

# **FIT1013 - Week 11 Resources**

**Querying a Database and Creating Advanced Queries**

# Week 11 Resources

1. Objectives .....	3
2. Create a query based on multiple tables.....	3
Updating a Database.....	3
Introduction to Queries .....	4
Updating Data Using a Query.....	5
Creating a Multitable Query.....	5
Sorting Data in a Query .....	6
Filtering Data.....	8
Activity 1 .....	9
Selection Criteria in Queries.....	9
Defining Record Selection Criteria for Queries .....	9
3. Using a Comparison Operator to Match a Range of Values.....	11
4. Use the And and Or logical operators in queries .....	13
Changing a Datasheet's Appearance .....	14
5. Creating a Calculated Field .....	15
6. Using Aggregate Functions .....	16
Working with Aggregate Functions Using the Total Row .....	17
Creating Queries with Aggregate Functions .....	17
Using Record Group Calculations .....	18
Working with the Navigation Pane .....	18
Activity 2 .....	19
7. Creating Advanced Queries and Enhancing Table Design .....	19
Reviewing the Clinic Database .....	19
Using Pattern Match in a Query .....	20
Using a List-of-Values Match in a Query.....	20
Using the Not Logical Operator in a Query.....	21
Using an AutoFilter to Filter Data .....	21
Assigning a Conditional Value to a Calculated Field .....	22
8. Creating a Parameter Query .....	24
Creating a More Flexible Parameter Query.....	25

9.	Use query wizards to create a crosstab query, a find duplicates query, and a find unmatched query .....	26
	Advanced Query Wizards .....	26
	Creating a Find Duplicates Query .....	29
	Creating a Find Unmatched Query.....	30
10.	Creating a Top Values Query.....	30
	Creating a Lookup Field.....	33
	Using the Input Mask Wizard.....	34
	Identifying Object Dependencies .....	36
	Defining Data Validation Rules .....	37
	Working with Long Text Fields.....	38
	Summary .....	39
11.	Practice and Apply .....	39

## Reference:

Microsoft Access 2016, New Perspectives Series, Shellman, Vodnik,  
Comprehensive Edn., Cengage Learning, **Module 3,5**

Sections © 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part, except for use as permitted in a license distributed with a certain product or service or otherwise on a password-protected website for classroom use.

# 1. Objectives

- Create a query based on multiple tables
- Use a comparison operator in a query to match a range of values
- Use the And and Or logical operators in queries
- Create and format a calculated field in a query
- Perform calculations in a query using aggregate functions and record group calculations
- Use the Like, In, Not, and & operators in queries
- Create a parameter query
- Use query wizards to create a crosstab query, a find duplicates query, and a find unmatched query
- Create a top values query

## 2. Create a query based on multiple tables

### Updating a Database

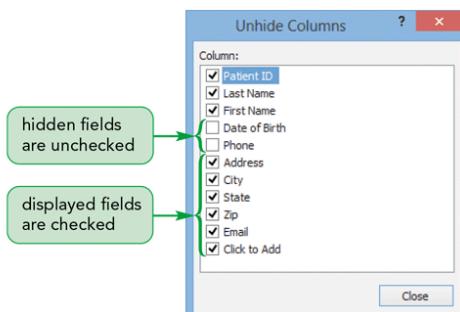
- Updating, or maintaining, a database is the process of adding, modifying, and deleting records in database tables to keep them current and accurate
- Modifying Records
  - To make minor changes, or select the field value to replace it entirely
    - The F2 key is a toggle that you use to switch between navigation mode and editing mode
  - In navigation mode, Access selects an entire field value. If you type while you are in navigation mode, your typed entry replaces the highlighted field value
  - In editing mode, you can insert or delete characters in a field value based on the location of the insertion point

Press	To Move the Selection in Navigation Mode	To Move the Insertion Point in Editing Mode
←	Left one field value at a time	Left one character at a time
→	Right one field value at a time	Right one character at a time
Home	Left to the first field value in the record	To the left of the first character in the field value
End	Right to the last field value in the record	To the right of the last character in the field value
↑ or ↓	Up or down one record at a time	Up or down one record at a time and switch to navigation mode
Tab or Enter	Right one field value at a time	Right one field value at a time and switch to navigation mode
Ctrl + Home	To the first field value in the first record	To the left of the first character in the field value
Ctrl + End	To the last field value in the last record	To the right of the last character in the field value

- Hiding and Unhiding Fields

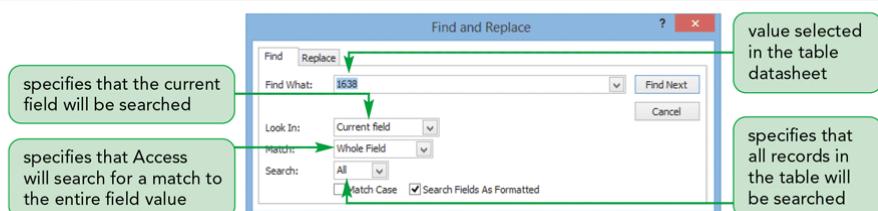
- When you are viewing a table or query datasheet in Datasheet view, you might want to temporarily remove certain fields from the displayed datasheet, making it easier to focus on the data you're interested in viewing
- The Hide Fields command removes the display of one or more fields
  - Can be especially useful in a table with many fields
- The Unhide Fields command redisplays any hidden fields

**Unhide Columns dialog box**



- Finding Data in a Table
  - Access provides options you can use to locate specific field values in a table
    - The Find command searches a table or query datasheet, or a form, to locate a specific field value or part of a field value

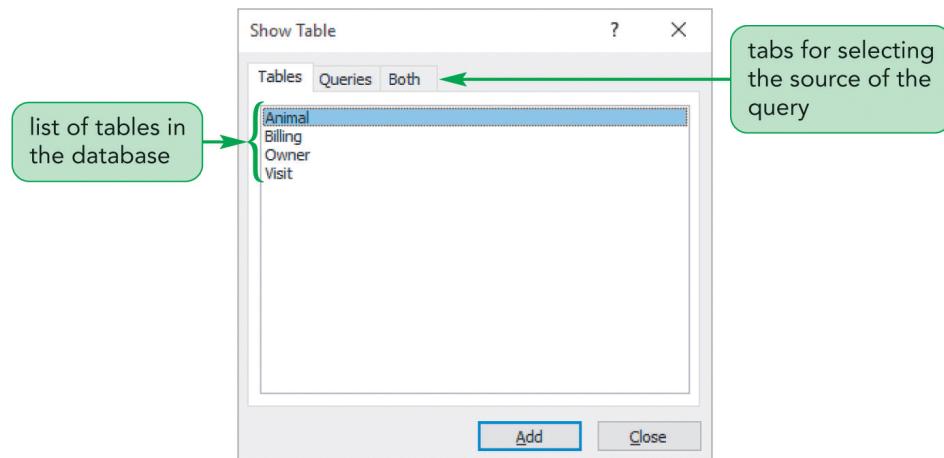
**Figure 3-4 Find and Replace dialog box**



- Deleting Records
  - To delete a record, you need to select the record in Datasheet view, and then delete it using the Delete button in the Records group on the HOME tab or the Delete Record option on the shortcut menu

## Introduction to Queries

- Access provides powerful query capabilities that allow you to do the following:
  - Display selected fields and records from a table
  - Sort records
  - Perform calculations
  - Generate data for forms, reports, and other queries
  - Update data in the tables in a database
  - Find and display data from two or more tables
- The answer to a select query is returned in the form of a datasheet
  - The result of a query is also referred to as a recordset because the query produces a set of records that answers your question



- Creating and Running a Query

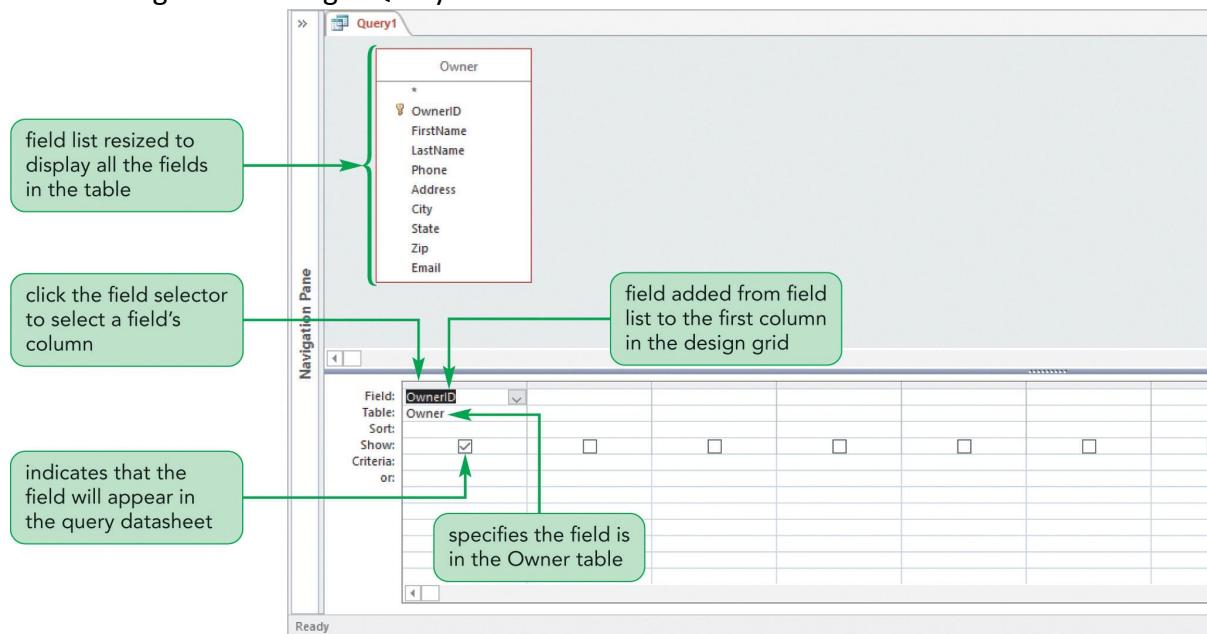


Figure 3-8 Datasheet displayed after running the query

Patient ID	Last Name	First Name	City	Email
22500	Student Last	Student First	Hartford	student1@example.edu
22501	Darcy	Edward	Bloomfield	edarcy@example.org
22502	Mendez	Andreas	Windsor	a.mendez@example.org
22504	Aguilar	Lilian	Hartford	laguilar@example.com
22505	Finnerty	Amber	Hartford	a.finnerty@example.com
22506	Gorski	Drea	Bloomfield	
22507	Weiss	Matthew	Hartford	matt.weiss@example.org

### Updating Data Using a Query

- A query datasheet is temporary and its contents are based on the criteria in the query design grid
  - You can still update the data in a table using a query datasheet
- Instead of making the changes in the table datasheet, you can make them in the PatientEmail query datasheet because the query is based on the Patient table
- The underlying Patient table will be updated with the changes you make

### Creating a Multitable Query

- A multitable query is a query based on more than one table

- If you want to create a query that retrieves data from multiple tables, the tables must have a common field

**Figure 3-9 Datasheet for query based on the Patient and Visit tables**

City	First Name	Last Name	Date of Visit	Reason/Diagnosis
Hartford	Student First	Student Last	11/17/2015	Migraine
Bloomfield	Edward	Darcy	11/30/2015	Influenza
Windsor	Andreas	Mendez	3/30/2016	Annual wellness visit
Hartford	Lilian	Aguilar	11/18/2015	Annual wellness visit
Hartford	Amber	Finnerty	1/26/2016	Annual wellness visit
Bloomfield	Drea	Gorski	4/1/2016	Fifth Disease
Bloomfield	Drea	Gorski	4/8/2016	Fifth Disease follow-up
Hartford	Matthew	Weiss	11/9/2015	Diabetes mellitus Type 2 - initial diagnosis
Hartford	Matthew	Weiss	2/9/2016	Diabetes mellitus Type 2 - serum glucose check
Hartford	Matthew	Weiss	4/7/2016	Diabetes mellitus Type 2 - serum glucose check
Hartford	Steve	Kervin	4/4/2016	Tinea pedis
Hartford	Steve	Kervin	4/18/2016	Tinea pedis follow-up
Hartford	Thomas	Booker	11/10/2015	Seborrheic dermatitis
Hartford	Thomas	Booker	3/1/2016	Seborrheic dermatitis follow-up
West Hartford	Daniel	Castro	12/3/2015	Annual wellness visit
West Hartford	Daniel	Castro	1/13/2016	Cardiac monitoring
Hartford	Lisa	Chang	1/5/2016	Annual wellness visit
Bloomfield	Troy	Smith	1/13/2016	Broken leg
Bloomfield	Troy	Smith	2/24/2016	Follow-up - cast removal
Hartford	Ian	Parker	12/15/2015	Influenza
Hartford	Ian	Parker	1/14/2016	Hypertension monitoring
Hartford	Susan	King	12/22/2015	COPD management visit
West Hartford	Henry	O'Brien	2/1/2016	Annual wellness visit
West Hartford	Henry	O'Brien	4/11/2016	Idiopathic abdominal pain
Hartford	Sera	Torres	4/1/2016	Conjunctivitis

## Sorting Data in a Query

- Sorting is the process of rearranging records in a specified order or sequence
  - Sometimes you might need to sort data before displaying or printing it to meet a specific request
- To sort records, you must select the sort field, which is the field used to determine the order of records in the datasheet

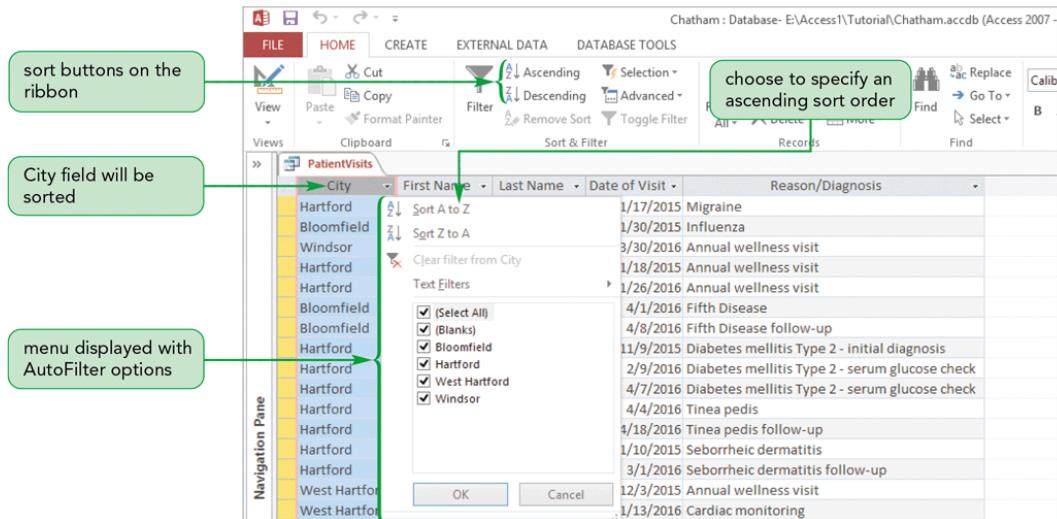
**Figure 3-10 Sorting results for different data types**

Data Type	Ascending Sort Results	Descending Sort Results
Short Text	A to Z (alphabetical)	Z to A (reverse alphabetical)
Number	lowest to highest numeric value	highest to lowest numeric value
Date/Time	oldest to most recent date	most recent to oldest date
Currency	lowest to highest numeric value	highest to lowest numeric value
AutoNumber	lowest to highest numeric value	highest to lowest numeric value
Yes/No	yes (checkmark in check box) then no values	no then yes values

