ACCESS

3

Querying a Database

One of the main goals of a database is to organize data so that information can be located and retrieved quickly. People in all types of businesses retrieve stored data and information daily, often at a moment's notice. In this chapter, you will search information that is stored in tables in a relational database and extract records that meet specific criteria using a query, a database object used to locate records based on the conditions you set.

LEARNING OBJECTIVES

▸ Create, save, and run select queries
▸ Create select queries using multiple tables
▸ Use simple query criteria
▸ Use AND and OR criteria in queries
▸ Use wildcard characters in query criteria
▸ Sort query results
▸ Create and format a calculated field

🗁 Project: Using Queries to Get Answers

You have been asked to query the Winchester Web Design database and compile two separate customer lists. The lists will be used to notify all past clients of updates to their website contact forms. The first list will include only the first and last names of the clients and their email addresses. The second list will include the first and last names of the clients and their mailing addresses, sorted by ZIP code. Additionally, you have been asked to build queries that instantly calculate the total income from all the Winchester Web Design services and from specific areas such as blogs or shopping carts.

Select Queries

A query asks a question, such as, *What are the customer addresses?* or *How much money did the company make last month?* The answer to the question is provided in a set of records.

All queries have common attributes:

▶ They function like a saved question you ask a database.

▶ They produce a subset of data from one or more tables.

▶ They are dynamic objects that display up-to-date data from tables.

▶ They can be used to create forms and reports with fields drawn from multiple tables.

▶ When you edit data in query results, you are actually editing the data stored in the source tables.

A select query is basically a database inquiry that selects only the records you want to see or edit, from one or more database tables, based on criteria that you set. The easiest way to create a select query is with the Query Wizard.

≡ Create→Queries→Query Wizard 🔍

DEVELOP YOUR SKILLS: A3-D1

In this exercise, you will use the Query Wizard to create a select query that generates a customer email list.

1. Open **A3-D1-WinDesign** from your **Access Chapter 3** folder and save it as:
   `A3-D1-WinDesignRev`

   Notice in the Navigation pane that the database currently has three queries.

**Note!** *When completing exercises, always choose to Enable Content.*

2. Choose **Create→Queries→Query Wizard** 📊.

The New Query dialog box appears, allowing you to select the Query Wizard you would like. The Wizard can help you create four types of queries, shown in the right pane of the dialog box.
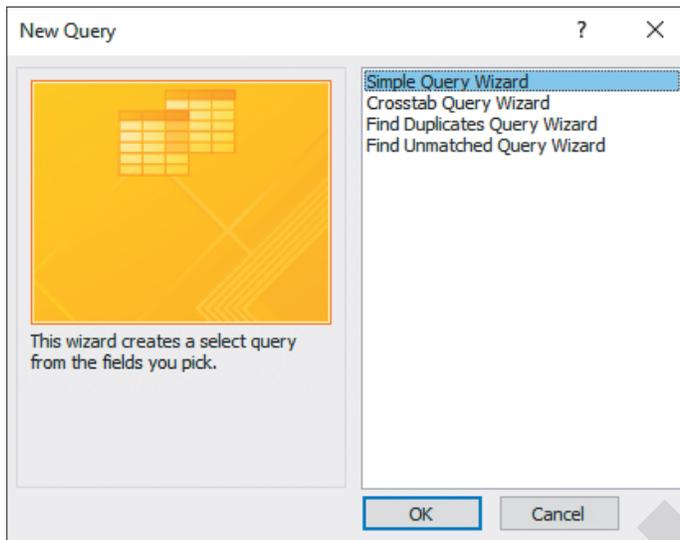
3. Click **OK** to accept the Simple Query Wizard.

4. Follow these steps to build the query:
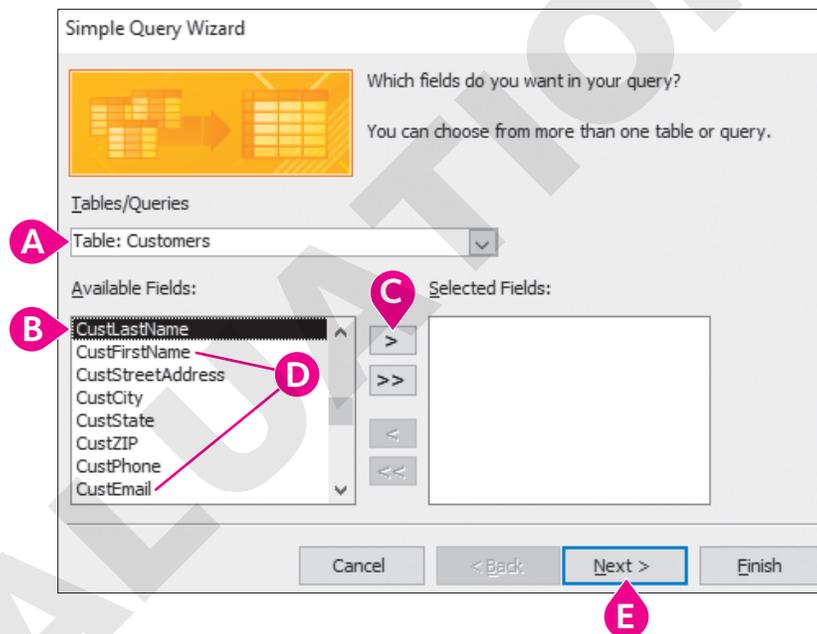
Ⓐ Make sure the **Customers** table is chosen in the Tables/Queries list. When building a query, you can use multiple tables and even existing queries.

Ⓑ Choose the **CustLastName** field from the Available Fields list.

Ⓒ Click the **Add** > button to add it to the Selected Fields list.

Ⓓ Now add the **CustFirstName** and **CustEmail** fields, scrolling to find them as necessary, to the Selected Fields list.

Ⓔ Click **Next**.

If you add the wrong field, double-click the name to move it back to the Available Fields list or select it and use the Move Back < or Move All Back << buttons.

Select Queries **73**

5. Type **Customers Email List** in the query title field at the top of the dialog box.

6. Make sure the **Open the Query to View Information** option is chosen and click the **Finish** button.

Notice the query results datasheet includes only the three fields you chose from the Customers list.

