" from the drop-down that appears.
9. Click into the "Criteria:" row underneath the "City" field in the QBE Grid, and type "Lansing."
10. Click the "Show:" check box for the "City" field in the QBE grid to remove the check from the checkbox.
11. Click "Save" in the Quick Access toolbar and name the query "qryLansingCustomers."
12. Click "OK."
13. Close the query.
14. Select the name of the query from the "Queries" group in the Navigation Pane.
15. Press the "Enter" key on your keyboard to view the query's result set.
16. Close the result set.
17. Close the database.

CHAPTER 8-
ADVANCED QUERIES

8.1- Using the "Between…And" Condition:

You can use the "BETWEEN…AND" condition to look for values within a field that are between and inclusive of "Value X" and "Value Y," as specified. For example, if you had a date/time field in your table and wished to view any records where the date/time value was on any date in January 2007, you might place this criteria into the criteria row: BETWEEN #01/01/2007# AND #1/31/2007#. This is an "inclusive" selection in that it also includes any records that are equal in value to the criteria specified. Essentially, it is equivalent in value to writing the criteria >= #01/01/2007# AND <= #1/31/2007#. It is simply a more intuitive way of expressing the same thing. This is handy for finding records when you are searching for a range of records that fall between two specified values.

8.2- Using Wildcard Characters in Criteria:

You can also use "wildcard characters" to add an additional level of flexibility to your queries. Wildcard characters represent unknown values. There are two main wildcard characters that you need to know: the asterisk "*" and the question mark "?." The asterisk represents multiple unknown characters. For example, the criteria "N*" would find all "N" words like "Nebraska," "Ned," "Not," "Never Ever," etc.

The question mark represents one unknown character. So using a criteria like "N?D" would only find three letter "N…D" words like "Ned" and "Nod." You can type as many question marks as you need to fill in the requisite unknown characters. So typing "N??D" will find words like "Need" and "Nerd."

When you type wildcard characters into the criteria row of the QBE grid, Access will place the word "Like" before them. This is simply the required syntax. It isn't necessary for you to type it in yourself, if you do not wish to, as Access will add it for you. But also, don't worry if it appears in your criteria either.

8.3- Creating a Calculated Field:

You can create calculated fields in queries. A calculated field is a field that is derived by performing some type of function upon values gathered from other table fields, or entered by hand. The data is displayed only for the duration of the query, and is not actually stored in the tables. They can perform almost any function and can use any available query field or data entered by hand as the basis for their calculations.

To enter a calculated field into a query, just open the query into which you want to insert the calculated field in design view. In the "Field Name" text box in the first available, blank column, type a name for the new calculated field, followed by the colon symbol (:).

Next, enter the expression or formula that you want to evaluate and display in the new field immediately following the colon symbol. If you are referencing fields, the name of the field must appear within brackets. If you happen to have two fields with the same name in two different tables, then you must specify the name of the table in brackets first ([]), then a period (.) followed by the field name enclosed in brackets ([]). For example: [Table 2].[Field4]. You can only refer to fields that are available in the tables within the query, not just from any table within your database.

8.4- Creating "Top Value" Queries:

You can also create "Top Value" queries that will return the top or bottom results of a query, instead of all results. For instance, you could create a query that would show you the top performing salespeople in your sales region. You could also find the bottom performing salespeople just as easily.

8.4- Creating "Top Value" Queries (cont'd.):

To create a "Top Value" query, you create the query as normal in design view and then just select a choice from the "Return:" drop-down in the "Query Setup" group on the "Design" tab of the "Query Tools" contextual tab within the Ribbon.

From the list you can select a number or a percentage of records to display. If the number of records or percentage of records that you want isn't shown in the drop-down list that appears, you can click into the box directly and then type your own number or percentage into the drop-down box and press "Enter" on your keyboard to set a custom amount to show.