© 2014 Cengage Learning

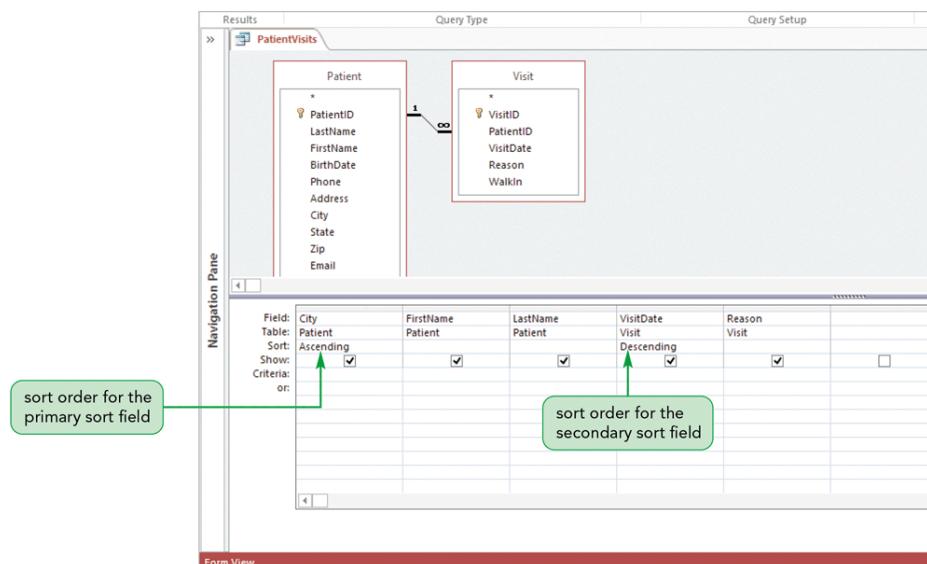
- When working in Datasheet view for a table or query, each column heading has an arrow to the right of the field name
  - Arrow gives you access to the AutoFilter feature, which enables you to quickly sort and display field values in various ways

**Figure 3-11** Using AutoFilter to sort records in the datasheet



- Sorting on Multiple Fields in Design View
  - Sort fields can be unique or nonunique
    - A sort field is unique if the value in the sort field for each record is different
    - A sort field is nonunique if more than one record can have the same value for the sort field
      - When the sort field is nonunique, records with the same sort field value are grouped together, but they are not sorted in a specific order within the group
      - To arrange these grouped records in a specific order, you can specify a secondary sort field, which is a second field that determines the order of records that are already sorted by the primary sort field (the first sort field specified)

**Figure 3-12** Selecting two sort fields in Design view



**Figure 3-13 Datasheet sorted on two fields**

City	First Name	Last Name	Date of Visit	Reason/Diagnosis
Bloomfield	Cathy	Engber	4/8/2016	Varicella
Bloomfield	Isobel	Kirk	4/8/2016	Hypertension monitoring
Bloomfield	Drea	Gorski	4/8/2016	Fifth Disease follow-up
Bloomfield	Pam	Fielder	4/6/2016	Pertussis
Bloomfield	Drea	Gorski	4/1/2016	Fifth Disease
Bloomfield	Isobel	Kirk	3/9/2016	Hypertension monitoring
Bloomfield	Siyang	Li	3/3/2016	Allergies - environmental follow-up
Bloomfield	Isobel	Kirk	2/25/2016	Hypertension monitoring
Bloomfield	Troy	Smith	2/24/2016	Follow-up - cast removal
Bloomfield	Isobel	Kirk	2/11/2016	Hypertension
Bloomfield	Troy	Smith	1/13/2016	Broken leg
Bloomfield	Pam	Fielder	12/1/2015	Influenza
Bloomfield	Edward	Darcy	11/30/2015	Influenza
Bloomfield	Siyang	Li	11/9/2015	Allergies - environmental
Hartford	Steve	Kervin	4/18/2016	Tinea pedis follow-up
Hartford	Michael	Caputo	4/15/2016	Removal of sutures from left eyelid
Hartford	Jane	Ropiak	4/14/2016	COPD management visit
Hartford	Susan	Hawes	4/12/2016	Joint pain/arthritis
Hartford	Michael	Caputo	4/8/2016	Laceration of left eyelid -sutured
Hartford	Sera	Torres	4/8/2016	Conjunctivitis follow-up
Hartford	Matthew	Weiss	4/7/2016	Diabetes mellitus Type 2 - serum glucose check
Hartford	Steve	Kervin	4/4/2016	Tinea pedis
Hartford	Sera	Torres	4/1/2016	Conjunctivitis
Hartford	Jessica	Wallner	3/28/2016	UTI follow-up
Hartford	Jerome	Taylor	3/23/2016	Oncocryoptosis follow-up

## Filtering Data

- A filter is a set of restrictions you place on the records to temporarily isolate a subset of the records
  - Lets you view different subsets of displayed records so that you can focus on only the data you need
  - An applied filter is not available the next time you run the query or open the form (unless it has been saved)
- The simplest technique for filtering records is Filter By Selection
  - Lets you select all or part of a field value in a datasheet or form, and then display only those records that contain the selected value in the field
  - Another technique for filtering records is to use Filter By Form, which changes your datasheet to display blank fields

**Figure 3-14 Using Filter By Selection**

City	First Name	Last Name	Date of Visit	Reason/Diagnosis
Hartford	Julia	Ingram	11/12/2015	Transverse fracture of left ulna
Hartford	Thomas	Booker	11/10/2015	Seborrheic dermatitis
Hartford	Matthew	Weiss	11/9/2015	Diabetes mellitus Type 2 - initial diagnosis
West Hartford	Henry	O'Brien	4/11/2016	Idiopathic abdominal pain
West Hartford	Magdalena	Cruz	3/18/2016	Migraine headache follow-up
West Hartford	Magdalena	Cruz	2/18/2016	Migraine headache
West Hartford	Anna	Diaz	2/2/2016	Brachial neuritis
West Hartford	Henry	O'Brien	2/1/2016	Annual wellness visit
West Hartford	Daniel	Castro	1/13/2016	Cardiac monitoring
West Hartford	Daniel	Shaw	1/11/2016	Nasopharyngitis
West Hartford	Malcolm	Belanger	1/8/2016	Hypertension monitoring
West Hartford	Daniel	Castro	12/3/2015	Annual wellness visit
West Hartford	Malcolm	Belanger	11/30/2015	Hypertension
Windsor	Chaney	Franklin	4/15/2016	Elevated blood lipids-monitoring meds
Windsor	Maria	Rodriguez	4/6/2016	Wound care for patellar abrasion
Windsor	Maria	Rodriguez	4/1/2016	Severe abrasion of left patella

**Figure 3-15 Datasheet after applying the filter**

City	First Name	Last Name	Date of Visit	Reason/Diagnosis
Windsor	Chaney	Franklin	4/15/2016	Elevated blood lipids-monitoring meds
Windsor	Maria	Rodriguez	4/6/2016	Wound care for patellar abrasion
Windsor	Maria	Rodriguez	4/1/2016	Severe abrasion of left patella
Windsor	Andreas	Mendez	3/30/2016	Annual wellness visit
Windsor	Chaney	Franklin	3/23/2016	Elevated blood lipids-monitoring meds
Windsor	Patrice	Lewis	3/21/2016	Eczema erythematous follow-up
Windsor	Alex	Delgado	3/18/2016	Hypertension monitoring
Windsor	Christina	Rowe	3/11/2016	Annual wellness visit
Windsor	Patrice	Lewis	3/7/2016	Eczema erythematous
Windsor	Chaney	Franklin	2/24/2016	Elevated blood lipids-monitoring meds
Windsor	Davis	Hallick	2/22/2016	Follow-up - gastric reflux
Windsor	Robert	Goldberg	2/9/2016	Sinusitis
Windsor	Davis	Hallick	2/8/2016	Gastric reflux
Windsor	Alex	Delgado	2/8/2016	Hypertension monitoring
Windsor	Christina	Rowe	1/28/2016	Nasopharyngitis
Windsor	Max	Sutherland	1/26/2016	Acute viral rhinopharyngitis
Windsor	Nancy	Bennett	1/25/2016	Nasopharyngitis
Windsor	Alex	Delgado	1/25/2016	Hypertension
Windsor	Patrice	Lewis	1/22/2016	Acute sinusitis follow-up
Windsor	Ashley	Garrett	1/15/2016	Annual wellness visit
Windsor	Patrice	Lewis	1/13/2016	Acute sinusitis
Windsor	Davis	Hallick	12/7/2015	Annual wellness visit
Windsor	Patrice	Lewis	11/25/2015	Influenza
Windsor	Amrita	Mehta	11/24/2015	Influenza

## Activity 1

- What is a Select Query?
  - A select query is a general query in which you specify the fields and records you want Access to select.
- How are a table datasheet and a query datasheet similar? How are they different?
  - A table datasheet and a query datasheet look the same, appearing in Datasheet view, and can be used to update data in a database. A table datasheet shows the permanent data in a table, whereas a query datasheet is temporary and its contents are based on the criteria you establish in the design grid.

## Selection Criteria in Queries

The diagram illustrates how selection criteria are applied in Access queries through four examples:

- BloomfieldPatients:** Shows a query grid where the "City" field has a criteria of "Bloomfield". A callout notes: "When creating queries in Design view, you can enter criteria so that Access will display only selected records in the query results." The results show only patients from Bloomfield.
- LargeInvoicedAmts:** Shows a query grid where the "InvoiceAmt" field has a criteria of "> 250". A callout notes: "The results of a query containing selection criteria include only the records that meet the specified criteria." The results show invoices with amounts greater than \$250.
- BetweenVisits:** Shows a query grid where the "VisitDate" field has a criteria of "Between #12/1/2015# And #12/31/2015#". A callout notes: "A condition usually consists of an operator, often a comparison operator, and a value. A comparison operator asks Access to compare the value in a field to the condition value and to select all the records for which the condition is true." The results show visits between December 1, 2015, and December 31, 2015.
- DecemberVisits:** Shows a query grid where the "Visit Date" field has a criteria of "12/1/2015 <= Visit Date <= 12/31/2015". A callout notes: "Most comparison operators, such as Between... And... and Access to select records that match a range of values for the condition—in this case, all records with dates that fall within the range shown." The results show visits in December 2015.

## Defining Record Selection Criteria for Queries

- To tell Access which records you want to select, you must specify a condition as part of the query

- A condition usually includes one of the comparison operators

**Figure 3-16** Access comparison operators

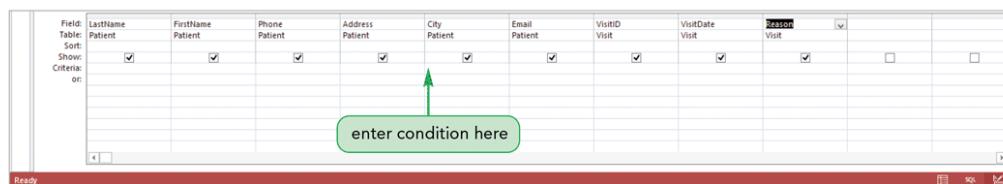
Operator	Meaning	Example
=	equal to (optional; default operator)	="Hall"
<>	not equal to	<>"Hall"
<	less than	<#1/1/99#
<=	less than or equal to	<=100
>	greater than	>"C400"
>=	greater than or equal to	>=18.75
Between ... And ...	between two values (inclusive)	Between 50 And 325
In ()	in a list of values	In ("Hall", "Seeger")
Like	matches a pattern that includes wildcards	Like "706*"

© 2014 Cengage Learning

#### ■ Specifying an Exact Match

- Create a query that will display specific records
- This type of condition is an exact match because the value in the specified field must match the condition exactly in order for the record to be included in the query results

**Figure 3-17** Design grid after adding fields from both tables



**Figure 3-18** Datasheet displaying selected fields and records

Last Name	First Name	Phone	Address	City	Email	Visit ID	Date of Visit	Reason/Diagnosis
Darcy	Edward	860-305-3985	723 Oxford Ave	Bloomfield	edarcy@example.org	1549	11/30/2015	Influenza
Gorski	Dre	860-305-8394	83 Everett Ln	Bloomfield		1667	4/1/2016	Fifth Disease
Gorski	Dre	860-305-8394	83 Everett Ln	Bloomfield		1683	4/8/2016	Fifth Disease follow-up
Smith	Troy	860-305-0384	16 Ravine Rd	Bloomfield	t_smith@example.edu	1575	1/13/2016	Broken leg
Smith	Troy	860-305-0384	16 Ravine Rd	Bloomfield	t_smith@example.edu	1626	2/24/2016	Follow-up - cast removal
Engber	Cathy	860-305-3048	58 Deering Pl	Bloomfield		1690	4/13/2016	Varicella
Li	Siyang	860-305-6548	225 Krauss Rd	Bloomfield	li_siwang@example.org	1527	11/5/2015	Allergies - environmental
Li	Siyang	860-305-6548	225 Krauss Rd	Bloomfield	li_siwang@example.org	1634	3/3/2016	Allergies - environmental follow-up
Fielder	Pam	860-305-2689	39 Unger Ave	Bloomfield	pfielder@example.org	1550	12/1/2015	Influenza
Fielder	Pam	860-305-2689	39 Unger Ave	Bloomfield	pfielder@example.org	1673	4/6/2016	Pertussis
Kirk	Isobel	860-305-7384	48 Grafton St	Bloomfield		1613	2/11/2016	Hypertension
Kirk	Isobel	860-305-7384	48 Grafton St	Bloomfield		1628	2/25/2016	Hypertension monitoring
Kirk	Isobel	860-305-7384	48 Grafton St	Bloomfield		1640	3/9/2016	Hypertension monitoring
Kirk	Isobel	860-305-7384	48 Grafton St	Bloomfield		1679	4/8/2016	Hypertension monitoring

14 records are selected

only records with a City field value of Bloomfield are selected

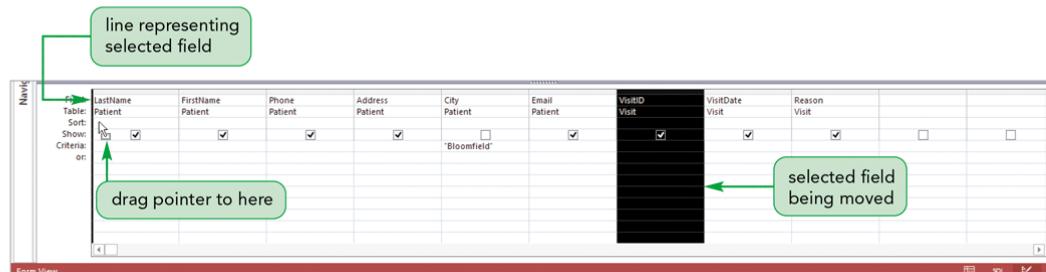
#### ■ Modifying a Query

- After you create a query and view the results, you might need to make changes to the query if the results are not what you expected or require

