Hands-On Exercises

Watch the Video for this Hands-On Exercise!

MyITLab® HOE1 Training

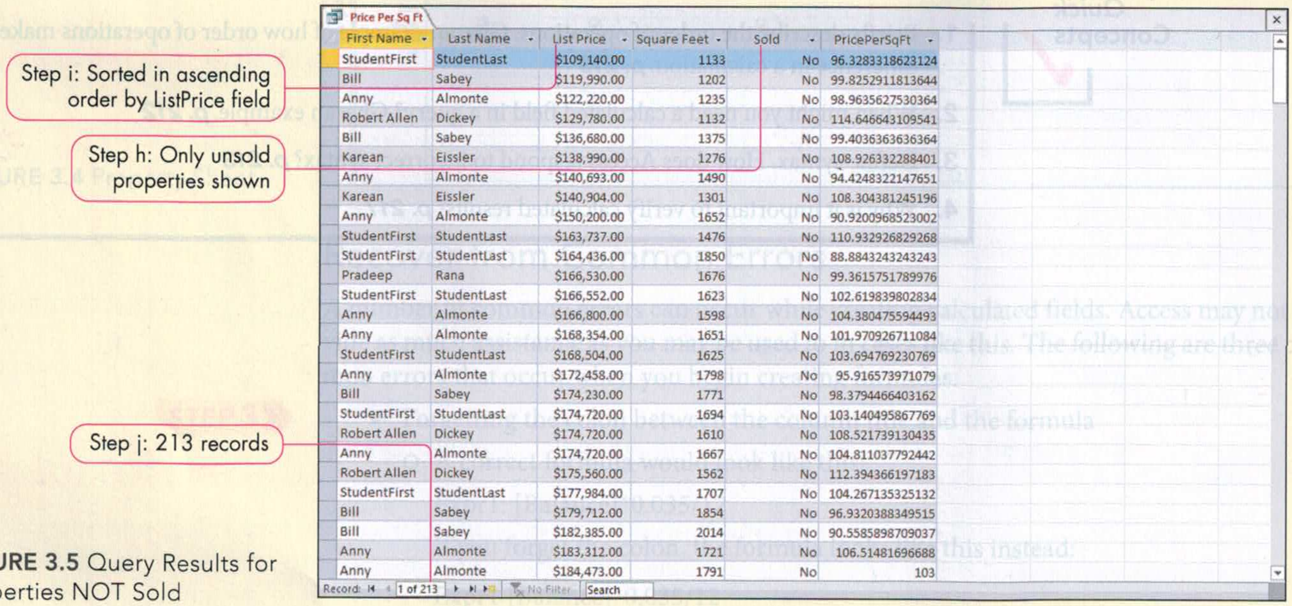
1 Calculations and Expressions

Using the data from the homes for sale lists that Don and Matt acquired, you are able to help them target properties that meet their criteria. As you examine the data, you discover other ways to analyze the properties. You create several queries and present your results to the two investors for their comments.

Skills covered: Create a Query with a Calculated Field • Format and Save Calculated Results • Recover from Common Errors • Verify the Calculated Results

STEP 1 »» CREATE A QUERY WITH A CALCULATED FIELD

You begin your analysis by creating a query using the Properties and Agents tables. The Properties table contains all the properties the investors will evaluate; the Agents table contains a list of real estate agents who represent the properties' sellers. In this exercise, add the fields you need and only show properties that have not been sold. Refer to Figure 3.5 as you complete Step 1.



Price Per Sq Ft

First Name	Last Name	List Price	Square Feet	Sold	PricePerSqFt
StudentFirst	StudentLast	$109,140.00	1133	No	96.3283318623124
Bill	Sabey	$119,990.00	1202	No	99.8252911813644
Anny	Almonte	$122,220.00	1235	No	98.9635627530364
Robert Allen	Dickey	$129,780.00	1132	No	114.646643109541
Bill	Sabey	$136,680.00	1375	No	99.4036363636364
Karean	Eissler	$138,990.00	1276	No	108.926332288401
Anny	Almonte	$140,693.00	1490	No	94.4248322147651
Karean	Eissler	$140,904.00	1301	No	108.304381245196
Anny	Almonte	$150,200.00	1652	No	90.9200968523002
StudentFirst	StudentLast	$163,737.00	1476	No	110.932926829268
StudentFirst	StudentLast	$164,436.00	1850	No	88.8843243243243
Pradeep	Rana	$166,530.00	1676	No	99.3615751789976
StudentFirst	StudentLast	$166,552.00	1623	No	102.619839802834
Anny	Almonte	$166,800.00	1598	No	104.380475594493
Anny	Almonte	$168,354.00	1651	No	101.970926711084
StudentFirst	StudentLast	$168,504.00	1625	No	103.694769230769
Anny	Almonte	$172,458.00	1798	No	95.916573971079
Bill	Sabey	$174,230.00	1771	No	98.3794466403162
StudentFirst	StudentLast	$174,720.00	1694	No	103.140495867769
Robert Allen	Dickey	$174,720.00	1610	No	108.521739130435
Anny	Almonte	$174,720.00	1667	No	104.811037792442
Robert Allen	Dickey	$175,560.00	1562	No	112.394366197183
StudentFirst	StudentLast	$177,984.00	1707	No	104.267135325132
Bill	Sabey	$179,712.00	1854	No	96.9320388349515
Bill	Sabey	$182,385.00	2014	No	90.5585898709037
Anny	Almonte	$183,312.00	1721	No	106.51481696688
Anny	Almonte	$184,473.00	1791	No	103

Record: 1 of 213 No Filter Search

Step i: Sorted in ascending order by ListPrice field

Step h: Only unsold properties shown

Step j: 213 records

FIGURE 3.5 Query Results for Properties NOT Sold

a. Open *a03h1Property*. Save the database as **a03h1Property_LastFirst**.

TROUBLESHOOTING: Throughout the remainder of this chapter and textbook, click Enable Content whenever you are working with student files.

TROUBLESHOOTING: If you make any major mistakes in this exercise, you can delete the *a03h1Property_LastFirst* file, repeat step a above, and then start this exercise over.

b. Open the Agents table and replace *Angela Scott* with your name. Close the table.

c. Click the **CREATE tab** and click **Query Design** in the Queries group to create a new query.

The Show Table dialog box opens so you can specify the table(s) and/or queries to include in the query design.

d. Select the **Agents table** and click **Add**. Select the **Properties table** and click **Add**. Close the Show Table dialog box.

e. Double-click the **FirstName** and **LastName fields** in the Agents table to add them to the design grid.

f. Double-click the **ListPrice**, **SqFeet**, and **Sold fields** in the Properties table to add them to the query design grid.

g. Click **Run** in the Results group to display the results in Datasheet view.

You should see 303 properties in the results.

h. Switch to Design view. Type **No** in the Criteria row of the Sold field.

i. Select **Ascending** from the Sort row of the ListPrice field.

j. Click **Run** to see the results.

You only want to see properties that were not sold. There should now be 213 properties in the datasheet.

k. Click **Save** on the Quick Access Toolbar and type **Price Per Sq Ft** as the Query Name in the Save As dialog box. Click **OK**.

l. Switch to Design view. Click in the top row of the first blank column of the query design grid and use **Shift+F2** to show the Zoom dialog box. Type **PricePerSqFt: ListPrice/SqFeet** and click **OK**.

Access inserts square brackets around the fields for you. The new field divides the values in the ListPrice field by the values in the SqFeet field. The : after *PricePerSqFt* is required.

m. Click **Run** in the Results group to view the results.

The new calculated field, PricePerSqFt, is displayed. Compare your results to those shown in Figure 3.5.

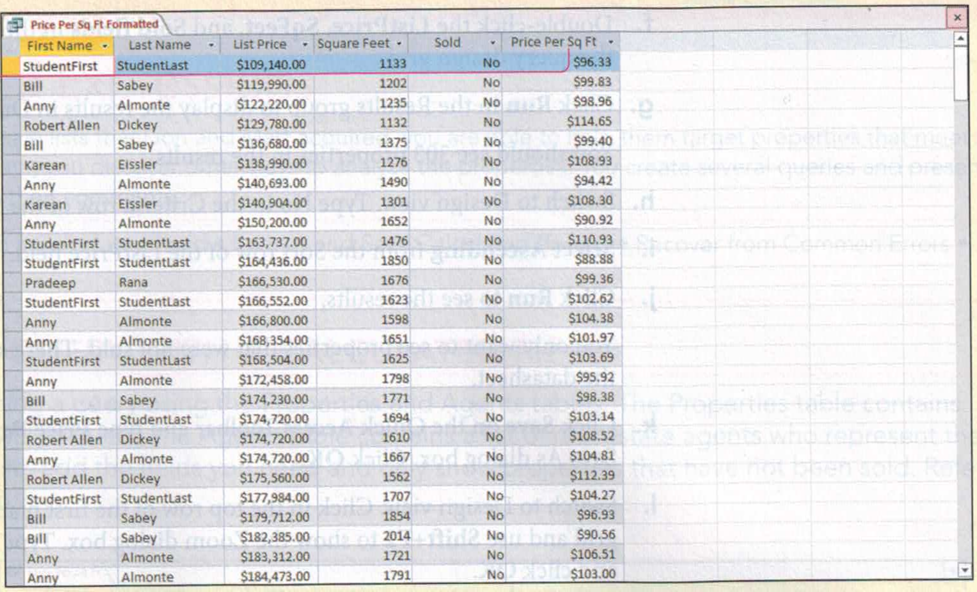
TROUBLESHOOTING: If you see pound signs (#####) in an Access column, use the vertical lines between column indicators to increase the width.

TROUBLESHOOTING: If, when you run the query, you are prompted for PricePerSqFt, cancel and return to Design view. Ensure you have entered the formula from step l to the field line of the query, not the criteria line.

n. Save the changes to the query and close the query.

STEP 2 ❯❯ FORMAT AND SAVE CALCULATED RESULTS

Don and Matt would like the field formatted differently. You will change the format to Currency and add a caption to the calculated field. Refer to Figure 3.6 as you complete Step 2.