Last Name	First Name	Email
Abrams	John	JPAbrams@email.com
Anders	Mark	AndersM@email.com
Blaser	Helen	BlasingHel@email.com
Davis	Peter	DavisAngie@email.com

Customers Email List

7. Click the **Close** × button to the right of the *Customers Email List* tab to close the query.

Creating a Select Query Using Query Design View

Some queries display just a few fields but report on every single record in the table. That may not be a problem for a small table, but, when thousands of records and multiple tables are involved, it is often necessary to choose only specific records by setting precise criteria. Using Query Design View, Access allows you to:

▶ Select fields from multiple tables

▶ Locate records using criteria from one or more fields

▶ Perform calculations

▶ Sort query results and show or hide fields in query results

≡ Create→Queries→Query Design 🖾

DEVELOP YOUR SKILLS: A3-D2

You have already created an email list for the Winchester Web Design customers and now need one for the company's employees. In this exercise, you will create a query to select fields from the Employees table in the Winchester Web Design database and then rearrange the columns in the query grid.

1. Choose **Create→Queries→Query Design** 🖾 to display the query design grid.

The Show Table list appears, showing the existing tables and queries in the database.

2. Choose the **Employees** table and click the **Add** button.

The Employees table appears in the design grid.

3. Close the Show Table dialog box and close the Property Sheet if it's open.

Next you will add fields from the Employees table to the grid.

4. Double-click the **EmpFirstName** field in the Employees table to add it to the grid.

5. Now add the **EmpLastName**, **EmpPhone**, and **EmpEmail** fields to the grid by either double-clicking or dragging them.

Tip! *Use the scroll bar located at the right of the table fields to access all available fields in the list.*
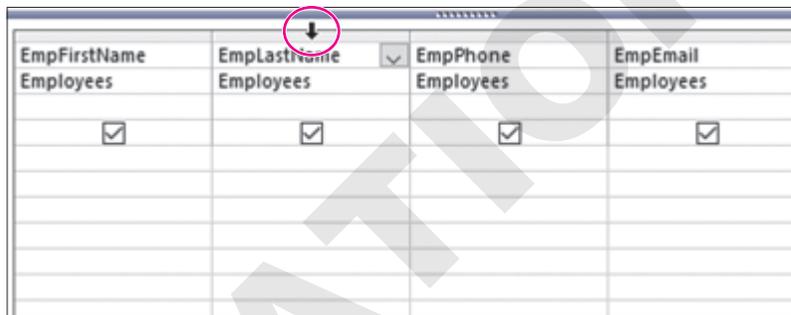
Field:	EmpFirstName	EmpLastName	EmpPhone	EmpEmail
Table:	Employees	Employees	Employees	Employees
Sort:				
Show:	☑	☑	☑	☑
Criteria:				
or:				

6. Choose **File→Save** or click the **Save** button on the Quick Access toolbar.

7. Type **Employee Contact Info** as the query name and click **OK**.

8. Choose **Query Tools→Design→Results→Run** ⚠.

   Access runs the query and displays four columns of data (First Name, Last Name, Telephone, and Email) for all Employee records.
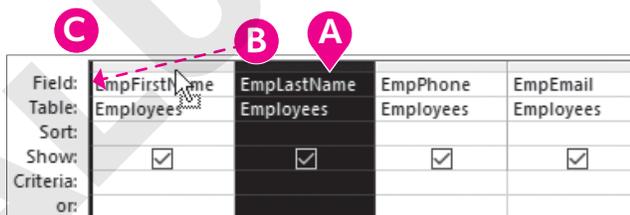
Rearrange Query Fields

9. Choose **Home→Views→View→Design View** 📝.

10. Select the **EmpLastName** column by placing your mouse pointer over the column heading until it becomes a downward-facing black arrow and then single-click.

EmpFirstName	EmpLastName ▼	EmpPhone	EmpEmail
Employees	Employees	Employees	Employees
☑	☑	☑	☑

   The entire column is selected, shown by shading it in black.

11. Follow these steps to rearrange the EmpFirstName and EmpLastName fields:

Field:	EmpFirstName	EmpLastName	EmpPhone	EmpEmail
Table:	Employees	Employees	Employees	Employees
Sort:				
Show:	☑	☑	☑	☑
Criteria:				
or:				

   A Click the **EmpLastName** column heading again, this time without releasing the button. The button will change to a white arrow with a silhouetted box, indicating you can move the column.

   B Drag the **EmpLastName** column to the left of the *EmpFirstName* column until the thick vertical bar is positioned as shown.

   C Release the mouse button to complete the rearrangement.

12. **Run** ⚠ the query.

   Last Name should now appear first in the results.

13. Close the query and save the changes.

Designing a Query Using Multiple Tables

Until now, the query results presented in the datasheets you have worked with have displayed data from only one table. There will be times when you need to view data contained in different tables within the same database. Multi-table queries allow you to do this.

Choosing Fields to Include in a Query

When you build a multi-table select query, you start in Query Design view. Using the Show Table dialog box, you can select only those tables and fields that you want to display in the query results datasheet and leave out those fields that have no impact on the data you want to view or that are confidential. By specifying only certain tables and fields in a database and displaying only the desired fields in a query, you can create a report or a form that presents only pertinent data.

Multiple tables are effective in a query only if the tables are related. Using related tables allows a query to provide results based on all the data contained in the related table fields selected. For example, if you want to find the names and addresses of customers who placed orders from a specific employee, you would need fields from both the Customers table and the Invoices table. This is because the Customers table does not include any Employee fields, and the Invoices table does not include the Customer fields. It would be impossible to answer the question using only the Customers or Invoice tables alone.
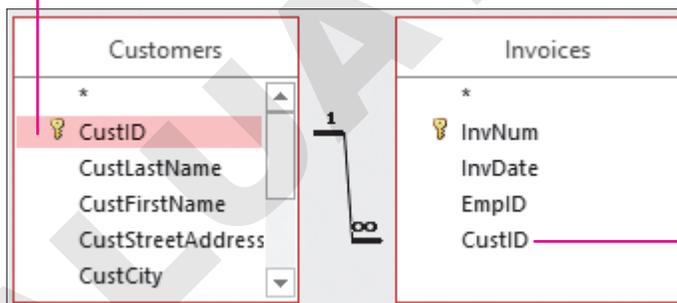
View the video "Create a Multi-Table Select Query."

Selecting a Field That Appears in Multiple Tables

Sometimes the same field occurs as a primary key in one table and as a foreign (or secondary) key in another table. If this occurs, always use the table with the primary key in your query.

Use the Customers table in the query because CustID is the primary key.

CustID is a foreign key in the Invoices table.

DEVELOP YOUR SKILLS: A3-D3

In this exercise, you will create a multi-table query using Query Design view to track the Winchester Web Design invoices by invoice number.

1. Choose **Create→Queries→Query Design** to display the query design grid.
2. Double-click the **Invoices**, **Invoice Details**, and **Products** tables to add them to the query.

