

2 The Prestige Hotel

The Prestige Hotel chain caters to upscale business travelers and provides state-of-the-art conference, meeting, and reception facilities. It prides itself on its international, four-star cuisine. Last year, it began a member reward club to help the marketing department track the purchasing patterns of its most loyal customers. All of the hotel transactions are stored in the database. Your task is to help the managers of the Prestige Hotel in Denver and Chicago identify their customers who stayed in a room last year and who had three persons in their party. Refer to Figure 2.57 as you complete this exercise.

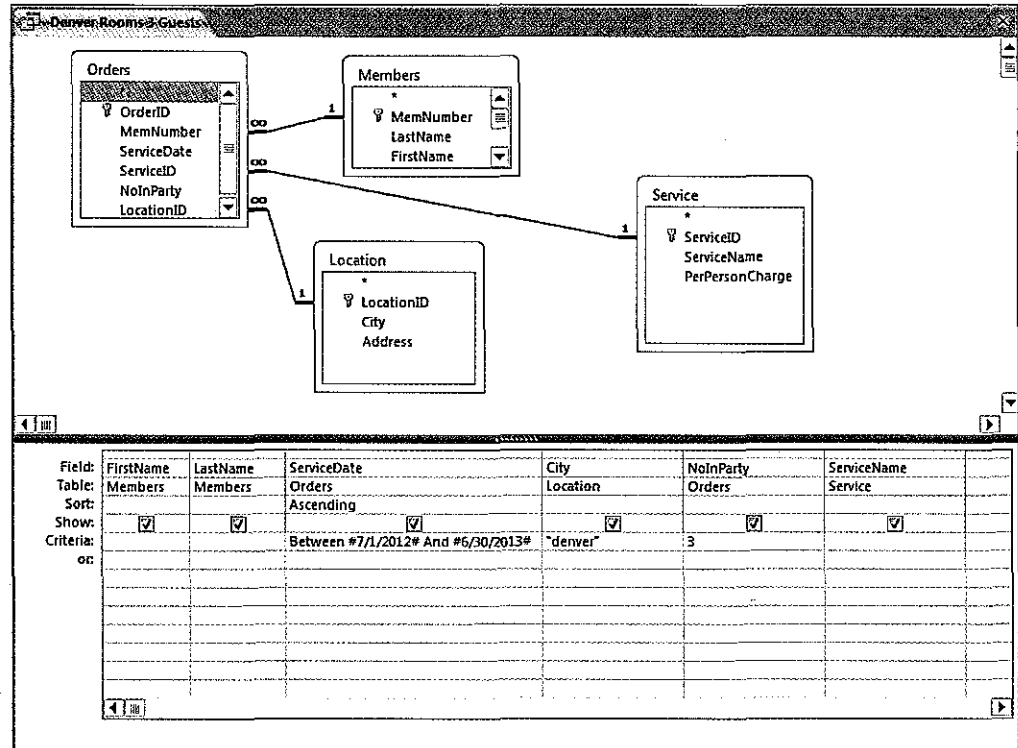


FIGURE 2.57 Denver Rooms
3 Guests Query ►

- Open *a02m2hotel*. Click the **File** tab, click **Save Database As**, and then type *a02m2hotel_LastnameFirstname*. Click **Save**. Review the data contained in the three tables. Specifically, look for the tables and fields containing the information you need: dates of stays in Denver suites, the members' names, and the numbers in the parties.
- Import the *Location* data from the Excel file *a02m2location* into your database as a new table. Set the *LocationID* Indexed property to **Yes (No Duplicates)**, and then set the data type to **Long Integer**. Select the *LocationID* field as the primary key. Name the table **Location**.
- Open the Relationships window, and then create a relationship between the *Location* table and the *Orders* table using the *LocationID* field. Enforce referential integrity, and then select **Cascade Update Related Fields**. Set the other two relationships as shown in Figure 2.57. Save and close the Relationships window.
- Open the *Members* table, and then find Bryan Gray's name. Replace his name with your own first and last name. Now find Nicole Lee's name, and then replace it with your name.