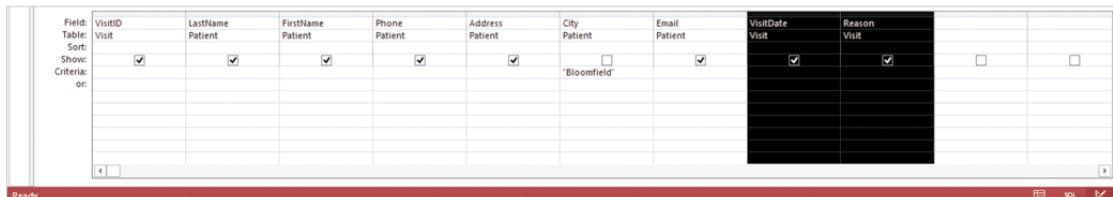
**Figure 3-19** Selected VisitID field

Show check box is cleared	VisitID field selector
entire column is selected	

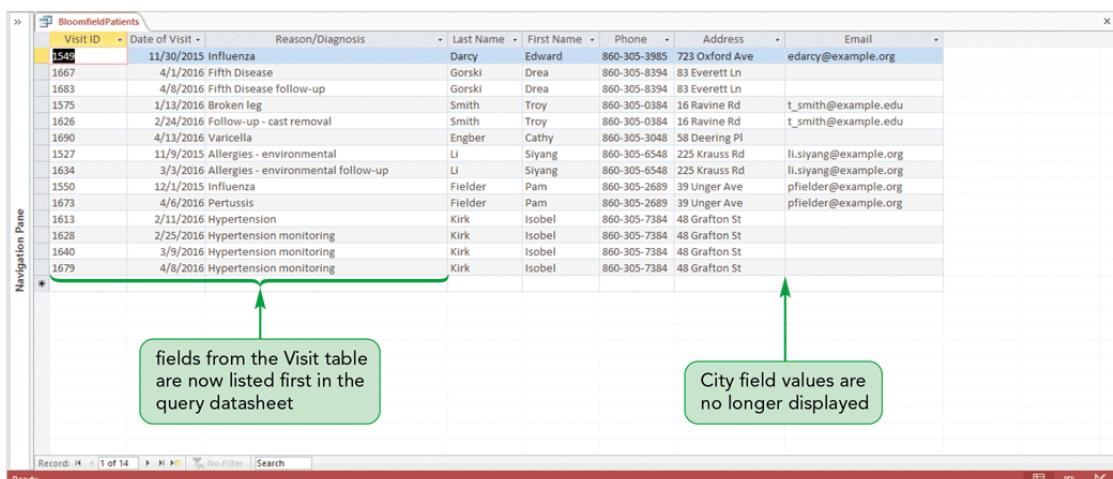
**Figure 3-20** Dragging the field in the design grid



**Figure 3-21** Multiple fields selected to be moved



**Figure 3-22** Results of the modified query



### 3. Using a Comparison Operator to Match a Range of Values

- After you create and save a query, you can double-click the query name in the Navigation Pane to run the query again
  - Click the View button to change its design
  - You can also use an existing query as the basis for creating another query

Figure 3-23 Criteria entered for the VisitDate field

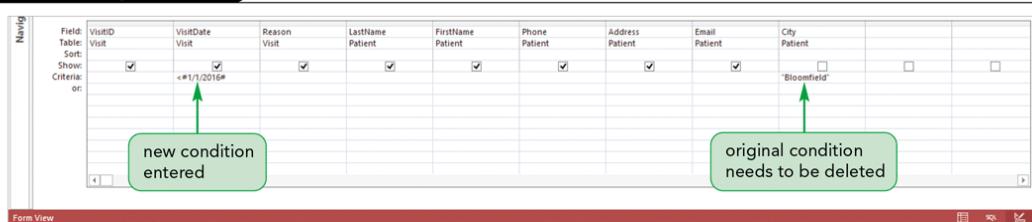


Figure 3-24 Design grid after moving the City field

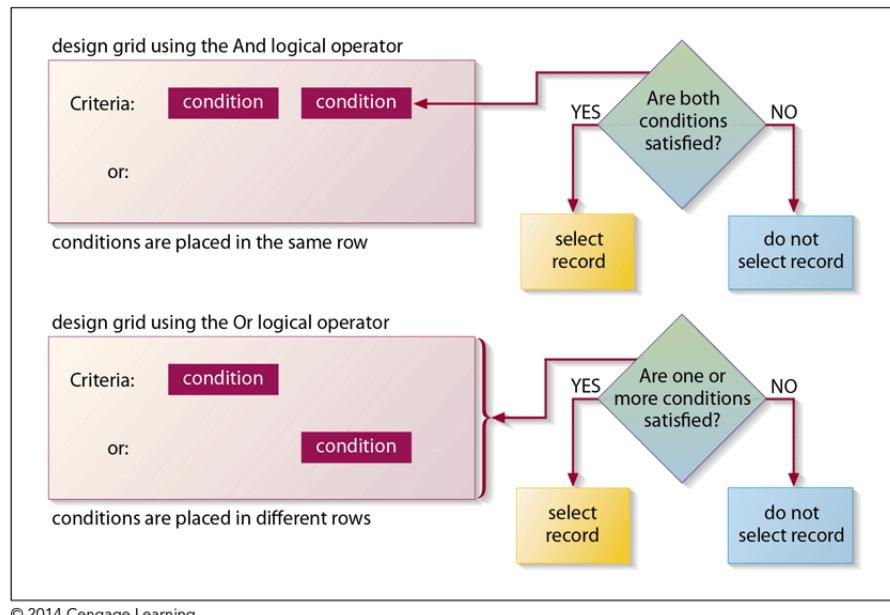
Field	VisitID	VisitDate	Reason	Lastname	FirstName	Phone	Address	City	Email
Table: Visit		Visit	Visit	Patient	Patient	Patient	Patient	Patient	Patient
Show:	<input checked="" type="checkbox"/>								
Criteria: or:			<#1/1/2016#						

Figure 3-25 Running the modified query

Visit ID	Date of Visit	Reason/Diagnosis	Last Name	First Name	Phone	Address	City	Email
1527	11/9/2015	Allergies - environmental	Li	Siyang	860-305-6548	225 Krauss Rd	Bloomfield	li.siyang@example.org
1528	11/9/2015	Diabetes mellitus Type 2 - initial diagnosis	Weiss	Matthew	860-426-7492	38 German St	Hartford	matt.weiss@example.org
1530	11/10/2015	Seborrheic dermatitis	Booker	Thomas	860-661-2539	59 Nicola Ave	Hartford	
1535	11/12/2015	Transverse fracture of left ulna	Ingram	Julia	860-661-4937	83 Kiefer Rd	Hartford	
1536	11/17/2015	Gastroenteritis	Svenson	Lucia	860-226-0293	83 Osage Ave	Hartford	
1538	11/17/2015	Migraine	Student Last	Student First	860-938-2822	501 Perkins Dr	Hartford	student1@example.edu
1539	11/18/2015	Annual wellness visit	Aguilar	Lilian	860-374-5724	329 Fairchild Ave	Hartford	lagular@example.com
1541	11/24/2015	Gastroenteritis - follow up	Svenson	Lucia	860-226-0293	83 Osage Ave	Hartford	
1542	11/24/2015	Influenza	Mehtra	Amrita	860-552-0375	23 Aiken Ave	Windsor	a.mehta@example.com
1544	11/25/2015	Influenza	Lewis	Patrice	860-552-5830	83 Highland St	Windsor	p.lewis5@example.org
1548	11/30/2015	Hypertension	Belanger	Malcolm	860-637-3927	723 Nicola Ave	West Hartford	
1549	11/30/2015	Influenza	Darcy	Edward	860-305-3985	723 Oxford Ave	Bloomfield	edarcy@example.org
1550	12/1/2015	Influenza	Fielder	Pam	860-305-2689	39 Unger Ave	Bloomfield	pfielder@example.org
1552	12/3/2015	Annual wellness visit	Castro	Daniel	860-637-0430	61 Osmond Way	West Hartford	d_castro@example.com
1555	12/7/2015	Annual wellness visit	Hallick	Davis	860-552-4495	84 Churchill Pl	Windsor	d.hallick@example.com
1557	12/10/2015	Annual wellness visit	Svenson	Lucia	860-226-0293	83 Osage Ave	Hartford	
1560	12/15/2015	Influenza	Parker	Ian	860-938-1873	12 Adelbert St	Hartford	ian_parker@example.org
1563	12/22/2015	COPD management visit	King	Susan	860-661-9347	48 Fenton St.	Hartford	susan.king@example.edu

- Multiple conditions require you to use logical operators to combine two or more conditions
  - Need to use the And logical operator
  - If you place conditions in separate fields in the same Criteria row of the design grid, all conditions in that row must be met in order for a record to be included in the query results
  - If you place conditions in different Criteria rows, a record will be selected if at least one of the conditions is met
  - If none of the conditions are met, no records are selected
  - When you place conditions in different Criteria rows, you are using the Or logical operator

**Figure 3-26** Logical operators And and Or for multiple selection criteria



## 4. Use the And and Or logical operators in queries

- The And Logical Operator
  - In the query design, both conditions you specify will appear in the same Criteria row; therefore, the query will select records only if both conditions are met

**Figure 3-27** Query to find older patients who have had influenza

And logical operator with conditions entered in the same row

Field:	FirstName	Patient	LastName	Patient	BirthDate	Patient	Phone	Patient	City	Patient	VisitDate	Visit	Reason	Visit
Table:			Sort:		Show:		Criteria:							
Criteria:	or:				<=#12/31/1956#								'Influenza'	

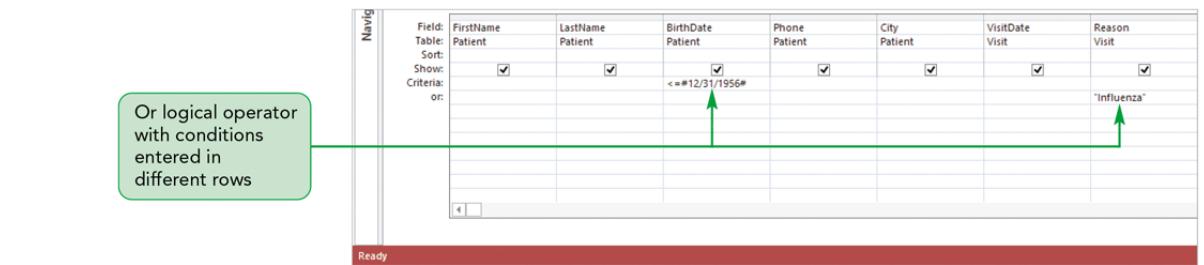
**Figure 3-28** Results of query using the And logical operator

Chatham : Database- E:\Access1\Tutorial\Chatham.accdb (Access 2007 - 2013 file format) - Access

First Name	Last Name	Date of Birth	Phone	City	Date of Visit	Reason/Diagnosis
Jan	Parker	6/3/1953	860-938-1873	Hartford	12/15/2015	Influenza
Jane	Ropiak	2/7/1950	860-938-7830	Hartford	3/3/2016	Influenza

- The Or Logical Operator
  - In the query design, either one of two conditions is satisfied or when both conditions are satisfied

**Figure 3-29** Query window with the Or logical operator



**Figure 3-30** Results of query using the Or logical operator

First Name	Last Name	Date of Birth	Phone	City	Date of Visit	Reason/Diagnosis
Patrice	Lewis	7/15/1991	860-552-5830	Windsor	11/25/2015	Influenza
Amrita	Mehta	6/3/1989	860-552-0375	Windsor	11/24/2015	Influenza
Edward	Darcy	7/15/1986	860-305-3985	Bloomfield	11/30/2015	Influenza
Pam	Fielder	12/6/1978	860-305-2689	Bloomfield	12/1/2015	Influenza
Lisa	Chang	10/5/1995	860-226-6034	Hartford	1/5/2016	Annual wellness visit
Caroline	Greene	6/12/1954	860-938-8295	Hartford	3/1/2016	Annual wellness visit
Chaney	Franklin	1/18/1954	860-552-2409	Windsor	4/15/2016	Elevated blood lipids-monitoring meds
Chaney	Franklin	1/18/1954	860-552-2409	Windsor	3/23/2016	Elevated blood lipids-monitoring meds
Chaney	Franklin	1/18/1954	860-552-2409	Windsor	2/24/2016	Elevated blood lipids-monitoring meds
Ian	Parker	6/3/1953	860-938-1873	Hartford	1/14/2016	Hypertension monitoring
Ian	Parker	6/3/1953	860-938-1873	Hartford	12/15/2015	Influenza
Malcolm	Belanger	10/17/1950	860-637-3927	West Hartford	11/30/2015	Hypertension
Malcolm	Belanger	10/17/1950	860-637-3927	West Hartford	1/8/2016	Hypertension monitoring
Jane	Ropak	2/7/1950	860-938-7830	Hartford	1/11/2016	COPD management visit
Jane	Ropak	2/7/1950	860-938-7830	Hartford	3/3/2016	Influenza
Jane	Ropak	2/7/1950	860-938-7830	Hartford	4/14/2016	COPD management visit
Amber	Finnerty	5/7/1946	860-226-4930	Hartford	1/26/2016	Annual wellness visit
Susan	Hawes	6/22/1945	860-661-7192	Hartford	4/12/2016	Joint pain/arthritis
Davis	Hallick	3/26/1944	860-552-4495	Windsor	2/8/2016	Gastric reflux
Davis	Hallick	3/26/1944	860-552-4495	Windsor	12/7/2015	Annual wellness visit
Davis	Hallick	3/26/1944	860-552-4495	Windsor	2/22/2016	Follow-up - gastric reflux
Olivia	Brown	11/24/1943	860-938-7482	Hartford	2/16/2016	Plantar facitis
Olivia	Brown	11/24/1943	860-938-7482	Hartford	1/24/2015	Gastroenteritis - follow up
Lucia	Swenson	5/1/1943	860-226-0293	Hartford	11/17/2015	Gastroenteritis
Lucia	Swenson	5/1/1943	860-226-0293	Hartford		

## Changing a Datasheet's Appearance

- You can make many formatting changes to a datasheet to improve its appearance or readability
  - Font type, size, color, alignment of text, apply different colors to the rows and columns
- Modifying the Font Size
  - Depending on the size of the monitor you are using or the screen resolution, you might need to increase or decrease the size of the font to view more or fewer columns of data
- Changing the Alternate Row Color in a Datasheet
  - Access uses themes to format the objects in a database. A theme is a predefined set of formats including colors, fonts, and other effects that enhance an object's appearance and usability
    - The Office theme, which formats every other row in a datasheet with a gray background color to distinguish one row from another, is the default

Figure 3-31 Gallery of color choices for alternate row color

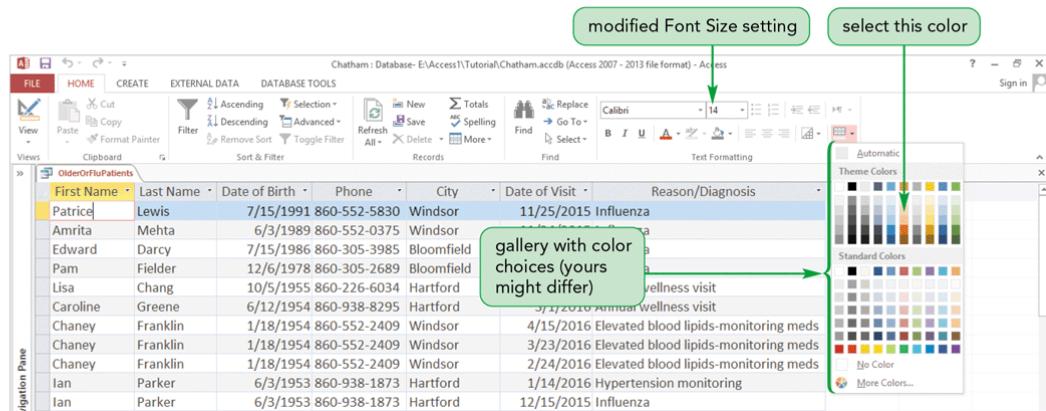
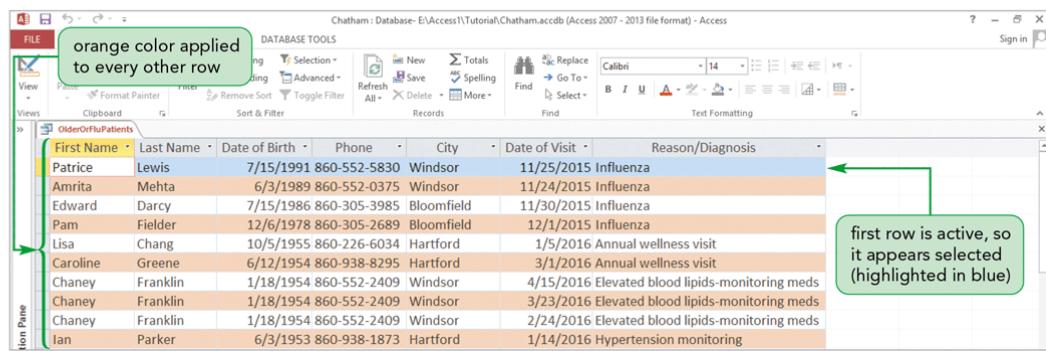


