FastCourse Microsoft® Access 2016 Level 2

BRIAN FAVRO

LABYRINTH
LEARNING™

Berkeley, CA

FastCourse Microsoft Access 2016: Level 2

LABYRINTH
LEARNING™

Labyrinth Learning
2560 9th Street, Suite 320
Berkeley, California 94710
800.522.9746
On the web at lablearning.com

Product Manager:
Jason Favro

Development Manager:
Laura Popelka

Senior Editor:
Alexandra Mummery

Junior Editor:
Alexandria Henderson

Assessment and Multimedia Content Development:
Ben Linford, Judy Mardar, Andrew Vaughnley

Production Manager:
Debra Grose

Compositor:
Happenstance Type-O-Rama

Indexer:
Valerie Perry

Interior Design:
Debra Grose

Cover Design:
Mick Koller

ITEM: 1-59136-964-9
ISBN-13: 978-159136-964-6

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

Table of Contents

ACCESS 2016 CHAPTER 1: REFINING TABLE DESIGN

ACCESS 2016 CHAPTER 2: CUSTOMIZING INPUT FORMS

ACCESS 2016 CHAPTER 3: CREATING COMPLEX QUERIES

ACCESS 2016 CHAPTER 4: CUSTOMIZING REPORTS

ACCESS 2016

Refining Table Design

It's important to understand what makes Access a relational database management system and why properly designed databases perform better. Well-designed databases reduce redundant data and create critical connections between the objects that help make them more efficient to use. In this chapter, you will develop important table relationships and use tools to help speed data entry and ensure data accuracy.

LEARNING OBJECTIVES

- Create and modify relationships
- Format a table datasheet layout
- Modify table structure
- Set field properties
- Use the Lookup Wizard

CHAPTER TIMING

- Concepts/Develop Your Skills: 2 hrs
- Self-Assessment: 15 mins
- Total: 2 hrs 15 mins

PROJECT: MAINTAINING AND FORMATTING DATABASES

You are tasked with maintaining the Winchester Web Design database. After reviewing the objects in the database, you decide to make some changes that will make the database more efficient and improve data entry. You will create a lookup field to streamline data entry. In the process, you will add some formatting to make the tables more colorful. You will then examine the relationships between tables to ensure that they define the database accurately.

Creating and Modifying Relationships

As you build tables and other objects in a relational database, Access creates some of the relationships between tables based upon the field structure of each table. However, it's a good idea to examine and edit these relationships manually. For example, you may choose to cascade updated or deleted records; that is, to automatically update or delete all affected records as part of a single operation. Cascade options can be invaluable in cases where a store pulls a product off of its shelf, and therefore needs to remove that product from its merchandise list, order list, inventory list, and advertising list. And in most cases, you also must enforce referential integrity to ensure that relationships between records in related tables are valid. Finally, it may be wise to create and display those relationships in a report to add to the database documentation.

Relationship Types

Database relationships connect data in one table to data stored in other tables. Access supports three different types of relationships:

- A one-to-one relationship means that each record in Table A can have only one matching record in Table B, and each record in Table B can have only one matching record in Table A. This is the least common relationship. A good example is a main Customers table linked to a CustPassword table. One customer has one password.
- A one-to-many relationship means that each record in Table A can have multiple matching records in Table B, but a record in Table B can have only one matching record in Table A. This is the most common relationship. Here's an example: One employee will have many sales, and a product will be sold many times.
- A many-to-many relationship occurs when two tables may each have many matches in the other table, but they do not share key fields, so they use a third junction table to tie the two tables together to complete the relationship. The junction table has a one-to-many relationship to each table. An example is a vendors table and a products table, where one vendor provides many different products and one product is available from many vendors.

Adding, Deleting, and Modifying Relationships

There are times when a database designer must add, delete, or change a relationship. To modify tables after relationships have been set, you must temporarily delete existing relationships so Access is free to make the revisions without violating integrity rules. For example, say you have an existing Short Text data type field, such as State, and you want to change it to a Lookup data type. If you attempt to change its data type, Access will display a warning message indicating that you must first delete its relationships to any other tables. After you delete the relationship and change the field's data type, you may have to reestablish the relationship and edit those relationship properties.

Referential Integrity Requirements

Perhaps the most important database relationship protocol is referential integrity, which is a set of rules used to maintain the validity of the related data in a database. It ensures that you don't delete a record or change a primary key that is related to data in a foreign table. It also requires the data types of the related fields (both the primary and foreign keys) to be the same or compatible.

Referential integrity is a critical part of a relational database, so let's look at it from several different views using real-life examples:

- If the ProdID primary key in the Products table has a Number data type (that is, Field Size property: Long Integer), then the ProdID foreign key in the Invoice Details table must also have the Number data type (that is, Field Size property: Long Integer).
- You cannot have a listing in the Invoice Details table for a product that you don't sell. This means that you cannot have a foreign key (that is, ProdID) in the Invoice Details table without a matching primary key (that is, ProdID) in the Products table.
- You cannot delete the primary key (that is, ProdID) from the Products table when there is a corresponding foreign key (that is, ProdID) in the Invoice Details table.
- You cannot change the primary key value (that is, 01HP) from the Products table when there is an existing and corresponding foreign key value (that is, 01HP) in the Invoice Details table, unless Cascade Update is enabled.

Relationship Cascade Options

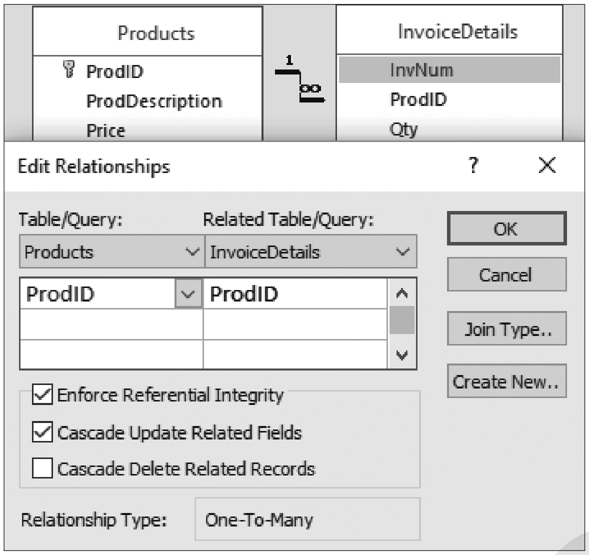
Two additional relationship options are available so that you can control updates to related tables: Cascade Update and Cascade Delete. Each has a unique function for maintaining database relationships, and it's important to know what they control before using them.

RELATIONSHIP CASCADE OPTIONS

Cascade Option	Description
Cascade Update	Updates the value in the key field of a related table if you change the primary key value in the primary table. For example, if you change a ProdID in the Products table, the ProdID field value in the Invoice Details table updates for each invoice.
Cascade Delete	Deletes records in a related table any time you delete related records in the primary table. You might consider this option if you deleted an employee from the Employees table and want to also delete their spouse from the Spouses table. However, use this option with caution because it would not be wise to delete all 2012 invoice records for an employee just because that employee retired in 2013.

The Edit Relationships Box

You can examine, create, and edit relationships between tables in the Relationships window. To create relationships manually, use the Edit Relationships box to identify the rules you want Access to enforce.



Table/Query: Products

Related Table/Query: InvoiceDetails

Table/Query	Related Table/Query
ProdID	ProdID

☑ Enforce Referential Integrity

☑ Cascade Update Related Fields

☐ Cascade Delete Related Records

Relationship Type: One-To-Many

Notice the one-to-many relationship join line between the ProdID fields, which indicates that referential integrity has been enforced.

☰ Database Tools→Relationships→Relationships

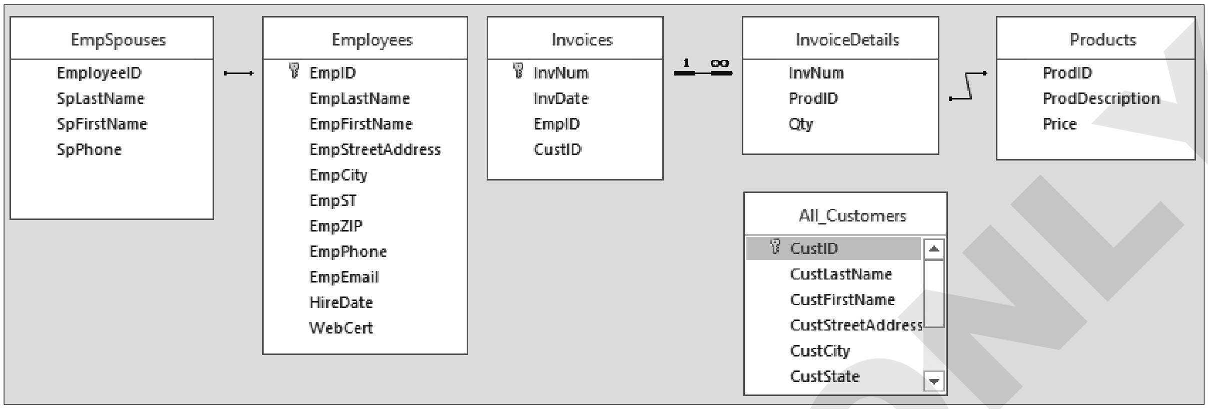
DEVELOP YOUR SKILLS: A1-D1

In this exercise, you will open the Relationships window, add a table, and create a relationship between tables. You will also set referential integrity for the relationship.

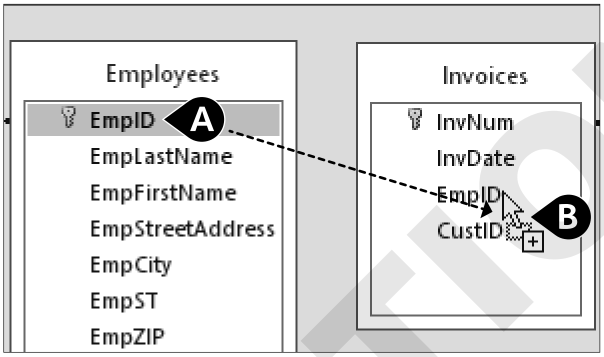
Before You Begin: Be sure to visit the Learning Resource Center at labyrinthelab.com/lrc to retrieve the exercise files for this course before beginning this exercise.

1. Open **A1-D1-WinWebDesign** from your **Access Chapter 1** folder and save it as **`A1-D1-WinWebDesignRevised`**. When completing exercises, always choose to Enable Content
2. Choose **Database Tools→Relationships→Relationships**.
3. Click the **Show Table** button.
4. Double-click the **EmpSpouses** and **All_Customers** tables to add them to the Relationships window and then close the Show Table box.

5. Rearrange the tables within the Relationships window by dragging their title bars until they are arranged as shown below.



6. Follow these steps to create a relationship between the Employees and Invoices tables:



Ⓐ Select the EmpID field in the Employees table and drag it to the **EmpID** field in the Invoices table.

Ⓑ When the mouse pointer is positioned as shown, release the mouse button.

7. Check the **Enforce Referential Integrity** box and click **Create**.

8. Drag the **CustID** primary key from the All_Customers table to the **CustID** field in the Invoices table.

9. Enforce referential integrity and click **Create** to complete the relationship.

10. Choose **File→Save** or click the **Save** button to save the database changes and leave the Relationships window open.

Editing Relationships

Setting cascade options can have a ripple effect on records and data in a database. So it's a good idea to back up a database before setting these options and then test the settings. That way you can restore the database from the backup if the changes you make to the settings result in data loss.

When to Review Relationships

Any time the structure of a table changes—whether it's through adding or removing fields, changing data types, or creating lookup fields—you should review and update the relationships among database tables.

DEVELOP YOUR SKILLS: A1-D2

In this exercise, you will edit the relationship between the Employees and Invoices tables so that if you change the Employee ID in the primary table (Employees), Access will update the Employee ID in the related foreign table (Invoices).

1. Right-click the **join line** between the Employees table and the Invoices table and choose **Edit Relationships**.
2. Check the **Cascade Update Related Fields** checkbox and click **OK**.
3. Choose **File→Save** or click the **Save** button to save the relationship change.

Documenting and Printing Relationships

After you have inspected the relationships, you may want to create a report to view a printable version of the relationships. You can also display the database objects that make use of, or are used by, other objects in the database. This is done through the Object Dependencies panel.

☰ Database Tools→Relationships→Relationships→Relationship Report

☰ Database Tools→Relationships→Object Dependencies

DEVELOP YOUR SKILLS: A1-D3

In this exercise, you will create a relationship report and examine object dependencies for the Employees table.