Steps d–e: PricePerSqFt field formatted as Currency with added caption

Price Per Sq Ft Formatted

First Name	Last Name	List Price	Square Feet	Sold	Price Per Sq Ft
StudentFirst	StudentLast	$109,140.00	1133	No	$96.33
Bill	Sabey	$119,990.00	1202	No	$99.83
Anny	Almonte	$122,220.00	1235	No	$98.96
Robert Allen	Dickey	$129,780.00	1132	No	$114.65
Bill	Sabey	$136,680.00	1375	No	$99.40
Karean	Eissler	$138,990.00	1276	No	$108.93
Anny	Almonte	$140,693.00	1490	No	$94.42
Karean	Eissler	$140,904.00	1301	No	$108.30
Anny	Almonte	$150,200.00	1652	No	$90.92
StudentFirst	StudentLast	$163,737.00	1476	No	$110.93
StudentFirst	StudentLast	$164,436.00	1850	No	$88.88
Pradeep	Rana	$166,530.00	1676	No	$99.36
StudentFirst	StudentLast	$166,552.00	1623	No	$102.62
Anny	Almonte	$166,800.00	1598	No	$104.38
Anny	Almonte	$168,354.00	1651	No	$101.97
StudentFirst	StudentLast	$168,504.00	1625	No	$103.69
Anny	Almonte	$172,458.00	1798	No	$95.92
Bill	Sabey	$174,230.00	1771	No	$98.38
StudentFirst	StudentLast	$174,720.00	1694	No	$103.14
Robert Allen	Dickey	$174,720.00	1610	No	$108.52
Anny	Almonte	$174,720.00	1667	No	$104.81
Robert Allen	Dickey	$175,560.00	1562	No	$112.39
StudentFirst	StudentLast	$177,984.00	1707	No	$104.27
Bill	Sabey	$179,712.00	1854	No	$96.93
Bill	Sabey	$182,385.00	2014	No	$90.56
Anny	Almonte	$183,312.00	1721	No	$106.51
Anny	Almonte	$184,473.00	1791	No	$103.00

FIGURE 3.6 Results of Calculated Field Creation

a. Make a copy of the Price Per Sq Ft query. Name the copy **Price Per Sq Ft Formatted**.

b. Open the Price Per Sq Ft Formatted query in Design view.

c. Click in the **PricePerSqFt calculated field cell**. Click **Property Sheet** in the Show/Hide group on the DESIGN tab.

The Property Sheet displays on the right side of your screen.

d. Click the **Format box**. Click the **Format property arrow** and select **Currency**.

e. Click in the **Caption box** and type **Price Per Sq Ft**. Press **Enter**. Close the Property Sheet.

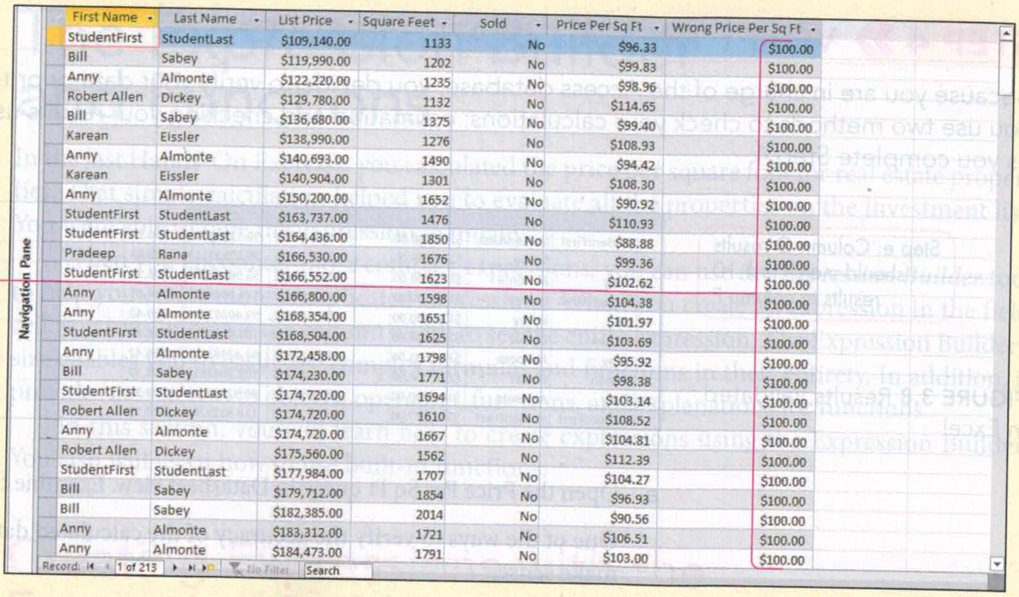
f. Click **Run** to view your changes.

The calculated field values are formatted as Currency, and the column heading displays *Price Per Sq Ft* instead of *PricePerSqFt*.

g. Save the changes to the query.

STEP 3 ❯❯ RECOVER FROM COMMON ERRORS

A few errors arise as you test the new calculated fields. You check the spelling of the field names in the calculated fields because that is a common mistake. Refer to Figure 3.7 as you complete Step 3.



Step g: Same results ($100.00) for every record

FIGURE 3.7 Results of a Misspelled Field Name

First Name	Last Name	List Price	Square Feet	Sold	Price Per Sq Ft	Wrong Price Per Sq Ft
StudentFirst	StudentLast	$109,140.00	1133	No	$96.33	$100.00
Bill	Sabey	$119,990.00	1202	No	$99.83	$100.00
Anny	Almonte	$122,220.00	1235	No	$98.96	$100.00
Robert Allen	Dickey	$129,780.00	1132	No	$114.65	$100.00
Bill	Sabey	$136,680.00	1375	No	$99.40	$100.00
Karean	Eissler	$138,990.00	1276	No	$108.93	$100.00
Anny	Almonte	$140,693.00	1490	No	$94.42	$100.00
Karean	Eissler	$140,904.00	1301	No	$108.30	$100.00
Anny	Almonte	$150,200.00	1652	No	$90.92	$100.00
StudentFirst	StudentLast	$163,737.00	1476	No	$110.93	$100.00
StudentFirst	StudentLast	$164,436.00	1850	No	$88.88	$100.00
Pradeep	Rana	$166,530.00	1676	No	$99.36	$100.00
StudentFirst	StudentLast	$166,552.00	1623	No	$102.62	$100.00
Anny	Almonte	$166,800.00	1598	No	$104.38	$100.00
Anny	Almonte	$168,354.00	1651	No	$101.97	$100.00
StudentFirst	StudentLast	$168,504.00	1625	No	$103.69	$100.00
Anny	Almonte	$172,458.00	1798	No	$95.92	$100.00
Bill	Sabey	$174,230.00	1771	No	$98.38	$100.00
StudentFirst	StudentLast	$174,720.00	1694	No	$103.14	$100.00
Robert Allen	Dickey	$174,720.00	1610	No	$108.52	$100.00
Anny	Almonte	$174,720.00	1667	No	$104.81	$100.00
Robert Allen	Dickey	$175,560.00	1562	No	$112.39	$100.00
StudentFirst	StudentLast	$177,984.00	1707	No	$104.27	$100.00
Bill	Sabey	$179,712.00	1854	No	$96.93	$100.00
Bill	Sabey	$182,385.00	2014	No	$90.56	$100.00
Anny	Almonte	$183,312.00	1721	No	$106.51	$100.00
Anny	Almonte	$184,473.00	1791	No	$103.00	$100.00

a. Switch to Design view of the Price Per Sq Ft Formatted query. Scroll to the first blank column of the query design grid and click in the top row.

b. Use **Shift+F2** to display the Zoom dialog box. Type **WrongPricePerSqFt: xListPrice/xSqFeet.** Click the **OK button** in the Zoom dialog box.

Be sure that you added the extra *x*'s to the field names. You are intentionally misspelling the field names to see how Access will respond. Access inserts square brackets around the field names for you.

c. Click **Property Sheet** in the Show/Hide group of the DESIGN tab. Click the **Format box.** From the menu, select **Currency.** Click in the **Caption box** and type **Wrong Price Per Sq Ft.** Close the Property Sheet.

d. Click **Run** in the Results group.

You should see the Enter Parameter Value dialog box. The dialog box indicates that Access does not recognize xListPrice in the tables defined for this query in the first record. When Access does not recognize a field name, it will ask you to supply a value.

e. Type **100000** in the first parameter box. Press **Enter** or click **OK.**

Another Enter Parameter Value dialog box displays, asking that you supply a value for xSqFeet. Again, this error occurs because the tables defined for this query do not contain an xSqFeet field.

f. Type **1000** in the second parameter box and press **Enter.**

The query has the necessary information to run and returns the results in Datasheet view.

g. Examine the results of the calculation for *Wrong Price Per Sq Ft.* You may have to scroll right to see the results.

All of the records show 100 because you entered the values 100000 and 1000, respectively, into the parameter boxes. The two values are treated as constants and give the same results for all the records.

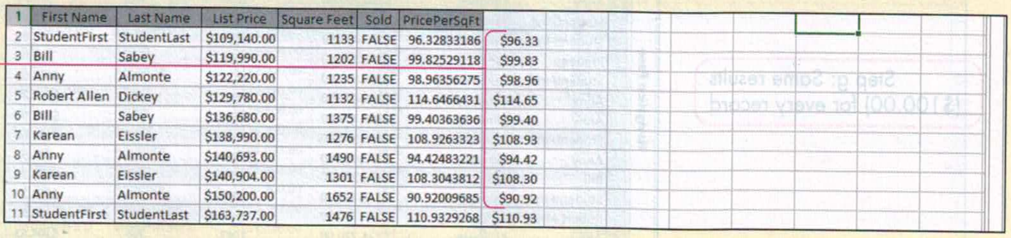
h. Return to Design view. Press **Shift+F2** to zoom. Correct the errors in the WrongPricePerSqFt field by changing the formula to **WrongPricePerSqFt: [ListPrice]/[SqFeet].** Click the **Close (X) button** in the top-right corner of the Zoom dialog box to close it.

i. Run and save the query. Close the query.