It is important to note that you **must** have the records **sorted by the column values for which you want to see the Top 10 values** in either "Ascending" or "Descending" order for this function to be of any use! If you have an "Ascending" sorted **number** field (1-10), requesting the "Top 10" will actually show you the smallest values! So it is important that you are aware of the way the query sorts its data. To display all records again, click the "Return" drop-down in the "Query Setup" group on the "Design" tab of the "Query Tools" contextual tab within the Ribbon again, and select "All" from the menu of choices.

8.5- Function Queries:

You can create summary queries that can perform a mathematical function on another grouped field in a query. These are usually shorter queries often used for reporting. For example, if you wanted to know the sum of sales for each salesperson in your company, you could show this in a summary query, assuming that you were recording the salesperson for each sale that occurred. You would need to first create a query that has the "Salesperson" field, followed by the "SalesAmount" field.

To create a summary query, open the query in design view and then click the "Totals" button in the "Show/Hide" group on the "Design" tab in the "Query Tools" contextual tab within the Ribbon. This will add an additional row into your query: the "Total:" row. Under each field in the query within this row, you will see the words "Group By."

The "Group By" value indicates that the query will group all records in that field that contain the exact same value. So, using our example, under the "Salesperson" fields you would leave the "Group by" value intact, which will then group all records where the "Salesperson" value is the same. Next, under the "SalesAmount" field you would click into the "Total:" row and select the function you wanted to perform on this field for each unique grouping created by the other field (or fields) by which you grouped the records. So, in this example, you would select the "Sum" function under the "SalesAmount" field. This query would then show you the "Sum" of the "SalesAmount" field for each set of records grouped by the values shown in the "Salesperson" fields. These type of functions are often called "aggregate functions," as they perform a function upon the aggregation of values in a field.

8.6- Parameter Queries:

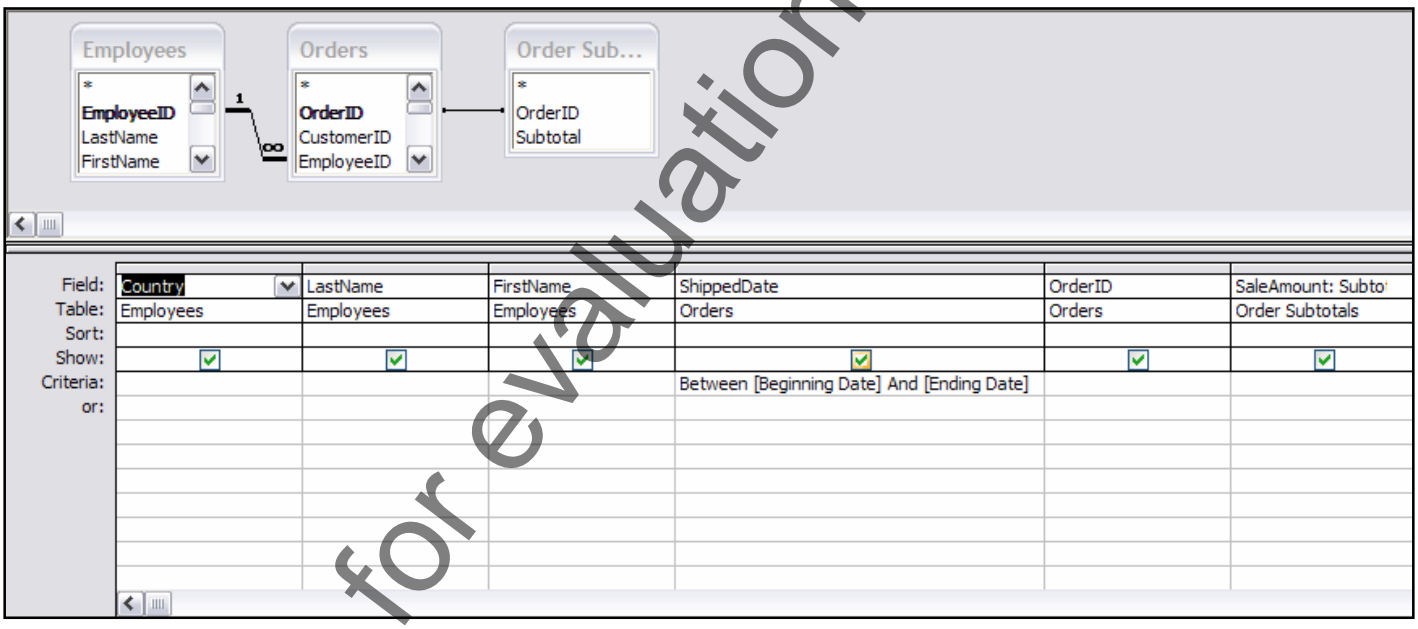
You can also create parameters in your query criteria that will prompt you to enter in the value which will then be used as the query criteria value for the query before returning the result set. This is tremendously helpful, as it prevents many hours of editing and changing query criteria. Each query with parameters in the criteria becomes more reusable, as each time the query is run it will prompt the user to input the parameter values used as the query criteria, which you can now easily change "on-the-fly" when you run the parameterized query. For example, suppose that every day hundreds of donation records were entered a non-profit organization's donations database. Let's also suppose that you were responsible for