1. In the Relationships window, if necessary, move the All_Customers table slightly, and any other tables as necessary, until all join lines are clearly visible.
2. Choose **Design→Tools→Relationship Report**.
3. Choose **File→Save** or click the **Save** button and save the report as `Relationships`.
4. Close the report and then close the Relationships window.
5. Click (but don't open) the **Employees** table in the Navigation pane.
6. Choose **Database Tools→Relationships→Object Dependencies**.
7. Review the Objects That Depend on Me option in the Object Dependencies panel.
8. Choose the **Objects That I Depend On** option.
9. Close the Object Dependencies panel.

Modifying Table Structures

Database integrity and data validity are important aspects of database maintenance. Access offers a number of features that enable you to modify table fields, control the data entered, and format the data to ensure consistent reporting. These features include, but are not limited to:

- Renaming tables, forms, and other database objects
- Adding and removing fields from tables
- Changing data types

Renaming Tables and Editing, Adding, and Deleting Table Fields

As you create tables, you define each field by setting the data type and entering the field name. Access works behind the scenes and sets default properties for the field that limit the number of characters in a field, as well as the format and data type of characters that are valid for the field. You can accept the default properties Access sets or modify the properties. Properties available depend on the data type selected for the field. Care must be taken when adding, editing, or deleting fields because of the impact such actions might have on the table data.

Renaming Tables

When you save a table, give it a name that describes the data it contains. You can later change the name without affecting its data. However, note that table names are often included in other database objects that use the table's data. So, renaming a table can impact other database objects. If you rename a table, make sure that every form, query, or report that uses that table still works.

Traditionally, it was wise to *not* include spaces in table names because referencing the table in a query could be confusing. For instance—is *Invoice Details* two objects or one object? Fortunately, Access will enclose a table name like *Invoice Details* within square brackets ([ ]) when it uses it in an expression.

Adding Fields to Existing Tables

Periodically you will need to create new fields in existing database tables and then add data to these fields. You can add a field either in Datasheet View or in Table Design View and then move it where you want it to be in the layout.

Deleting Fields

When you delete a field that contains data, Access displays a message warning you that deleting the field will remove all its data. If you delete a field in Design View and have not saved the table, you can recover the deleted field using the Undo command. If, however, you save the table after deleting the field, the data is lost and you have to add the field name to the Table Design and then reenter all field data in the table to restore the data. Fields are deleted in Design View by clicking the field header and tapping Delete.

Editing Field Data Types

Many Access data types start with a different letter, which means you can type a letter and the data type that begins with that letter will display. For example, if you want to change the data type of a field from Short Text to Number, you click in the field's Data Type and type *N*.

Any time you change the data type of a field that contains values that fail to conform to the new data type, Access deletes any nonconforming data. For example, if you change a field's data type from Short Text to a Number data type and someone has accidentally entered *1O* (using a capital *O*) instead of *10* (using zero), Access will warn you that you are about to delete data that did not conform. The great thing about this is that Access will only allow valid field data, which results in more accurate data.

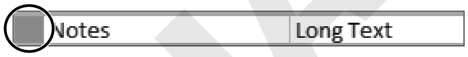
The Yes/No Data Type

The Yes/No data type sets the field so that only two entries are possible—Yes/No, True/False, or On/Off. When you set the Yes/No data type for a field, Access places a checkbox for the field in the datasheet and on forms where the data appears. Checking the checkbox indicates a value of Yes, True, On, etc.; clearing the checkbox indicates a value of No, False, or Off.

DEVELOP YOUR SKILLS: A1-D4

In this exercise, you will rename a table, delete a table field, add a table field, and modify the data type of a field.

1. Right-click the **All_Customers** table in the Navigation pane and choose **Rename**.
2. Type **`Customers`** and tap Enter.
3. Open the **Customers** table and switch to **Design View**.
4. Click the **field selection** box on the left edge of the Notes field to select the field.



5. Tap Delete and choose **Yes** to confirm the deletion.
6. Right-click the **CustStreetAddress** field selection box and choose **Insert Rows**.
7. Click in the empty **Field Name** box of your new field and type **`Business`** as the name.
8. Tap Tab and set the Data Type to **Yes/No**.
9. Choose **File→Save** or click the **Save** button to save the table.
10. Switch to **Datasheet View**.
11. Check the **Business** field boxes for the records with the following CustIDs: **DavisP**, **HassanA**, and **KleinJ**.
12. Close the Customers table.

Formatting a Table Datasheet Layout

It's difficult to plan and prepare for all of the possibilities that may occur as a database is first developed. For instance, if field values are longer than anticipated, Access will display only the portion of the data that fits within the column width, causing some of the data to be unseen. Or the opposite scenario may occur, in which one or two fields were added, and you need to display all the fields on one screen, which means that you may have to modify the width of each column to fit the screen. Alternatively, you can maximize the Access window or close the Navigation pane to provide more room without having to modify the width of each column.

Changing the Width of Columns

Access offers some useful techniques to adjust the width of each column in a datasheet to display all data in the column:

- **Drag a column border:** Drag a column border to make the column on the left of the border wider or narrower.
- **Double-click a column heading border:** Double-click the right border of a column to change the width of the column on the left to fit either the longest data entry in the column or column heading, whichever is wider.
- **Right-click a field heading and choose Field Width:** Select the Field Width command from the context menu to open the Column Width dialog box and type the width, reset the standard width, or select Best Fit to automatically size the field width to the longest entry.

Moving and Hiding Data Columns

There will be times when you want to reposition a column of data in a table layout—perhaps to display the email address before the telephone number. When you rearrange the columns in a datasheet, the table layout remains the same but the fields display in a different order in the datasheet. You may also want to hide some columns so you can better view other field columns. When you hide columns, Access temporarily removes them from display. The data, however, remains in the table—it is not deleted. If you want to view data in hidden columns at a later time, you unhide the column.

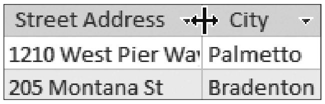
Saving a Table Layout

Changing the layout of a table datasheet has no real effect on table data or structure; however, when you make changes to a table datasheet, Access recognizes the differences between the structure of the table and its layout, and prompts you to save the changes to the layout when you close the table. If you abandon the changes, the next time you open the table datasheet, the column widths will return to their original size and any columns that were hidden will show. If you save the changes, the next time you open the table datasheet, Access recalls the layout and displays the formatting changes.

DEVELOP YOUR SKILLS: A1-D5

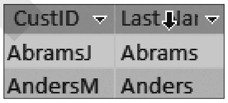
In this exercise, you will adjust column width to allow for the best display of data in a datasheet, rearrange columns, and hide a column.

1. Display the **Customers** table in **Datasheet View**.
2. Position the mouse pointer between the Street Address and City column headings and double-click when the adjust pointer appears as shown here.



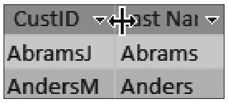
Street Address	City
1210 West Pier Wa	Palmetto
205 Montana St	Bradenton

3. Position the mouse pointer on the **CustID** column heading and drag right until both the CustID and Last Name columns are selected as shown here. Release the mouse button.



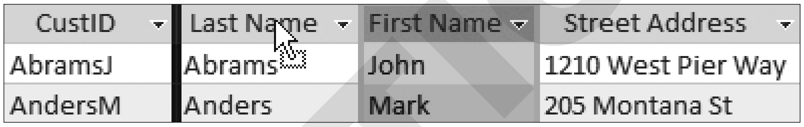
CustID	Last Nai
AbramsJ	Abrams
AndersM	Anders

4. Position the mouse pointer between the column headings and double-click when the adjust pointer appears.



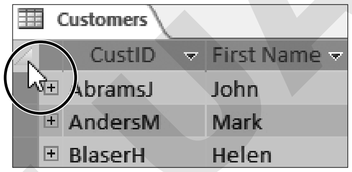
CustID	st Nai
AbramsJ	Abrams
AndersM	Anders

5. Click the **First Name** column heading to select the column.
6. Drag the **First Name** column to the left of the Last Name column and release the mouse button when the black vertical bar is positioned as shown here.



CustID	Last Name	First Name	Street Address
AbramsJ	Abrams	John	1210 West Pier Way
AndersM	Anders	Mark	205 Montana St

7. Click the table selection button to select all data as shown here.



Customers

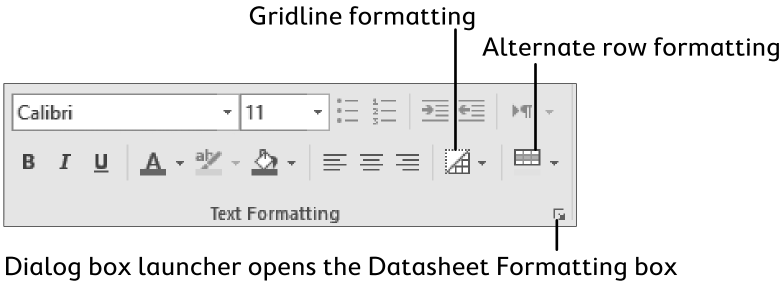
CustID	First Name
AbramsJ	John
AndersM	Mark
BlaserH	Helen

8. Double-click the border between any of the selected column headings to Best Fit all columns.
9. Close the Customers table, saving your changes.

Enhancing a Datasheet

Changing the datasheet layout enables you to make necessary adjustments, such as widening a field so that a longer value can be fully displayed. Enhancing the datasheet layout enables you to improve its readability. Some of the features you can apply to enhance a datasheet include gridlines, font size and color, and background color.

As you apply enhancements to the datasheet, Access formats all data and gridlines to match the format you choose. The Text Formatting group on the Home tab displays tools for enhancing the most commonly formatted features on a datasheet such as fonts, colors, fills, and alignments. And it also contains tools for formatting alternate rows and gridlines. Finally, a Datasheet Formatting dialog box can be launched from the Text Formatting group from which additional datasheet formatting can be applied.

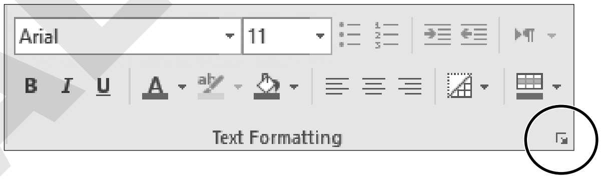


☰ Home→Text Formatting

DEVELOP YOUR SKILLS: A1-D6

In this exercise, you will use the Text Formatting tools to set datasheet enhancement options for the Customers table.

1. Open the **Customers** table in **Datasheet View**.
2. Choose **Home→Text Formatting→Alternate Row Color ▦ menu button ▼**.
3. Choose the **Blue, Accent 1** color, which is the 5th color on the top row of Theme Colors.
4. Choose **Home→Text Formatting→Font menu button ▼** and choose **Arial** from the font list.
5. Choose **Home→Text Formatting→Gridlines ▨ menu button ▼** and choose **Gridlines: Horizontal**.
6. Click the **Home→Text Formatting** group dialog box launcher as shown below to open the Datasheet Formatting dialog box.



7. Click the **Gridline Color menu** button ▼ and choose **Black, Text 1** (2nd color on the top row of Theme Colors).
8. Click **OK** to apply the black gridlines.
9. Choose **File→Save** and save the changes to the Customers table.

Setting Field Properties

Field properties are settings that enable you to further define the properties of each field. The Field Properties pane appears in the lower portion of the Table Design View or on the Fields tab in Datasheet View. The most frequently used properties are identified in the following table.

COMMON FIELD PROPERTIES

Field Property	Description
Field Size	Sets a field length for the number of characters each field can hold.
Format	Sets a predefined display layout for fields (that is, currency or percent).
Input Mask	Identifies the format of values entered—with hyphens or without, alphabetic or numeric, uppercase or lowercase, etc.
Caption	Sets a column heading title to describe the data content better than the actual field name. Includes spaces where appropriate.
Default Value	Adds a default value for a specific field in each record, such as FL for the State field, abbreviated as “ST.”
Validation Rule	Controls actual values entered into a field, such as less than 100 or greater than 01/01/2017.
Validation Text	Provides a tip that identifies valid data entries, such as “All dates must be after 01/01/2017.”
Required	Sets the field as required to ensure a value is entered in the field.

Why Set Field Properties?

Different people add data to databases—and they often enter the data differently. For example, some people type parentheses around the area code when entering phone numbers. Others may separate the area code from the number using a hyphen. Both formats are accurate, but displaying mismatched data can be distracting. Entering parentheses or hyphens can also be time-consuming. Setting field properties to control how data appears helps maintain data consistency throughout a database.

Set Field Sizes, Captions, and Default Values

Maintaining database integrity, data validity, and data format are important considerations when building a database. You should make every effort to ensure that data is entered consistently, contains the required number of characters, and falls within valid data ranges.

Setting Field Size

Rather than using a default field size, you can set the Field Size property to limit the number of characters that can be entered into the field for each record. For example, you can limit data entry of state names to the two-character state abbreviation.