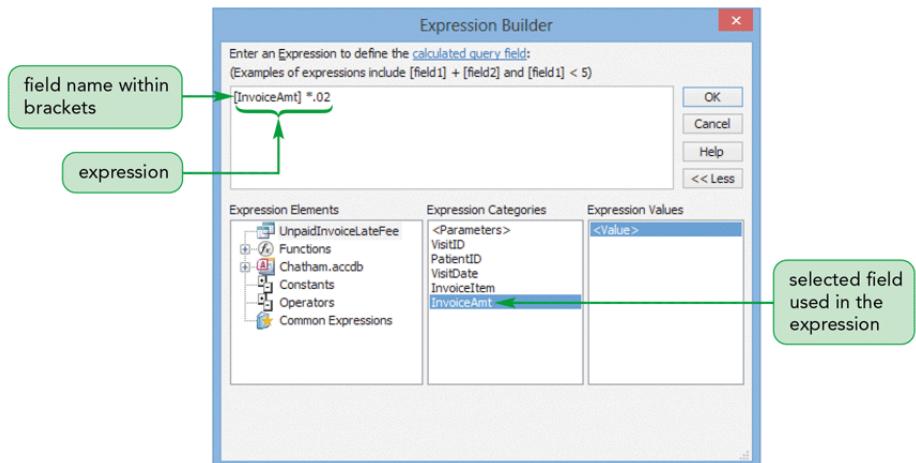
Figure 3-32 Datasheet formatted with alternate row color



## 5. Creating a Calculated Field

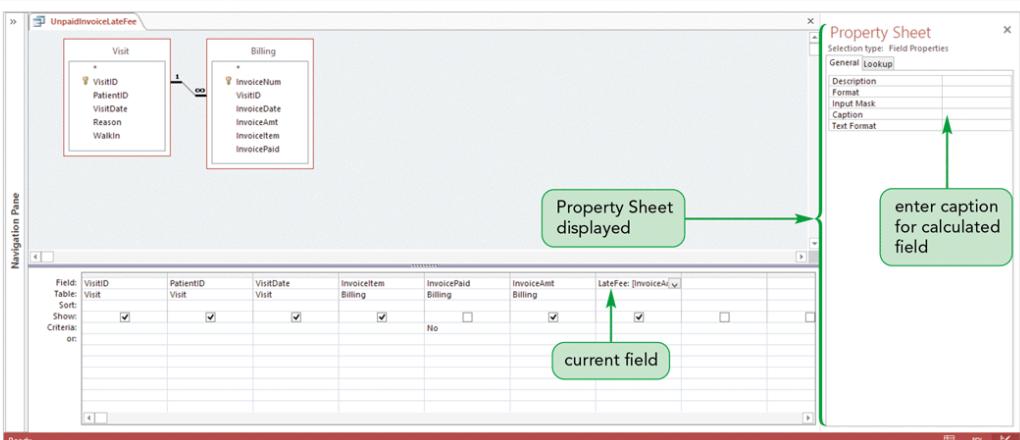
- Queries can perform calculations
  - Must define an expression containing a combination of database fields, constants, and operators
  - A calculated field is a field that displays the results of an expression but it does not exist in a database
  - The Zoom box is a dialog box that you can use to enter text, expressions, or other values
  - Expression Builder is an Access tool that makes it easy for you to create an expression
    - It contains a box for entering the expression, an option for displaying and choosing common operators, and one or more lists of expression elements, such as table and field names

**Figure 3-33** Completed expression for the calculated field



- **Formatting a Calculated Field**
  - You can specify a particular format for a calculated field, just as you can for any field, by modifying its properties

**Figure 3-34** Property Sheet for the calculated field



**Figure 3-35** Datasheet displaying the calculated field

The screenshot shows the Microsoft Access datasheet with the following details:

- The ribbon tabs include Home, Insert, Design, Tools, and View.
- The "Design" tab is selected.
- The "Table/Query" section of the ribbon shows "UnpaidInvoiceLateFee".
- The "Navigation Pane" on the left displays tables: Visit and Billing.
- The datasheet displays a table with columns: Visit ID, Patient ID, Date of Visit, Invoice Item, Invoice Amt, and Late Fee.
- A callout bubble labeled "specified caption for the calculated field" points to the "Late Fee" column header.
- A callout bubble labeled "calculated field values" points to the data entries in the "Late Fee" column.
- The status bar at the bottom shows "Record: 1 of 76" and other navigation icons.

## 6. Using Aggregate Functions

- You can calculate statistical information, such as totals and averages, on the records displayed in a table datasheet or selected by a query

- Use the Access Aggregate functions which perform arithmetic operations on selected records in a database

Figure 3-36 Frequently used aggregate functions

Aggregate Function	Determines	Data Types Supported
Average	Average of the field values for the selected records	AutoNumber, Currency, Date/Time, Number
Count	Number of records selected	AutoNumber, Currency, Date/Time, Long Text, Number, OLE Object, Short Text, Yes/No
Maximum	Highest field value for the selected records	AutoNumber, Currency, Date/Time, Number, Short Text
Minimum	Lowest field value for the selected records	AutoNumber, Currency, Date/Time, Number, Short Text
Sum	Total of the field values for the selected records	AutoNumber, Currency, Date/Time, Number

© 2014 Cengage Learning

### Working with Aggregate Functions Using the Total Row

- To quickly perform a calculation using an aggregate function in a table or query datasheet, you can use the Totals button in the Records group on the HOME tab
- When you click this button, a row labeled “Total” appears at the bottom of the datasheet
- Choose one of the aggregate functions

Figure 3-37 Using aggregate functions in the Total row

The screenshot shows a Microsoft Access datasheet for the 'Billing' table. The table contains columns for Invoice Num, Visit ID, Invoice Date, Invoice Amt, Invoice Item, Invoice Paid, and Click to A. A 'Navigation Pane' is visible on the left. A 'Total' row is added at the bottom of the table. The 'Invoice Amt' column in the total row has a dropdown menu open, displaying options: None, Sum, Average, Count, Maximum, Minimum, Standard Deviation, and Variance. A callout bubble points to this menu with the text 'menu of aggregate functions for a Currency field'. Another callout bubble points to the 'Invoice Amt' cell in the total row with the text 'Total row in the datasheet'. A third callout bubble points to the 'Invoice Amt' cell in the first data row with the text 'current field'.

### Creating Queries with Aggregate Functions

- Aggregate functions operate on the records that meet a query's selection criteria
- You specify an aggregate function for a specific field, and the appropriate operation applies to that field's values for the selected records

Figure 3-38 Total row inserted in the design grid

The screenshot shows the Microsoft Access Query Design grid. The 'Totals' row is highlighted in red. The 'InvoiceAmt' field is listed three times in the 'Field' column, each with 'Billing' in the 'Table' column and 'Group By' in the 'Group By' column. The 'Criteria' column for all three entries has a checkbox checked. A callout bubble points to the 'Total' row with the text 'Total row'. Another callout bubble points to the 'InvoiceAmt' field in the first row with the text 'InvoiceAmt field included three times in the design grid'.

Figure 3-39 Query with aggregate functions entered

The screenshot shows the Microsoft Access query design grid. A green callout box labeled "functions entered and columns resized" points to the first column of the grid. The first column contains the following data:

Field	MinimumInvoiceAmt: InvoiceAmt	AverageInvoiceAmt: InvoiceAmt	MaximumInvoiceAmt: InvoiceAmt
Table: Billing	Billing	Avg	Billing
Total: Min			Max
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			
or:			

Figure 3-40 Result of the query using aggregate functions

The screenshot shows the Microsoft Access datasheet. The results of the query are displayed in three columns:

Minimum Invoice Amt	Average Invoice Amt	Maximum Invoice Amt
\$30.00	\$101.49	\$450.00

## Using Record Group Calculations

- In addition to calculating statistical information on all or selected records, you can calculate statistics for groups of records
- The **Group By operator** divides the selected records into groups based on the values in the specified field
  - Those records with the same value for the field are grouped together, and the datasheet displays one record for each group
  - Aggregate functions, which appear in the other columns of the design grid, provide statistical information for each group

Figure 3-41 Aggregate functions grouped by WalkIn

The screenshot shows the Microsoft Access datasheet. The results are grouped by the "Walk-in?" field. A green callout box labeled "record groups" points to the "Walk-in?" column header. A green callout box labeled "aggregate function results" points to the data rows under the "Walk-in?" column.

Walk-in?	Minimum Invoice Amt	Average Invoice Amt	Maximum Invoice Amt
□	\$35.00	\$115.78	\$450.00
✓	\$30.00	\$88.16	\$150.00

## Working with the Navigation Pane

- The Navigation Pane is the main area for working with the objects in a database
- Provides options for grouping database objects in various ways to suit your needs
- Divides database objects into categories, and each category contains groups
- The default category is **Object Type**, which arranges objects by type—tables, queries, forms, and reports
- The default group is **All Access Objects**, which appears at the top of the Navigation Pane

Figure 3-42 Navigation Pane menu

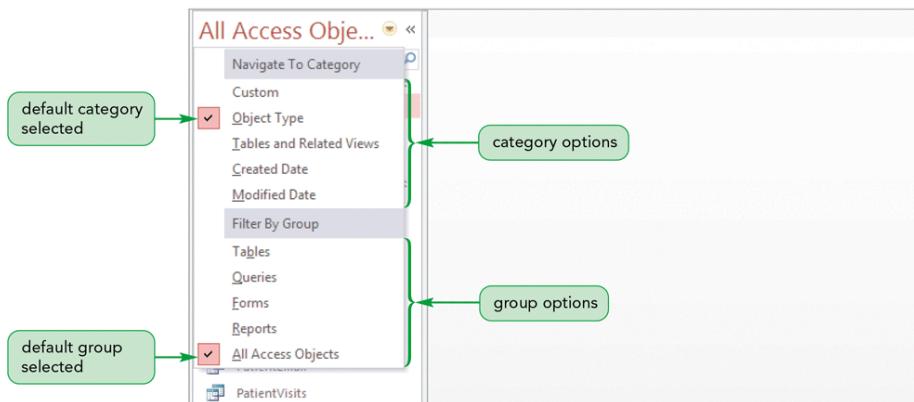
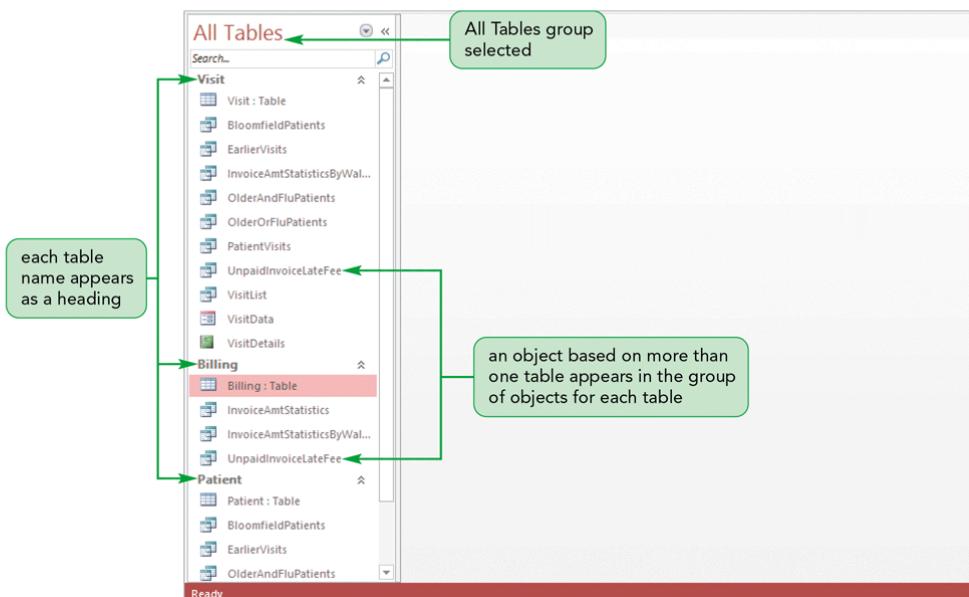


Figure 3-43 Database objects grouped by table in the Navigation Pane



## Activity 2

- In the design grid, where do you place the conditions for 2 different fields when you use the AND logical operator, and where do you place them when you use the OR operator?
  - In the same Criteria row; in different Criteria rows
- Which Access tool do you use to create an expression for a calculated field in a query?
  - Expression builder

## 7. Creating Advanced Queries and Enhancing Table Design

### Reviewing the Clinic Database

- The Navigation Pane displays the objects grouped by object type
  - Each object name has a prefix tag—a `tbl` prefix tag for tables, a `qry` prefix tag for queries, a `frm` prefix tag for forms, and a `rpt` prefix tag for reports
  - All three characters in each prefix tag are lower case. The word immediately after the three-character prefix begins with an upper case letter

- Using object prefix tags, you can readily identify the object type, even when the objects have the same base name
- Object names have no spaces, because other database management systems do not permit making it easy during conversions to those systems

### Using Pattern Match in a Query

- A **pattern match** selects records with a value for the designated field that matches the pattern of a simple condition value
- The **Like comparison operator** selects records by matching field values to a specific pattern that includes one or more of these wildcard characters: asterisk (\*), question mark (?), and number symbol (#)
- The asterisk represents any string of characters, the question mark represents any single character, and the number symbol represents any single digit

Figure 5-2 Record selection based on matching a specific pattern

The screenshot shows the Microsoft Access query design grid. The grid has columns for Field, Table, Sort, Show, Criteria, and Or. In the Criteria column, there is a checkmark next to the 'Phone' field, with an arrow pointing from the text 'Like "860"' to this checkmark. A green callout box labeled 'pattern match selection criterion' is positioned over the criteria row. The 'Or' column is empty.

Field:	PatientID tblPatient	LastName tblPatient	FirstName tblPatient	Parent tblPatient	BirthDate tblPatient	Phone tblPatient
Show:	<input checked="" type="checkbox"/>					
Criteria:						
Or:						

Figure 5-3 tblPatient table records for area code 860

The screenshot shows the Microsoft Access datasheet for the 'qry860AreaCode' query. The table has columns: Patient ID, Last Name, First Name, Parent, Date of Birth, Phone, Address, City, State, Zip, and EmailAddress. A green callout box labeled '46 records total' is at the bottom left. An arrow points from this box to the bottom of the table. Another arrow points from the text 'scroll down to see more records that match the criteria' to the bottom right of the table. The table contains 46 records, mostly for patients with area code 860.