8.6- Parameter Queries (cont'd.):

finding all records entered from the previous day and double-checking their spelling accuracy and address information. You certainly wouldn't want to have to sort through all of the records in the database tables to find the ones that were entered the day prior. Even if you created a query that found the records for the previous day's data entry, without parameters you would have to open the query in design view each day and edit the criteria values before running it. Instead, you can create a parameterized query that will prompt you to enter a value for a specified field (in this case the "EntryDate" field), that the query will then use to find those records that match the value which you entered when you opened the query.

To create a parameter query, construct your query as usual. Then click into the QBE grid underneath the field for which you want to set up a criteria parameter. To enter the parameter, type an open bracket ([), followed by what you want the parameter prompt to display, and ended with a closed bracket (]). (E.g.. [Please type the date here:]).

Parameters can be also be used in conjunction with other operators and conditions, like the "BETWEEN…AND" clause. This can be helpful in finding records in a range that you specify when you run the query. The syntax would be similar to the following: BETWEEN [Type the first date here:] AND [Type the second date here:].



ACTIONS-
ADVANCED QUERIES

USING THE "BETWEEN…AND" CONDITION:

1. Open the query into which you would like to insert the "Between…And" condition in query design view.
2. Click into the "Criteria:" row underneath the field into which you want to place the criteria.
3. Type "Between," then the first criteria value, followed by "And," ending with the last criteria value.
4. Click "Save" in the Quick Access toolbar to save your changes.

USING WILDCARD CHARACTERS IN QUERY CRITERIA:

1. Open the query into which you would like to insert wildcard character criteria in query design view.
2. Click into the "Criteria:" row underneath the field into which you want to place the criteria.
3. Type the criteria for which you want to search, using the appropriate wildcard characters:
   ? = a single unknown character.
   ***** = multiple unknown characters.
4. Click "Save" in the Quick Access toolbar to save your changes.

CREATING A CALCULATED FIELD:

1. Open the query into which you would like to insert a calculated field in query design view.
2. Click into the "Field:" row in the first available blank column in the query.
3. Type the name you want to give to the new calculated field, followed by a colon (:).
4. Next, type the expression (formula) required to give the field a value. Remember that when you reference field names they must be enclosed in brackets ([]).
   E.g. NewField: [Field5]*20%
5. Click "Save" in the Quick Access toolbar to save your changes.

CREATING A "TOP VALUE" QUERY:

1. Open the query which you wish to restrict to displaying the top values in query design view.
2. Click into the "Sort:" row under the field by which you wish to sort the records and select the desired sort order: "Ascending" or "Descending."
3. Click the "Return" drop-down in the "Query Setup" group on the "Design" tab of the "Query Tools" contextual tab within the Ribbon and then select the desired value from the list, or click into the "Return" drop-down and type your own value as a number or a percentage. If you enter your own value or percentage, press "Enter" on your keyboard when you are finished.
4. Click "Save" in the Quick Access toolbar to save your changes.