The calculated values in the last two columns should be the same.

STEP 4 » VERIFY THE CALCULATED RESULTS

Because you are in charge of the Access database, you decide to verify your data prior to showing it to the investors. You use two methods to check your calculations: estimation and checking your results using Excel. Refer to Figure 3.8 as you complete Step 4.



1	First Name	Last Name	List Price	Square Feet	Sold	PricePerSqFt	
2	StudentFirst	StudentLast	$109,140.00	1133	FALSE	96.32833186	$96.33
3	Bill	Sabey	$119,990.00	1202	FALSE	99.82529118	$99.83
4	Anny	Almonte	$122,220.00	1235	FALSE	98.96356275	$98.96
5	Robert Allen	Dickey	$129,780.00	1132	FALSE	114.6466431	$114.65
6	Bill	Sabey	$136,680.00	1375	FALSE	99.40363636	$99.40
7	Karean	Eissler	$138,990.00	1276	FALSE	108.9263323	$108.93
8	Anny	Almonte	$140,693.00	1490	FALSE	94.42483221	$94.42
9	Karean	Eissler	$140,904.00	1301	FALSE	108.3043812	$108.30
10	Anny	Almonte	$150,200.00	1652	FALSE	90.92009685	$90.92
11	StudentFirst	StudentLast	$163,737.00	1476	FALSE	110.9329268	$110.93

Step e: Column G results should match first 10 results in column F

FIGURE 3.8 Results Validated in Excel

a. Open the Price Per Sq Ft query in Datasheet view. Examine the PricePerSqFt field.

One of the ways to verify the accuracy of the calculated data is to ask yourself if the numbers make sense.

b. Locate the second record with *Bill Sabey* as the listing agent, an asking price of *$119,990*, and square footage of *1202*. Ask yourself if the calculated value of *$99.83* makes sense.

The sale price is $119,990, and the square footage is 1202. You can verify the calculated field easily by rounding the two numbers (to 120,000 and 1,200) and dividing the values in your head (120,000 divided by 1,200 = 100) to verify that the calculated value, $99.83 per square foot, makes sense.

TROUBLESHOOTING: If the second record is not the one listed above, ensure you have sorted the query by the List Price in ascending order, as specified in Step 1i.

c. Open a new, blank workbook in Excel and switch to Access. Drag over the record selector for the first 10 records (the tenth record has a list price of $163,737). Click **Copy** in the Clipboard group on the HOME tab.

You will verify the calculation in the first 10 records by pasting the results in Excel.

d. Switch to Excel and, click **Paste** in the Clipboard group on the HOME tab.

The field names display in the first row, and the 10 records display in the next 10 rows. The fields are located in columns A–F. The calculated field results are pasted in column F as values rather than as a formula.

TROUBLESHOOTING: If you see pound signs (#####) in an Excel column, use the vertical lines between column indicators to increase the width.

e. Type **=C2/D2** in **cell G2** and press **Enter**. Copy the formula from **cell G2** and paste it into **cells G3 to G11**.

The formula divides the list price by the square feet. Compare the results in columns F and G. The numbers should be the same, except for a slight difference due to rounding.

TROUBLESHOOTING: If the values differ, look at both the Excel and Access formulas. Determine which is correct, and then find and fix the error in the incorrect formula.

f. Save the Excel workbook as **a03Property_LastFirst**. Exit Excel.

g. Keep the database open if you plan to continue with the next Hands-On Exercise. If not, close the database and exit Access.

Hands-On Exercises

Watch the Video for this Hands-On Exercise!

MyITLab® HOE2 Training

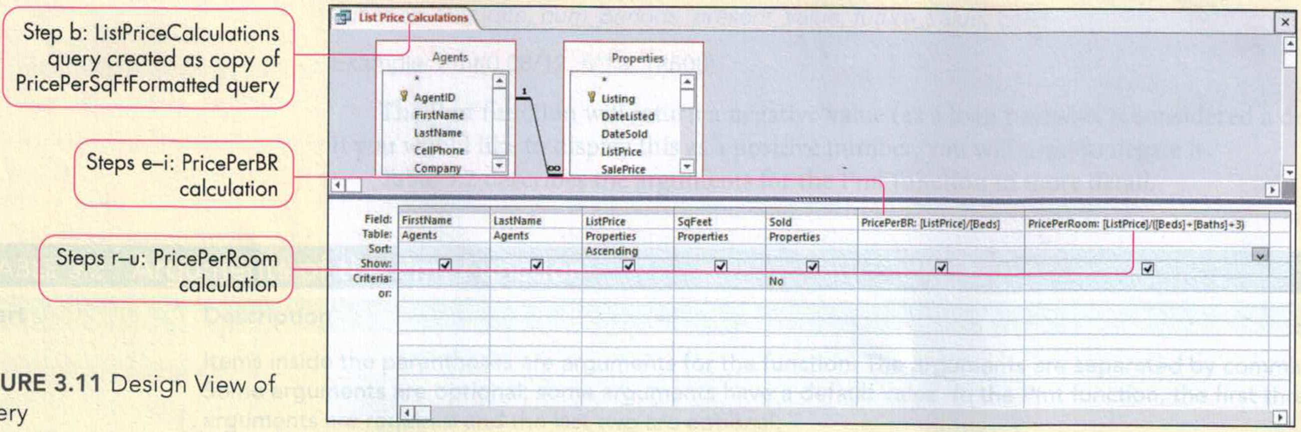
2 The Expression Builder and Functions

When Don and Matt ask you to calculate the price per bedroom and the price per room for each property, you use the Expression Builder to make the task easier. You also add two additional fields that calculate the days on market and the estimated commission for each property.

Skills covered: Use the Expression Builder to Add and Modify a Field • Use Built-In Functions

STEP 1 » USE THE EXPRESSION BUILDER TO ADD AND MODIFY A FIELD

You create a copy of the Price Per Sq Ft Formatted query from the previous Hands-On Exercise and paste it using a new name. You will add a few more calculated fields to the new query. You will create one calculation to determine the price per bedroom for each house. You will create a second field to calculate the price per room. For this calculation, you will assume that each property has a kitchen, a living room, a dining room, and the listed bedrooms and bathrooms. The calculations you will create are shown in Figure 3.11. Your expected output is shown in Figure 3.12.



Step b: ListPriceCalculations query created as copy of PricePerSqFtFormatted query

Steps e–i: PricePerBR calculation

Steps r–u: PricePerRoom calculation

Field:	FirstName	LastName	ListPrice	SqFeet	Sold	PricePerBR: [ListPrice]/[Beds]	PricePerRoom: [ListPrice]/([Beds]+[Baths]+3)
Table:	Agents	Agents	Properties	Properties	Properties		
Sort:			Ascending				
Show:	☑	☑	☑	☑	☑	☑	☑
Criteria:					No		
or:							

FIGURE 3.11 Design View of Query

a. Open *a03h1Property_LastFirst* if you closed it at the end of Hands-On Exercise 1 and save it as **a03h2Property_LastFirst**, changing *h1* to *h2*.

b. Create a copy of the Price Per Sq Ft Formatted query with the name **List Price Calculations**.

The new query is displayed in the Navigation Pane. The name of the query suggests it should contain calculations based on each property's list price.

c. Open the List Price Calculations query in Design view. Click the **WrongPricePerSqFt field**. Click **Delete Columns** in the Query Setup group on the QUERY TOOLS DESIGN tab.

> **TROUBLESHOOTING:** If instead of the column being deleted, a new row named *Delete* appears on the bottom half of the screen, close the query without saving, open in Design view once more, and ensure you are clicking Delete Columns in the Query Setup group. If you click Delete under Query Type, you will get very different results.

d. Click in the top cell in the PricePerSqFt column and click **Builder** in the Query Setup group.

The Expression Builder dialog box opens, displaying the current formula.

TROUBLESHOOTING: If the Builder is gray and cannot be clicked, ensure you have clicked in the top cell of the column.

e. Change the PricePerSqFt field name to **PricePerBR**.

f. Double-click the **[SqFeet] field** in the expression and press **Delete**.

g. Click the **plus sign** (+) under Expression Elements, next to the *a03h2Property_LastFirst* database in the Expression Elements box, to expand the list. Click + next to *Tables* and click the table named **Properties**.

The fields from the Properties table are now listed in the middle column (Expression Categories).

h. Double-click the **Beds field** to add it to the expression box.

The expression now reads *PricePerBR: [ListPrice]/[Properties]![Beds]*.

i. Highlight the **[Properties]! prefix** in front of *Beds* and press **Delete**.

The expression now reads *PricePerBR: [ListPrice]/[Beds]*. As the Beds field name is unique within our query, the table name is not necessary. Removing this makes the query easier to read. If a field named Beds appeared in more than one table in our query, removing the table name would cause problems.

j. Click **OK** and click **Run** to view the query results.

Notice the column heading still reads Price Per Sq Ft. Also notice the column's contents are formatted as Currency. These settings were copied when we copied the query.

k. Switch to Design view and ensure the **PricePerBR field** is selected. Click **Property Sheet** in the Show/Hide group and change the **Caption** to **Price Per Bedroom**. Close the Property Sheet. Run the query and examine the changes.

The PricePerBR column now has an appropriate caption.

l. Switch to Design view. Select the entire **PricePerBR expression**, right-click the selected expression, and then select **Copy**. Right-click in the top cell of the next blank column and select **Paste**.

You will edit the copy so that it reflects the price per room. As stated already, you assume the kitchen, living room, dining room, and the bedrooms and bathrooms will make up the number of rooms. Your final formula would be the list price divided by the total number of rooms, which is the number of bedrooms (in the Beds field), plus the number of bathrooms (found in the Baths field), plus 3 (a constant representing the kitchen, living room, and dining room).

m. Click **Builder** in the Query Setup group.

n. Change the PricePerBR field name to **PricePerRoom**.

o. Add **parentheses** before the [Beds] portion of the formula. Type a **plus sign** (+) after *[Beds]*.