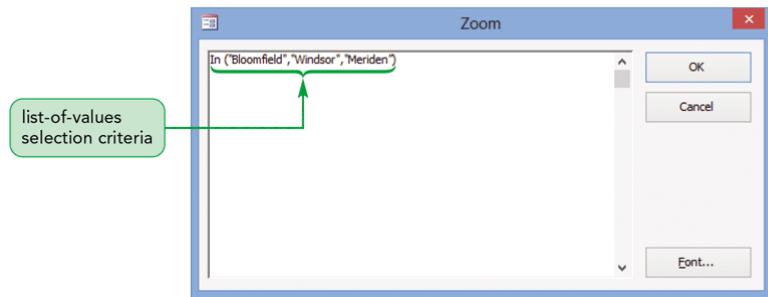
Patient ID	Last Name	First Name	Parent	Date of Birth	Phone	Address	City	State	Zip	EmailAddress
22500	Student	Last	Student	2/28/1994	8609382832	501 Perkins Dr	Hartford	CT	06120	student2@example.com
22501	Darcy	Edward		7/15/1986	8603053985	723 Oxford Ave	Bloomfield	CT	06002	edarcy@cengage.com
22502	Mendez	Andreas		3/1/1994	8605526893	57 Wallace Rd	Windsor	CT	06095	amendlos@example.com
22504	Aguilar	Lillian		8/16/1998	8603745724	329 Fairchild Ave	Hartford	CT	06114	lagular@cengage.com
22505	Finnerty	Amber		5/7/1996	8602264930	37 Noyes Ct	Hartford	CT	06112	ambergl@EXAMPLE.COM
22506	Gorski	Drea	Samson, Ma	2/19/2005	8603058394	83 Everett Ln	Bloomfield	CT	06002	
22507	Weiss	Matthew	Weiss, Jordi	6/7/1997	8604267492	38 German St	Hartford	CT	06109	matt.weiss@cengage.com
22509	Kervin	Steve		4/6/1993	8609380025	49 Davenport St	Hartford	CT	06120	skerwin@example.com
22511	Castro	Daniel		9/23/1993	8606370430	61 Osmond Way	West Hartford	CT	06117	d_castro@cengage.com
22512	Chang	Lisa		10/5/1995	8602266034	731 Macon Rd	Hartford	CT	06112	lchang14@example.com
22513	Smith	Troy		1/31/1996	8603050384	16 Ravine Rd	Bloomfield	CT	06002	troysmith9@example.net
22514	Parker	Ian		6/3/1998	8609381873	12 Adelbert St	Hartford	CT	06120	ian_parker@example.com
22517	O'Brien	Henry		12/10/1940	8606379203	58 Redmond Dr	West Hartford	CT	06117	hobrien32@example.com
22518	Torres	Sera	Torres, Gina	4/9/2008	8609382098	27 Reno Dr	Hartford	CT	06120	serat83@example.com
22519	Belanger	Malcolm		10/17/1950	8606373927	723 Nicola Ave	West Hartford	CT	06117	
22520	Hallick	Dave		3/26/1944	8605524495	84 Churchill Pl	Windsor	CT	06095	dhallick@EXAMPLE.COM
22521	Engber	Cathy	Engber, Jim	4/7/2006	8603053048	58 Deering Pl	Bloomfield	CT	06002	cengber2@cengage.com
22522	Li	Siyang		7/25/1986	8603056548	225 Krauss Rd	Bloomfield	CT	06002	lisiy3@example.com
22523	Fraser	Nancy		11/8/1977	8605527392	7 Quinn Dr	Windsor	CT	06095	
22526	Swenson	Lucia		5/1/1943	8602260293	83 Osage Ave	Hartford	CT	06112	
22527	Lee	Hwan		8/25/1987	8606613974	153 Agnes Ct	Hartford	CT	06105	hw.lee@example.org
22529	Calderon	Robert		5/30/1961	8605522873	92 Gaston Ave	Windsor	CT	06095	
22534	Hawes	Susan		2/28/1980	8603742887	74 Brayton Dr	Hartford	CT	06114	jesswallner@example.net
22535	Delgado	Alex		6/22/1945	8605521793	26 Edwin Ct	Windsor	CT	06095	crowebblack@example.net
22536	Caputo	Michael		7/16/1960	8605521793	48 Warwick St	Windsor	CT	06105	susan.hawes4@example.com
				10/19/1998	8603749347	96 Vega Dr	Hartford	CT	06114	

### Using a List-of-Values Match in a Query

- A list-of-values match selects records whose value for the designated field matches one of two or more simple condition values
  - Could include several Or conditions in the design grid, but there is an easier and clearer way to do this
    - The In comparison operator lets you define a condition with a list of two or more values for a field that select and include those record in the query results

Figure 5-4

Record selection based on matching field values to a list of values

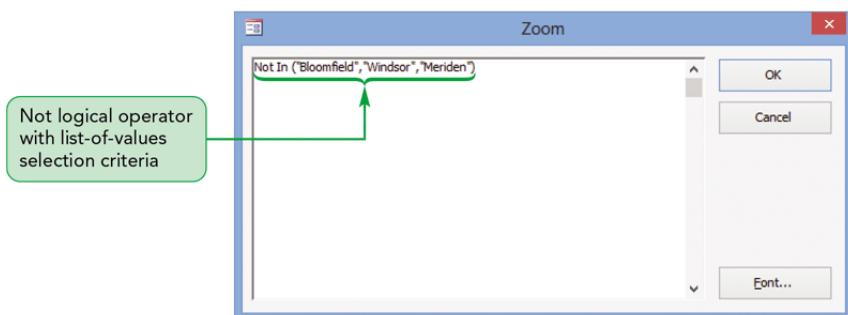


### Using the Not Logical Operator in a Query

- The **Not logical operator** negates a criterion or selects records for which the designated field does not match the criterion

Figure 5-5

Record selection based on not matching a list of values



### Using an AutoFilter to Filter Data

- You can use the AutoFilter feature to choose restrictions faster and with more flexibility than using the condition and Or logical operator

Figure 5-6

Using an AutoFilter to filter records in the query recordset

current field

filter choices for current field

Last Name	First Name	City	Date of Visit	Walk-in?	Reason/Diagnosis
Li	Siyang	Bloomfield	1/26/2016	<input checked="" type="checkbox"/>	rhinitis - environmental
Weiss	Matthew	Hartford	1/28/2016	<input checked="" type="checkbox"/>	etes mellitis Type 2 - initial diagnosis
Ingram	Julia	Hartford	2/1/2016	<input checked="" type="checkbox"/>	sverse fracture of left ulna
Student Last	Student First	Hartford	2/2/2016	<input checked="" type="checkbox"/>	aine
Mehta	Amrita	Windsor	2/8/2016	<input checked="" type="checkbox"/>	ienza
Lewis	Patrice	Windsor	2/9/2016	<input checked="" type="checkbox"/>	ienza
Darcy	Edward	Bloomfield	2/11/2016	<input checked="" type="checkbox"/>	ienza
Fielder	Pam	Bloomfield	2/18/2016	<input checked="" type="checkbox"/>	ienza
Parker	Ian	Hartford	3/7/2016	<input checked="" type="checkbox"/>	ienza
Ingram	Julia	Hartford	3/9/2016	<input checked="" type="checkbox"/>	ow-up - cast removal
Chang	Lisa	Hartford			ual wellness visit
Shaw	Daniel	West Hartford			opharyngitis
Lewis	Patrice	Windsor			e sinusitis
Smith	Troy	Bloomfield			en leg
Phillips	Aquon	Hartford			e viral rhinopharyngitis
Fraser	Nancy	Windsor			opharyngitis
Sutherland	Max	Windsor			Acute viral rhinopharyngitis
Rowe	Christina	Windsor			Nasopharyngitis
Brown	Olivia	Meriden			Plantar faciitis
Diaz	Anna	West Hartford			Brachial neuritis
Hallick	Davis	Windsor			Gastric reflux
Lee	Hwan	Hartford			UTI
Goldberg	Robert	Windsor			Sinusitis
Kirk	Isobel	Bloomfield			Hypertension
Cruz	Magdalena	West Hartford			Migrane headache
Lewis	Patrice	Windsor			Eczema erythematous
Taylor	Jerome	Hartford			Onychocryptosis

Figure 5-7

Using an AutoFilter to filter records in the query recordset

The screenshot shows a Microsoft Access window with the ribbon menu open. The 'HOME' tab is selected. In the 'Sort & Filter' group, the 'Filter' button is highlighted with a red box, and a green callout bubble points to it with the text 'toggles filter off and on'. Below the ribbon, a query results grid titled 'qryJanuaryWalkin' is displayed. The grid has columns for Last Name, First Name, City, Date of Visit, Walk-in?, and Reason/Diagnosis. A green callout bubble points to the 'Navigation Pane' icon on the left side of the grid with the text 'Hartford and West Hartford patients selected'. At the bottom of the grid, there is a 'Filtered' indicator with a green arrow pointing down to it.

Last Name	First Name	City	Date of Visit	Walk-in?	Reason/Diagnosis
Student Last	Student First	Hartford	11/17/2015	<input checked="" type="checkbox"/>	Migraine
Weiss	Matthew	Hartford	11/9/2015	<input checked="" type="checkbox"/>	Diabetes mellitus Type 2 - initial diagnosis
Kervin	Steve	Hartford	4/4/2016	<input checked="" type="checkbox"/>	Tinea pedis
Chang	Lisa	Hartford	1/5/2016	<input type="checkbox"/>	Annual wellness visit
Parker	Ian	Hartford	12/15/2015	<input checked="" type="checkbox"/>	Influenza
O'Brien	Henry	West Hartford	4/11/2016	<input checked="" type="checkbox"/>	Idiopathic abdominal pain
Torres	Sera	Hartford	4/1/2016	<input checked="" type="checkbox"/>	Conjunctivitis
Lee	Hwan	Hartford	2/9/2016	<input checked="" type="checkbox"/>	UTI
Wallner	Jessica	Hartford	3/14/2016	<input checked="" type="checkbox"/>	UTI
Hawes	Susan	Hartford	4/12/2016	<input checked="" type="checkbox"/>	Joint pain/arthritis
Caputo	Michael	Hartford	4/8/2016	<input checked="" type="checkbox"/>	Laceration of left eyelid - sutured
Taylor	Jerome	Hartford	3/9/2016	<input checked="" type="checkbox"/>	Onychocryptosis
Diaz	Anna	West Hartford	2/2/2016	<input checked="" type="checkbox"/>	Brachial neuritis
Ingram	Julia	Hartford	11/12/2015	<input checked="" type="checkbox"/>	Transverse fracture of left ulna
Ingram	Julia	Hartford	1/4/2016	<input type="checkbox"/>	Follow-up - cast removal
Cruz	Magdalena	West Hartford	2/18/2016	<input checked="" type="checkbox"/>	Migraine headache
Phillips	Aquon	Hartford	1/25/2016	<input checked="" type="checkbox"/>	Acute viral rhinopharyngitis
Shaw	Daniel	West Hartford	1/11/2016	<input checked="" type="checkbox"/>	Nasopharyngitis

## Assigning a Conditional Value to a Calculated Field

- Null Fields are fields in a record does not contain any information at all
- A field in a record that contains any data at all—even a single space—is nonnull
  - To combine nonnull LastName and FirstName fields, you can use the expression LastName & ", " & FirstName
    - The **& (ampersand)** operator is a concatenation operator that joins text expressions
    - **Concatenation** refers to joining two or more text fields or characters encapsulated in quotes
- The **IIf** (Immediate If) function assigns one value to a calculated field or control if a condition is true, and a second value if the condition is false
  - The **IIf function** has three parts: a condition that is true or false, the result when the condition is true, and the result when the condition is false
  - Each part of the IIf function is separated by a comma
  - The **IsNull function** tests a field value or an expression for a null value; if the field value or expression is null, the result is true; otherwise, the result is false

Figure 5-8 IIf function inserted for the calculated field

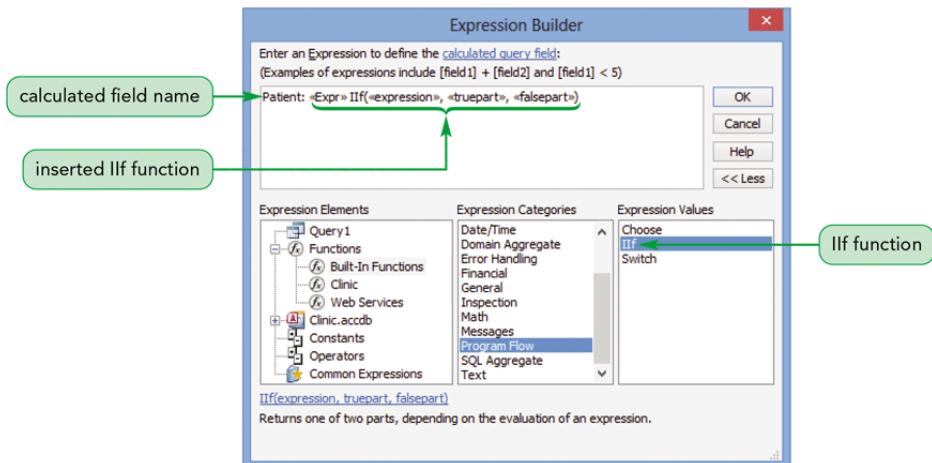


Figure 5-9 After entering the condition for the calculated field's IIf function

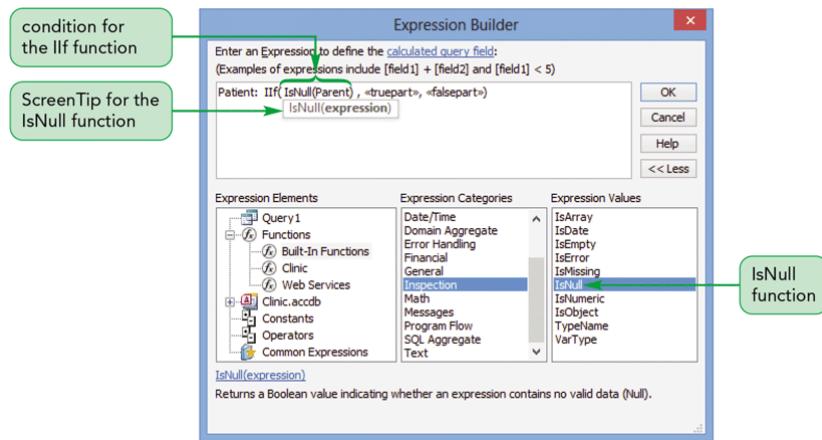
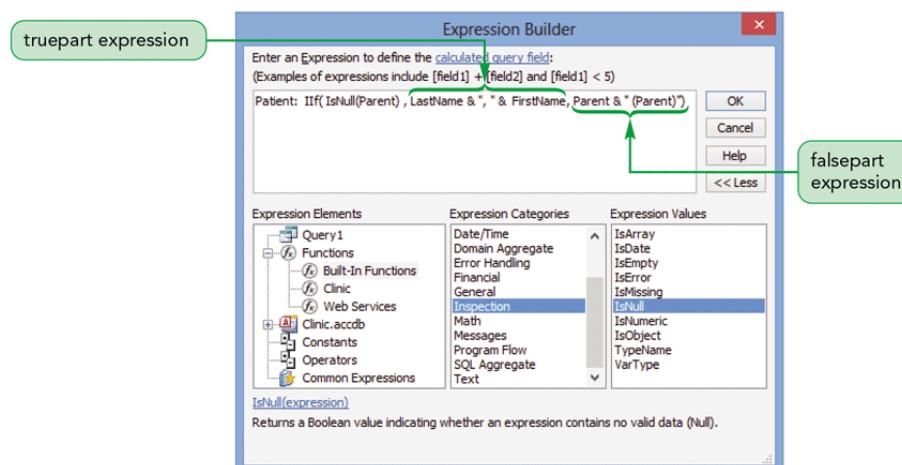


Figure 5-10 Completed calculated field



**Figure 5-11** Property sheet for the Patient calculated field

The screenshot shows the Microsoft Access interface with a query named 'Query1'. The 'tblPatient' table is selected. In the query grid, there is a calculated field named 'Patient' with the formula: `IF([Parent] IS NULL,[PatientID]&[LastName], [FirstName])`. The 'Field' column shows the formula, and the 'Properties' pane on the right displays the properties for this calculated field.

**Figure 5-12** Completed query displaying the Patient calculated field

The screenshot shows the completed query 'qryPatientsByName'. The results show patient names concatenated from LastName and FirstName. A note indicates that for null Parent values, the FirstName is null. Another note indicates that for nonnull Parent values, the additional '(Parent)' text is added to the name. The table includes columns for Patient ID, Last Name, First Name, Parent, and Date of Birth.