ACTIONS-
ADVANCED QUERIES

CREATING FUNCTION FIELDS IN QUERIES:

1. Open the query into which you would like to insert a function field in query design view.
2. Click the “Totals” button in the “Show/Hide” group on the “Design” tab in the “Query Tools” contextual tab within the Ribbon to add the “Totals:” row to the QBE grid.
3. Click into the “Totals:” row under the fields by which you want to group the records and select “Group By” from the drop-down menu.
4. Click into the “Totals:” row under the field upon whose values you wish to perform a function for each unique grouping created by the other field, and then select the mathematical function that you want to perform from the drop-down menu.
5. Click “Save” in the Quick Access toolbar to save your changes.

CREATING PARAMETER QUERIES:

1. Open the query into which you would like to insert criteria parameters in query design view.
2. Click into the “Criteria:” row underneath the field into which you want to place the criteria parameters.
3. Type an comparison operators or conditions, as needed, and then type an open bracket ([), the text for the parameter prompt, and a closed bracket (]) to finish creating the criteria.
4. Click “Save” in the Quick Access toolbar to save your changes.

EXERCISES-
ADVANCED QUERIES

Purpose:

1. To be able to create an advanced query on your data.

Exercises:

1. Open your Access application.
2. Open the "test" database you completed from the Exercise at the end of the previous chapter.
3. Click the "Query Design" button in the "Queries" group ("Other" group in 2007) on the "Create" tab in the Ribbon.
4. In the "Show Table" dialog box, select "Employees" and click "Add."
5. In the "Show Table" dialog box, select "Sales" and click "Add."
6. In the "Show Table" dialog box, select "SalesDetails" and click "Add."
7. Click "Close" in the "Show Table" dialog box.
8. Double-click on the "FirstName" field in the "Employees" table to place it in the QBE grid below.
9. Double-click on the "LastName" field in the "Employees" table to place it in the QBE grid below.
10. Double-click on the "Saledate" field in the "Sales" table to place it in the QBE grid below.
11. Click into the next available column in the QBE Grid and click into the "Field:" row.
12. In that cell, type "SaleAmount: [Price]*[Quantity]."
13. Click the "Totals" button in the "Show/Hide" group on the "Design" tab in the "Query Tools" contextual tab within the Ribbon.
14. In the "Totals" row, under the "Saledate" column, select "Where" from the drop-down.
15. In the "Criteria" row, under the "Saledate" column, type "Between #1/1/2007# And #1/31/2007#."
16. In the "Totals" row, under the "SaleAmount" column, select "Sum" from the drop-down.
17. Click the "Save" button in the Quick Access toolbar and name the query "qryJanuaryTotalSalesByEmployee."
18. Click the "OK" button to save the query.
19. Close the query.
20. In the Navigation Pane, double-click the name of the query that you just created to view the results.
21. Close the result set.
22. Close the database.

qryJanuaryTotalSalesByEmployee

FirstName	LastName	SaleAmount
Fred	Smith	$662.00
Greg	King	$159.00
Jack	Wells	$357.50
Joe	Smith	$366.00
Mary	Jones	$254.00