Sometimes, when you reduce an existing field size to limit data entry, Access displays a warning that data may be lost due to the reduced field size. In most cases, you are familiar with the data, so you can choose *Yes* to continue. For instance, truncating Florida to FL would not create invalid data. However, if you are uncertain, you should choose *No*, check the data to ensure that it fits the new limit, and then set the field size.

Identifying Field Size for Number Fields

Number fields are identified by special formats in the Properties panel. In general, number fields should be set to define the largest value anticipated for the field. Setting the proper field size controls for number fields helps optimize database performance. The following table identifies each Number field format and describes the type of data each stores.

NUMBER FIELD FORMATS

Field Size Property	Description
Byte	Stores whole numbers between 0 and 255 using one byte and allows no fractions or decimal points; uses the minimum amount of memory, allowing for the fastest processing.
Integer	Stores whole numbers between –32,768 and 32,767 using 2 bytes rather than the standard 7 bytes normally used for high values.
Long Integer	Stores whole numbers between –2,147,483,648 and 2,147,483,647 using 4 bytes rather than the standard 14 bytes normally used for high values.
Single	Stores positive and negative numbers to exactly seven decimal places using 4 bytes.
Double	Stores positive and negative numbers to exactly 15 decimal places using 8 bytes.
Replication ID	Identifies replication of tables, records, and other objects in Access 2003 or earlier databases using 16 bytes.
Decimal	Stores positive and negative numbers to exactly 28 decimal places using 12 bytes.

Setting Text to Uppercase or Lowercase Format

Text fields have unique field properties available for formatting data. Access provides a Format field that enables you to force a specific format to all characters in the field. The most common format characters are used to force uppercase (>) and lowercase (<). Using the Text Format property eliminates the need to spend valuable time entering multiple characters in the Input Mask property.

Setting Captions

As you may have noticed, many field names contain no spaces or include an underscore, such as LastName or Last_Name. The Caption field property enables you to type a more descriptive name for a field that is more suitable for display on forms, in datasheets, and on reports—such as Last Name.

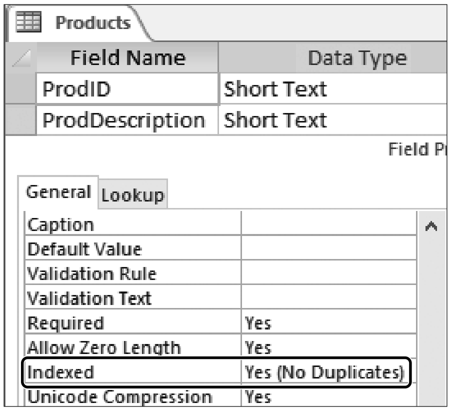
Setting Default Field Values

Validation rules control the data you enter in table fields. Setting a default value for a field automatically enters the most common data value and can save time and help reduce the number of errors made during data entry.

For instance, all the employees at Winchester Web Design live in Florida. Consequently, it saves time and reduces inconsistency when the default value for the State field is set to FL. The default value appears whenever a new record is added. If you need to enter a different state, you simply type in the new state's two-character abbreviation to replace the default value.

Making a Field Required

Whenever you create a primary key field, its properties are automatically set to be required and indexed, allowing no duplicates. A database index is a structure whose main function is to speed up database operations. An index that is set on key fields enables faster searches and retrieval of data.



Products

Field Name	Data Type
ProdID	Short Text
ProdDescription	Short Text

Field Pr

Caption	
Default Value	
Validation Rule	
Validation Text	
Required	Yes
Allow Zero Length	Yes
Indexed	Yes (No Duplicates)
Unicode Compression	Yes

A key field must have a value; by default, every other field does not require that a value be entered. There are times, however, when non-key fields must have values. For instance, you must include an employee last name and first name when entering a new record into your Employees table. The Required field property helps to easily accomplish this.

DEVELOP YOUR SKILLS: A1-D7

In this exercise, you will set field sizes, captions, and default values in the Customers table.

1. If necessary, open the **Customers** table and switch to **Design View**.
2. Click anywhere in the **CustState** field and set the following Field Properties:

   Field Size: **2**
   Format: **>**
   Caption: **ST**
   Default Value: **FL**

3. Click anywhere in the **CustLastName** field and change the field size to **25** and the required property to **Yes**.
4. Click anywhere in the **CustPhone** field and change the field size to **15** and the required property to **Yes**.
5. Choose **File→Save**, choose **Yes** when the Some Data May Be Lost message appears, and choose **Yes** again when the Data Integrity Rules Have Been Changed message appears.

6. Switch to **Datasheet View** and enter this data in a new record:

Field	Data to Enter
CustID	`JonesK`
First Name	`Ken`
Last Name	Leave this field blank.
Business	Check the box to set it to **Yes**.
Street Address	`2300 Maple Ave.`
City	`Palmetto`
ST	This field will already be set to FL, as you set FL as the default value.
ZIP	`34628`
Telephone	Leave this field blank.
Email	`KJones@email.com`

7. Tap Tab after entering the email address and you will be prompted to enter a value for the Customers.CustLastName field, because you made it a required field.
8. Click **OK** and then type `Jones` in the Last Name field.
9. Tap Tab repeatedly until you are prompted to enter a value for the Customers.CustPhone field, since it is also a required field.
10. Click **OK** and then enter `9415553232` as the telephone number, tapping Tab when you are finished.

Custom Text and Memo Field Formats

Standard field formats in Access cannot meet the needs of every text or memo field contained in every database. That's why Access provides tools for creating custom formats. Custom formats for Text and Memo fields can contain two sections:

- **Section 1:** Contains a symbol and is followed by a semicolon when a second section is entered.
- **Section 2:** Contains the value of the alternate value when no value is entered. This alternate is a *null* value and is enclosed in quotation marks with no space between: "".

An example of a two-section format for a text field would look like this: @; "N/A"

The @ symbol tells Access to display the field data if a value is entered, and N/A tells Access to display N/A (Not Applicable) if no value is entered. The @ symbol displays all the characters that will fit the Field Size property, and if there are fewer characters than the Field Size value, Access pads the rest of the field with blank spaces.

Short Text and Long Text Field Unique Properties

Text and Memo fields are formatted to hold text characters (abc), symbols (#$%), and numbers (123) on which no mathematical calculations will be performed, such as FirstName, LastName, City, and also ZipCode, PhoneNumber, and SocSecNumber. Because of the broad scope of data that these data types can contain, Access provides several field properties for controlling and formatting data entry in the field.

SHORT TEXT AND LONG TEXT FIELD UNIQUE PROPERTIES

Property	Description
Allow Zero Length	Allows data entry of zero length in a field. Data is entered as open and close parentheses with no character or space between: (). The purpose of this entry is to show that there is no value to enter. For example, if you have a field in a Customers table that requires a land phone number and the customer has no land phone, you would enter () in the field.
Text Format	This property will set the text in a Long Text field as Plain Text or Rich Text. Rich text fields can be formatted with different fonts, font sizes, and colors.
Text Align	Positions the text on the left, center, or right side of the field box or column. The Distribute setting spreads out the text to fill the column or text box size.
Append Only	Adds a series of date-stamped comments to a single Long Text field, making it easy to create a history log of comments added to the fields. These comments are stored in a separate table and accessed through the Append Only Long Text field.

Entering Field Properties

Access provides three basic techniques for setting field properties:

- Type the value into the property box.
- Choose the value from the property list. (For example, click the drop-down menu button to select a valid entry from the list.)
- Click the Build button that appears at the right side of a field property to open the Wizard associated with that property. Then choose the settings you want to apply. For example, click the Build button to open the Input Mask Wizard to format the display of text and field dates.

DEVELOP YOUR SKILLS: A1-D8

In this exercise, you will set additional properties to require the entry of a customer's first name.

1. Display the **Customers** table in **Design View**.
2. Click anywhere in the **CustFirstName** field and type **@** for the Format field property to display all characters that fit within the field size and pad any remaining positions with spaces.
3. Choose **Yes** in the Required field property and **No** in the Allow Zero Length field property to prohibit a null value from being entered.

4. Choose **File→Save** or click the **Save** button and choose **Yes** when advised that the data integrity rules have been changed.
5. Switch to **Datasheet View**.
6. Click in the **CustID** field of the new, blank record at the bottom of the table and type **`SmithA`**.
7. Type **`Smith`** in the Last Name field and then close the Customers table.
8. Click **OK** to dismiss the message and then click **Yes** to close the database object (the table) now.

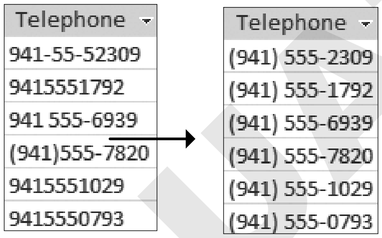
Formatting Data Using Input Masks

Consistency of data format is important for visual aesthetics; it also helps ensure accuracy in searches, queries, and sorts. You can control data formats using the field property input mask. Using input masks, you can set the characters you want displayed in fields, such as the parentheses in an area code, and Access requires the user to enter the data within that format.

The Input Mask Wizard

The Input Mask Wizard is a valuable tool for setting the most common formats used in databases. You can also set input masks to require a specific number of characters in a field or to convert characters to capital or lowercase.

Setting input masks ensures that the data format in tables is consistent. Because the table data is consistent, data displayed in forms and reports will also be consistent.



Telephone
941-55-52309
9415551792
941 555-6939
(941)555-7820
9415551029
9415550793

Telephone
(941) 555-2309
(941) 555-1792
(941) 555-6939
(941) 555-7820
(941) 555-1029
(941) 555-0793

Input masks can automatically format unformatted data.

A Build ··· button appears at the right end of the Input Mask box when you click the box. It starts the Input Mask Wizard, which helps you build the mask.

Input Mask Symbols

When you use the Input Mask Wizard, Access places the necessary coding into the Field Properties pane. Access uses several symbols to control the appearance of data.

INPUT MASK SYMBOLS

Symbol	Description	Example
0	Requires a numeric digit	(000) 000-0000 requires an area code as part of the phone number.
9	Data is optional but must be a digit	(999) 000-0000 requires a seven-digit phone number with an optional three-digit area code.
#	Restricts data to a digit, +, -, or space	#99.99 permits + or – in the position of the #.
L	Requires an alphabetic character (that is, a letter)	LL requires the entry of two alphabetic characters in the State field.
?	Restricts, but does not require, data to alphabetic characters	L????L requires two alphabetic characters, one on each end of the data, but permits four additional alphabetic characters in between.
A	Requires an alphabetic or numeric character	000-AAAA permits a phone number to be entered either as 555-1234 or 555-HOME.
a	Allows, but does not require, alphabetic or numeric characters	(aaa) AAA-AAAA requires a seven-digit phone number but not the area code.
&	Requires any alphanumeric character (a letter or a number) or a space	&&&& permits data entry such as a four-character ID such as 01HP, 1 HP, or 1234.
C	Allows, but does not require, any character or space	CCCC could contain an entry such as 01HP, 1 HP, HP, etc.
.,:;-/	Characters used to separate parts of numeric, date, time, and currency values	#,###.## permits numeric data. 99/99/00 permits date data. 99:00:00 permits time date.
<	Converts characters to lowercase	<aaa permits entry of three characters such as *ABC* and converts data to lowercase *abc*.
>	Converts characters to uppercase	>aa permits entry of two characters such as *fl* and converts the data to *FL*.
!	Displays input mask characters from right to left	!(#) 000-0000 right-aligns the phone number so that if only seven numbers are entered, the area code is left blank. This affects fields defined with the Number data type.
\	Causes characters that follow the \ to display as literal characters	(\A) appears as (A).
"Literal Text"	Places text that appears between the quotation marks into the field value at the identified position	"ID-"0000 places *ID-* before the numbers entered. A space may be enclosed in quotes to ensure it appears in the value.
Password	Creates a password entry text box. Any character typed in the text box is stored as a character but displays as an asterisk (*) as the password is entered	When passWord1! is typed, Access shows **********.

Storing Input Mask Characters

Access provides two methods for storing the input mask with the table data—with or without the symbols. Storing the symbols with the data increases the size of the database file. Therefore, companies that store extremely large volumes of data often prefer storing the data without the input mask symbols. You can choose one of these methods while running the Input Mask Wizard.

Using Smart Tags

As you work in Access, you will periodically see smart tags, such as the Paste Options smart tag, which you may have seen in Word and other Microsoft applications. Smart tags allow you to apply formatting changes you make to a field in one table to the same field anywhere else it occurs in the database. For example, if you modify the field format properties in a table, the Property Options smart tag lets you apply the same format changes to the field when it appears in other forms, queries, and reports. This helps ensure the consistency of data throughout the database.

DEVELOP YOUR SKILLS: A1-D9

In this exercise, you will set the primary key, change a field size, and apply a custom input mask to a field in the Products table. Then you will apply a standard telephone input mask to a field in the Employees table.