As you want the addition to be done first, the order of operations states we must enclose the addition in parentheses. The expression box should read *PricePerRoom: [ListPrice]/([Beds]+*

p. Click the **plus sign** (+) next to the *a03h2Property_LastFirst* database in the Expression Elements box to expand the list. Click the **plus sign** (+) next to *Tables* and click the **Properties table**.

The fields from the Properties table are now listed in the Expression Categories box.

q. Double-click the **Baths field** to add it to the expression box.

The expression now reads *PricePerRoom: [ListPrice]/([Beds]+[Properties]![Baths]*.

r. Type another plus sign after *[Baths]* and type **3**).

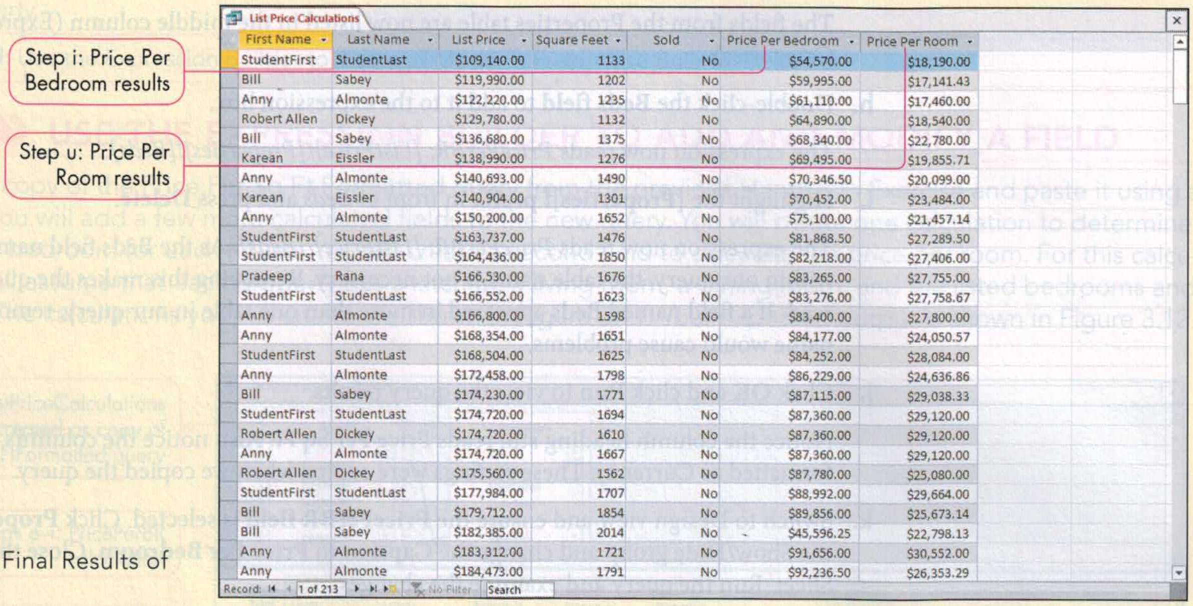
The expression now reads *PricePerRoom: [ListPrice]/([Beds]+[Properties]![Baths]+3).*

s. Delete the [Properties]! portion of the expression and click **OK** to close the Expression Builder.

The expression now reads *PricePerRoom: [ListPrice]/([Beds]+[Baths]+3).*

t. Click **Property Sheet**. Type **Price Per Room** in the **Caption box**. Close the Property Sheet.

u. Run the query. Widen the PricePerRoom column if necessary in order to see all the values.


Step i: Price Per Bedroom results
Step u: Price Per Room results
List Price Calculations
Record: 1 of 213
No Filter
Search

First Name	Last Name	List Price	Square Feet	Sold	Price Per Bedroom	Price Per Room
StudentFirst	StudentLast	$109,140.00	1133	No	$54,570.00	$18,190.00
Bill	Sabey	$119,990.00	1202	No	$59,995.00	$17,141.43
Anny	Almonte	$122,220.00	1235	No	$61,110.00	$17,460.00
Robert Allen	Dickey	$129,780.00	1132	No	$64,890.00	$18,540.00
Bill	Sabey	$136,680.00	1375	No	$68,340.00	$22,780.00
Karean	Eissler	$138,990.00	1276	No	$69,495.00	$19,855.71
Anny	Almonte	$140,693.00	1490	No	$70,346.50	$20,099.00
Karean	Eissler	$140,904.00	1301	No	$70,452.00	$23,484.00
Anny	Almonte	$150,200.00	1652	No	$75,100.00	$21,457.14
StudentFirst	StudentLast	$163,737.00	1476	No	$81,868.50	$27,289.50
StudentFirst	StudentLast	$164,436.00	1850	No	$82,218.00	$27,406.00
Pradeep	Rana	$166,530.00	1676	No	$83,265.00	$27,755.00
StudentFirst	StudentLast	$166,552.00	1623	No	$83,276.00	$27,758.67
Anny	Almonte	$166,800.00	1598	No	$83,400.00	$27,800.00
Anny	Almonte	$168,354.00	1651	No	$84,177.00	$24,050.57
StudentFirst	StudentLast	$168,504.00	1625	No	$84,252.00	$28,084.00
Anny	Almonte	$172,458.00	1798	No	$86,229.00	$24,636.86
Bill	Sabey	$174,230.00	1771	No	$87,115.00	$29,038.33
StudentFirst	StudentLast	$174,720.00	1694	No	$87,360.00	$29,120.00
Robert Allen	Dickey	$174,720.00	1610	No	$87,360.00	$29,120.00
Anny	Almonte	$174,720.00	1667	No	$87,360.00	$29,120.00
Robert Allen	Dickey	$175,560.00	1562	No	$87,780.00	$25,080.00
StudentFirst	StudentLast	$177,984.00	1707	No	$88,992.00	$29,664.00
Bill	Sabey	$179,712.00	1854	No	$89,856.00	$25,673.14
Bill	Sabey	$182,385.00	2014	No	$45,596.25	$22,798.13
Anny	Almonte	$183,312.00	1721	No	$91,656.00	$30,552.00
Anny	Almonte	$184,473.00	1791	No	$92,236.50	$26,353.29

FIGURE 3.12 Final Results of Query

v. Save and close the query.

TIP Switching Between Object Views

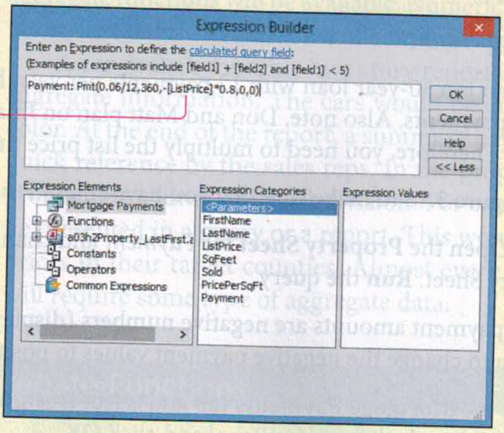
You can switch between object views quickly by clicking View, or you can click the View arrow and select the desired view from the list. Another way to switch between views is to right-click the object tab and select the view from the shortcut menu.

TIP Expression Builder and Property Sheet

You can launch the Expression Builder by either clicking Builder in the Query Setup group on the Design tab or by right-clicking in the top row of the query design grid and selecting Build. Similarly, you can display the Property Sheet by clicking Property Sheet in the Show/Hide group on the Design tab or by right-clicking the top row of the query design grid and selecting Properties from the shortcut menu.

STEP 2 » USE BUILT-IN FUNCTIONS

Don and Matt feel like they are close to making an offer on a house. They would like to restrict the query to houses that cost $150,000 or less. They would also like to calculate the estimated mortgage payment for each house. You create this calculation using the Pmt function. You will use the Pmt function to calculate an estimated house payment for each of the sold properties. You make the following assumptions: 100% of the sale price will be financed, a 30-year term, monthly payments, and a fixed 6.0% annual interest rate. Refer to Figures 3.13 and 3.14 as you complete Step 2.



Enter an Expression to define the calculated query field:
(Examples of expressions include [field1] + [field2] and [field1] < 5)

Payment: Pmt(0.06/12,360,-[ListPrice]*0.8,0,0)

Steps i–j: Pmt function as it should appear

FIGURE 3.13 Pmt Function Calculating Mortgage Costs

a. Create a copy of the Price Per Sq Ft Formatted query named **Mortgage Payments**.

   The new query is displayed in the Navigation Pane.

b. Right-click **Mortgage Payments** and select **Design View**.

c. Delete the WrongPricePerSqFt field.

   The WrongPricePerSqFt field is not needed for this query.

TROUBLESHOOTING: If you do not see the WrongPricePerSqFt field, ensure you copied the correct query.

d. Type **<=150000** in the Criteria row of the ListPrice column. Press **Enter**.

   The query, when it is run, will show only the houses that cost $150,000 or less.

e. Click in the top cell of the first blank column. Click **Builder** in the Query Setup group to open the Expression Builder dialog box.

f. Double-click **Functions** in the Expression Elements box and click **Built-In Functions**.

g. Click **Financial** in the Expression Categories box.

h. Double-click **Pmt** in the Expression Values box.

   The expression box displays:

   Pmt(«rate», «num_periods», «present_value», «future_value», «type»)

i. Position the insertion point before the Pmt function. Type **Payment:** to the left of the Pmt function.

   The expression box now displays:

   Payment:Pmt(«rate», «num_periods», «present_value», «future_value», «type»)

TROUBLESHOOTING: If you forget to add the calculated field name to the left of the expression, Access will add *Expr1* to the front of your expression for you. You can edit the Expr1 name later, after the Expression Builder is closed.

j. Click each argument to select it and substitute the appropriate information. Make sure there is a comma between each argument.