   If the Show Table dialog box is not visible, choose Query Tools→Design→Query Setup→Show Table.

3. Close the Show Table dialog box.

4. Double-click the **InvNum**, **InvDate**, and **EmpID** fields in the Invoices table to add those fields to the query grid.

5. Add the **ProdDescription** and **Price** fields from the Products table.

6. Add the **Qty** field from the Invoice Details table.

Your query field list should look like this.

Field:	InvNum	InvDate	EmpID	ProdDescription	Price	Qty
Table:	Invoices	Invoices	Invoices	Products	Products	Invoice Details
Sort:						
Show:	☑	☑	☑	☑	☑	☑
Criteria:						
or:						

7. Click the **Sort** cell for the InvNum field, click the **menu** button ▼, and choose **Ascending**.

Field:	InvNum	InvDate
Table:	Invoices	Invoices
Sort:	Ascending	
Show:	☑	☑
Criteria:		
or:		

8. Choose **Query Tools**→**Design**→**Results**→**Run** [!] to run the query.

The query results are now sorted by invoice number in the first column.

9. Click the **Save** button on the Quick Access toolbar; then name the query **InvoicesList** and click **OK** to save the query.

Using Criteria in Queries

Queries let you specify criteria, which are conditions that field values must meet. Only records meeting the criteria are returned when the query is run.

Field:	InvNum	InvDate	EmpID	ProdDescription	Price	Qty
Table:	Invoices	Invoices	Invoices	Products	Products	Invoice Details
Sort:	Ascending					
Show:	☑	☑	☑	☑	☑	☑
Criteria:			"JFW"			
or:						

In this query, the EmpID criteria is set to *JFW*.

Invoice Details Query

InvNum	Invoice Date	Emp ID	Description	Price	Qty
1	3 /14/2017	JFW	Home Page, Nav, CSS, Design	$400.00	1
1	3 /14/2017	JFW	Secondary Page	$200.00	6
1	3 /14/2017	JFW	Image, Custom Designed	$40.00	11
5	6 /18/2017	JFW	Home Page, Nav, CSS, Design	$400.00	1

Only records where EmpID is *JFW* are returned.

ACCESS

Criteria are commonly used with text, numeric, currency, and date fields. Review the table for examples of how criteria are used.

TYPES OF CRITERIA

Field Type	Criteria	Examples of How Records Are Returned
Text	Smith	Last name is Smith
	> =Smith	Last names are from Smith through the end of the alphabet
	Not Smith	Last name is not Smith
Numeric & Currency	> 123	Numeric value is greater than 123
	> =123	Numeric value is greater than or equal to 123
Date	Date( )	Date is today's date
	< Date( ) – 30	The Date field is 30 days or more prior to today's date

Tip! *Search for* Query Criteria *in Access help for more criteria examples.*

DEVELOP YOUR SKILLS: A3-D4

In this exercise, you will add criteria to the query grid and run the query.

1. With the InvoicesList query open, switch to **Design View** .

2. Click the **Criteria** cell for the EmpID field.

3. Type **JFW** and tap Enter .

   Access will apply quotation marks indicating this is a literal value.

Field:	InvNum	InvDate	EmpID
Table:	Invoices	Invoices	Invoices
Sort:	Ascending		
Show:	☑	☑	☑
Criteria:			"JFW"
or:			

4. Choose **Query Tools→Design→Results→Run** ! to run the query.

   The query results now include only records where the EmpID is equal to JFW.

5. Close the query and choose **No** when prompted to save the changes.

   Saving changes to the query at this time would save the JFW criteria as part of the query. However, you will continue to use the query for all employees.

Wildcard Characters

Wildcard characters are used to locate records that have similar but not identical data. They help you locate records that match a pattern. For example, you might want to find all customers with last names that begin with the letter *B* or all products that begin with the word *Blog*.

COMMON WILDCARD CHARACTERS

Wildcard Symbol	How It Is Used
Asterisk (*)	Substitutes for a group of characters that appear at the position of the asterisk
	Example: *R** in the last name field will locate all last names beginning with *R* regardless of how many characters make up the name. In this case, *Rogers, Rich,* and *Rodriquez* would all appear in the results datasheet.
Question mark (?)	Substitutes for a single character that might appear at the position of the question mark
	Example: *m?s* will locate records containing values such as *mrs, ms,* and *mbs*.
Open/close brackets [ ]	Matches text or individual characters placed within the brackets individually
	Example: *ca[rt]* will find *cat* and *car* but not *cab* or *cad*.

Tip! *Search for* wildcard characters *in Access help for more wildcard symbols and examples.*

AND and OR Criteria

In some cases, you may need to locate records that meet multiple criteria. This can be done using AND and OR conditions. For example, you may want to locate all records where the employee is web certified AND lives in Sarasota. Or you may want to locate all employees who live in Sarasota OR Bradenton.

Field:	CustID	CustFirstName	CustLastName	CustCity
Table:	Customers	Customers	Customers	Customers
Sort:				
Show:	☑	☑	☑	☑
Criteria:				"Sarasota"
or:				"Bradenton"

Create an OR condition by adding a second criterion to the Or row of a field.

Field:	InvNum	InvDate	ProdDescription	Price	Qty
Table:	Invoices	Invoices	Products	Products	Invoice Details
Sort:	Ascending				
Show:	☑	☑	☑	☑	☑
Criteria:			"Image"		>10

Create an AND condition by adding another criterion to a different field on the Criteria row.

View the video "Create a Query with Criteria."

Using Criteria in Queries **79**

DEVELOP YOUR SKILLS: A3-D5

In this exercise, you will use wildcards to locate variable data and set multiple criteria in a query to find out which customers have gotten blogs and which customers have added more than ten images at a time to their websites.

1. Open the **Invoices Query** query in **Design View**.
2. Follow these steps to use wildcard characters and to use AND and OR criteria:

Field:	InvNum	InvDate	ProdDescription	Price	Qty	LineTotal: [Qty]*[Price]
Table:	Invoices	Invoices	Products	Products	Invoice Details	
Sort:						
Show:	☑	☑	☑	☑	☑	☑
Criteria:			Like "Blog*"			
or:			Like "Image*"		>10	

Ⓐ In the ProdDescription **Criteria** cell, type **Blog*** and tap ⎡Enter⎤.

Access converts Blog* *to:* Like "Blog*"

Ⓑ In the ProdDescription **Or** cell, type **Image*** and tap ⎡Enter⎤.