Patient	Patient ID	Last Name	First Name	Parent	Date of Birth
Aguilar, Lilian	22504	Aguilar	Lilian		8/16/1938
Belanger, Malcolm	22519	Belanger	Malcolm		10/17/1950
Billings, Claire	22541	Billings	Claire		11/16/1990
Booker, Thomas	22510	Booker	Thomas		8/25/1966
Boucher, Sam	22543	Boucher	Sam		3/11/1975
Brown, Olivia	22530	Brown	Olivia		11/24/1943
Caputo, Michael	22536	Caputo	Michael		10/19/1998
Castro, Daniel	22511	Castro	Daniel		9/23/1933
Chang, Lisa	22512	Chang	Lisa		10/5/1955
Cruz, Magdalena	22550	Cruz	Magdalena		7/24/1984
Darcy, Edward	22501	Darcy	Edward		7/15/1986
Delgado, Alex	22535	Delgado	Alex		7/16/1960
Diaz, Anna	22542	Diaz	Anna		9/25/1987
Engber, Jim (Parent)	22521	Engber	Cathy	Engber, Jim	4/7/2006
Fielder, Pam	22549	Fielder	Pam		12/6/1978
Finnerty, Amber	22505	Finnerty	Amber		5/7/1946
Franklin, Chaney	22551	Franklin	Chaney		1/18/1954
Fraser, Nancy	22523	Fraser	Nancy		11/8/1977
Garrett, Ashley	22552	Garrett	Ashley		3/24/1989

## 8. Creating a Parameter Query

- A parameter query displays a dialog box that prompts the user to enter one or more criteria values when the query is run
  - The value entered into the prompt causes the query to select only those records with field value from the table

**Figure 5-13** Specifying the prompt for the parameter query

The screenshot shows the 'Enter Parameter Value' dialog box. It contains a text input field with the placeholder 'Type the city:' and two buttons: 'OK' and 'Cancel'. A callout bubble points to the text input field with the label 'prompt text enclosed in brackets'.

**Figure 5-14** Enter Parameter Value dialog box



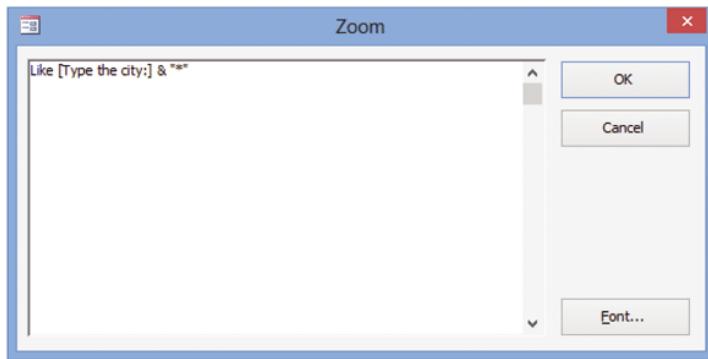
**Figure 5-15 Results of the parameter query**

Patient	Patient ID	Last Name	First Name	Parent	Date of Birth	Phone	Address	City	State	Zip	EmailAddress
Garrett, Ashley	22552	Garrett	Ashley		3/24/1989	4755528429	372 Higbee Ct	Waterbury	CT	06704	agarrett@example.org
Rodriguez, Maria	22525	Rodriguez	Maria		2/11/1936	4755529023	624 Noyes St	Waterbury	CT	06704	marodrig13@example.net

### Creating a More Flexible Parameter Query

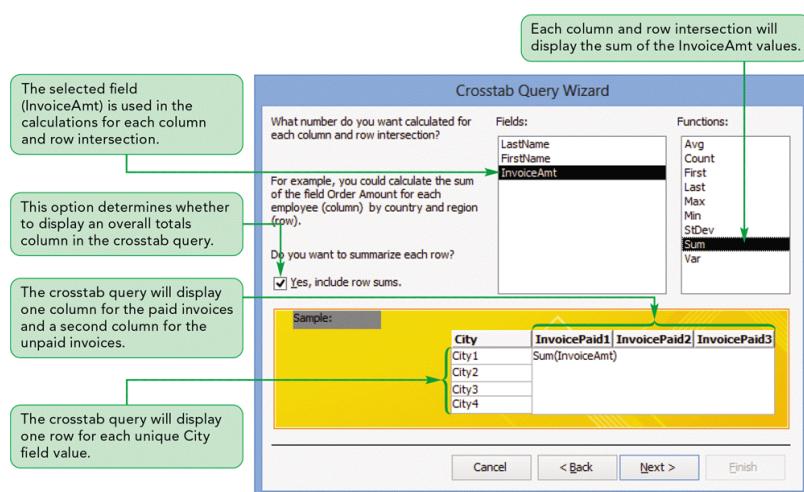
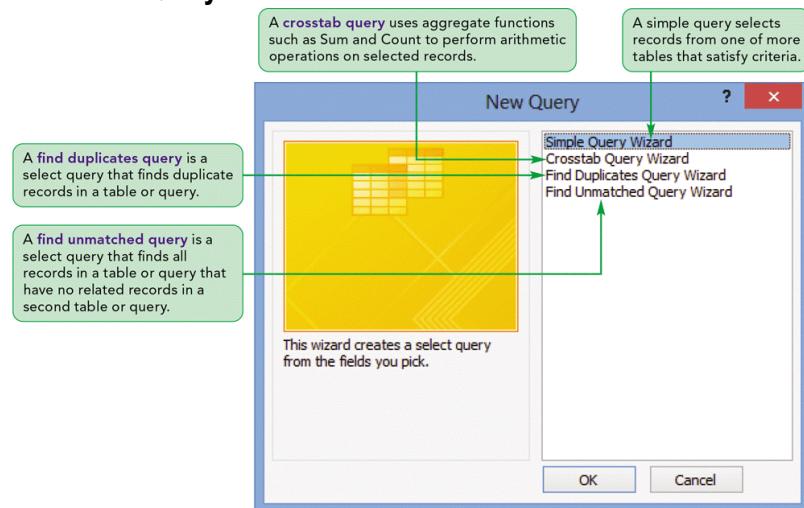
- Most users want a parameter query to display the records that match the parameter value the user enters or to display all records when the user doesn't enter a parameter value
  - To provide this functionality, you can change the value in the Criteria box in the design grid for the specified column
  - Prefix the Like operator to the original criterion and concatenate the criterion to a wildcard character

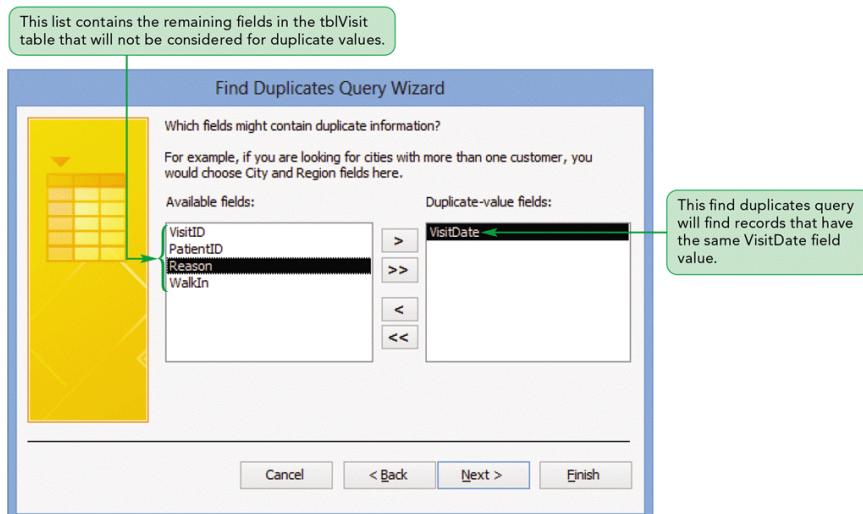
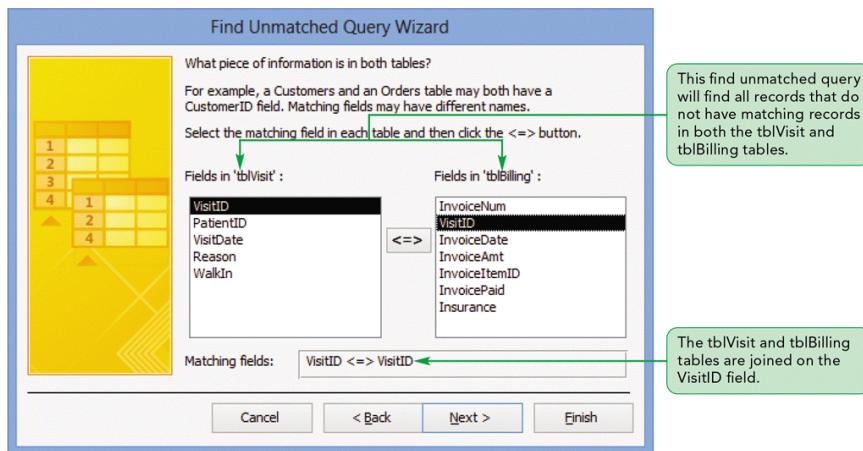
Modified City Criteria value in the Zoom dialog box



## 9. Use query wizards to create a crosstab query, a find duplicates query, and a find unmatched query

### Advanced Query Wizards





## ■ Creating a Crosstab Query

**Figure 5-17 Aggregate functions used in crosstab queries**

Aggregate Function	Definition
Avg	Average of the field values
Count	Number of the nonnull field values
First	First field value
Last	Last field value
Max	Highest field value
Min	Lowest field value
StDev	Standard deviation of the field values
Sum	Total of the field values
Var	Variance of the field values

©2014 Cengage Learning

**Figure 5-18 Comparing a select query to a crosstab query**

The figure shows two side-by-side tables in Microsoft Access.

**Select Query Results:** A grid titled "qryPatientsAndInvoices". It has columns: Last Name, First Name, City, Invoice Amt, and Invoice Paid. A green box labeled "results of a select query" points to the top-left of the grid. A green box labeled "individual West Hartford records" points to a row where the City is "West Hartford". A green box labeled "West Hartford record with unpaid invoice" points to a row where the City is "West Hartford" and the Invoice Paid column is empty. A green box labeled "West Hartford records with paid invoices" points to a row where the City is "West Hartford" and the Invoice Paid column contains a checkmark.

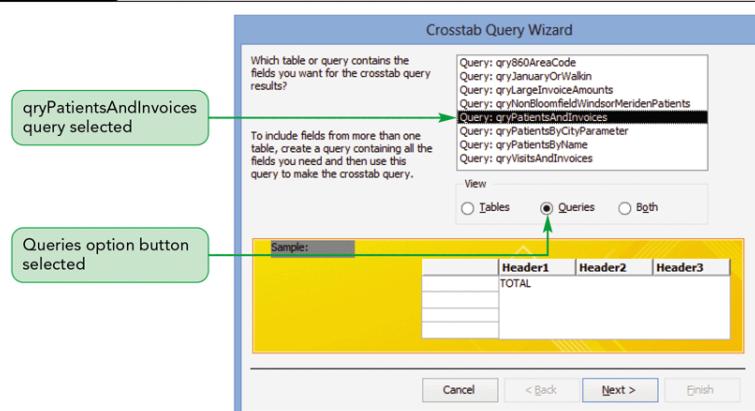
Last Name	First Name	City	Invoice Amt	Invoice Paid
Lewis	Patrice	Windsor	\$85.00	✓
Lewis	Patrice	Windsor	\$48.00	✓
Belanger	Malcolm	West Hartford	\$100.00	□
Darcy	Edward	Bloomfield	\$100.00	✓
Darcy	Edward	Bloomfield	\$85.00	✓
Fielder	Pam	Bloomfield	\$100.00	✓
Fielder	Pam	Bloomfield	\$85.00	□
Fielder	Pam	Bloomfield	\$250.00	□
Castro	Daniel	West Hartford	\$100.00	✓
Hallick	Davis	Windsor	\$100.00	□
Hallick	Davis	Windsor	\$72.00	□
Swenson	Lucia	Hartford	\$100.00	✓
Swenson	Lucia	Hartford	\$85.00	✓
Swenson	Lucia	Hartford	\$45.00	□
Swenson	Lucia	Hartford	\$32.00	□
Parker	Ian	Hartford	\$100.00	✓
Parker	Ian	Hartford	\$85.00	✓
King	Susan	Meriden	\$100.00	□
King	Susan	Meriden	\$150.00	□
Ingram	Julia	Hartford	\$100.00	✓
Ingram	Julia	Hartford	\$75.00	□
Chang	Lisa	Hartford	\$100.00	✓
Chang	Lisa	Hartford	\$85.00	✓
Ropak	Jane	Hartford	\$100.00	✓
Ropak	Jane	Hartford	\$150.00	✓
Shaw	Daniel	West Hartford	\$100.00	✓

**Crosstab Query Results:** A grid titled "qryPatientsAndInvoicesCrosstab". It has columns: City, Total Of InvoiceAmt, Paid, and Unpaid. A green box labeled "results of a crosstab query" points to the top-left of the grid. A green box labeled "one row for West Hartford invoice amounts" points to the row where the City is "West Hartford". A green box labeled "paid invoices" points to the "Paid" column header, and a green box labeled "unpaid invoices" points to the "Unpaid" column header.

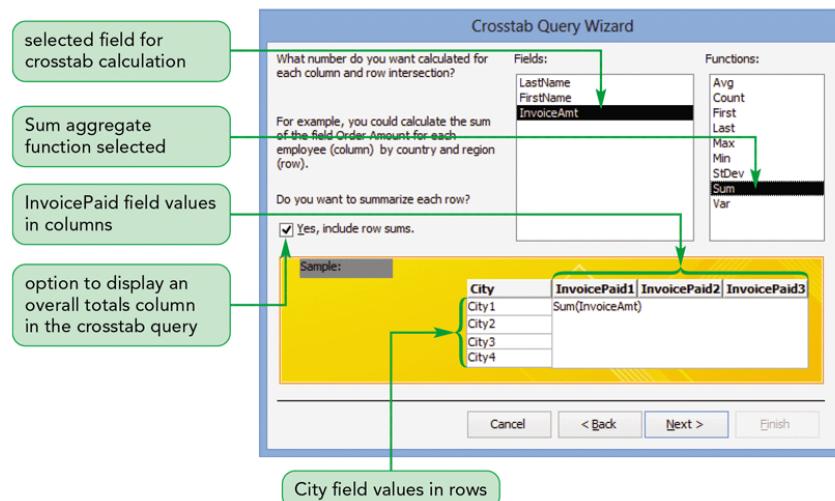
City	Total Of InvoiceAmt	Paid	Unpaid
Bloomfield	\$2,953.00	\$1,273.00	\$1,680.00
Hartford	\$9,040.00	\$4,681.00	\$4,359.00
Meriden	\$1,336.00	\$748.00	\$588.00
Waterbury	\$528.00	\$460.00	\$68.00
West Hartford	\$2,153.00	\$1,395.00	\$758.00
Windsor	\$4,693.00	\$3,591.00	\$1,102.00

- The quickest way to create a crosstab query is to use the **Crosstab Query Wizard**

**Figure 5-19 Choosing the query for the crosstab query**



**Figure 5-20** Completed crosstab query design



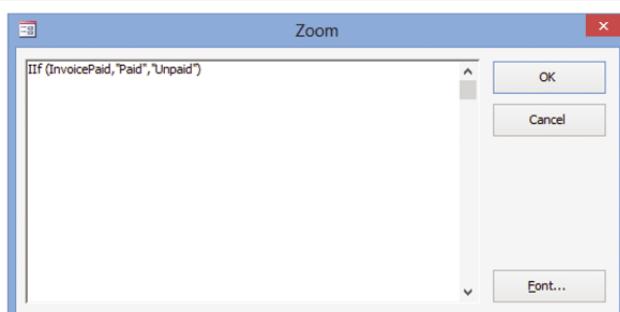
**Figure 5-21** Crosstab query recordset