1. Open the **Products** table in **Datasheet View**.
2. Switch to **Design View** and click the **Primary Key** button to make **ProdID** the primary key field.
3. Click in the **Input Mask** field properties box and type **`"PROD-"00AA`** (use zeros not the letter O for 00).
4. Choose **File→Save** or click the **Save** button, and the Property Update Options smart tag will appear next to the input mask you just entered.
5. Click the **smart tag** and choose **Update Input Mask Everywhere ProdID Is Used**.
6. Click **Yes** to update the Products Report object.
7. Switch to **Datasheet View**.
8. Close the Products table.
9. Open the **Employees** table in **Datasheet View**.
10. Switch to **Design View** and click anywhere in the **EmpPhone** field.
11. Click in the **Input Mask** field property box and then click the ... button on the right side of the box.
12. Choose **Phone Number** as the input mask and click **Finish**.
13. Choose **File→Save** to save the table and then switch to **Datasheet View**.
14. Switch back to **Design View**.

15. Change these field properties to set the field sizes, input masks, and captions shown below:

Field	Field Size	Input Mask	Caption
EmpLastName	25	>L<????????????????????????	Last Name
EmpFirstName	25	>L<????????????????????????	First Name
EmpStreetAddress	30		Street Address
EmpCity	15	>L<??????????????	City
EmpST	2	>LL	State
EmpZIP			ZIP
EmpEmail			Email
HireDate			Hire Date
WebCert			Web Cert

16. Choose **File→Save** or click the **Save** button and choose **Yes** when advised that some data may be lost.
17. Switch to **Datasheet View**.
18. Close the Employees table.

Setting Validation Rules

A validation rule is a field property that enables you to limit the values entered into the field in order to reduce inaccurate data entry. You could, for example, set a validation rule to limit the value typed into an HoursWorked field to fifty or fewer, or the value of Pay Rate to less than $60.

Setting Appropriate Data Types for Validation Rules

For validation rules to be effective, it is important that the field for which you are setting the rule be formatted appropriately for the data type that should be entered. For example, if you set a validation rule requiring a four-digit number, the data type for the field should be set to Number. If you are requiring dates that occur before a specific date, the data type for the field should be Date/Time.

Validation Text Messages

When you set a validation rule for a field, it is also a good idea to set validation text, which contains instructions or valid data values to help guide data entry. Access displays the text as a message each time an invalid value is entered in the field.

Setting Different Types of Validation Rules

Validation rules are used to examine data entered into tables and forms. You can set comparison rules. Samples of comparison rules you can set to determine if the value is within a valid range are shown in the following table.

VALIDATION RULES

Comparison	Validation Rule Example	Validation Text Example
Greater than	>100	Enter a value greater than 100.
Less than	<100	Enter a value less than 100.
Equal to	=1 Or =2	Enter a value of 1 or 2.
Date after a date	>#1/1/2017#	Enter a date after January 1, 2017.
Greater than or equal to	>=100	Enter a value of 100 or more.
Less than or equal to	<=100	Enter a value of 100 or less.
Like	Like “ID-0000”	Enter a 4-digit value starting with *ID-*.
Between	Between 1 And 8	Enter a value from 1 to 8.

The same wildcards used to enter input masks are used in validation rules. For example, the question mark is substituted for each character that is required, such as in *ID-????*. The asterisk (*) can substitute for a group of characters that may vary, such as in *ID-**.

DEVELOP YOUR SKILLS: A1-D10

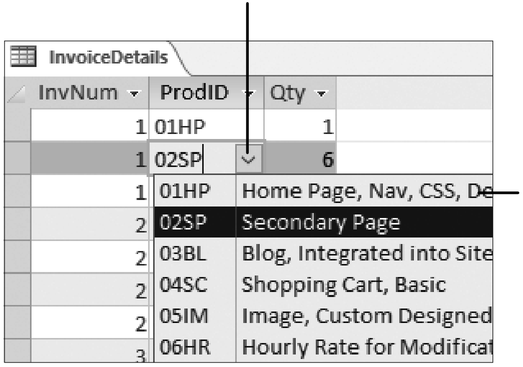
In this exercise, you will set validation rules for data entered into fields in the Products table.

1. Display the **Products** table in **Design View**.
2. Click anywhere in the **Price** field and then click in the **Validation Rule** field property box.
3. Type **`>=25`** as the validation rule.
4. Click in the **Validation Text** box and type **`All prices must be at least $25`**.
5. Choose **File→Save** or click the **Save** button and choose **Yes** when the warning message appears notifying you that the date integrity rules have been changed.
6. Switch to **Datasheet View**.
7. Click in the **ProdID** field for the new, blank record and type **`07SW`**.
8. Tap Tab, type **`Switchboard Page`** as the description, and tap Tab again to move to the Price field.
9. Type **`20`** in the Price field and tap Tab.
10. Click **OK** to acknowledge the error message, type **`30`** for the price, and tap Tab.
11. Close the Products table.

Setting Lookup Fields with the Lookup Wizard

All tables in a relational database are related in some way to each other, as well as to other objects in the database. Data from one table is often used in another table. A lookup field enables you to select a field value in one table by looking up values from another table; or you could select from a list of values entered by the database designer. The list of valid entries appears in a drop-down menu in the table accessing the values.

A lookup field displays a menu arrow at the right end of the field during data entry.



InvoiceDetails

InvNum	ProdID	Qty	
1	01HP	1	
1	02SP	6	
1	01HP		Home Page, Nav, CSS, De
2	02SP		Secondary Page
2	03BL		Blog, Integrated into Site
2	04SC		Shopping Cart, Basic
2	05IM		Image, Custom Designed
3	06HR		Hourly Rate for Modificat

This example shows the Products list displaying selectable values.

Using a lookup value also enables you to look up values from one field and return a value from a different field in the connected table. For example, you can look up a product number by typing the common product name.

Examining the Benefits of Lookup Tables

Adding a lookup field to a table serves three primary purposes:

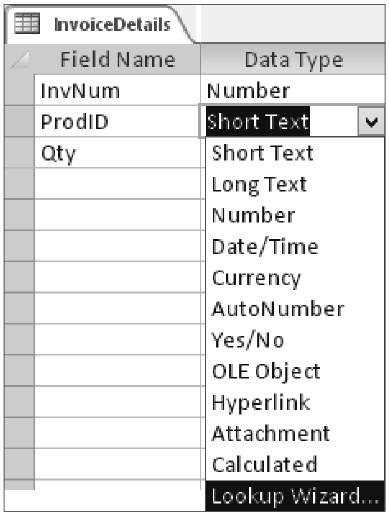
- It reduces the time required to enter the data repeatedly.
- It reduces errors associated with data entry.
- It restricts data to valid entries.

For example, if you are processing time card data before issuing employee checks, setting a lookup field of valid employee IDs helps ensure that only valid employees receive checks. Lookup fields also help reduce the number of redundant fields contained in database tables.

Performing a Lookup

Access provides the following two ways to use the Lookup feature:

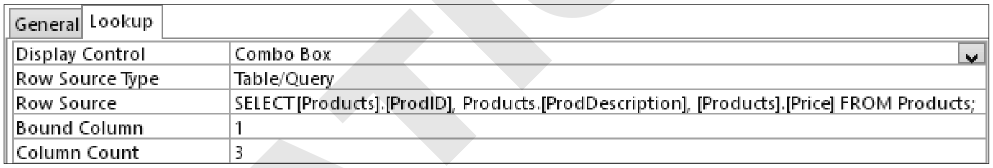
- **Lookup Wizard:** A data type that launches the Lookup Wizard, which walks you through the process of setting up a lookup field.



InvoiceDetails

Field Name	Data Type
InvNum	Number
ProdID	Short Text
Qty	

- **Lookup tab:** An option in the Design View Field Properties pane that sets the data source containing the values you want to display in the field.



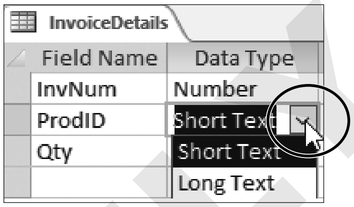
Display Control	Combo Box
Row Source Type	Table/Query
Row Source	SELECT [Products].[ProdID], Products.[ProdDescription], [Products].[Price] FROM Products;
Bound Column	1
Column Count	3

DEVELOP YOUR SKILLS: A1-D11

In this exercise, you will delete the relationship between two tables and create a lookup field in the Products table that displays a list of valid products. You will then use the lookup field to enter data into the Invoice Details table.

1. Make sure the Products table is closed, as you cannot make relationship changes to an open table.
2. Choose **Database Tools→Relationships→Relationships** .
3. Right-click the **join line** linking the Products and InvoiceDetails tables.
4. Choose **Delete** and then click **Yes** to confirm the deletion.
5. Using the same procedure, delete the relationship between the **EmpID** fields in the Employees and Invoices tables, and between **EmpID** in the Employees table and **EmployeeID** in the EmpSpouses table.

6. Close the Relationships window and save the changes.
7. Display the **InvoiceDetails** table in **Design View**.
8. Click in the **Data Type** box of the **ProdID** field and click the drop-down **menu** button ▼.
9. Choose **Lookup Wizard** from the menu.
10. Click **Next** to accept the current setting **I Want the Lookup Field to Get the Values from Another Table or Query**.



Field Name	Data Type
InvNum	Number
ProdID	Short Text
Qty	

11. Choose **Table: Products** in the next Wizard screen, and then click **Next**.
12. Move all three fields to the **Selected Fields** list and click **Next**.
13. Choose **ProdID** as the sort field, leave the sort order as **Ascending**, and click **Next**.
14. Uncheck the **Hide Key Column** checkbox.
15. Double-click the right borders of all three column headings to Best Fit the columns and then click **Next**.
16. Click **Next** again to choose **ProdID** as the field that uniquely identifies the row.
17. In the next Wizard screen, check the **Enable Data Integrity** box and then choose the **Cascade Delete** option.
18. Click **Finish** and choose **Yes** in the warning box that appears.
19. Choose **Yes** again in the Some Data May Be Lost warning box.
20. Switch to **Datasheet View**.
21. Click in the **ProdID** field for the 02SP record (second record in the table).
22. Click the drop-down **menu** button ▼ and then choose the product **07SW**.
23. Close the InvoiceDetails table.

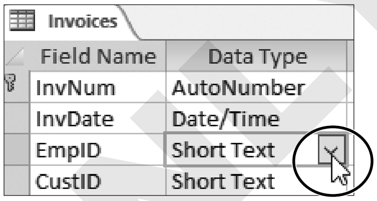
Creating Lookup Fields for Multiple Values

You have already created a lookup field that enabled you to select a single item from a list. You can also set up lists that allow you to select multiple values to enter for each lookup field. If, for example, an inventory item is available from more than one supplier, you can set up the field to allow you to select all suppliers for an item. To create a selection list, simply check the Multiple Items option as you move through the Lookup Wizard screens.

DEVELOP YOUR SKILLS: A1-D12

In this exercise, you will create a lookup field in the Invoices table that allows you to assign two or more employees to an inventory item.

1. Display the **Invoices** table in **Design View**.
2. Click in the **Data Type** box of the **EmpID** field and click the **menu** button ▼.
3. Choose **Lookup Wizard** from the menu.
4. Click **Next** to accept the current setting, **I Want the Lookup Field to Get the Values from Another Table or Query**.



Invoices

Field Name	Data Type
InvNum	AutoNumber
InvDate	Date/Time
EmpID	Short Text
CustID	Short Text

5. Choose **Table: Employees** and click **Next**.
6. Move the **EmpID**, **EmpLastName**, and **EmpFirstName** fields to the Selected Fields list and click **Next**.
7. Choose **EmpID** as the sort field, leave the sort order as **Ascending**, and click **Next**.
8. Click **Next** to accept the default width settings for the columns.
9. In the final Wizard screen, check the **Allow Multiple Values** box, leave the label set to **EmpID**, and click **Finish**.
10. Choose **Yes** in the message box to confirm that you want to store multiple values.
11. Choose **Yes** again to save the table and choose **Yes** one last time to confirm the Some Data May Be Lost message.
12. Switch to **Datasheet View**.
13. Click in the **Emp ID** field for the **third record** (invoice 3).
14. Click the **drop-down menu** button ▼ to display the list of employees.
15. Check the boxes for **Winchester** and **Mansfield** and click **OK**.
16. Double-click the border between the Emp ID and Cust ID columns to Best Fit the Emp ID column.
17. Choose **File→Close** to close the database and save the changes to the Invoices table.

Self-Assessment

Check your knowledge of this chapter's key concepts and skills by completing the Self-Assessment. The answers to these questions can be found at the back of this book.