Ⓒ Enter **> 10** in the Qty **Or** cell. Be sure to type in the same row as *Like "Image*"* (the *Or* row).

These criteria will choose records where ProdDescription begins with Blog *OR ProdDescription begins with* Image *AND the Qty is greater than 10.*

3. **Run** ⎡!⎤ the query.

Access displays the records that meet the specified criteria: either a blog or a transaction with more than ten images.

4. Close the query and save the changes.

Date Criteria

You can set date criteria to determine age, hired date, invoice date, and so forth. Access acknowledges the same comparison criteria for performing date comparisons that it does for locating other types of data, regardless of the format used to enter dates.

DATE CRITERIA

Criterion	Examples of How Records Are Returned
06/22/2019	Date is 06/22/2019
<22-Oct-2019	Dates that occur before 22/Oct/2019
>01/01/19	Dates that occur after 01/01/19
Between 01/01/19 and 06/30/19	Dates between 01/01/2019 and 06/30/2019

DEVELOP YOUR SKILLS: A3-D6

Winchester Web Design needs to track all invoices issued in 2018. In this exercise, you will query the database to locate customers with invoices dated from January 1, 2018, through December 31, 2018.

1. Choose **Create→Queries→Query Design** 🖾.
2. Use double clicks to add the **Customers**, **Invoices**, **Invoice Details**, and **Products** tables to the query.

3. Close the Show Table dialog box.

4. In the Invoices table double-click **InvNum** and **InvDate** to add those fields to the query grid.

5. From the Customers table add the **CustID** field.

6. From the Invoice Details table add the **Qty** field.

7. From the Products table add the **ProdDescription** and **Price** fields.

8. Hover your mouse pointer on the right edge of the InvDate column heading so a black, two-direction arrow appears.

InvDate	CustID
Invoices	Customers
☑	☑

9. Click and drag the column heading to the right until the column is about three times the original width.

   You will enter a long entry in the next step and widening the InvDate column will allow you to see the entire entry.

10. Click in the **Criteria** cell for the InvDate field and type: `Between January 1, 2018 And December 31, 2018`

    Access formats the expression. Your query grid should now match this example. Regardless of how you type the dates—whether January 1, 2018; 01/01/18; or 1-1-2018—Access formats the data after you enter it so it appears as #1/1/2018#.

InvNum	InvDate	CustID	Qty	ProdDescription	Price
Invoices	Invoices	Customers	Invoice Details	Products	Products
☑	☑	☑	☑	☑	☑
	Between #1/1/2018# And #12/31/2018#				

11. **Run** ! the query.

    Notice that only records with a date in 2018 appear in the results.

12. Choose **File→Save** or click the **Save** button on the Quick Access toolbar.

13. Save the query as `Invoices2018` and then close it.

Sorting, Showing, and Limiting Results

The query grid contains a Sort row that lets you sort the query results. At times you may also want to use fields to specify criteria but may not want those fields to be displayed in the query results. This can be accomplished by unchecking the Show box for the desired field(s).

Limiting the Number of Results Displayed

Large databases with thousands of records often return so many records that it can be challenging to find what you are looking for. Limiting the number of records displayed can be beneficial, especially when these records are sorted.

ACCESS

For example, if you set up a query to sort in descending order and then limit the number of items displayed to ten, you would, in effect, have a list of the top ten items in the table being queried. The Return feature lets you set the number of records to be displayed, or returned, in the query results.

≡ Design→Query Setup→Return 🖩

DEVELOP YOUR SKILLS: A3-D7

In this exercise, you will create a query that sets a sort order and hide a field from displaying in the query results. You will also limit the number of records returned.

1. Choose **Create→Queries→Query Design** 📇.

2. Use double clicks to add the **Customers**, **Invoices**, **Invoice Details**, and **Products** tables to the query.

3. Close the Show Table dialog box.

4. In the Customers table double-click the **CustID**, **CustFirstName**, and **CustLastName** fields to add them to the design grid.

5. From the Invoices table add the **InvDate** field.

6. From the Products table add the **ProdDescription** field.

7. From the Invoice Details table add the **Qty** field.

8. Follow these steps to set a criterion and set the sort order:

Field:	CustID	CustFirstName	CustLastName	InvDate	ProdDescription	Qty	
Table:	Customers	Customers	Customers	Invoices	Products	Invoice Details	
Sort:						Descending	
Show:	☑	☑	☑	☑	☑	☑	
Criteria:					Like "Image*"		
or:							

Ⓐ In the ProdDescription **Criteria** cell, type **Image*** and tap ⌷Enter⌷. Access converts *Image** to *Like "Image*"*. This criterion will choose only records where the product description begins with *Image*.

Ⓑ In the **Sort** cell for the Qty field, choose **Descending** from the list of sort options.

9. **Run** ❗ the query.

The records are now sorted in descending order (largest to smallest) by quantity.

CustID is an important key to have in the query because it is a primary key field. But it isn't needed in the query results because it contains the same information that appears in the CustFirstName and CustLastName fields, so you will hide it from the query results.

10. Choose **Home→Views→Design View** 📐 to switch back to Design View.

11. Uncheck the **Show** box for the CustID field and **Run** ❗ the query.

The CustID field is still part of the query design, but it no longer shows in the query results.

12. Switch back to **Design View** and choose **Query Tools→Design→Query Setup→ Return** 🖩 **menu button** ▼.

13. Choose **5** from the list and run the query.

The query returns seven records (not five). This is because the query returns all records with the five largest quantities. But three records had a Qty of 14, which is the fifth highest amount, so all those records were returned, increasing the total to seven records.

14. Save the query as **Most Images** and then close it.

Calculated Fields

Calculated fields are fields containing formulas that perform calculations. Formulas used in calculated fields are often based on other fields within the query. Calculated fields are added as an additional field to a query and are not part of the underlying query tables. They are added to the query design grid and their calculated results then appear in the query results. A calculated field:

▸ Creates a new field in a query that can also be used in a form or report

▸ Can be used to perform mathematical operations, such as addition and multiplication

▸ Has a name and can be formatted with properties just like a regular field

▸ Enables you to combine values in two text fields into one field, such as LastName and FirstInitial

▸ Updates and recalculates each time you run the query

Identifying Parts of a Calculated Field

The structure of a calculated field includes a field name and a mathematical expression. An example of a calculated field in an Access query is Wage: 12.00 * 40, where Wage is the calculated field name and 12.00 * 40 is the calculation to be performed. Another example is Total: Price * Quantity, where Total is the calculated field name and Price * Quantity are the calculations performed using the data in those query fields.