- e. Create a query with fields ServiceDate, City, NoInParty, ServiceName, FirstName, and LastName. Set the criteria to limit the output to **denver**. Use the Between command to only show services from 7/1/2012 to 6/30/2013. Set the Number in Party criterion to 3. Sort the results in ascending order by the Service Date.
- f. Run the query, and then examine the number of records in the status bar at the bottom of the query. It should display 154. If your number of records is different, examine the criteria.
- g. Change the order of the query fields so that they display as FirstName, LastName, ServiceDate, City, NoInParty, and ServiceName. Compare your results to Figure 2.57.
- h. Save the query as **Denver Rooms 3 Guests**. Close the query, and then copy it and rename the new query **Chicago Rooms 3 Guests**. One of your colleagues in Chicago asked for your help in analyzing the guest data.
- i. Open the Chicago Rooms 3 Guests query in Design view, and then change the criterion for denver to **chicago**. Run and save the changes.
- j. Combine the two previous queries into a third query named **Denver and Chicago Rooms 3 Guests**. Use the criteria from the two individual queries to create a combination AND – OR condition. The records in the combined query should equal the sum of the records in the two individual queries.
- k. Compact, repair, and back up the database file. Close the database.
- l. Submit based on your instructor's directions.

DISCOVER



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 are 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 each employee's performance bonus for employees who have been employed at least one year. This exercise follows the same set of skills as used in Hands-On Exercises 1 and 2 in the chapter. Refer to Figure 3.24 as you complete this exercise.

Last Name	First Name	Performance	Salary	2012 Increase	NewSalary	Bonus
Lacher	Tom	Good	\$31,200.00	3.00%	\$32,136.00	\$50.00
Fantis	Laurie	Good	\$28,000.00	3.00%	\$28,840.00	\$50.00
Fleming	Karen	Average	\$41,100.00	3.00%	\$42,333.00	\$50.00
McKey	Boo	Good	\$39,600.00	3.00%	\$40,788.00	\$50.00
Daniels	Phil	Good	\$42,600.00	3.00%	\$43,878.00	\$50.00
Park	Johnny	Excellent	\$48,400.00	3.00%	\$49,852.00	\$1,000.00
Johnson	Debbie	Excellent	\$39,700.00	3.00%	\$40,891.00	\$1,000.00
Drubin	Lolly	Good	\$37,000.00	3.00%	\$38,110.00	\$50.00
Titley	David	Good	\$40,200.00	3.00%	\$41,406.00	\$50.00
Grippando	Joan	Average	\$26,100.00	3.00%	\$26,883.00	\$50.00
Block	Leonard	Excellent	\$26,200.00	3.00%	\$26,986.00	\$1,000.00
Mills	Jack	Average	\$44,600.00	3.00%	\$45,938.00	\$50.00
			\$46,200.00	3.00%	\$47,586.00	\$50.00
			\$33,400.00	3.00%	\$33,475.00	

FIGURE 3.24 Raises and Bonuses

- Open *a03p1insurance*. Click the **File** tab, click **Save Database As**, and then type *a03p1insurance_LastnameFirstname*. Click **Save**.
- Click the **Database Tools** tab, and then click **Relationships** in the Relationships group. Examine the table structure, relationships, and fields. Once you are familiar with the database, close the Relationships window.
- Click the **Create** tab, and then click **Query Design** in the Queries group to start a new query. The Show Table dialog box opens. Add the **Employees** and **Titles** tables. Close the Show Table dialog box.
- Add the **LastName**, **FirstName**, **Performance**, and **Salary** fields to the query. From the **Titles** table, add the **2012Increase** field to the query.
- Click the top row of the first blank column in the query design grid, and then type `NewSalary:[Salary]*[2012Increase]+[Salary]` to create a calculated field.
- Click **Run** in the Results group to run the query. (If you receive the Enter Parameter Value dialog box, check your expression carefully for spelling errors.) Look at the output in the Datasheet view. Verify that your answers are correct. Notice that the fourth column heading displays *2012 Increase*. This is the caption for the **2012Increase** 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.
- Click **View** in the Views group to switch back to Design view. Open the Property Sheet, click in the **NewSalary** calculated field, and then change the format to **Currency**. Type **New Salary** in the **Caption** box. Close the Property Sheet.
- Save the query as **Raises and Bonuses**.
- Click the top row of the first blank column, and then click **Builder** in the Query Setup group. In the Expression Elements box, double-click the folder for **Functions**. Select the **Built-In Functions** folder. Scroll down the Expression Values box to locate the **IIf** function. Double-click **IIf** to insert the function.

- j. Click <<expression>>, and then replace it with **Performance = "Excellent"**. Click <<truepart>>, and then replace it with **1000**. Click <<>falsepart>>, and then replace it with **50**.
- k. Type **Bonus:** to the left of *IIf*, as the calculated field name. Click **OK**.
- l. Change the format of the Bonus field to **Currency** in the Property Sheet.
- m. Run the query. Save and close the query.
- n. Click the **File tab**, and then click **Compact and Repair Database**.
- o. Click the **File tab**, click **Save & Publish**, and then double-click **Back Up Database**. Click **Save** to accept the default backup file name.
- p. Click the **File tab**, and then click **Exit** (to exit Access).
- q. Submit based on your instructor's directions.