1. A one-to-many relationship would be an appropriate relationship type to use when linking a single employee to multiple sales. *True* *False*

2. A one-to-one relationship would be an appropriate relationship type to use when linking a single employee to an employee password table. *True* *False*

3. Referential integrity requires that the data types of related fields are the same or compatible. *True* *False*

4. A good example of a one-to-many relationship is where one employee has many sales. *True* *False*

5. Hiding columns in a datasheet permanently removes the underlying table data. *True* *False*

6. When used in an input mask, the < symbol forces all characters to uppercase. *True* *False*

7. A lookup field enables you to select a field value in one table by looking up values from another table. *True* *False*

8. A validation text message is displayed whenever invalid data is entered in a field that has a validation rule and validation text in place. *True* *False*

9. Which data type results in a checkbox being displayed in the datasheet?

A. Short Text
B. Day/Time
C. AutoNumber
D. Yes/No

10. Which tool helps ensure data consistency?

A. Input mask
B. Hiding columns
C. Captions
D. Smart tags

11. Which database feature lets you restrict or limit data entry?

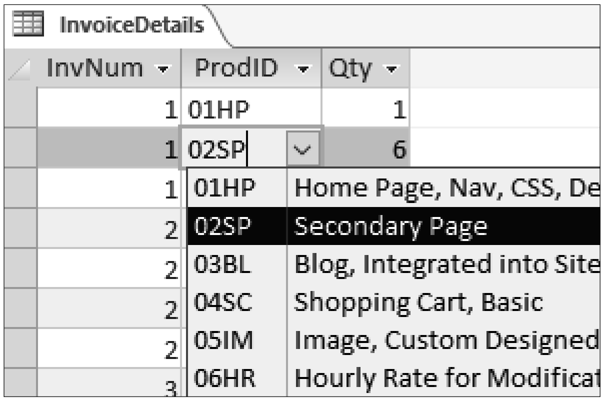
A. Input mask
B. Lookup field
C. Validation rule
D. All of these options

12. Which tool is used to document relationships?

A. Relationships report
B. Input Mask Wizard
C. Referential integrity
D. Lookup table

(continued)

13. Which database feature is being used in the ProdID field in the following image?



InvNum	ProdID	Qty
1	01HP	1
1	02SP	6
1		
2		
2		
2		
2		
3		

01HP	Home Page, Nav, CSS, De
02SP	Secondary Page
03BL	Blog, Integrated into Site
04SC	Shopping Cart, Basic
05IM	Image, Custom Designed
06HR	Hourly Rate for Modificat

A. Input mask

B. Lookup field

C. Validation rule

D. None of these options

2

ACCESS 2016

Customizing Input Forms

In this chapter, you will learn how to add a subform to a main form, which is a handy technique used to include data from a different source. You will also explore how to create calculated fields in a form.

LEARNING OBJECTIVES

- Create a form that contains a subform
- Add a calculated control to a form
- Add a total row to a form

CHAPTER TIMING

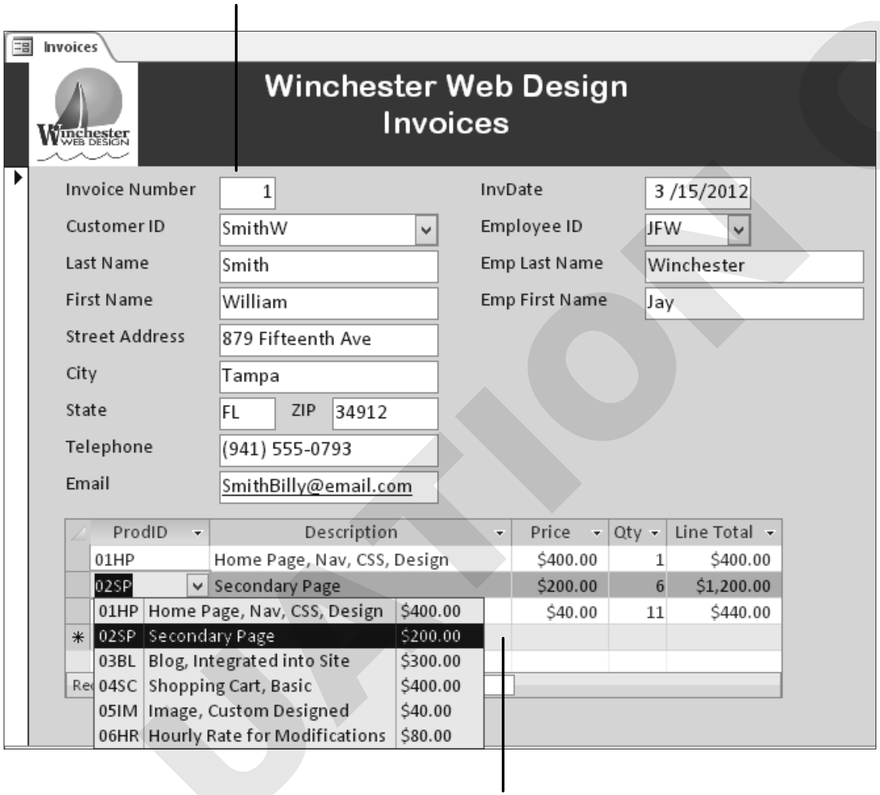
- Concepts/Develop Your Skills: 1 hr 30 mins
- Self-Assessment: 15 mins
- Total: 1 hr 45 mins

PROJECT: FORMATTING FUNCTIONAL FORMS

Winchester Web Design has seen sales increase over recent months and wants to simplify data entry. You must design advanced forms to make data entry easier and less prone to errors. You will create an Invoice form that contains a subform containing invoice details. Your form will also contain a calculated field for creating totals. The database relationships will be critical for setting up these forms.

Subforms

Although many forms are designed to enter data into a single table, there are times when you may need forms that perform actions such as processing customer invoices, calculating totals, and locating data from multiple tables. One of the best ways to accomplish this is through a subform, which is a secondary form placed on the main form. Subforms work well when one-to-many relationships are set, allowing the user to use multiple tables on a single form. Subforms are simply subsets of data linked by parent fields on the main form to child fields on the subform.



Winchester Web Design
Invoices

Invoice Number	1	InvDate	3 /15/2012
Customer ID	SmithW	Employee ID	JFW
Last Name	Smith	Emp Last Name	Winchester
First Name	William	Emp First Name	Jay
Street Address	879 Fifteenth Ave		
City	Tampa		
State	FL ZIP 34912		
Telephone	(941) 555-0793		
Email	SmithBilly@email.com		

ProdID	Description	Price	Qty	Line Total
01HP	Home Page, Nav, CSS, Design	$400.00	1	$400.00
02SP	Secondary Page	$200.00	6	$1,200.00
		$40.00	11	$440.00

01HP	Home Page, Nav, CSS, Design	$400.00
02SP	Secondary Page	$200.00
03BL	Blog, Integrated into Site	$300.00
04SC	Shopping Cart, Basic	$400.00
05IM	Image, Custom Designed	$40.00
06HR	Hourly Rate for Modifications	$80.00

Main form displaying customer and employee information

Detail data from related tables appears in a subform.

Creating Subforms

The easiest way to create a subform is with the Form Wizard, which contains an option to add a subform. Subforms can be added to existing forms using the Subform control in the Property Sheet. When this option is used, you are able to specify the position and size of the subform on the main form. A Subform Wizard is then launched, in which you specify the subform properties.

☰ Create→Forms→Form Wizard ; Design→Controls→Subform

DEVELOP YOUR SKILLS: A2-D1

In this exercise, you will use the Form Wizard to create an invoice form that contains an invoice details subform.

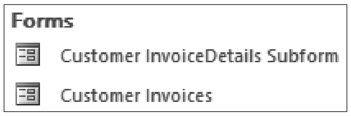
1. Open **A2-D1-WinWebDesign** from your **Access Chapter 2** folder and save it as **`A2-D1-WinWebDesignRevised`**.
2. Click the **Invoices** table in the Navigation pane.
3. Choose **Create→Forms→Form Wizard** .
4. Add **InvNum** and **CustID** to the Selected Fields list.
5. Choose **Table: Customers** from the Tables/Queries list then add the following fields to the Selected Fields list: **CustLastName**, **CustFirstName**, **CustStreetAddress**, **CustCity**, **CustState**, **CustZIP**, **CustPhone**, and **CustEmail**.
6. Add the fields from the following tables to the Selected Fields list in the order shown here:

Table	Fields
Invoices	InvDate
Employees	EmpID
	EmpLastName
	EmpFirstName
InvoiceDetails	ProdID
Products	ProdDescription
	Price
InvoiceDetails	Qty

7. Click **Next** and notice the data is arranged by Invoices and the **Form with Subform(s)** option is already chosen.
8. Click **Next** to accept the settings and then click **Next** again to accept the Datasheet layout for the subform.
9. In the final Wizard screen, name the form **`Customer Invoices`**, name the subform **`Customer InvoiceDetails Subform`**, and click **Finish**.
10. Double-click the right border of each subform column heading to autofit the columns.
11. Click in the empty **ProdID** cell in the new row at the bottom of the subform.
12. Click the drop-down **menu** button ▼ and choose **03BL**.
13. Enter **`1`** in the Qty column and tap Tab to complete the record.

Modifying Subforms

When a subform is created with the Form Wizard, Access creates both the subform and a main form with the subform embedded within it, and it displays both in the Navigation pane. The subform can then be opened and modified by itself in either Layout View or Design View. Or the main form can be opened and both it and the embedded subform can be modified together.


Forms
Customer InvoiceDetails Subform
Customer Invoices

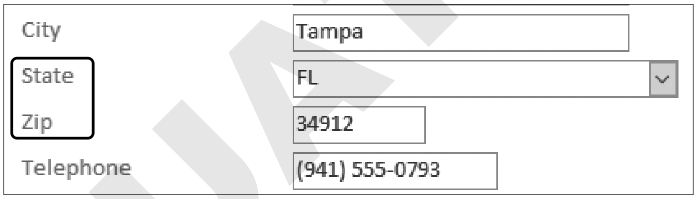
The Customer InvoiceDetails Subform and the Customer Invoices main form are displayed in the Navigation pane.

Subforms and their controls have their own Property Sheets that allow you to precisely control the subform layout and design. And you can use the same keystroke and mouse techniques to arrange and size subform controls that you use with main forms.

DEVELOP YOUR SKILLS: A2-D2

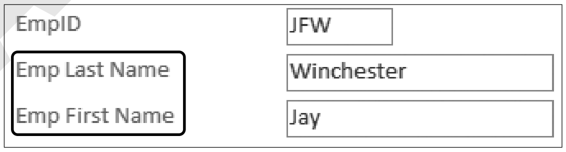
In this exercise, you will modify the Customer Invoices form and subform by deleting, moving, and sizing controls, and adding a graphic to the form.

1. Switch to **Layout View**.
2. Click the **ST** label in the main form to select it and then click inside the label and change it to **`State`**.
3. Change the *ZIP* label to **`Zip`**.



City	Tampa
State	FL
Zip	34912
Telephone	(941) 555-0793

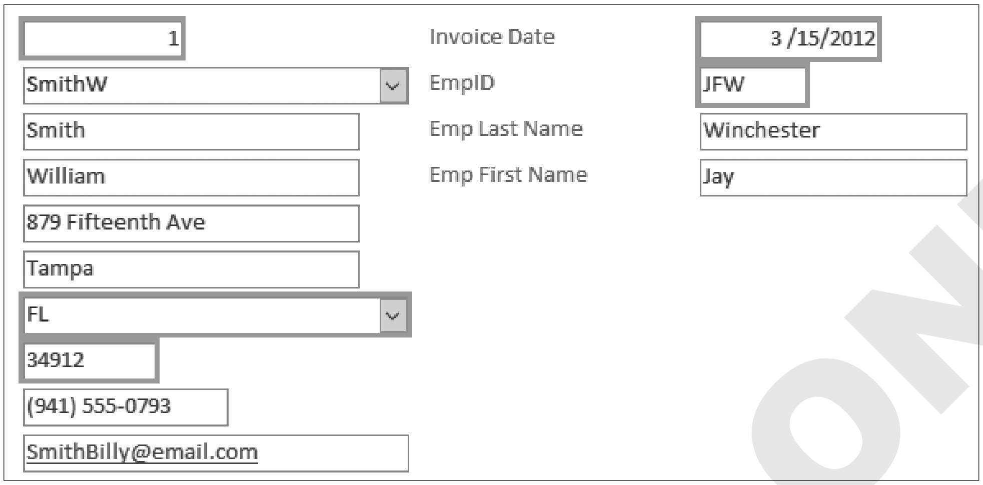
4. Locate the *EmpID* label and notice the *Last Name* and *First Name* labels below it.
5. Change the employee *Last Name* label to **`Emp Last Name`** and the employee *First Name* label to **`Emp First Name`**.