Price	Qty	LineTotal: [Price]*[Qty]
Products	Invoice Details	
☑	☑	☑

Price ▾	Qty ▾	LineTotal ▾
$200.00	6	$1,200.00
$40.00	11	$440.00
$400.00	1	$400.00
$40.00	15	$600.00

The LineTotal calculated field multiplies Price * Qty.   **The query results.**

Each calculated field can contain the following elements:

CALCULATED FIELD ELEMENTS	
Element	**Description**
Calculated field name	This is the unique name you assign to the field and is followed by a colon (:) to separate the field name from the expression.
Field names from existing tables	Field names from the query can be added to the calculated field expression. Access adds brackets [ ] around field names.
Arithmetic or comparison operators	Use +, -, *, /, ( ), ^, <, =, and > to perform mathematical operations or compare values.
Concatenation (i.e., linking together)	An ampersand (&) can be used to join text values from multiple fields. For example, FirstName&LastName.
	Spaces can be added between fields by using quotation marks around a single space (" "). For example, the quotation marks in FirstName& " " &LastName create a space between the first and last names in the query results.

ACCESS

Calculated Field Properties

You can set field properties such as size, number format, and default values within tables. Likewise, you can set field properties in calculated fields. This is almost always needed in calculated fields as the query results need to be formatted with the correct number of decimal places, commas, currency format, and other formatting as needed. Field properties are set using the field Property Sheet.

≡ Design→Show/Hide→Property Sheet 🗒

DEVELOP YOUR SKILLS: A3-D8

In this exercise, you will create and format a calculated field.

1. Open the **InvoicesList** query and switch to **Design View**.
2. Click in the first cell of the blank column next to the Qty field.

Price	Qty		
Products	Invoice Details		
☑	☑	☐	
	>10		

3. Type the calculated field expression, taking care to include the colon between *LineTotal* and *Price:*
   LineTotal:Price * Qty
4. Tap **Enter** to accept the entry and allow Access to format the expression by adding brackets to the field names.

   Access does not always format your expression by adding brackets to field names. Brackets are required for Access to identify the entry as a field within the query. If you want to use field data within a calculated field expression, you need to include the name of your calculated field within brackets. In this example our calculated field is named LineTotal*, and it will multiply the data in the Price field by the data in the Qty field in each record when the query is run. Your completed field should match the following.*

Field:	InvNum	InvDate	EmpID	ProdDescription	Price	Qty	LineTotal: [Price]*[Qty]
Table:	Invoices	Invoices	Invoices	Products	Products	Invoice Details	
Sort:	Ascending						
Show:	☑	☑	☑	☑	☑	☑	☑
Criteria:							
or:							

5. Right-click anywhere in the column of your calculated field and choose **Properties**.

   The Property Sheet opens.
6. Click the **Format** field, then click the **menu** button ▾ and choose **Currency** from the list.
7. Click the **Caption** field and type: **Line Total**

   The currency format will display the calculated results with a dollar sign and two decimals. The caption will become the column heading for your calculated field in the query results.

8. Run [!] the query, and your calculated field results will appear as shown:

InvoiceList						
InvNum	Invoice Date	Emp ID	Description	Price	Qty	Line Total
1	3/14/2017	JFW	Secondary Page	$200.00	6	$1,200.00
1	3/14/2017	JFW	Image, Custom Designed	$40.00	11	$440.00
1	3/14/2017	JFW	Home Page, Nav, CSS, Design	$400.00	1	$400.00
2	4/1/2017	MJW	Image, Custom Designed	$40.00	15	$600.00
2	4/1/2017	MJW	Home Page, Nav, CSS, Design	$400.00	1	$400.00
2	4/1/2017	MJW	Secondary Page	$200.00	7	$1,400.00

9. Save and close the query and then close the database.

Self-Assessment

Check your knowledge of this chapter's key concepts and skills using the Self-Assessment in your ebook or online (eLab course or Student Resource Center).

ACCESS

📌 Reinforce Your Skills

Create Queries Using Criteria and Wildcards

Kids for Change is planning to fine-tune its database by adding queries that enable it to track activities as well as staff/volunteer availability. In this exercise, you will create various queries that will yield the desired information.

1. Open **A3-R1-K4C** from your **Access Chapter 3** folder and save it as: **A3-R1-K4CRev**
2. Choose **Create→Queries→Query Wizard** 🔲.
3. Choose **Simple Query Wizard** and click **OK**.
4. Add the **Activity**, **Location**, **Day**, and **MeetTime** fields from the Activities table to the Selected Fields list and click **Next**.
5. Name the query **Activities List** and click **Finish**.
6. Review the query results and then close the query.

Create a Query in Design View

7. Choose **Create→Queries→Query Design** 🔲 to start a new query.
8. Add the **Volunteers** table and then close the Show Table dialog box.
9. Add the **VolLastName**, **VolFirstName**, **VolPhone**, and **VolDay** fields to the query design grid (in that order).
10. Run the query and take a moment to review the results.

    Now you will change the field order.
11. Switch to **Design View** 🔲.
12. Click and drag the **VolDay** field, dropping it in front of the *VolLastName* field.

Field:	VolDay	VolLastName	VolFirstName	VolPhone
Table:	Volunteers	Volunteers	Volunteers	Volunteers
Sort:				
Show:	☑	☑	☑	☑
Criteria:				
or:				

13. **Run** ⚠ the query and review the results.

    You may notice the field names appear differently from those used in the query grid. Field names in queries retain any caption labels previously set in table properties.
14. Save the query as **Volunteer List** and then close the query.

Create a Multi-Table Query

15. Create a new query in **Query Design** view, add the **Activities** and **Staff** tables to the query, and then close the Show Table dialog box.

16. Move the fields from the indicated tables to the query design grid:

From This Table	Add These Fields
Activities	• **Activity** • **Day** • **MeetTime**
Staff	• **StaffLastName** • **StaffFirstName** • **StaffPhone**

17. Set the Sort option for the Activity field to **Ascending**.

18. **Run** ❗ the query and view the results.

19. Save the query as `Activity Staffing List` and then close it.

Add Wildcard and AND/OR Criteria to a Query

20. Right-click the **Activity Staffing List** query in the Navigation pane and choose **Design View**.

You can open a query in Design View using this method or you can run it first and then switch to Design View. Remember to try right-clicks if you are having trouble finding commands.

21. Create a Saturday or Sunday OR condition in the Day field.

Typing the quotation marks " " isn't necessary, as Access will add them for you.