Access Keyboard Shortcuts

Category: General Shortcuts	
Command	**Key**
Open a database	Ctrl + O
Create a new database	Ctrl + N
Exit Access	Alt + F4
Print	Ctrl + P
Open “Page Setup” from “Print Preview”	S
Cancel “Print Preview” or “Layout Preview”	C or Esc
Save	Ctrl + S or Shift + F12
Open “Save As” dialog box	F12
Open a combo box (drop-down box)	Alt + Down Arrow or F4
Refresh a lookup field	F9
Move down one line in combo or list box	Down Arrow
Move down one page in combo or list box	Page Down
Move up one line in combo or list box	Up Arrow
Move up one page in combo or list box	Page Up
Exit the combo or list box	Tab
Open “Find” from Datasheet or Form View	Ctrl + F
Open “Replace” from Datasheet or Form View	Ctrl + H
Switch between Edit Mode and Navigation Mode in a datasheet	F2
Exit Navigation Mode in a form or report	Esc
Show/Hide Property Sheet in Design View	Alt + Enter or F4
Cycle between panes in Design View	F6
Open “Choose Builder” or Visual Basic Editor in Form and Report Design View	F7
Switch between Visual Basic Editor and Form or Report Design View windows	Shift + F7 or Alt + F11
Copy	Ctrl + C
Cut	Ctrl + X
Paste	Ctrl + V
Move selected form or report control right	Right Arrow or Ctrl + Right Arrow
Move selected form or report control left	Left Arrow or Ctrl + Left Arrow
Move selected form or report control up	Up Arrow or Ctrl + Up Arrow
Move selected form or report control down	Down Arrow or Ctrl + Down Arrow
Increase height of control	Shift + Down Arrow
Reduce height of control	Shift + Up Arrow
Increase width of control	Shift + Right Arrow
Decrease width of control	Shift + Left Arrow
Show/Hide Navigation Pane	F11
Cycle between open windows	Ctrl + F6
Close the active window	Ctrl + W
Switch focus of controls	Tab
“Next” button in wizards	Alt + N
“Back” button in wizards	Alt + B
“Finish” button in wizards	Alt + F
Display hyperlink address	F2
Check Spelling	F7
Open “Zoom” when performing data entry in small fields	Shift + F2
Switch forward to next object view, if available	Ctrl + , (comma)
Switch back to previous object view, if available	Ctrl + . (period)

Category: Using the Navigation Pane	
Command	**Key**
Rename selected Navigation Pane object	F2
Open (Run) selected Navigation Pane object	Enter
Design selected Navigation Pane object	Ctrl + Enter

Access Keyboard Shortcuts

Category: Using Menus	
Command	**Key**
Show pop-up menu	Shift + F10
Show access keys in Ribbon	Alt or F10
Show application menu	Alt + Spacebar
Close menus	Alt

Category: Using Windows and Dialog Boxes	
Command	**Key**
Switch to next open application	Alt + Tab
Switch to previous application	Alt + Shift + Tab
Shown Windows “Start” Menu	Ctrl + Esc
Close active window in database	Ctrl + W
Switch to next open database window	Ctrl + F6
Switch to previous open database window	Ctrl + Shift + F6
Switch to next tab in dialog box	Ctrl + Tab
Switch to previous tab in dialog box	Ctrl + Shift + Tab
Move to next option or option group	Tab
Move to previous option or option group	Shift + Tab
Perform selected button action or toggle status of selected checkbox	Spacebar
Close dialog box	Alt + F4
Show/Hide the Field List Pane	Alt + F8
Send selected Navigation Pane object as an email message	Alt + F + E

Category: Working with Text and Data	
Command	**Key**
Extend text selection one character right	Shift + Right Arrow
Extend text selection one word right	Ctrl + Shift + Right Arrow
Extend text selection one character left	Shift + Left Arrow
Extend text selection one word left	Ctrl + Shift + Left Arrow
Undo typing	Ctrl +Z or Alt + Backspace
Undo changes to field, then undo changes to record	Esc once to undo field, Esc twice to undo record
Insert current date	Ctrl + ; (semicolon)
Insert current time	Ctrl + : (colon)
Insert default field value	Ctrl + Alt + Spacebar
Insert previous record’s field value	Ctrl + ‘ (apostrophe)
Add new record	Ctrl + + (plus sign)
Delete current record	Ctrl + - (minus sign)
Save changes to current record	Shift + Enter
Switch between options in a checkbox or option box	Spacebar
Insert a new line	Ctrl + Enter
Recalculate fields	F9
Requery underlying data	Shift + F9