City	Total Of InvoiceAmt	-1	0
Bloomfield	\$2,953.00	\$1,273.00	\$1,680.00
Hartford	\$9,040.00	\$4,681.00	\$4,359.00
Meriden	\$1,336.00	\$748.00	\$588.00
Waterbury	\$528.00	\$460.00	\$68.00
West Hartford	\$2,153.00	\$1,395.00	\$758.00
Windsor	\$4,693.00	\$3,591.00	\$1,102.00

**Figure 5-22** Crosstab query in the design grid

Field:	Table:	Total:	Crosstab:	Sort:	Criteria:
[City]	qryPatientsAndInvoices				
			Group By		
			Row Heading		

**Figure 5-23** IIf function for the crosstab query column headings



## Creating a Find Duplicates Query

- A find duplicates query is a select query that finds duplicate records in a table or query
  - You can create this type of query using the **Find Duplicates Query Wizard**
- A find duplicates query searches for duplicate values based on the fields you select when answering the Wizard's questions

## Query recordset for duplicate visit dates

Date of Visit	Visit ID	Patient ID	Reason/Diagnosis	Walk-in?
11/9/2015	1528	22507	Diabetes mellitus Type 2 - initial diagnosis	<input checked="" type="checkbox"/>
11/9/2015	1527	22522	Allergies - environmental	<input checked="" type="checkbox"/>
11/17/2015	1536	22526	Gastroenteritis	<input type="checkbox"/>
11/17/2015	1538	22500	Migraine	<input checked="" type="checkbox"/>
11/24/2015	1541	22526	Gastroenteritis - follow up	<input type="checkbox"/>
11/24/2015	1542	22537	Influenza	<input checked="" type="checkbox"/>
11/30/2015	1548	22519	Hypertension	<input type="checkbox"/>
11/30/2015	1549	22501	Influenza	<input checked="" type="checkbox"/>
1/11/2016	1569	22558	COPD management visit	<input type="checkbox"/>
1/11/2016	1570	22561	Nasopharyngitis	<input checked="" type="checkbox"/>
1/13/2016	1573	22511	Cardiac monitoring	<input type="checkbox"/>
1/13/2016	1575	22513	Broken leg	<input checked="" type="checkbox"/>
1/13/2016	1572	22560	Acute sinusitis	<input checked="" type="checkbox"/>
1/25/2016	1586	22523	Nasopharyngitis	<input checked="" type="checkbox"/>
1/25/2016	1588	22535	Hypertension	<input type="checkbox"/>
1/25/2016	1585	22555	Acute viral rhinopharyngitis	<input checked="" type="checkbox"/>
1/26/2016	1590	22505	Annual wellness visit	<input type="checkbox"/>
1/26/2016	1591	22544	Acute viral rhinopharyngitis	<input checked="" type="checkbox"/>
2/1/2016	1597	22517	Annual wellness visit	<input type="checkbox"/>
2/1/2016	1598	22530	Plantar faciitis	<input checked="" type="checkbox"/>
2/8/2016	1605	22535	Hypertension monitoring	<input type="checkbox"/>
2/8/2016	1606	22520	Gastric reflux	<input checked="" type="checkbox"/>
2/9/2016	1610	22529	Sinusitis	<input checked="" type="checkbox"/>
2/9/2016	1608	22527	UTI	<input checked="" type="checkbox"/>
2/9/2016	1607	22507	Diabetes mellitus Type 2 - serum glucose che	<input type="checkbox"/>
2/24/2016	1626	22513	Follow-up - cast removal	<input type="checkbox"/>
2/24/2016	1625	22551	Elevated blood lipids-monitoring meds	<input type="checkbox"/>

## Creating a Find Unmatched Query

- A find unmatched query is a select query that finds all records in a table or query that have no related records in a second table or query
  - Use the Find Unmatched Query Wizard to create this type of query

Figure 5-25 Selecting the common field

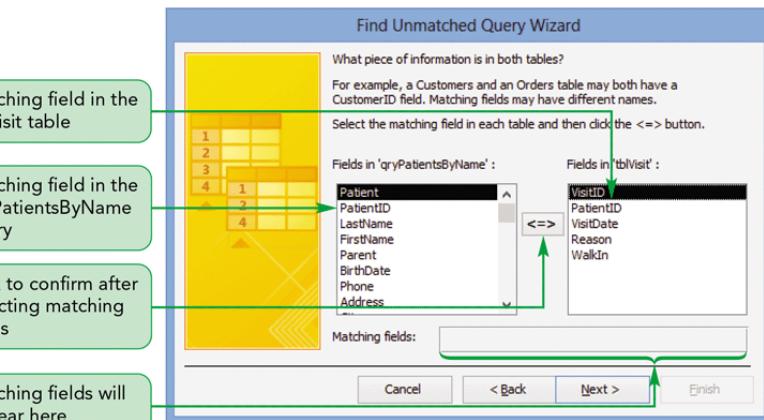


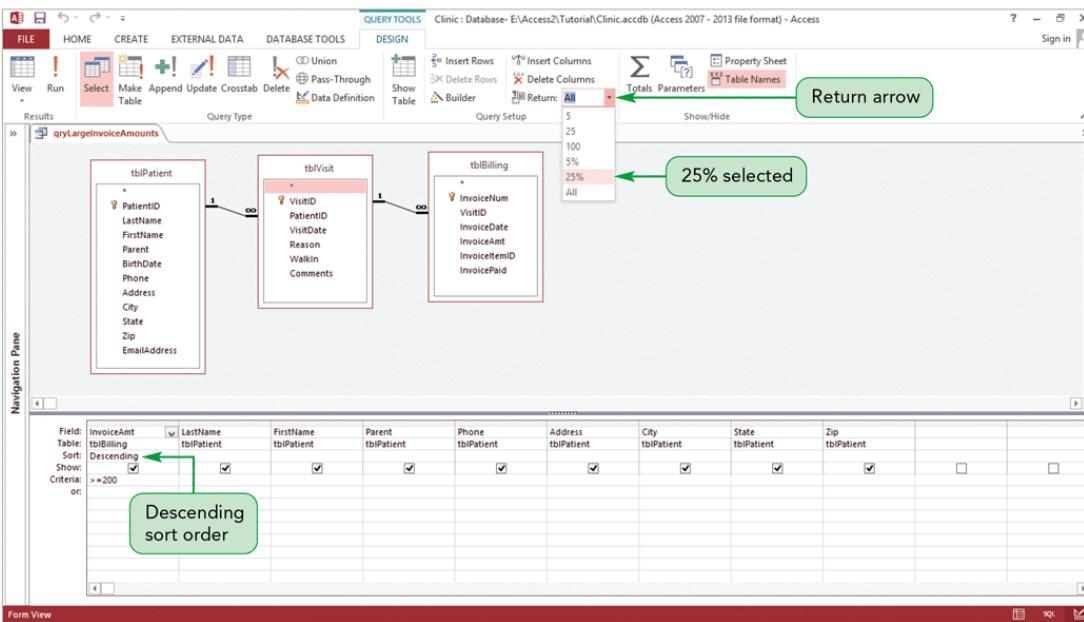
Figure 5-26 Query recordset displaying four patients without visits

Patient	Patient ID	Last Name	First Name	Parent	Date of Birth	Phone	Address	City	State	Zip	EmailAddress
Billings, Claire	22541	Billings	Claire		11/16/1990	8606612053	786 Kidder Rd	Hartford	CT	06105	
Boucher, Sam	22543	Boucher	Sam		3/11/1975	8602265920	92 Bunnell Pl	Hartford	CT	06112	sambchr2@engage.com
Grus, Mateo	22556	Grus	Mateo		3/11/1959	8602268472	18 Norman Rd	Hartford	CT	06112	matgrus5@example.com
Morales, Jose	22559	Morales	Jose		12/25/1965	8606372015	251 Lilac St	West Hartford	CT	06117	

## 10. Creating a Top Values Query

- Users might want to limit the number to a more manageable size by displaying, for example, just the first 10 records
  - The Top Values property for a query lets you limit the number of records in the query results

**Figure 5-27** Creating the top values query



**Figure 5-28** Top values query recordset

The screenshot shows the Microsoft Access ribbon with tabs FILE, HOME, CREATE, EXTERNAL DATA, and DATABASE TOOLS. The HOME tab is selected. The main area displays a table named 'qryLargeInvoiceAmounts' with the following data:

Invoice Amt	Last Name	First Name	Parent	Phone	Address	City	State	Zip
\$450.00	Caputo	Michael		8600349347	96 Vega Dr	Hartford	CT	06114
\$450.00	Taylor	Jerome		8602262037	598 Bellflower Ave	Hartford	CT	06112
\$300.00	Smith	Troy		8603050384	16 Ravine Rd	Bloomfield	CT	06002
\$300.00	Ingram	Julia		8606614937	834 Kiefer Rd	Hartford	CT	06105
*								

- Lookup Fields and Input Masks

The tblInvoiceItem query supplies the field values for the lookup field in the tblBilling table. A **lookup field** lets the user select a value from a list of possible values to enter data into the field.

Invoice Item ID	Invoice Item
DG111	Lab work
DG115	Lab work - culture
DG118	Lab work - glycated hemoglobin (A1C)
DG119	Lab work - urine glucose
DG225	Lab - culture
DG287	Lab - serum glucose
DG424	EKG with interpretation
DG532	Radiograph
OST145	Bone setting and cast
OST150	Cast of fracture

The tblBilling table contains the lookup field.

The InvoiceItemID and InvoiceItem fields from the tblInvoiceItem table are used to look up InvoiceItemID values in the tblBilling table.

Invoice Num	Visit ID	Invoice Date	Invoice Amt	Invoice Item	Invoice Paid	Insurance
35801	1527	11/10/2015	\$100.00	Office visit	<input checked="" type="checkbox"/>	\$50.00
35802	1528	11/10/2015	\$100.00	Lab - culture	<input type="checkbox"/>	\$0.00
35803	1528	11/10/2015	\$45.00	Lab - serum glucose	<input type="checkbox"/>	\$0.00
35804	1528	11/13/2015	\$238.00	EKG with interpretation	<input type="checkbox"/>	\$0.00
35805	1528	11/13/2015	\$48.00	Radiograph	<input type="checkbox"/>	\$0.00
35808	1530	11/12/2015	\$100.00	Bone setting and cast	<input type="checkbox"/>	\$0.00
35809	1530	11/12/2015	\$85.00	Cast of fracture	<input type="checkbox"/>	\$0.00
35810	1530	11/12/2015	\$65.00	Cast removal	<input type="checkbox"/>	\$0.00
35811	1530	11/13/2015	\$48.00	Pharmacy	<input type="checkbox"/>	\$0.00
35813	1535	11/13/2015	\$100.00	Office visit	<input checked="" type="checkbox"/>	\$0.00
35814	1535	11/13/2015	\$45.00	IM injection	<input type="checkbox"/>	\$0.00
35815	1535	11/16/2015	\$300.00	Physical therapy	<input type="checkbox"/>	\$0.00
35816	1535	11/16/2015	\$250.00	Phlebotomy	<input type="checkbox"/>	\$0.00
35818	1536	11/18/2015	\$100.00	Influenza vaccine	<input type="checkbox"/>	\$0.00
35819	1536	11/18/2015	\$65.00	Respiratory therapy	<input type="checkbox"/>	\$100.00
35821	1538	11/18/2015	\$100.00	Surgery	<input type="checkbox"/>	\$0.00
35822	1538	11/18/2015	\$125.00	Suture removal	<input type="checkbox"/>	\$0.00
35825	1539	11/19/2015	\$100.00	Pharmacy	<input checked="" type="checkbox"/>	\$0.00
				Office visit	<input checked="" type="checkbox"/>	\$0.00

Values in the lookup field appear in alphabetical order, sorted by Invoice Item.

Only the InvoiceItemID values are stored in the InvoiceItemID field in the tblBilling table even though the user also sees the InvoiceItem values in the datasheet.

The tblPatient table contains the field that displays values with an input mask. An **input mask** is a predefined format that is used to enter and display data in a field.

Field Name	Data Type	
PatientID	Short Text	Primary key
LastName	Short Text	
FirstName	Short Text	
Parent	Short Text	Parent or Guardian
BirthDate	Date/Time	
Phone	Short Text	
Address	Short Text	
City	Short Text	
State	Short Text	
Zip	Short Text	
EmailAddress	Short Text	

Field Properties

General	Lookup
Field Size	14
Format	
Input Mask	999-000-0000;;
Caption	
Default Value	
Validation Rule	
Validation Text	
Required	No
Allow Zero Length	Yes
Indexed	No
Unicode Compression	No
IME Mode	No Control
IME Sentence Mode	None
Text Align	General

## Creating a Lookup Field

- Data entry is easier if users do not need to remember the correct values for fields
- A lookup field lets the user select a value from a list of possible values
- Use a **Lookup Wizard** field in Access to create a lookup field in a table

Figure 5-29      Warning message for an existing table relationship



Figure 5-30      Selecting the lookup fields

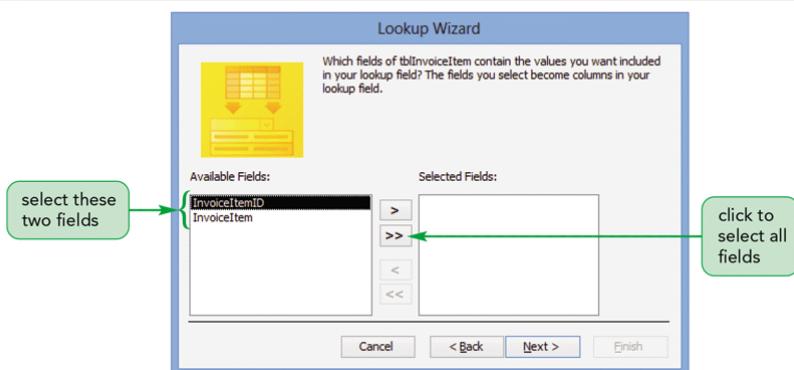


Figure 5-31 Adjusting the width of the lookup column

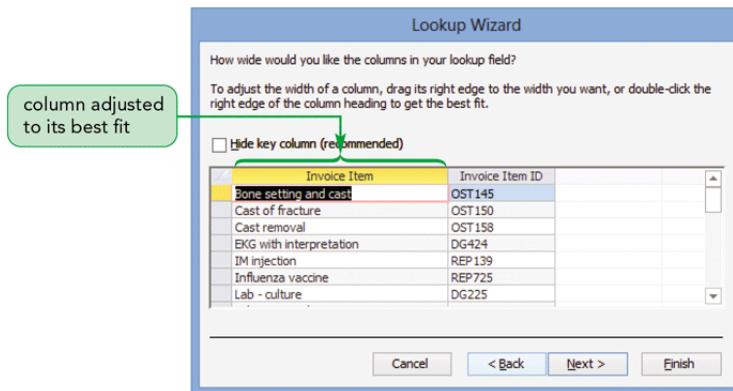


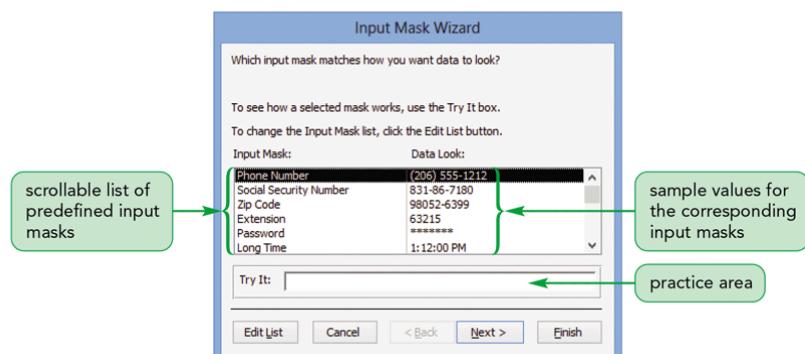
Figure 5-32 List of InvoiceItem and InvoiceItemID field values

The screenshot shows a Microsoft Access form for 'tblBilling'. In the 'Invoice Item' column of a data grid, a dropdown arrow is clicked, revealing a scrollable list of values. A callout box labeled 'scrollable list of values for the lookup field' points to this list. The list includes items like 'Office visit', 'Pharmacy', 'Lab - culture', etc.