Field:	Activity	Day	MeetTime
Table:	Activities	Activities	Activities
Sort:	Ascending		
Show:	☑	☑	☑
Criteria:		"Saturday"	
or:		"Sunday"	

22. Run the query.

Only activities for Saturday or Sunday should be displayed.

23. Switch to **Design View** and delete **Sunday** to remove the OR condition.

24. Enter `12:00` in the **MeetTime Criteria** field, tapping [Enter] when finished.

This creates a Saturday AND 12:00 meet-time condition. Access will format the 12:00 condition like this: #12:00:00 PM#

Field:	Activity	Day	MeetTime
Table:	Activities	Activities	Activities
Sort:	Ascending		
Show:	☑	☑	☑
Criteria:		"Saturday"	#12:00:00 PM#
or:			

25. Run the query.

Because both conditions must be met, just one activity meeting (a car wash) should be returned by the query.

26. Switch to **Design View** and remove both the **Saturday** and **12:00** criteria.

Reinforce Your Skills **87**

27. Type **S*** in the Criteria cell for the Day field and tap ⎡Enter⎤.

    Access recognizes the asterisk * wildcard character and formats the condition as Like "S*". The query will return all records where the name of the day begins with S (Saturday and Sunday) and should produce the same results as when you used the Saturday OR Sunday condition earlier in this exercise.

28. Run the query and take a moment to observe the results.

29. Save the changes and close the query.

Add Date Criteria to a New Query

Now you will create a query that returns the records of the youngest children so you can determine which children may need more supervision.

30. Create a new query in **Design View**, adding the **Children** table and the fields **ChildLastName**, **ChildFirstName**, and **BirthDate**.

31. Run the query and take a moment to observe the results.

    Now you will add a condition.

32. Switch to **Design View**, type **>January 1, 2010** in the **BirthDate Criteria** field, and tap ⎡Enter⎤.

    Once again Access will apply formatting to the criterion.

33. Run the query.

    Only records where the child was born after January 1, 2010, should be displayed.

34. Choose **File→Save** or click the **Save** button on the Quick Access toolbar and save the query as: **Younger Children**

35. Close the query and then close the database.

REINFORCE YOUR SKILLS: A3-R2

Limit the Records Returned and Use Calculated Fields

Kids for Change is planning to fine-tune its database further by adding queries that will produce calculated results. You are in charge of the IT department. In this exercise, you will generate the desired query results.

1. Open **A3-R2-K4C** from your **Access Chapter 3** folder and save it as: **A3-R2-K4CRev**

2. Run (open) the **Children List** query.

    The query returns the records of all children in the database in alphabetical order by last name.

3. Switch to **Design View** and choose **Query Tools→Design→Query Setup→ Return** 🔢 **menu button** ▾.

4. Choose **5** from the list.

5. Click in the **Sort** cell for the BirthDate field and choose **Descending**.

6. Run the query.

    Only the records for the five youngest children should be displayed.

7. Close the query, saving the changes.

Add a Calculated Field and Format the Field

As part of its community give-back policy, Kids for Change puts 10% of all donations into a scholarship fund. Now you will add a field that calculates 10% of each donation.

8. Run the **Donations Query** query and take a moment to observe the results.

9. Switch to **Design View** and use the scrollbar at the bottom of the grid to scroll the query grid to the right until the first empty column is visible.

   You will enter a calculated field in this column.

10. Type **ScholarFund:Amount*.1** in the first cell (the Field cell) of the empty column, being sure to include the colon between *ScholarFund* and *Amount*.

11. Tap Enter to complete the calculated field, and if necessary, widen the column so you can see the entire calculated field.

12. If the Property Sheet is not open, right-click anywhere in your calculated field column and choose **Properties**.

13. Click in the **Format** field and choose **Currency** from the drop-down list.

14. Type **Scholar Fund** in the Caption field.

15. Run the query and take a moment to ensure that the calculated field is calculating correctly and is formatted with the Currency format.

16. Close the query, saving the changes, and then close the database.

REINFORCE YOUR SKILLS: A3-R3

Create Select Queries Using Criteria and Calculated Fields

In this exercise, you will help Kids for Change further develop its database by adding queries that will produce calculated and formatted results based on specific search criteria.

1. Open **A3-R3-K4C** from your **Access Chapter 3** folder and save it as: **A3-R3-K4CRev**

2. Use the **Query Wizard** to create a simple query using the **Donors** table and the **DonorLName**, **DonorFName**, **DonorPhone**, and **DonorEmail** fields.

3. Use **Donor Contact List** as the query name and finish the query.

4. Review the query results and then close the query.

5. Using **Query Design**, create a new query with the **Staff** table and the **StaffLastName, StaffFirstName**, **StaffStreet**, **StaffCity**, **StaffST**, and **StaffZIP** fields.

6. Save the query as: **Staff Mailing List**

7. Run the query, review the results, and then close the query.

Create a Multi-Table Query

8. Create a new query using **Query Design** view and add the **Activities** and **Children** tables to the design grid.

9. Move the fields from the indicated tables to the query design grid:

From This Table	Add These Fields
Activities	• **Activity**
	• **Day**
	• **MeetTime**
Children	• **ChildLastName**
	• **ChildFirstName**
	• **ChildPhone**

10. Save the query as: **Participant List**

11. Run the query and review the results.

Add Criteria Including Wildcards and Dates

Now you will add criteria to the Participant List query to list the children signed up for 9:00 AM Saturday activities.

12. Switch to **Design View**.

13. Create an AND condition by setting **Saturday** as a criterion in the Day field and **9:00** as a criterion in the MeetTime field.

14. Run the query.

The only records returned are those where the day is Saturday AND the meet time is 9:00.

15. Close the query, saving the changes.

Now you will use a wildcard to select nearby donors so they can be invited to local activities.

16. Right-click the **Donations Query** query in the Navigation pane and choose **Design View**.

17. If necessary, scroll right through the field list until you locate the DonorZIP field.

18. Enter **34*** in the DonorZIP Criteria field.

The asterisk is a wildcard character.

19. Run the query.

Only records where the ZIP code begins with 34 are returned by the query.