Argument	Replacement Value
«rate»	0.06/12
«num_periods»	360
«present_value»	[ListPrice]*0.8
«future_value»	0
«type»	0

Note the loan is a 30-year loan with 12 payments per year, hence the 360 value for the number of payments. Also note, Don and Matt plan on financing 80% of the cost, putting 20% down. Therefore, you need to multiply the list price times 0.8 (80%).

k. Examine Figure 3.13 to make sure that you have entered the correct arguments. Click **OK**.

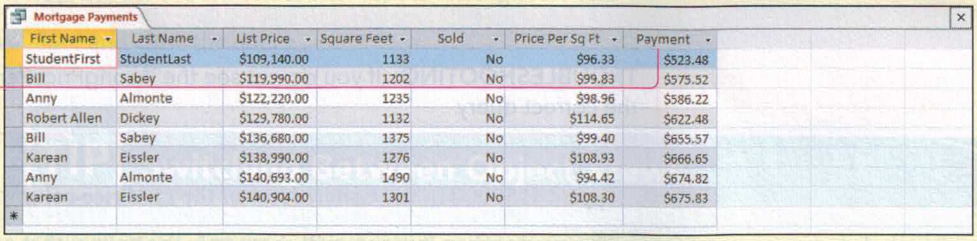
l. Click **OK**. Open the **Property Sheet** for *Payment* and change the format to **Currency**. Close the Property Sheet. **Run** the query.

Notice the payment amounts are negative numbers (displayed in parentheses). You will edit the formula to change the negative payment values to positive.

m. Right-click the **Mortgage Payments tab** and select **Design View**. Click **Builder.** Add a **minus sign** (–) to the left of *[ListPrice]* and click **OK**.

By adding the negative sign in front of the ListPrice field, you ensure the value is displayed as a positive number. The expression now reads *Payment: Pmt(0.06/12,360, –[ListPrice]*0.8,0,0)*.

The calculated field values should now appear as positive values formatted as currency, as shown in Figure 3.14.



Mortgage Payments

First Name	Last Name	List Price	Square Feet	Sold	Price Per Sq Ft	Payment
StudentFirst	StudentLast	$109,140.00	1133	No	$96.33	$523.48
Bill	Sabey	$119,990.00	1202	No	$99.83	$575.52
Anny	Almonte	$122,220.00	1235	No	$98.96	$586.22
Robert Allen	Dickey	$129,780.00	1132	No	$114.65	$622.48
Bill	Sabey	$136,680.00	1375	No	$99.40	$655.57
Karean	Eissler	$138,990.00	1276	No	$108.93	$666.65
Anny	Almonte	$140,693.00	1490	No	$94.42	$674.82
Karean	Eissler	$140,904.00	1301	No	$108.30	$675.83
*						

Steps l–n: Payment field displayed as a positive number, formatted as Currency

FIGURE 3.14 Results of Mortgage Payments Query

n. Click **OK**. Run the query and examine the results.

The query displays a column containing the calculated monthly mortgage payment, formatted as currency.

o. Save and close the query. Keep the database open if you plan to continue with the next Hands-On Exercise. If not, close the database and exit Access.

Hands-On Exercises

Watch the Video for this Hands-On Exercise!

MyITLab® HOE3 Training

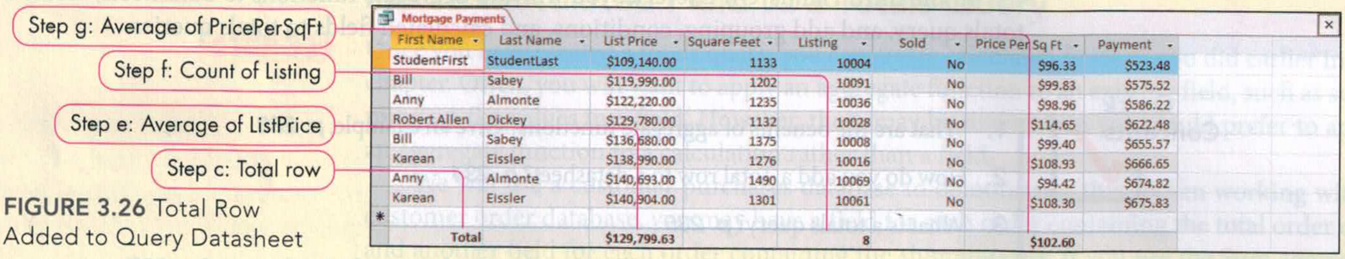
3 Aggregate Functions

The investors decide it would be helpful to analyze the property lists they purchased. Some of the lists do not have homes that match their target criteria. The investors will either need to purchase new lists or alter their criteria. You create several totals queries to evaluate the property lists.

Skills covered: Add Aggregate Functions to Datasheets • Create a Totals Query with Grouping and Conditions • Add a Calculated Field to a Totals Query

STEP 1 » ADD AGGREGATE FUNCTIONS TO DATASHEETS

You begin your property list analysis by creating a total row in the Datasheet view of the Mortgage Payments query. This will give you a variety of aggregate information for each column. Refer to Figure 3.26 as you complete Step 1.



First Name	Last Name	List Price	Square Feet	Listing	Sold	Price Per Sq Ft	Payment
StudentFirst	StudentLast	$109,140.00	1133	10004	No	$96.33	$523.48
Bill	Sabey	$119,990.00	1202	10091	No	$99.83	$575.52
Anny	Almonte	$122,220.00	1235	10036	No	$98.96	$586.22
Robert Allen	Dickey	$129,780.00	1132	10028	No	$114.65	$622.48
Bill	Sabey	$136,680.00	1375	10008	No	$99.40	$655.57
Karean	Eissler	$138,990.00	1276	10016	No	$108.93	$666.65
Anny	Almonte	$140,693.00	1490	10069	No	$94.42	$674.82
Karean	Eissler	$140,904.00	1301	10061	No	$108.30	$675.83
*							
Total		$129,799.63		8		$102.60	

FIGURE 3.26 Total Row Added to Query Datasheet

a. Open *a03h2Property_LastFirst* if you closed it at the end of Hands-On Exercise 2 and save it as **a03h3Property_LastFirst**, changing *h2* to *h3*.

b. Right-click the **Mortgage Payments query** in the Navigation Pane and select **Design View**. Drag the **Listing field** from the Properties table to the fifth column.

   The Listing field is now in the fifth column, between the SqFeet and Sold fields. The other columns shift to the right.

> **TROUBLESHOOTING:** If you drag the Listing field to the wrong position, you can drag it again to the correct location.

c. Switch to Datasheet view. Click **Totals**, in the Records group on the HOME tab to display the Total row.

   The Total row displays as the last row of the query results.

d. Click in the cell that intersects the Total row and the List Price column.

e. Click the arrow and select **Average** to display the average value of all the properties that have not sold. Widen the List Price column if you can't see the entire total value.

   The average list price of all properties is $129,799.63.

f. Click the arrow in the Total row in the Listing column and select **Count** from the list.

   The count of properties in this datasheet is 8.

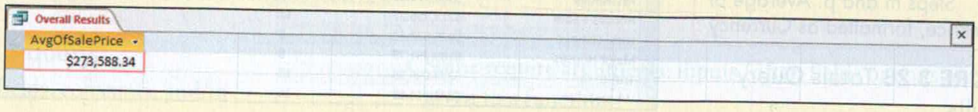
g. Click in the **Total row** in the Price Per Sq Ft column. Click the arrow and select **Average** to display the average price per square foot.

   The average price per square foot is $102.60.

h. Save and close the query.

STEP 2 » CREATE A TOTALS QUERY WITH GROUPING AND CONDITIONS

You create a totals query to help Don and Matt evaluate the properties in groups. Refer to Figure 3.27 and Figure 3.28 as you complete Step 2.


Overall Results

AvgOfSalePrice
$273,588.34

FIGURE 3.27 Overall Results Query

a. Click **Query Design** in the Queries group of the CREATE tab.

You create a new query in Query Design; the Show Table dialog box opens.

b. Add the Properties table from the Show Table dialog box. Close the Show Table dialog box.

c. Add the SalePrice and Sold fields from the Properties table to the query design grid.

d. Click **Totals** in the Show/Hide group of the QUERY TOOLS DESIGN tab to show the Total row.

A new row labeled Totals displays at the bottom of the screen in the design grid, between the Table and Sort rows. Each field will have Group By listed in the new row by default.

e. Click the **Group By arrow** in the SalePrice column Total row and select **Avg.**

f. Click the **Group By arrow** in the Sold column Total row and select **Where**. Type **Yes** in the Criteria row.

This criterion will limit the results to sold houses only.

g. Click in the **SalePrice field** and click **Property Sheet** in the Show/Hide group. Change the SalePrice format to **Currency**. Close the Property Sheet. Run the query and compare your results to Figure 3.27.

The results show an overall average of $273,588.34 for the sold properties in the database.

h. Click **Save** on the Quick Access Toolbar and type **Overall Results** as the Query Name in the Save As dialog box. Click **OK**. Close the query.

i. Click **Query Design** in the Query group of the HOME tab to create a new query.

j. Add the Properties table and the Lists table from the Show Table dialog box. Close the Show Table dialog box.

k. Add the NameOfList field from the Lists table and the SalePrice, Listing, and Sold fields from the Properties table to the query design grid.

l. Click **Totals** on the QUERY TOOLS DESIGN tab in the Show/Hide group to show the Total row.

A new row labeled Total appears at the bottom of the screen in the design grid between the Table and Sort rows.

m. Change the Total row for *SalePrice* to **Avg.**

n. Change the Total row for *Listing* to **Count**.

o. Change the Total row for *Sold* to **Where**. Type **Yes** in the Criteria row.

This criterion will limit the results to sold houses only.

p. Click in the **SalePrice field** and click **Property Sheet** in the Show/Hide group. Change the SalePrice format to **Currency**.