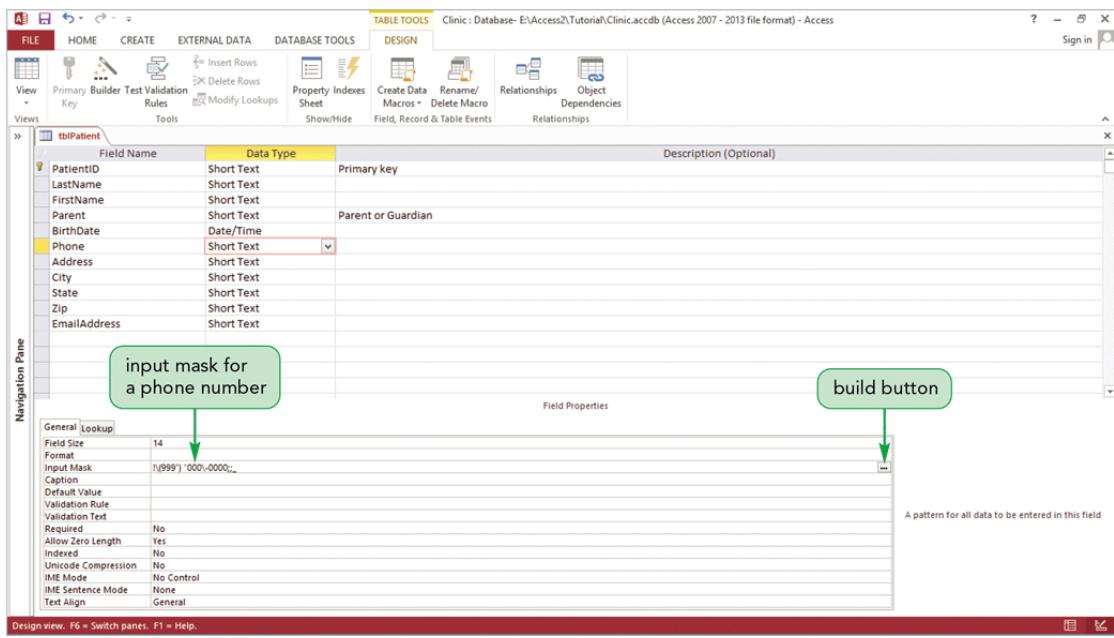
## Using the Input Mask Wizard

- A literal display character is a special character that automatically appears in specific positions of a field value – like hyphens in a social security number
  - To include these characters, you need to create an input mask, which is a predefined format used to enter and display data in a field
- An easy way to create an input mask is to use the **Input Mask Wizard** which guides you in creating a predefined format for a field
  - You must be in Design view to use the Input Mask Wizard

Figure 5-33 Input Mask Wizard dialog box



**Figure 5-34** Phone number input mask created by the Input Mask Wizard



**Figure 5-35** Input mask characters

Input Mask Character	Description
0	Digit only must be entered. Entry is required.
9	Digit or space can be entered. Entry is optional.
#	Digit, space, or a plus or minus sign can be entered. Entry is optional.
L	Letter only must be entered. Entry is required.
?	Letter only can be entered. Entry is optional.
A	Letter or digit must be entered. Entry is required.
a	Letter or digit can be entered. Entry is optional.
&	Any character or a space must be entered. Entry is required.
C	Any character or a space can be entered. Entry is optional.
>	All characters that follow are displayed in uppercase.
<	All characters that follow are displayed in lowercase.
"	Enclosed characters treated as literal display characters.
\	Following character treated as a literal display character. This is the same as enclosing a single character in quotation marks.
!	Input mask is displayed from right to left, rather than the default of left to right. Characters typed into the mask always fill in from left to right.
::	The character between the first and second semicolons determines whether to store the literal display characters in the database. If the value is 1 or if no value is provided, the literal display characters are not stored. If the value is 0, the literal display characters are stored. The character following the second semicolon is the placeholder character that appears in the displayed input mask.

© 2014 Cengage Learning

Figure 5-36 Property Update Options button menu

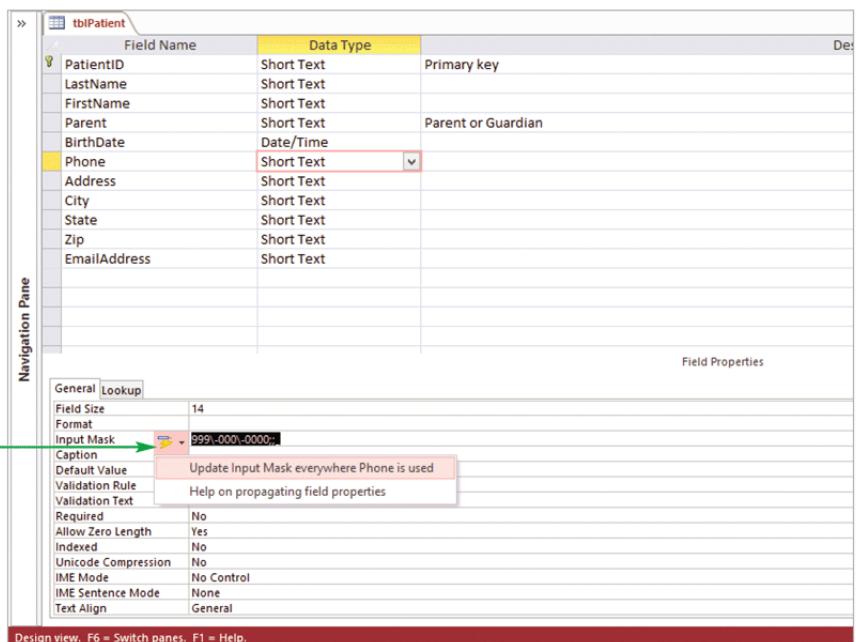
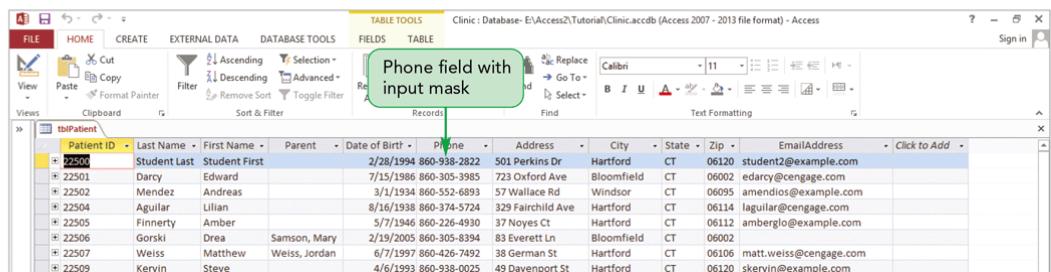


Figure 5-37 Update Properties dialog box



Figure 5-38 After changing the Phone field input mask



## Identifying Object Dependencies

- An **object dependency** exists between two objects when a change to the properties of data in one object affects the properties of data in the other object
  - Dependencies between Access objects, such as tables, queries, and forms, can occur as relationships or using a query to obtain values from more than one table.
  - Any form or report that uses fields from a query is directly dependent on the query and is indirectly dependent on the tables that provide the data to the query
- The **Object Dependencies pane** displays a collapsible list of the dependencies among the objects in an Access database

Figure 5-39 After opening the Object Dependencies pane

The screenshot shows the Microsoft Access ribbon with the 'DATABASE TOOLS' tab selected. In the 'Relationships' group, the 'Object Dependencies' button is highlighted. The main window displays a table named 'tblPatient' with various patient records. To the right, the 'Object Dependencies' pane is open, showing a tree view of objects and their dependencies. A green callout bubble with the text 'drag this edge to the left' points to the top edge of the pane. Another green callout bubble with the text 'warning messages and help' points to a section at the bottom of the pane.

## Defining Data Validation Rules

### Defining Field Validation Rules

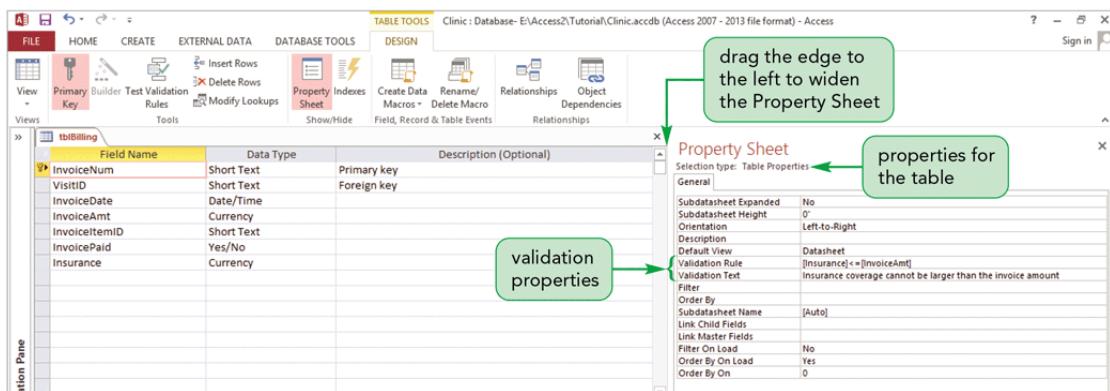
- To prevent a user from entering an unacceptable value in a field, you can create a field validation rule that verifies a field value by comparing it to a constant or to a set of constants
  - You create a field validation rule by setting the Validation Rule and the Validation Text field properties
  - The Validation Rule property value specifies the valid values that users can enter in a field.
  - The Validation Text property value will be displayed in a dialog box if a user enters an invalid value

Figure 5-40 Validation properties for the InvoiceAmt field

The screenshot shows the Microsoft Access ribbon with the 'DESIGN' tab selected. In the 'Fields' group, the 'Field Properties' button is highlighted. The main window displays a table named 'tblBilling' with various fields. The 'InvoiceAmt' field is selected, and its properties are shown in the 'Field Properties' dialog. A green callout bubble with the text 'current field' points to the 'InvoiceAmt' entry in the list. Another green callout bubble with the text 'validation properties' points to the 'Validation Rule' and 'Validation Text' entries in the 'Field Properties' dialog.

- Defining Table Validation Rules
  - To make sure that the value a user enters is not larger than the maximum field value, you can create a **table validation rule**
  - Use the Validation Rule and Validation Text properties and set these properties for the table instead of for an individual field
  - Use a table validation rule because this validation involves multiple fields
  - A field validation rule is used when the validation involves a restriction for only the selected field, and does not depend on other fields

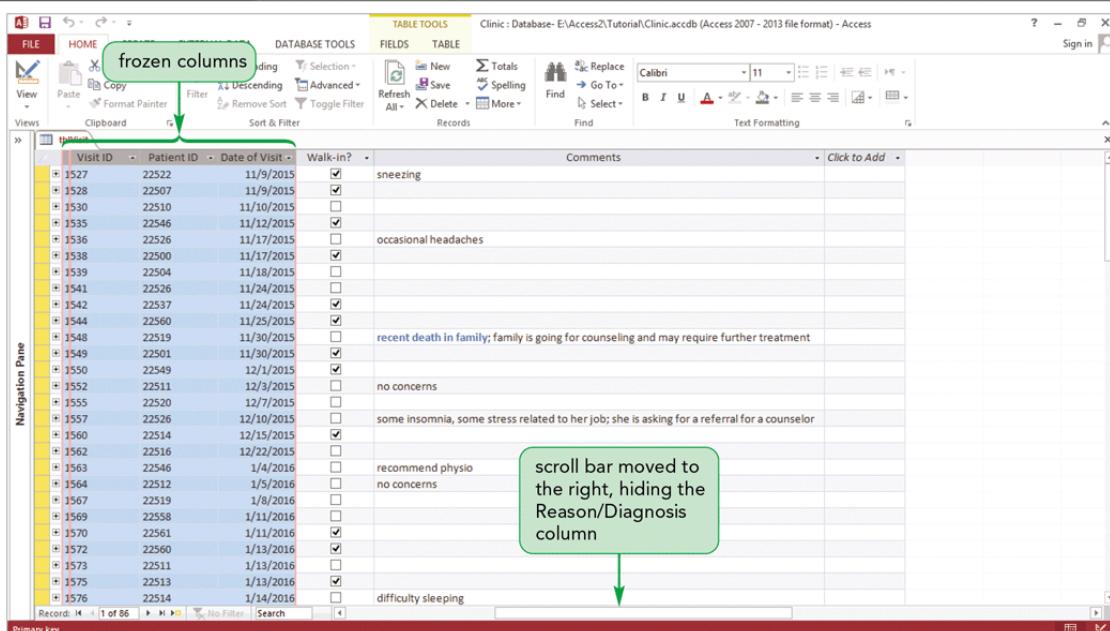
**Figure 5-41** Setting table validation properties



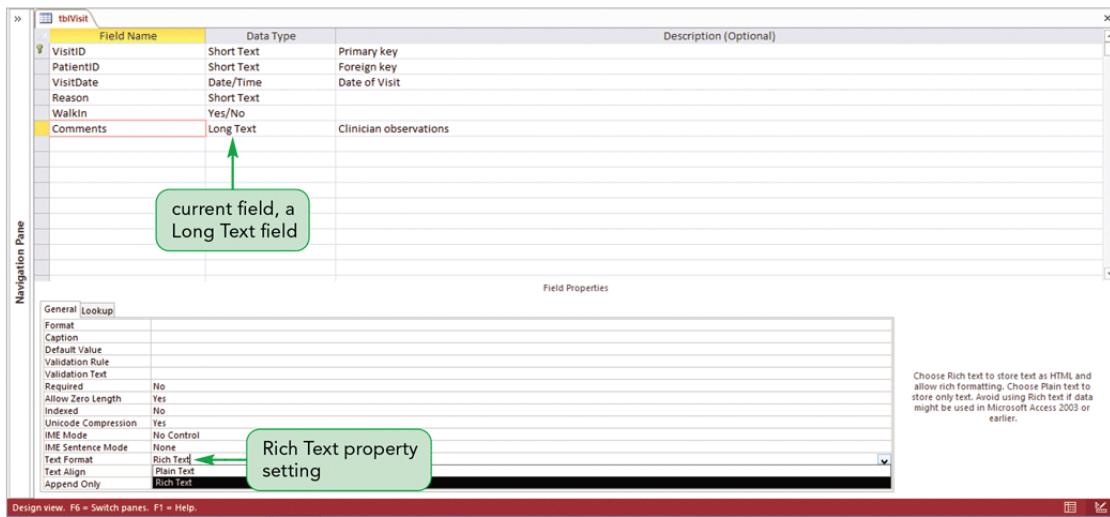
## Working with Long Text Fields

- Use a Long Text field to store long comments and explanations
- Short Text fields are limited to 255 characters, but Long Text fields can hold up to 65,535 characters
  - Short Text fields limit you to plain text with no special formatting
  - Long Text fields store plain text similar to Short Text fields or to store rich text, which you can selectively format with options such as bold, italic, and different fonts and colors

**Figure 5-42** Freezing three datasheet columns



**Figure 5-43** Viewing the properties for a Long Text field



## Summary

- Create different types of queries based on multiple tables
- Use operators in queries
- Create and format a calculated field in a query
- Perform calculations in a query

## 11. Practice and Apply

- Create a query based on multiple tables
- Use a comparison operator in a query to match a range of values
- Use the And and Or logical operators in queries
- Create and format a calculated field in a query
- Perform calculations in a query using aggregate functions and record group calculations
- Use the Like, In, Not, and & operators in queries
- Create a parameter query
- Use query wizards to create a crosstab query, a find duplicates query, and a find unmatched query
- Create a top values query
- Complete Tutorial 11 Exercises