20. Switch to **Design View** and remove the criteria from the DonorZIP field.

21. Enter **>01/01/2018** in the DonationDate Criteria field.

22. Run the query and review the results.

Sort and Limit Query Results

23. Switch to **Design View**.

24. Set the DonationDate field to sort in **Descending** order.

25. Use the **Query Tools→Design→Query Setup→Return** 🔢 **menu button** ▾ list to limit the records returned to **5**.

26. Run the query and review the results.

Add a Calculated Field and Format the Field

27. Switch to **Design View** and set the Return number back to **All**.

28. Create a calculated field by entering `NetAmt:Amount-ScholarFund` in the first empty column's Field row.

29. Right-click in the new calculated field column and open the Property Sheet.

30. Set the Format to Currency and type `Net Donation` as the Caption.

31. Run the query and review the results.

32. Close the query, saving the changes, and then close the database.

✎ Apply Your Skills

Create Queries Using Criteria and Wildcards

The new CEO of Universal Corporate Events has asked you to refine a number of queries to be more selective in data output. In this exercise, you will create queries; add criteria, wildcards, and AND/OR conditions to a query; and add date criteria to a query.

1. Open **A3-A1-UCE** from your **Access Chapter 3** folder and save it as: **A3-A1-UCERev**
2. Use the **Query Wizard** and this table to create a simple select query:

Table to Use	Fields to Add	Query Name
Personnel	• **PerLastName**	**Personnel Contact List**
	• **PerFirstName**	
	• **PerPhone**	
	• **PerEmail**	

3. Review the results and then close the query.
4. Create a query in **Design View** that uses the tables and fields indicated:

From This Table	Use These Fields
Events	• **EventName**
Schedules	• **VenueID**
	• **ContactID**
	• **EventDate**
	• **Guests**
Menus	• **MenuPlan**
	• **Chg/PP**

5. Run the query and review the results.
6. Save the query as **Event List** and then close it.

Use Wildcards and AND/OR Criteria

UCE is planning a recruiting event in Sarasota and would like to contact employees from greater Sarasota (area code 941) to involve them in the planning. You will modify a query to return the records of personnel who live in the Sarasota area.

7. Open the **Personnel Contact List** query in **Design View**.
8. Enter the wildcard text ***941*** in the PerPhone Criteria field.
9. Run the query and verify that each telephone number in the query results contains *941* somewhere in the number.
10. Close the query, saving the changes.

11. Create a new query in **Design View** from the **Venues** table that includes the **VenueName**, **VenueCity**, **VenuePhone**, and **VenueWebSite** fields.

12. Enter **Sarasota** in the VenueCity Criteria field and **Tampa** in the Or row of the VenueCity field.

13. Run the query and verify that the city is *Sarasota* or *Tampa* in each record.

14. Save the query as **Tampa-Sarasota Venues** and then close it.

Add Date Criteria

15. Run the **Event List** query and notice the range of dates.

16. Switch to **Design View** and type **>May 1, 2019** in the EventDate Criteria field.

17. Sort the query in **Ascending** order on the **EventDate** field.

18. Run the query and make sure it produces the intended results.

19. Close the query, saving the changes, and then close the database.

APPLY YOUR SKILLS: A3-A2

Limit the Records Returned and Use Calculated Fields

You've been asked to improve Universal Corporate Events' data retrieval and formatting. In this exercise, you will sort and limit records returned in query results and create a query using a calculated field.

1. Open **A3-A2-UCE** from your **Access Chapter 3** folder and save it as: **A3-A2-UCERev**

2. Run the **Event Revenue** query and review the results.

3. Switch to **Design View** and set the sort order of the TotalRev calculated field to **Descending**.

4. Set the Return number to **5** to limit the number of records returned by the query to the top five.

5. Run the query and review the results.

6. Switch to **Design View** and change the Return value back to **All**.

Add a Calculated Field and Format the Field

7. Create a new calculated field using the name and the expression: **Comm:TotalRev*.08**

8. Open the Property Sheet for the new calculated field and set the Format to **Currency** and use **Commission** as the Caption.

9. Run the query and review the results.

10. Close the query, saving the changes, and then close the database.

ACCESS

APPLY YOUR SKILLS: A3-A3

Create Select Queries Using Criteria and Calculated Fields

In this exercise, you will create and modify a number of queries for more precise, targeted data selection for Universal Corporate Events.

1. Open **A3-A3-UCE** from your **Access Chapter 3** folder and save it as: **A3-A3-UCERev**

   To begin, you will create a query to list contact information for the event venues that have an 800 telephone number so they can be reached by phone at no charge to the caller.

2. Create a simple query named **TollFreeVenues** that uses the **Venues** table to generate a list of venue names and their corresponding phone numbers and websites.

3. In **Design View**, add the wildcard text ***800*** to the Criteria row to return only records for which the venue phone number includes *800*.

4. Run the query and resize the columns in the query results so all data is visible.

5. Close the query, saving the changes.

Add Wildcard and Date Criteria and Sort the Query

Because June is the most popular month for weddings, UCE wants to pay special attention to weddings scheduled for June so they can hire extra part-time workers.

6. Using the **Query Wizard**, create a simple query that uses all fields from the Event List query.

7. Leave the Wizard's Detail or Summary option set to **Detail**.

8. Name the query **June Weddings** and finish the query.

9. Switch to **Design View**.

10. Add the wildcard text **Wed*** (for *Weddings*) to the EventName Criteria field.

11. Set the sort order of the EventDate field to **Ascending**.

12. In the EventDate Criteria field, enter: **Between June 1, 2019 And June 30, 2019**

13. Run the query and review the results.

14. Close the query, saving the changes.

Limit the Number of Records in Query Results

Now you will sort the Location Scheduling query by the largest number of guests and return the ten highest values so the company can focus extra personnel and resources to those events if the guests are scheduled for a full menu plan.

15. Display the **Location Scheduling** query in **Design View**.

16. Sort the query in **Descending** order by **Guests**.

17. Set the number of records returned to: **10**

   You'll need to click in the Return cell and type 10. If Access changes the 10 to 100, delete the extra zero (0).

18. Run the query and review the results.

19. Close the query, saving the changes.

Add and Format Calculated Fields

Now you will add a calculated field that subtracts the venue contact's commission from the total revenue to result in a net revenue amount.

20. Display the **Event Revenue** query in **Design View**.

21. Add a calculated field named **NetRev** that subtracts Comm from TotalRev.

22. Format the new field as **Currency** and set the Caption as: **Net Revenue**

23. Add a criterion to the TotalRev field to choose only records where the TotalRev is greater than 3000.

24. Run the query and review the results.

25. Close the query, saving the changes, and then close the database.

ACCESS

Apply Your Skills **95**

📑 Project Grader

If your class is using eLab (labyrinthelab.com), you may upload your completed Project Grader assignments for automatic grading. You may complete these projects even if your class doesn't use eLab, though you will not be able to upload your work.

PROJECT GRADER: A3-P1

Taylor Games: Creating Queries

Taylor Games wants to evaluate where levels of inventory are too high. You will use the inventory data to create several queries, leverage wildcard characters, and add a calculated field.