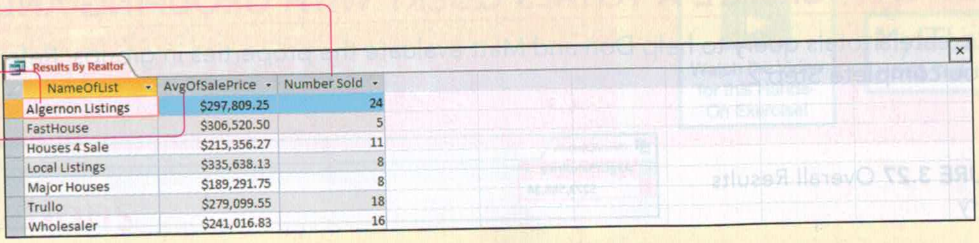
q. Change the caption of the Listing column to **Number Sold**. Run the query and widen the columns as shown in Figure 3.28.

Notice Major Houses has the only average sale price under $200,000. As Don and Matt are hoping to focus on inexpensive properties, they will focus on properties offered by this source. Notice the query results show the number of properties sold in each source, in addition to the average sale price. This will help determine which sources have been more effective.

Steps n and q: Count of Listing, with caption of Number Sold

Step k: Grouped by NameOfList

Steps m and p: Average of SalePrice, formatted as Currency



Results By Realtor

NameOfList	AvgOfSalePrice	Number Sold
Algernon Listings	$297,809.25	24
FastHouse	$306,520.50	5
Houses 4 Sale	$215,356.27	11
Local Listings	$335,638.13	8
Major Houses	$189,291.75	8
Trullo	$279,099.55	18
Wholesaler	$241,016.83	16

FIGURE 3.28 Totals Query Results

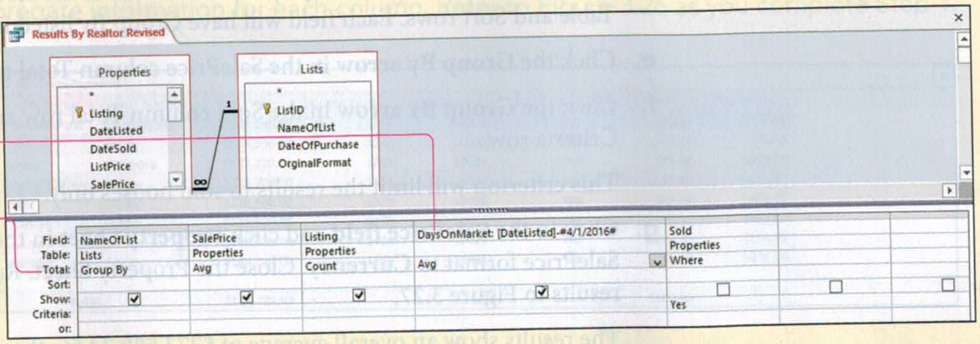
r. Click **Save** on the Quick Access Toolbar and type **Results By Realtor** as the Query Name in the Save As dialog box. Click **OK**. Keep the query open for the next step.

STEP 3 » ADD A CALCULATED FIELD TO A TOTALS QUERY

The previous query shows the average value of the properties by realtor. However, Don and Matt learned at the seminar they attended that the longer a property has been on the market, the better your chances of negotiating a better price. You will revise the query to include the average number of days on the market for each realtor. Refer to Figure 3.29 as you complete Step 3.

Step d: Formula entered

Step c: Total row



Results By Realtor Revised

Field:	NameOfList	SalePrice	Listing	DaysOnMarket: [DateListed]-#4/1/2016#	Sold		
Table:	Lists	Properties	Properties		Properties		
Total:	Group By	Avg	Count	Avg	Where		
Sort:							
Show:	☑	☑	☑	☑	□	□	□
Criteria:					Yes		
or:							

FIGURE 3.29 Results By Realtor Revised Query

a. Click the **FILE tab**, click **Save As**, and then click **Save Object As**. Click **Save As**, and in the *Save 'Results By Realtor' to:* box, type **Results By Realtor Revised**.

b. Switch to the Datasheet view for the new Results By Realtor Revised query, if necessary.

c. Click **Totals** in the Records group of the HOME tab. Change the Total to **Sum** in the Number Sold column.

The total number of houses sold (90) now displays at the bottom of the Number Sold column.

d. Switch to Design view. In the first blank column, type **DaysOnMarket: [DateListed] – #4/1/2016#** to create a new calculated field. Change the Total row to **Avg**.

The DaysOnMarket field will show the average number of days on the market for each realtor's listings.

e. Display the Property Sheet for the DaysOnMarket field and change the Format property to **Fixed**. Change the Decimal Places property to **0**. Close the Property Sheet.

TROUBLESHOOTING: If you do not see a Decimal Places property immediately beneath the Format property, change the format to Fixed, save and close the query, and then reopen the query. Refer to Figure 3.4 for the location of Decimal Places.

f. Run the query and examine the DaysOnMarket field.

Major Houses listings have an average of 117 days on the market. The combination of inexpensive prices and properties that do not sell quickly may help Don and Matt negotiate with the realtor.

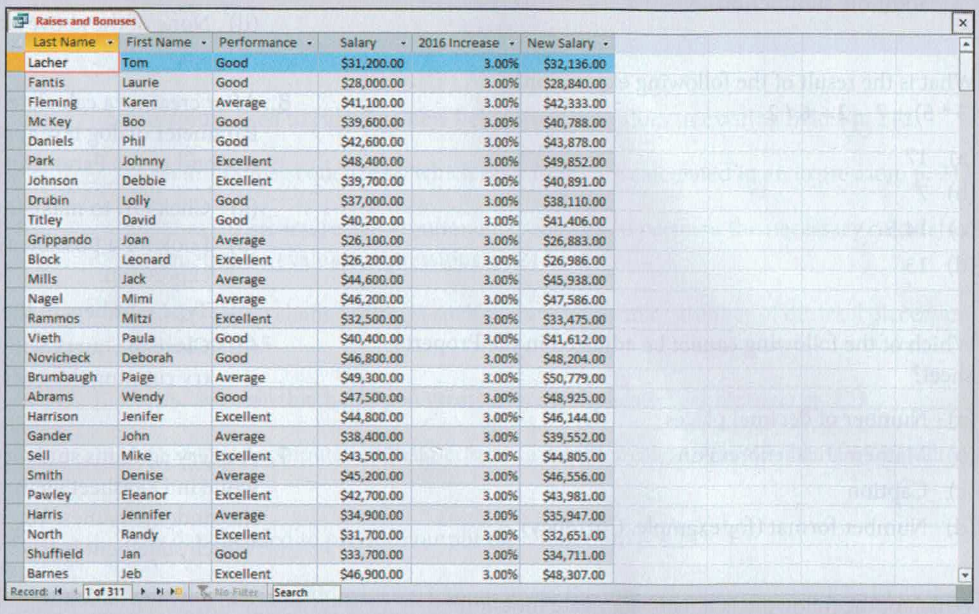
g. Save and close the query.

h. Exit Access. Submit based on your instructor's directions.

Practice Exercises

1 Comfort Insurance

The Comfort Insurance Agency is a mid-sized company with offices located across the country. Each employee receives a performance review annually. The review determines employee eligibility for salary increases and the annual performance bonus. The employee data is stored in an Access database, which is used by the human resources department to monitor and maintain employee records. Your task is to calculate the salary increase for each employee; you will also calculate the average salary for each position. This exercise follows the same set of skills as used in Hands-On Exercises 1 and 2 in the chapter. Refer to Figure 3.30 as you complete this exercise.



Last Name	First Name	Performance	Salary	2016 Increase	New Salary
Lacher	Tom	Good	$31,200.00	3.00%	$32,136.00
Fantis	Laurie	Good	$28,000.00	3.00%	$28,840.00
Fleming	Karen	Average	$41,100.00	3.00%	$42,333.00
Mc Key	Boo	Good	$39,600.00	3.00%	$40,788.00
Daniels	Phil	Good	$42,600.00	3.00%	$43,878.00
Park	Johnny	Excellent	$48,400.00	3.00%	$49,852.00
Johnson	Debbie	Excellent	$39,700.00	3.00%	$40,891.00
Drubin	Lolly	Good	$37,000.00	3.00%	$38,110.00
Titley	David	Good	$40,200.00	3.00%	$41,406.00
Grippando	Joan	Average	$26,100.00	3.00%	$26,883.00
Block	Leonard	Excellent	$26,200.00	3.00%	$26,986.00
Mills	Jack	Average	$44,600.00	3.00%	$45,938.00
Nagel	Mimi	Average	$46,200.00	3.00%	$47,586.00
Rammos	Mitzi	Excellent	$32,500.00	3.00%	$33,475.00
Vieth	Paula	Good	$40,400.00	3.00%	$41,612.00
Novicheck	Deborah	Good	$46,800.00	3.00%	$48,204.00
Brumbaugh	Paige	Average	$49,300.00	3.00%	$50,779.00
Abrams	Wendy	Good	$47,500.00	3.00%	$48,925.00
Harrison	Jenifer	Excellent	$44,800.00	3.00%	$46,144.00
Gander	John	Average	$38,400.00	3.00%	$39,552.00
Sell	Mike	Excellent	$43,500.00	3.00%	$44,805.00
Smith	Denise	Average	$45,200.00	3.00%	$46,556.00
Pawley	Eleanor	Excellent	$42,700.00	3.00%	$43,981.00
Harris	Jennifer	Average	$34,900.00	3.00%	$35,947.00
North	Randy	Excellent	$31,700.00	3.00%	$32,651.00
Shuffield	Jan	Good	$33,700.00	3.00%	$34,711.00
Barnes	Jeb	Excellent	$46,900.00	3.00%	$48,307.00

FIGURE 3.30 Raises and Bonuses Query

a. Open *a03p1Insurance*. Save the database as **a03p1Insurance_LastFirst**.

b. Examine the Relationships for the database. Notice the table structure, relationships, and fields. Once you are familiar with the database, close the Relationships window.

c. Create a new query in Design view. Add the Employees and Titles tables.

d. Add the LastName, FirstName, Performance, and Salary fields from the Employees table to the query. Add the 2016Increase field from the Titles table to the query.

e. Click the top row of the first blank column in the query design grid and type **NewSalary:[Salary]+[Salary]*[2016Increase]** to create a calculated field that adds the existing salary to the increase. You may opt to use the Expression Builder if you prefer.

f. Click **Run** in the Results group to run the query. Look at the output in the Datasheet view. Verify that your answers are correct. Notice that the fourth column heading displays *2016 Increase*.