EmpID	JFW
Emp Last Name	Winchester
Emp First Name	Jay

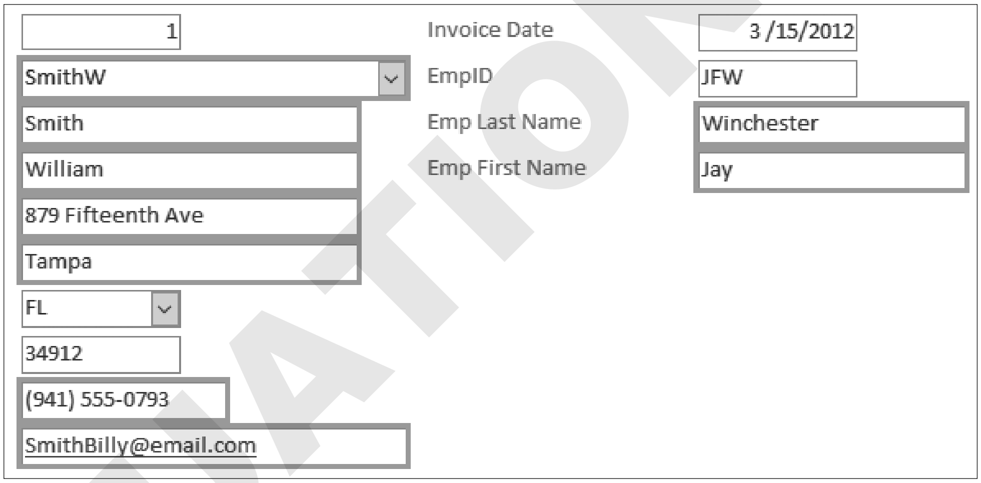
6. Display the Property Sheet by choosing **Design→Tools→Property Sheet** .

7. Select the text boxes shown here by pressing and holding the Ctrl key while clicking the boxes, and then set the Width property to `1"`.



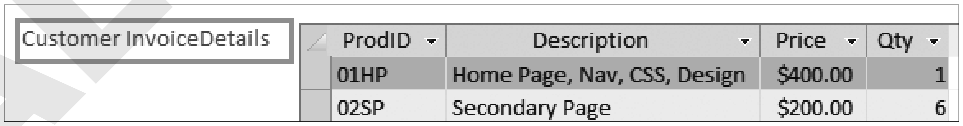
1	Invoice Date	3 /15/2012
SmithW	EmpID	JFW
Smith	Emp Last Name	Winchester
William	Emp First Name	Jay
879 Fifteenth Ave		
Tampa		
FL		
34912		
(941) 555-0793		
SmithBilly@email.com		

8. Select the text boxes shown here and set the Width property to `2"`.



1	Invoice Date	3 /15/2012
SmithW	EmpID	JFW
Smith	Emp Last Name	Winchester
William	Emp First Name	Jay
879 Fifteenth Ave		
Tampa		
FL		
34912		
(941) 555-0793		
SmithBilly@email.com		

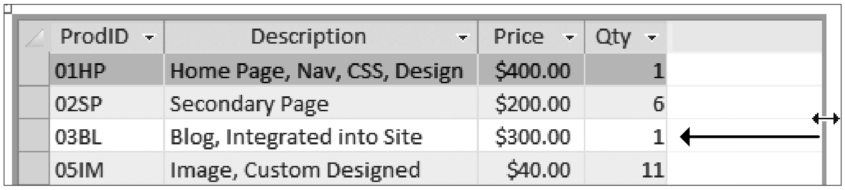
9. Select the *CustomerInvoiceDetails* label on the subform and tap Delete to remove it.



Customer InvoiceDetails

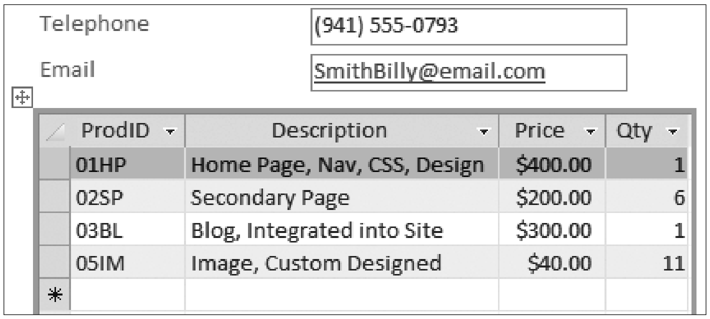
ProdID	Description	Price	Qty
01HP	Home Page, Nav, CSS, Design	$400.00	1
02SP	Secondary Page	$200.00	6

10. Click the right border of the subform frame and then drag left until the frame is slightly wider than the subform.



ProdID	Description	Price	Qty
01HP	Home Page, Nav, CSS, Design	$400.00	1
02SP	Secondary Page	$200.00	6
03BL	Blog, Integrated into Site	$300.00	1
05IM	Image, Custom Designed	$40.00	11

11. With the subform frame still selected, use the left arrow ← key to nudge the subform to the left until it aligns with the main form labels.



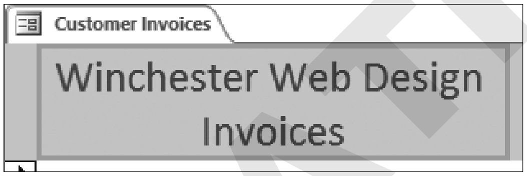
Telephone (941) 555-0793

Email SmithBilly@email.com

ProdID	Description	Price	Qty
01HP	Home Page, Nav, CSS, Design	$400.00	1
02SP	Secondary Page	$200.00	6
03BL	Blog, Integrated into Site	$300.00	1
05IM	Image, Custom Designed	$40.00	11
*			

12. Use these guidelines to modify the form as shown here:

- Click just in front of *Invoices* in the title box and then press Shift + Enter to move *Invoices* to a new line.
- Click on the right border of the title box and then drag right to widen the box by about **0.5**".
- Replace *Customer* with **`Winchester Web Design`**
- Adjust the box width again so it is just wide enough to keep *Winchester Web Design* on a single line.
- Select both lines and use **Home→Text Formatting→Center** to center the title within the box.



Winchester Web Design
Invoices

13. Click to the right of the title box in an empty part of the Form Header.

14. If necessary, display the Property Sheet and click in the **Back Color** property box.

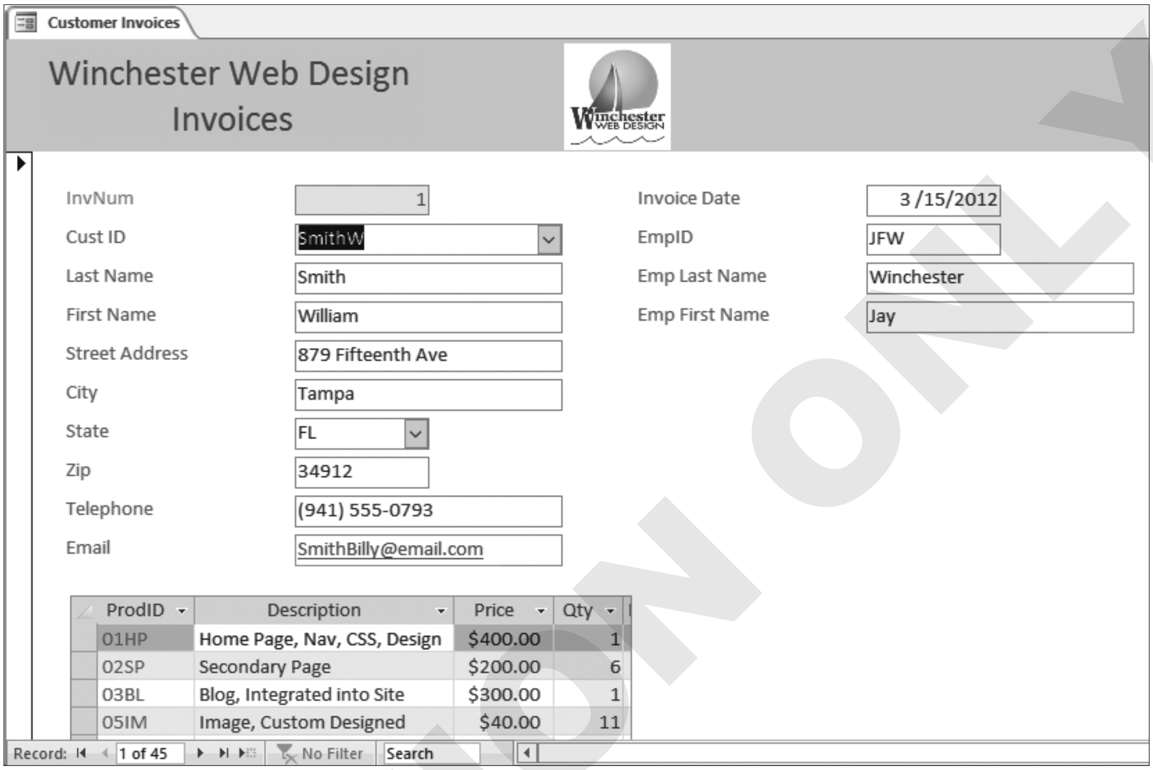
15. Click the **Build** ··· button and choose a light blue color or whatever color you feel looks best in the Form Header.

16. Choose **Design→Header/Footer→Logo**.

17. Navigate to your **Access Chapter 2** folder, choose **WWD-Logo.jpg**, and click **OK**.

18. Set the Width and Height properties to **`0.8"`** and the Left property to **`4"`**.

19. Review your completed form and then close it, saving the changes to both the form and subform.



Winchester Web Design
Invoices

InvNum	1	Invoice Date	3 /15/2012
Cust ID	SmithW	EmpID	JFW
Last Name	Smith	Emp Last Name	Winchester
First Name	William	Emp First Name	Jay
Street Address	879 Fifteenth Ave		
City	Tampa		
State	FL		
Zip	34912		
Telephone	(941) 555-0793		
Email	SmithBilly@email.com		

ProdID	Description	Price	Qty
01HP	Home Page, Nav, CSS, Design	$400.00	1
02SP	Secondary Page	$200.00	6
03BL	Blog, Integrated into Site	$300.00	1
05IM	Image, Custom Designed	$40.00	11

Adding Calculations to Forms

There are several ways to add calculations to forms. The easiest way is to build a form based on a query that already has a calculated field. If your form is based on one or more tables or is based on a query without a calculated field, then other methods are used.

Applying Totals to Forms in Datasheet Layout

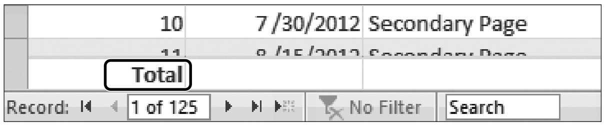
The Form Wizard has an option that lets you create a form in Datasheet Layout View. A form in Datasheet Layout View looks just like a table in Datasheet View. The Totals feature is available for tables in Datasheet View and for forms in Datasheet Layout View. The Totals feature lets you easily use aggregate functions such as count, sum, and average to create totals for numeric fields in the datasheet.

☰ Home → Records→Totals Σ

DEVELOP YOUR SKILLS: A2-D3

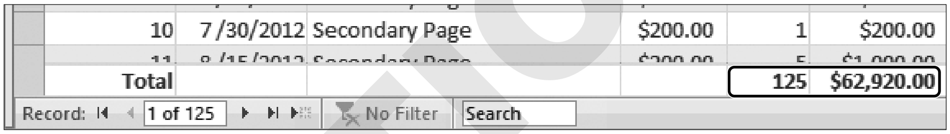
In this exercise, you will create a form that uses the Totals feature to count the number of individual line items on customer invoices and to total the amounts of all invoices.

1. Choose **Invoices Query** in the Navigation pane and then choose **Create→Forms→ Form Wizard**.
2. Move all fields to the Selected Fields list and then click **Next**.
3. Choose **Datasheet** as the layout and click **Next**.
4. Name the new form **`Invoices Query Form`** and click **Finish**.
5. **Choose Home→Records→Totals** Σ to make a Total line appear at the bottom of the datasheet layout.



```
10   7/30/2012 Secondary Page
11   8/15/2012 Secondary Page
Total
Record: 1 of 125   No Filter   Search
```

6. Click in the **Qty** cell on the Total row and choose **Count,** and then click in the **LineTotal** cell and choose **Sum**.



```
10   7/30/2012 Secondary Page   $200.00   1     $200.00
11   8/15/2012 Secondary Page   $200.00   5   $1,000.00
Total                                   125  $62,920.00
Record: 1 of 125   No Filter   Search
```

7. Scroll down through the datasheet's 125 rows and notice that the Total row remains fixed at the bottom of the window.
8. Close the Invoices Query Form, saving the changes if prompted.