Northwind Traders, an international gourmet food distributor, is concerned about shipping delays over the last six months. Review the orders over the past six months and identify any order that was not shipped within 30 days. Each customer that falls within that time frame will be called to inquire about any problems the delay may have caused. In addition, an order summary and an order summary by country will be created.

Database File Setup

Open the food database, use **Save As** to make a copy of the database, and then use the new database to complete this capstone exercise. You will replace an existing employee's name with your name.

- Locate and open *a03c1food*.
- Click the **File** tab, click **Save Database As**, and then type *a03c1food_LastnameFirstname*.
- Click **Save**.
- Open the **Employees** table.
- Replace *Rachael Eliza* with your name. Close the table.

DaysToShip Query

You need to create a query to calculate the number of days between the date an order was placed and the date the order was shipped for each order. As you create the query, run the query at several intervals so you can verify that the data looks correct. The result of your work will be a list of orders that took more than three weeks to ship. The salespeople will be calling each customer to see if there was any problem with their order.

- Create a query using **Query Design**. Include the fields **CompanyName**, **ContactName**, **ContactTitle**, **Phone**, **OrderID**, **LastName**, **OrderDate**, and **ShippedDate**. Use the **Relationships** window to determine which tables you need before you begin.
- Run the query, and then examine the records. Save the query as **Shipping Efficiency**.
- Add a calculated field named **DaysToShip** to calculate the number of days taken to fill each order. (*Hint: The expression will include the OrderDate and the ShippedDate; the results will not contain negative numbers.*)
- Run the query, and then examine the results. Does the data in the **DaysToShip** field look accurate? Save the query.
- Add criteria to limit the query results to include any order that took more than 30 days to ship.
- Add the **ProductID** and **Quantity** fields to the **Shipping Efficiency** query. Sort the query by ascending **OrderID**. When the sales reps contact these customers, these two fields will provide useful information about the orders.

- Switch to **Datasheet** view to view the final results. This list will be distributed to the sales reps so they can contact the customers. In **Design** view, add the **Sales Rep caption** to the **LastName** field.
- Save and close the query.

Order Summary Query


You need to create an **Order Summary** that will show the total amount of each order in one column and the total discount amount in another column. This query will require four tables: **Orders**, **Order Details**, **Products**, and **Customers**. Query to determine if employees are following the employee discount policy. You will group the data by employee name, count the orders, show the total dollars, and show the total discount amount. You will then determine which employees are following the company guidelines.

- Create a query using **Query Design** and add the four tables above plus the **Products** table. Add the fields **OrderID** and **OrderDate**. Click **Totals** in the **Show/Hide Group**; the **Total** row for both fields should be **Group By**.
- In the third column, add a calculated field: **ExtendedAmount: Quantity*UnitPrice**. Format the calculated field as **Currency**. This calculation will calculate the total amount for each order. Change the **Total** row to **Sum**.
- In the fourth column, add a calculated field: **DiscountAmount: Quantity*UnitPrice*Discount**. Format the calculated field as **Currency**. This will calculate the total discount for each order. Change the **Total** row to **Sum**.
- Run the query. Save the query as **Order Summary**. Return to **Design** view.
- Enter the expression **Between 1/1/2012 And 12/31/2012** in the criteria of **OrderDate**. Change the **Total** row to **Where**. This expression will display only orders that were created in 2012.
- Run the query and view the results. Save the query.
- Add the **Total Dollars caption** to the **ExtendedAmount** field and add the **Discount Amt caption** to the **DiscountAmount** field.
- Run the query. Save and close the query.

Order Summary by Country Query

You need to create one additional query based on the **Order Summary** query you created in the previous step. This new query will enable you to analyze the orders by country.

- Select the **Order Summary** query, and then use **Save Object As** to create a new query named **Order Summary by Country**.

- 
- b. In Design view of the new query, replace the OrderID field with the Country field.
 - c. Run the query, and then examine the summary records; there should be 21 countries listed.
 - d. In Design view, change the sort order so that the country with the highest Total Dollars is first, and the country with the lowest Total Dollars is last.
 - e. Run the query and verify the results.
 - f. Save and close the query, and then close the database and exit Access.
 - g. Submit based on your instructor's directions.