This is the caption for the 2016Increase field in the Titles table that was carried over to the query. When a caption exists for a field in the table Design view, the caption also displays in the Query Datasheet view instead of the field name in the query.

g. Switch back to Design view. Click in the **NewSalary calculated field**, display the Property Sheet, and then change the format to **Currency**. Type **New Salary** in the **Caption box**. Close the Property Sheet.

h. Save the query as **Raises and Bonuses**. Close the query.

i. Create a new query in Design view. Add the Employees and Titles tables.

You will create a query to show the average salary by position.

j. Add the TitleName field from the Titles table. From the Employees table, add the Salary field.

k. Display the Total row. Change the Total row for Salary to **Avg**. Leave the TitleName field set to **Group By**.

l. Click the **Salary field** and display the Property Sheet. Change the format for the field to **Currency**.

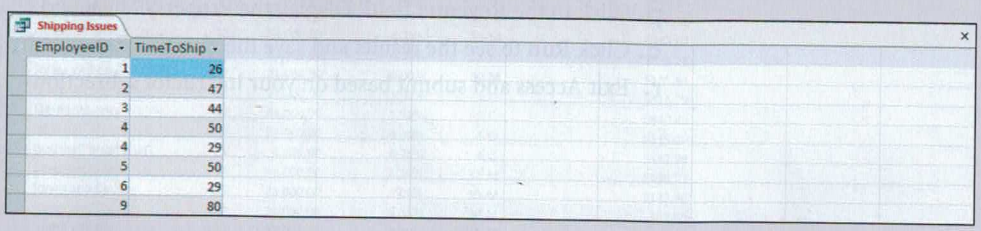
m. Run the query. Save the query as **Average Salary By Position** and close the query.

n. Exit Access and submit based on your instructor's directions.

2 Analyze Orders

FROM SCRATCH

You are the marketing manager of your company, and you must use the order information from an Access database to analyze sales trends. You need to determine the order revenue for all orders, grouped by Ship Country. The company would also like to check to see if there are order delays related to a specific employee. You must analyze shipping performance based on the number of days it takes to ship each order. This exercise follows the same set of skills as used in Hands-On Exercises 2 and 3 in the chapter. Refer to Figure 3.31 as you complete this exercise.


Shipping Issues

EmployeeID	TimeToShip
1	26
2	47
3	44
4	50
4	29
5	50
6	29
9	80

FIGURE 3.31 Shipping Issues Query

a. Create a new blank desktop database named **a03p2Orders_LastFirst**.
You will be shown a blank table in Datasheet view.

b. Click **View** in the Views group to switch to Design view. Save the table as **Orders**.

c. Change the first Field Name to **OrderID** and change the Data Type to **Number**. Type **CustomerID** in the second row and press **Tab**. Accept **Short Text** as the Data Type. Type **EmployeeID** in the third row and press **Tab**. Select **Number** for the Data Type.

d. Type and format the remainder of the fields as follows:

OrderDate	Date/Time
ShippedDate	Date/Time
ShipVia	Number
Revenue	Currency
ShipCountry	Short Text

e. Click **View** in the Views group to switch to Datasheet view. Click **Yes** to save the table. Add the three records as shown in the following table. Press **Tab** to move to the next field.

Order ID	Customer ID	Employee ID	Order Date	Shipped Date	Ship Via	Revenue	Ship Country
10248	WILMK	5	1/6/2017	1/19/2017	1	$142.86	Belgium
10249	TRADH	6	1/7/2017	1/10/2017	2	$205.38	Germany
10250	HANAR	4	1/10/2017	1/30/2017	2	$58.60	Venezuela

f. Open the *a03p2Orders* Excel file and click **Enable Editing**, if necessary. Click and hold **row 2** and drag through **row 828** so that all of the data rows are selected. Click **Copy** in the Clipboard group.

g. Return to Access and click on the **asterisk** (*) on the fourth row of the Orders table. Click **Paste** in the Clipboard group and click **Yes** to confirm that you want to paste all 827 rows into the Orders table. Save and close the table, and then close the spreadsheet and Excel. If prompted to save the data in the clipboard, click **No**.

h. Click the **CREATE tab** and click **Query Design** in the Queries group to start a new query. The Show Table dialog box opens. Add the Orders table and close the Show Table dialog box.

i. Add EmployeeID to the query and sort the table by EmployeeID in ascending order.

j. Use the Expression Builder to create a new calculated field. Type the following: **TimeToShip: [ShippedDate]-[OrderDate]**

   Run the query and verify that TimeToShip is displaying valid values.

k. Switch back to Design view. Add the criteria **>21** to the TimeToShip field. Run the query and compare your results with Figure 3.31.

   The results do not show a pattern of one employee's orders being delayed.

l. Save the query as **Shipping Issues**. Close the query.

m. Click the **CREATE tab** and click **Query Design** in the Queries group to start a new query. The Show Table dialog box opens. Add the Orders table and close the Show Table dialog box. Click **Totals** in the Show/Hide group.

n. Insert the ShipCountry and Revenue fields from the Orders table.

o. Verify the value for ShipCountry is set to **Group By** in the Totals row in Design view and verify that the value for the Revenue field is set to **Sum**.

p. Click in the **Revenue field**. Display the Property Sheet and change the caption to **Total Revenue**.

q. Click **Run** to see the results and save the query as **Revenue by Ship Country**. Close the query.

r. Exit Access and submit based on your instructor's directions.

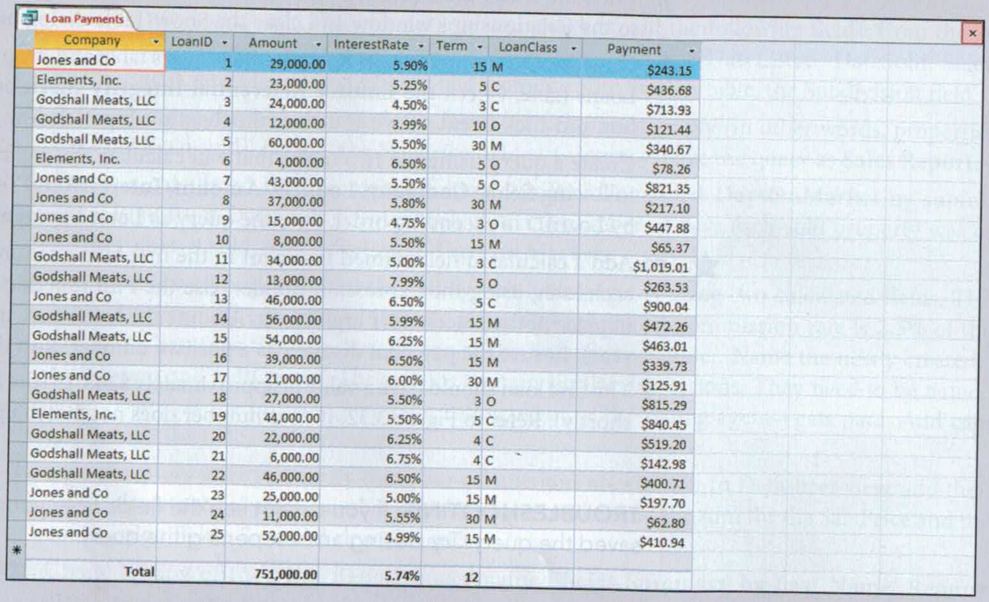
Mid-Level Exercises

1 Small Business Loans

ANALYSIS CASE

FROM SCRATCH

You are the manager of the small business loan department for the U.S. government. You need to calculate the payments for the loans that are currently on the books. To do this, you will need to create a query and add the Pmt function to calculate the loan payments for each loan. You will also summarize each loan by loan type (M=Mortgage, C=Car, and O=Other). Refer to Figure 3.32 as you complete this exercise.



Loan Payments

Company	LoanID	Amount	InterestRate	Term	LoanClass	Payment
Jones and Co	1	29,000.00	5.90%	15	M	$243.15
Elements, Inc.	2	23,000.00	5.25%	5	C	$436.68
Godshall Meats, LLC	3	24,000.00	4.50%	3	C	$713.93
Godshall Meats, LLC	4	12,000.00	3.99%	10	O	$121.44
Godshall Meats, LLC	5	60,000.00	5.50%	30	M	$340.67
Elements, Inc.	6	4,000.00	6.50%	5	O	$78.26
Jones and Co	7	43,000.00	5.50%	5	O	$821.35
Jones and Co	8	37,000.00	5.80%	30	M	$217.10
Jones and Co	9	15,000.00	4.75%	3	O	$447.88
Jones and Co	10	8,000.00	5.50%	15	M	$65.37
Godshall Meats, LLC	11	34,000.00	5.00%	3	C	$1,019.01
Godshall Meats, LLC	12	13,000.00	7.99%	5	O	$263.53
Jones and Co	13	46,000.00	6.50%	5	C	$900.04
Godshall Meats, LLC	14	56,000.00	5.99%	15	M	$472.26
Godshall Meats, LLC	15	54,000.00	6.25%	15	M	$463.01
Jones and Co	16	39,000.00	6.50%	15	M	$339.73
Jones and Co	17	21,000.00	6.00%	30	M	$125.91
Godshall Meats, LLC	18	27,000.00	5.50%	3	O	$815.29
Elements, Inc.	19	44,000.00	5.50%	5	C	$840.45
Godshall Meats, LLC	20	22,000.00	6.25%	4	C	$519.20
Godshall Meats, LLC	21	6,000.00	6.75%	4	C	$142.98
Godshall Meats, LLC	22	46,000.00	6.50%	15	M	$400.71
Jones and Co	23	25,000.00	5.00%	15	M	$197.70
Jones and Co	24	11,000.00	5.55%	30	M	$62.80
Jones and Co	25	52,000.00	4.99%	15	M	$410.94
*						
Total		751,000.00	5.74%	12		

FIGURE 3.32 Loan Payments Query Results

a. Open Access and create a new blank desktop database named **a03m1Loans_LastFirst**. Access will display a table named Table1 with one field, ID.

b. Switch to Design view. Type **Customers** in the **Save As dialog box** and click **OK**.

c. Change the first Field Name to **CustomerID** and accept **AutoNumber** as the Data Type. Type **Company** in the second row and press **Tab**. Accept **Short Text** as the Data Type. Type **FirstName** in the third row and press **Tab**. Accept **Short Text** as the Data Type.

d. Type the remainder of the fields:

LastName	Short Text
City	Short Text
State	Short Text
Zip	Short Text

e. Verify the first field is set as the primary key.

f. Switch to Datasheet view. Click **Yes** to save the table. Add the records as shown in the following table. Note you will allow Access to assign an ID. Once you have entered the records, close the Customers table.