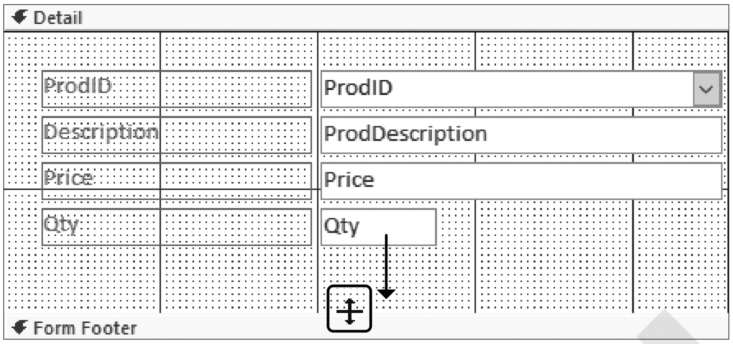
Creating Calculated Controls in Forms

The Totals feature is useful when you want to create totals for all records in a datasheet. But sometimes it is necessary to display calculations in Form View, which displays just a single record at a time. This is done by inserting a new text box control on the form and inserting a formula in the Control Source property. The formula references other controls on the form that are bound to underlying database fields. For example, you would use the formula =Price*Qty to display the total amount of a transaction, with both the Price and Qty shown on the form.

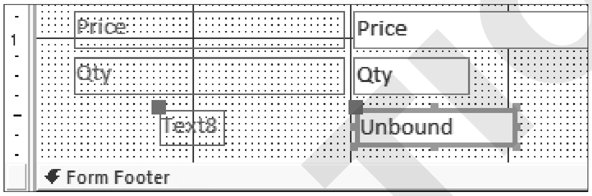
DEVELOP YOUR SKILLS: A2-D4

In this exercise, you will add a calculated control to the Customer InvoiceDetails subform.

1. Display the **Customer InvoiceDetails** subform in **Design View**.
2. Position the mouse pointer on the top edge of the **Form Footer** section bar until the resize pointer appears and then drag down slightly to make room for a new text box.

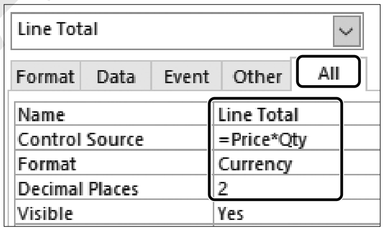


3. Choose **Design→Controls→Text Box** .
4. Click in the space you just created below the Qty control to insert a new control as shown here.



5. Use the arrow keys as needed to nudge the control so it is aligned with the Qty control.
6. Make sure the new control is still selected and, if necessary, choose **Design→Tools→ Property Sheet** to display the Property Sheet.
7. Click the **All** tab in the Property Sheet box and set these four properties:

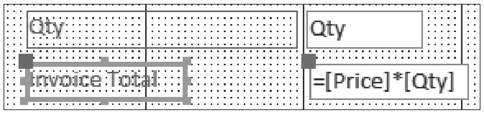
   Name: `Line Total`
   Control Source: `=Price*Qty`
   Format: `Currency`
   Decimal Places: `2`



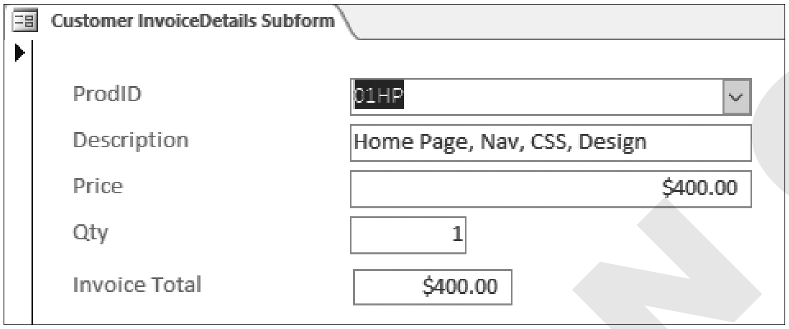
Name	Line Total
Control Source	=Price*Qty
Format	Currency
Decimal Places	2
Visible	Yes

8. Click the *Text8* label on the subform and set the following properties:

   Caption: **`Invoice Total`**
   Width: **`1"`**
   Left: **`0.25"`**



9. Switch to **Form View** to see your new calculated control in action.


Customer InvoiceDetails Subform

ProdID 01HP
Description Home Page, Nav, CSS, Design
Price $400.00
Qty 1
Invoice Total $400.00

10. Navigate to record 16 and other records where the Qty is greater than 1, and notice your calculated control always performs the correct calculations.

11. Close the form when you have finished and save the changes.

12. Choose **File→Close** to close the database, saving the changes to any unsaved forms.

Self-Assessment

Check your knowledge of this chapter's key concepts and skills by completing the Self-Assessment. The answers to these questions can be found at the back of this book.

1. A subform is a secondary form that is placed on a main form. *True* *False*

2. Subforms can be created using the Form Wizard. *True* *False*

3. The only way to modify a subform is to open it by itself in Design View. *True* *False*

4. Property sheets are not available for subform controls. *True* *False*

5. Forms can display calculated fields from queries. *True* *False*

6. For forms, the Totals feature is only available in Datasheet Layout View. *True* *False*

7. A calculated control is created in a form by inserting a formula in the Control Source property of a text box. *True* *False*

8. A subform is linked to the parent form by placing the parent field on the subform. *True* *False*

9. You may include fields from multiple tables on a single form by creating subforms. *True* *False*

10. An aggregate function may be added to a form in Datasheet Layout View. *True* *False*

11. Which function is available through the Totals feature?

A. Count
B. Sum
C. Average
D. All of these options

12. In which property would you insert a formula on a form?

A. Control Source
B. Design View
C. Value
D. Formula

3

ACCESS 2016

Creating Complex Queries

In this chapter, you will explore queries designed to enhance the timeliness and accuracy of large relational databases. You will create crosstab queries and use parameter queries that prompt you to enter values to generate or modify records. And you will create action queries to update databases and automate database tasks.

LEARNING OBJECTIVES

- Create a crosstab query
- Create a find unmatched query
- Create a find duplicates query
- Create and run parameter queries
- Create and run action queries

CHAPTER TIMING

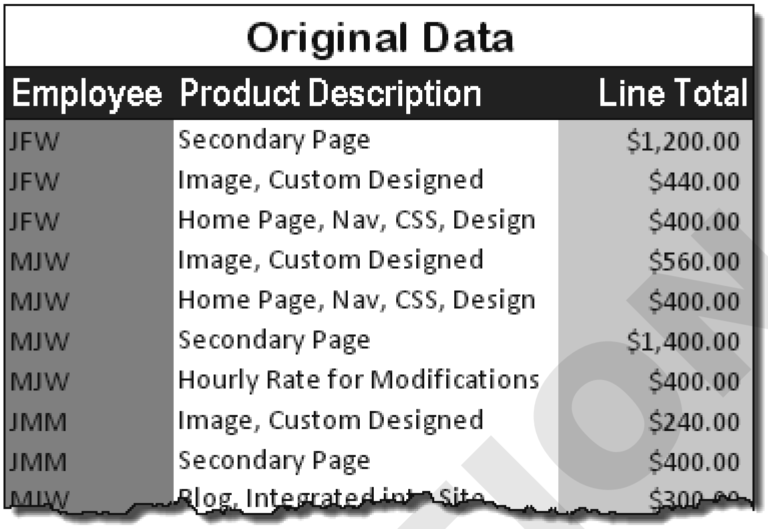
- Concepts/Develop Your Skills: 2 hrs
- Self-Assessment: 15 mins
- Total: 2 hrs 15 mins

PROJECT: HANDLING GROWING DATABASES

You are responsible for analyzing the data retrieval processes for the growing Winchester Web Design database. You decide to develop queries to increase the efficiency of data entry and updates and to better analyze data. The tools you will use include crosstab queries for data analysis, parameter queries that prompt the user for input, and action queries to update and maintain the database.

Crosstab Queries

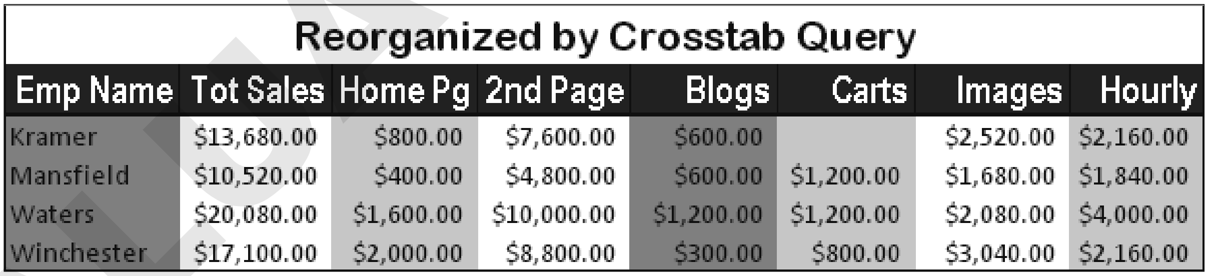
Crosstab queries allow you to easily analyze data. A crosstab query lists the fields to be grouped on the left side of the datasheet. It arranges the fields to be summarized across the top so you can calculate sums, averages, counts, or totals by group and subgroup. For example, if you have a database that contains sales records for your employees, the description of each product they sell, and their total sales for each product, you could create a crosstab query to display the total sales by product for each employee. Such a grouping and summarization might appear as shown in the following illustrations.



Original Data

Employee	Product Description	Line Total
JFW	Secondary Page	$1,200.00
JFW	Image, Custom Designed	$440.00
JFW	Home Page, Nav, CSS, Design	$400.00
MJW	Image, Custom Designed	$560.00
MJW	Home Page, Nav, CSS, Design	$400.00
MJW	Secondary Page	$1,400.00
MJW	Hourly Rate for Modifications	$400.00
JMM	Image, Custom Designed	$240.00
JMM	Secondary Page	$400.00
MJW	Blog, Integrated into Site	$300.00

The original data format is arranged by record.



Reorganized by Crosstab Query

Emp Name	Tot Sales	Home Pg	2nd Page	Blogs	Carts	Images	Hourly
Kramer	$13,680.00	$800.00	$7,600.00	$600.00		$2,520.00	$2,160.00
Mansfield	$10,520.00	$400.00	$4,800.00	$600.00	$1,200.00	$1,680.00	$1,840.00
Waters	$20,080.00	$1,600.00	$10,000.00	$1,200.00	$1,200.00	$2,080.00	$4,000.00
Winchester	$17,100.00	$2,000.00	$8,800.00	$300.00	$800.00	$3,040.00	$2,160.00

Using a crosstab query, you can display the data grouped by employee with totals for the various products.

Both tables and queries can be used as the basis of a crosstab query. Crosstab queries can be created while working with an existing query in Design View and choosing the Crosstab option or they can be created using the Query Wizard.

≡ Design→Query Type→Crosstab

DEVELOP YOUR SKILLS: A3-D1

In this exercise, you will create a crosstab query that lists each employee and their total invoice amount generated by product.

1. Open **A3-D1-WinWebDesign** from your **Access Chapter 3** folder and save it as `A3-D1-WinWebDesignRevised`.
2. Double-click the **Employee Sales** query to run the query and display the resulting datasheet.
3. Close the Employee Sales datasheet.
4. Choose **Create→Queries→Query Wizard**.
5. Choose **Crosstab Query Wizard** and click **OK**.
6. Choose the **Queries** View option then choose **Query: Employee Sales** from the query list.
7. Click **Next** to accept the Employees Sales query as the basis of your crosstab query.
8. Choose **EmpLastName** from the Available fields list and add it to the Selected Fields list.
9. Click **Next** and choose **ProdDescription** for the column headings.
10. Click **Next** and choose **LineTotal** from the Fields list and then choose **Sum** from the Functions list.
11. Leave the **Yes, Include Row Sums** option checked and click **Next**.
12. Leave the query name as Employee Sales_Crosstab and click **Finish**.
13. Close the query when you have finished.

Find Queries

Database tables often contain common fields that are used to link or relate the tables. For example, the product id field from a Products table also appears in an Invoices table so that the invoice displays the appropriate products. Thus, it is important that records entered in one table have a matching record in the related table. For example, an invoice should never list a product that is not in the products table.

Sometimes databases are poorly designed, allowing data to be incorrectly entered. Data is also sometimes imported from other data sources, also resulting in incorrect or duplicate data. Fortunately, Access provides two additional Query Wizard options to help resolve these types of data conflicts.

Find Unmatched Query

The find unmatched query locates records in one table that have no related records in another table. For example, you could create a find unmatched query to ensure that each record in an invoice table has a corresponding record in a Customers or Products table.

Find Duplicates Query

A find duplicates query locates records containing duplicate field values in a single table or query datasheet. For example, you could create a find duplicates query to locate all customers with the same last name in a Customers table or to find all customers located in a particular state or ZIP code.