1. Download and open your Project Grader starting file.
   - *Using eLab:* Download **A3_P1_eStart** from the Assignments page. You *must* start with this file or your work cannot be automatically graded.
   - *Not using eLab:* Open **A3_P1_Start** from your **Access Chapter 3** folder.

2. Use **Query Design** to create a simple query using the following guidelines:
   - Add all fields from the **Inventory** table in the same sequence as they appear in the table.
   - Set the criteria to include all records where (Quantity is greater than 75) AND (Total Cost is greater than 250).
   - Save the query as **Overstock** and run it.

Item	SKU	Quantity	Cost	Total Cost
Vinyl 3-hole card holders	5273359	621	$0.76	$471.96
Replacement pieces - Monopoly	5273363	88	$3.25	$286.00

3. Add criteria to the Overstock query so that the query logic now becomes (Quantity is greater than 75 AND Total Cost is greater than 250) OR (Item contains the word *dice* AND Quantity is greater than 75) and then run the query.

Item	SKU	Quantity	Cost	Total Cost
Vinyl 3-hole card holders	5273359	621	$0.76	$471.96
Replacement pieces - Monopoly	5273363	88	$3.25	$286.00
20-sided dice - White	5273368	247	$0.38	$93.86

4. Add a calculated field named **Overstock Qty** that subtracts 75 from the Quantity field in each record.

5. Sort the query results on the **Overstock Qty** field in descending order.

Item	SKU	Quantity	Cost	Total Cost	Overstock
Vinyl 3-hole card holders	5273359	621	$0.76	$471.96	546
20-sided dice - White	5273368	247	$0.38	$93.86	172
Replacement pieces - Monopoly	5273363	88	$3.25	$286.00	13

6. Save and close the query.

7. Save your database.
   - *Using eLab:* Save it to your **Access Chapter 3** folder as **A3_P1_eSubmission** and attach the file to your eLab assignment for grading.
   - *Not using eLab:* Save it to your **Access Chapter 3** folder as: **A3_P1_Submission**

PROJECT GRADER: A3-P2

WebVision: Querying a Database

WebVision would like to create a monthly query that will calculate the Sales Rep commissions for each order. You will use the data in multiple tables to create a select query and add a calculated field.

1. Download and open your Project Grader starting file.
   - *Using eLab:* Download **A3_P2_eStart** from the Assignments page. You *must* start with this file or your work cannot be automatically graded.
   - *Not using eLab:* Open **A3_P2_Start** from your **Access Chapter 3** folder.

2. Use the **Query Wizard** to create a simple query using the following guidelines:
   - Add the **RepID**, **LastName**, and **SalesTeam** fields from the **Sales Reps** table in that sequence.
   - Add the **OrderID**, **Date**, and **Amount** fields from the **Orders** table in that sequence.
   - Use the **Detail** option.
   - Name the query: **June Commissions**

3. Set the Date criteria to include records between **6/1/2019 And 6/30/2019**.

4. Sort the query results on the **RepID** field in **Ascending** order.

5. Add a calculated field named **Commissions** that multiplies the Amount field in each record by: **0.02**

6. Apply the **Currency** number format to the Commissions field and then run, save, and close the query.

Rep ID	Last Name	Sales Team	Order Numb	Date	Amount	Commission
S101	Franks	North	9	6/7/2019	$38,024	$760.48
S101	Franks	North	5	6/15/2019	$29,382	$587.64
S102	Edmunds	Central	10	6/3/2019	$62,569	$1,251.38
S102	Edmunds	Central	6	6/14/2019	$52,063	$1,041.26
S102	Edmunds	Central	2	6/23/2019	$60,093	$1,201.86
S103	Berry	West	11	6/2/2019	$36,759	$735.18
S103	Berry	West	7	6/11/2019	$46,146	$922.92
S103	Berry	West	3	6/20/2019	$53,933	$1,078.66
S104	Lifestone	South	8	6/8/2019	$35,249	$704.98
S104	Lifestone	South	4	6/19/2019	$63,958	$1,279.16

7. Save your database.
   - *Using eLab:* Save it to your **Access Chapter 3** folder as **A3_P2_eSubmission** and attach the file to your eLab assignment for grading.
   - *Not using eLab:* Save it to your **Access Chapter 3** folder as: **A3_P2_Submission**

ACCESS

◢ Extend Your Skills

These exercises challenge you to think critically and apply your new skills in a real-world setting. You will be evaluated on your ability to follow directions, completeness, creativity, and the use of proper grammar and mechanics. Save files to your chapter folder. Submit assignments as directed.

A3-E1   That's the Way I See It

Blue Jean Landscaping needs queries to better manage its customer and equipment lists and you've volunteered to assist. Open **A3-E1-BJL** and save it as: **A3-E1-BJLRev**

Create a query named **813 Area Code** that uses all fields from the Customers table. The query should return only customers with a phone area code of 813 sorted in ascending order by city. Create another query named **Equipment Value** that uses all fields from the Equipment table. Use a calculated field named **EquipValue** to determine the total value of equipment by multiplying the Cost by the quantity In Stock. Sort the results with the largest Equipment Values appearing first and format the EquipValue field using the Currency format.

A3-E2   Be Your Own Boss

Blue Jean Landscaping wants to devise more targeted data retrieval. Open **A3-E2-BJL** and save it as: **A3-E2-BJLRev**

Create a query that returns a contact list for BJL's customers sorted by last name. Create another query that creates a customer mailing list sorted by ZIP code. Use a wildcard to select only records where the ZIP code begins with 33. Add a calculated field to the Sales Invoices query that multiplies Cost by Qty Sold to produce a total. Format the new field as Currency and assign it a caption. Finally, limit the number of records returned to the largest five invoice totals, so those customers can be targeted for preferred customer offers.

A3-E3   Demonstrate Proficiency

You've been asked by the management at Stormy BBQ to query its database. Open **A3-E3-StormyBBQ** and save it as: **A3-E3-StormyBBQRev**

Create a query that uses data from the DailyReceipts table and determines the total revenue received for each item using the ItemPrice and QtySold fields. Include all fields from the table in the query and sort in descending order on the field that is used to perform the daily total calculations. Create another query using the Merchandise table that contains all fields from the Merchandise table and a sequence of calculated fields. For each item, the calculated fields should determine the Stock Cost of that item (Cost * Stock), the List Price Revenue if all items were sold at list price (Listprice * Stock), and the Profit, which is the difference between the revenue and cost.