Company	FirstName	LastName	City	State	Zip
Jones and Co	Robert	Paterson	Greensboro	NC	27401
Elements, Inc.	Merve	Kana	Paterson	NJ	07505
Godshall Meats, LLC	Francisco	De La Cruz	Beverly Hills	CA	90210

DISCOVER

g. Click the **External Data tab** and click **Excel** in the Import & Link group. Click **Browse** to locate the *a03m1Loans* spreadsheet. Select the database and click **Open** at the bottom of the dialog box.

h. Ensure the *Import the source data into a new table in the current database option* is selected and click **OK**. Click **Next** three times, until you are asked to add a primary key. From the *Choose my own Primary Key* menu, select **LoanID** (this should be the default option). Click **Next** once more and click **Finish**. Click **Close** in the Save Import Steps dialog box.

i. Open the Loans table in Design view. Select the **InterestRate field** and change the format to **Percent**. Change the field size for the CustomerID field to **Long Integer**. Click **Yes** when prompted that some data may be lost. Save and close the table.

j. Click the **Database Tools tab** and click **Relationships** in the Relationships group. Add both tables to the Relationships window and close the Show Table dialog box.

k. Drag the **CustomerID field** from the Customers table and drop it onto the CustomerID field in the Loans table. Check the **Enforce Referential Integrity check box** in the Edit Relationships dialog box and click **Create**. Save and close the Relationships window.

l. Create a query using the two tables that will calculate the payment amount for each loan. Add the following fields: **Company**, **LoanID**, **Amount**, **InterestRate**, **Term**, and **LoanClass**. Sort the query by LoanID in ascending order. Save the query as **Loan Payments**.

★ m. Add a calculated field named **Payment** in the first blank column to calculate the loan payment for each loan, using the Expression Builder. Use the Pmt function. Insert the appropriate field names in place of the placeholder arguments. Assume the loans have monthly payments (12 payments per year). Ensure the payment displays as a positive number. Run the query.

The first loan should have a value of approximately $243.15 (the extra decimal places will be removed shortly). Refer to Figure 3.32. If your number does not match up, reexamine your formula.

TROUBLESHOOTING: If you cannot see the fields from your current query, ensure you have saved the query. Try closing and reopening the query.

n. Switch to Design view and change the display to **Currency format**. Run the query again to verify your changes. Compare your results to Figure 3.32.

o. Switch to Datasheet view and add a **Totals row**. Use it to calculate the sum of the amount column, the average interest rate, and the average term. Save and close the query.

p. Create a copy of Loan Payments. Save the new query as **Loan Payments Summary**.

q. Open the Loan Payments Summary query in Design view and rearrange the columns as follows: LoanClass, LoanID, Amount, and InterestRate. Delete columns CompanyName, Term, and Payment. Click **Totals** in the Show/Hide group. Change the Total row from left to right as follows: Group By, Count, Sum, and Avg. Run the query.

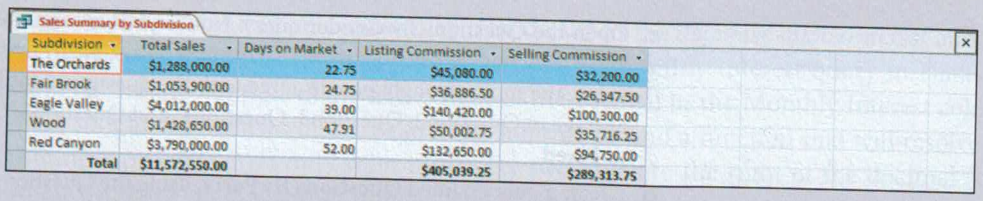
As we sorted the previous query by LoanID in Ascending order, this query will have the same sort by default.

r. Switch to Design view and display the Property Sheet. For the LoanID field, change the caption to **Loans**. For the Amount field, change the caption to **Total Amount** and change the format to **Currency**. For the InterestRate field, change the caption to **Avg Interest Rate** and change the format to **Percent**. Run the query. Save and close the query.

s. Exit Access, and submit based on your instructor's directions.

2 Investment Properties

You are in charge of LGS Investment's database, which contains all of the information on the properties your firm has listed and sold. Your task is to determine the length of time each property was on the market before it sold. You also need to calculate the sales commission from each property sold. Two agents will receive commission on each transaction: the listing agent and the selling agent. You also need to summarize the sales data by employee and calculate the average number of days each employee's sales were on the market prior to selling and the total commission earned by the employees. Refer to Figure 3.33 as you complete this exercise.



Sales Summary by Subdivision

Subdivision	Total Sales	Days on Market	Listing Commission	Selling Commission
The Orchards	$1,288,000.00	22.75	$45,080.00	$32,200.00
Fair Brook	$1,053,900.00	24.75	$36,886.50	$26,347.50
Eagle Valley	$4,012,000.00	39.00	$140,420.00	$100,300.00
Wood	$1,428,650.00	47.91	$50,002.75	$35,716.25
Red Canyon	$3,790,000.00	52.00	$132,650.00	$94,750.00
Total	$11,572,550.00		$405,039.25	$289,313.75

FIGURE 3.33 Sales Summary Query

a. Open *a03m2Homes*. Save the database as **a03m2Homes_LastFirst**.

b. Create a new query, add the necessary tables, and then add the following fields: from the Agents table, add the LastName field; from the Properties table, the DateListed, DateSold, SalePrice, SellingAgent, and ListingAgent fields; and from the SubDivision table, the Subdivision field.

c. Add criteria to the table to ensure the DateSold field is not empty (in other words, properties that have not been sold). Format the SalePrice field as **Currency**. Save the query as **Sales Report**.

d. Create a calculated field using the Expression Builder named **DaysOnMarket** by subtracting DateListed from DateSold. This will calculate the number of days each sold property was on the market when it sold. Add a caption of **Days on Market**.

e. Calculate the commissions for the selling and listing agents using two calculated fields. The listing commission rate is **3.5%** of the sale price, and the selling commission rate is **2.5%** of the sale price. You can type these in directly or use the Expression Builder. Name the newly created fields **ListComm** and **SellComm**. These fields contain similar expressions. They need to be named differently so that the proper agent—the listing agent or the selling agent—gets paid. Add captions and format the fields as **Currency**.

f. Save the query after you verify that your calculations are correct. In Datasheet view, add the Total row. Calculate the average number of days on the market and the sum for the SalePrice and the two commission fields. Save and close the query.

g. Create a copy of the Sales Report query named **Sales Summary by Last Name**. Remove the DateListed, SellingAgent, ListingAgent, and Subdivision fields.

h. Display the Total row. Group by LastName and change the DateSold field Total row to **Where**, so the condition carries over. Show the sum of SalePrice, the average of DaysOnMarket, and the sum for both ListComm and SellComm. Change the caption for the SalePrice field to **Total Sales** and format the DaysOnMarket field as **Fixed**. Run the query. Adjust column widths as necessary.

i. Adjust the Total row in the Datasheet view so it shows the sum of TotalSales. Save the query.

DISCOVER

j. Create a copy of the Sales Summary by Last Name query named **Sales Summary by Subdivision**. Modify the query so the grouping is based on the Subdivision field, not LastName. Sort the query results so the fewest Days on Market is first and the most Days on Market is last. Limit the results to the top five rows.

k. Exit Access and submit based on your instructor's directions.

3 Political Pollsters

COLLABORATION CASE

You are working with a group that would like to analyze survey results. You are specifically looking for trends in the data based on gender, political affiliation, and income level. To demonstrate the power of Access, you and your group will perform a small survey, add the results to a database, and create some queries to demonstrate how grouping can help get results.

a. Individually, open the *a03t1Survey.docx* file. Collect 10 responses each (unless directed to do otherwise by your instructor). You should try to survey a diverse group of people. You can do this survey via e-mail, Facebook, or another appropriate method. Bring the collected data to your group.

b. Open the *a03t1Survey* database and save the database as **a03t1Survey_GroupName**. Use the Enter New Survey Result form to enter all of your information into the existing database. There are four records to start your database.

c. Open the Questions By Gender query. Notice your average for question 1 is a number between 1 and 3. As your survey document listed Agree as a 3, Neutral as 2, and Disagree as 1, the higher the value, the more strongly people agree with the question. Modify the query so that you display the average of Question2, Question3, Question4, and Question5. Change the format for the new fields to **Fixed**.

d. Create a query named **Questions By Party**, using the Questions By Gender query as a guide, grouping by the PoliticalAffiliation field rather than the Gender field.

e. Create a query named **Questions By Income Level**, using the Questions By Gender query as a guide, grouping by the IncomeLevel field rather than the Gender field.

f. Examine your results. Discuss the results with your group and type up your conclusions in a new Word document named **a03t1SurveyResults_GroupName**. You should be able to make around five conclusions. An example of a conclusion might be that people of lower income levels are less interested in being taxed to support free Internet than people of higher income.

g. Exit Access and Word and submit based on your instructor's directions.