DEVELOP YOUR SKILLS: A3-D2

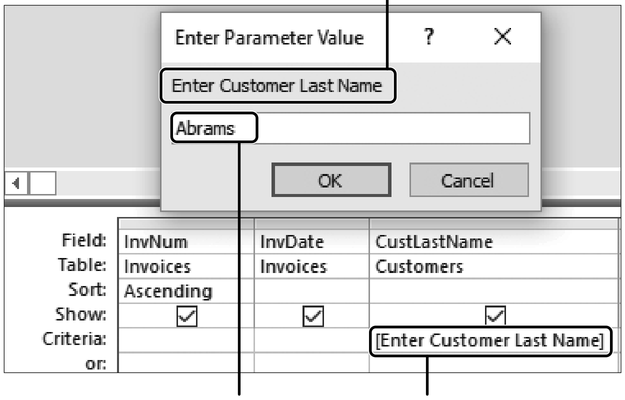
In this exercise, you will first create a query to locate records in the Customers table that do not have a matching CustomerID in the Invoices table. You will then create a query to identify records with duplicate customer last names.

1. Choose **Create→Queries→Query Wizard**, choose **Find Unmatched Query Wizard**, and click **OK**.
2. Click **Next** to choose the Customers table.
3. Choose the **Invoices** table and click **Next**.
4. Click **Next** to accept CustID as the matching field.
5. Add **CustLastName**, **CustFirstName**, **CustPhone**, and **CustEmail** to the Selected Fields list.
6. Click **Next** and then click **Finish** to accept the default query name Access assigns.
7. Close the datasheet after you have finished reviewing it.
8. Choose **Create→Queries→Query Wizard**.
9. Choose **Find Duplicates Query Wizard** and click **OK**.
10. Click **Next** to choose the Customers table as the query you want to use to search for duplicate field values.
11. Add **CustLastName** to the Duplicate Value Fields list and click **Next**.
12. Add **CustFirstName** and **CustPhone** to the Additional Query Fields list and click **Next**.
13. Name the query `Customers with the Same Last Name` and click **Finish**.
14. Close the query when you finished reviewing the results.

Parameter Queries

A parameter query is a select query that prompts the user to enter new criteria values each time the query is run. The query then generates results based on the value(s) entered. For example, a parameter query that searches for customers with a specific last name might prompt the user to enter the desired last name when the query is run. The query then returns only records containing the last name entered by the user. Parameter queries are created by enclosing the desired prompt text with square brackets, [ ], in the query Criteria row.

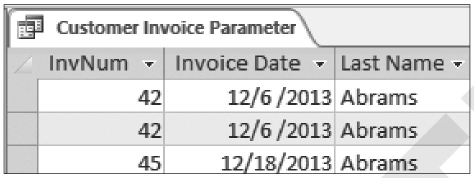
The prompt text appears here when the query is run.



Field:	InvNum	InvDate	CustLastName
Table:	Invoices	Invoices	Customers
Sort:	Ascending		
Show:	☑	☑	☑
Criteria:			[Enter Customer Last Name]
or:			

The user enters the desired parameter value here.

A parameter query is created by enclosing the desired prompt text in square brackets in a Criteria cell.

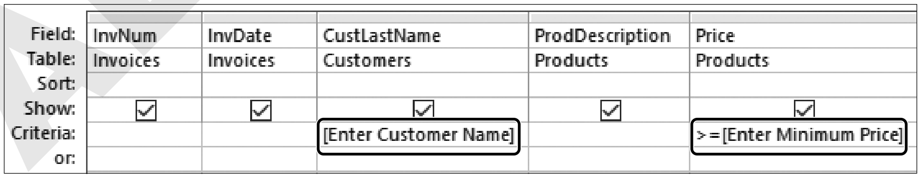


InvNum	Invoice Date	Last Name
42	12/6 /2013	Abrams
42	12/6 /2013	Abrams
45	12/18/2013	Abrams

Only records where the customer last name is Abrams are returned.

Complex Parameter Queries

Suppose you want to see all items purchased by a particular customer and those that are equal to or greater than a particular price. For example, all items purchased by Abrams that have a price greater than or equal to $300. This is done by creating an AND condition using parameters in the customer last name and price fields.



Field:	InvNum	InvDate	CustLastName	ProdDescription	Price
Table:	Invoices	Invoices	Customers	Products	Products
Sort:					
Show:	☑	☑	☑	☑	☑
Criteria:			[Enter Customer Name]		>=[Enter Minimum Price]
or:					

A parameter query with two parameters

You can also create expressions with prompts for multiple values in the same query field or include logical criteria such as greater than (>), less than (<), and equal to (=). The following table shows examples of more complex single-field parameter query expressions.

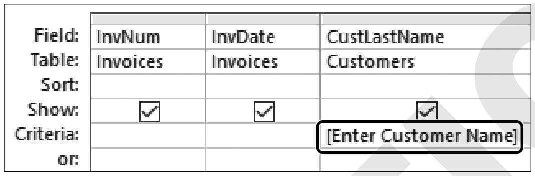
EXAMPLES OF PARAMETER QUERY CRITERIA FOR A SINGLE FIELD

Parameter Criteria	Result
Between [What is the start date?] And [What is the end date?]	Prompts the user to enter a start date and end date. Access recognizes the Between and And expressions and returns dates that are within the range entered by the user.
>=[Enter minimum price]	Displays the prompt "Enter minimum price" and returns only records that are greater than or equal to the price entered by the user.

DEVELOP YOUR SKILLS: A3-D3

In this exercise, you will use parameters to return customer records based on user input.

1. Display the **Customer Invoice Parameter** query in **Design View**.
2. Click in the **CustLastName** criteria field and enter the criterion **[Enter Customer Name]**.



Field:	InvNum	InvDate	CustLastName
Table:	Invoices	Invoices	Customers
Sort:			
Show:	☑	☑	☑
Criteria:			[Enter Customer Name]
or:			

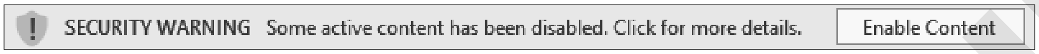
3. Run the query, type **Roberts** in the parameter box that appears, and click **OK**.
4. Switch to **Design View** and enter the following parameter in the Price criteria box **>= [Enter Minimum Price]**.
5. Run the query and enter **Roberts** in the first parameter box that appears and **300** in the second box.
6. Close the query, saving the changes.

Action Queries

An action query performs an action that modifies a database table or a group of records in a table. Action queries can modify, move, update, or delete groups of records with a single action. You can even use an action query to create a new table by adding various fields and data from other tables.

An action query is run whenever it is opened. So, if you create an update query designed to increase prices by 10 percent on all items in a table, Access will increase those prices every time you run the query. Action queries do this without opening the underlying tables that are being modified by the query. For this reason, an action query may accidentally be run more than once, inadvertently changing the underlying table data multiple times. Thus, it is good practice to delete action queries after running them. This will help maintain the validity of the database as changes to the underlying data cannot easily be undone.

Action queries require that content within a database be enabled. As a result, if you did not click the Enable Content button found at the top of the Access window when you first opened the database, Access will display an error message advising you to enable content before you can create or run action queries.



SECURITY WARNING Some active content has been disabled. Click for more details. Enable Content

Make Table Queries

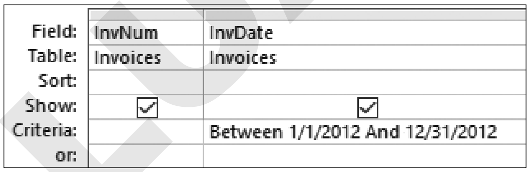
A make table query is an action query that can create a new table using data from multiple database tables. It's also a great way to move data produced from a calculated query field into a table. When you create a new table using a make table query, Access prompts you for a table name and even allows you to save the data in another database. A reason to move records to another database, for example, would be in order to archive them when they become obsolete, such as when a product is no longer available. If you rerun a make table query, Access will replace the table that was created with the previous running of the query. To retain the previously created table you must first rename it so that it isn't replaced.

≡ Design→Query Type→Make Table

DEVELOP YOUR SKILLS: A3-D4

In this exercise, you will create a make table action query to save all of the 2012 invoice records in a new table.

1. Open **Invoices Query** in **Design View**.
2. Enter **`Between 1/1/2012 And 12/31/2012`** as the criteria for the InvDate field.



Field:	InvNum	InvDate
Table:	Invoices	Invoices
Sort:		
Show:	☑	☑
Criteria:		Between 1/1/2012 And 12/31/2012
or:		

3. Choose **Design→Query Type→Make Table**.
4. Enter **`2012 Invoices`** as the table name and click **OK**.
5. **Run** ! the query and choose **Yes** to paste 61 rows into a new table.
7. Close the 2012 Invoices table and then close the invoices query without saving the changes.
8. Now open the **Invoices** table in **Datasheet View**.
9. Close the Invoices table.

Append Queries

An append query adds a group of records from one or more tables to the end of one or more tables in the same or in another database. For example, if you want to offer a new set of products, you could use an append action query to add the new items from a new products table to the existing products table. Or you might use an append query to automatically add new customers to the Customers table the first time a customer places an order.

Formatting the Source and Destination Tables

In an append query, the table that records are drawn from is called the source table. The table receiving the records is the destination table. To successfully run an append query, the structures, field names, data types, and field order for both tables should be the same.

Identifying the Source and Destination Tables

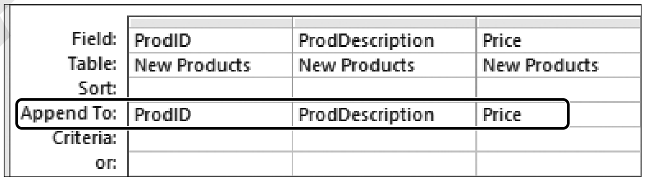
Append queries are created in the database that contains the source table. When the query is run, the Append dialog box prompts you to identify the destination database and table. Access identifies the destination table in the Append To row of the query grid.

☰ Design→Query Type→Append +!

DEVELOP YOUR SKILLS: A3-D5

In this exercise, you will create an action query to append records from the New Products table to the existing Products table.

1. Open the **Products** table and notice that it contains 6 records.
2. Open the **NewProducts** table to see the 5 records that will be appended to the Products table.
3. Close both tables.
4. Choose **Create→Queries→Query Design** to create a new query.
5. Add the **New Products** table to the query window and then close the Show Table box.
6. Add all fields from the New Products table to the query grid.
7. Choose **Design→Query Type→Append** +!.
8. Click the **Table Name menu** button ▼, choose **Products**, and click **OK** to add an Append To row to the query



Field:	ProdID	ProdDescription	Price
Table:	New Products	New Products	New Products
Sort:			
Append To:	ProdID	ProdDescription	Price
Criteria:			
or:			

9. **Run** ! the query and choose **Yes** to append the five rows to the Products table.
10. Display the **Products** table in **Datasheet View** to verify that the new records were appended.

11. Close the Products table.
12. Close the new append query, saving it as `Append Products`.

Update Queries

An update query is an action query that makes global changes to a group of records in one or more tables. For example, you can use an update query to increase the prices for every product in a specific category or to update the area code for phone numbers that change when the phone company adds or changes an area code. To ensure that the corresponding fields in related tables are updated consistently, check the Cascade Update Related Fields checkbox in the Edit Relationships window.

Identifying the Query Grid Update Row

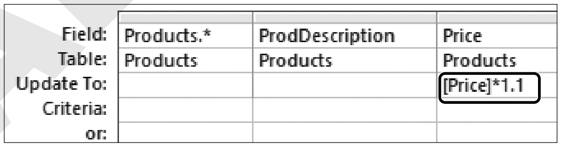
Append, update, crosstab, and delete queries all add a query-specific row to the query grid. The update query places an Update To row in the query grid so that you can tell Access how to update the desired field(s). In most cases, this will be changing one value to another by substitution, mathematical operation, formula, or comparison.

≡ Design→Query Type→Update

DEVELOP YOUR SKILLS: A3-D6

In this exercise, you will create an update action query that increases the prices of every item in the Products table by 10 percent.

1. Open the **Products** table in **Datasheet View** and notice the Home Page price is $400.
2. Close the Products table.
3. Choose **Create→Queries→Query Design**.
4. Add the **Products** table to the query window and then close the Show Table box.
5. Add all fields from the Products table to the query grid.
6. Choose **Design→Query Type→Update**.
7. Click in the **Update To** cell for the Price field and enter **`[Price]*1.1`**.



Field:	Products.*	ProdDescription	Price
Table:	Products	Products	Products
Update To:			[Price]*1.1
Criteria:			
or:			

8. **Run** ! the query and choose **Yes** when the warning prompt appears.
9. Close the query without saving it.
10. Open the **Products** table in **Datasheet View** and notice the Home Page price went from $400 to $440 (an increase of 10%).
11. Close the Products table.

Delete Queries

A delete query deletes a group of records from one or more tables. For example, you could create a delete query to remove records for a discontinued line of products or to delete records you have appended to another table to prevent inadvertently running an append query multiple times.

Preparing for Delete Queries

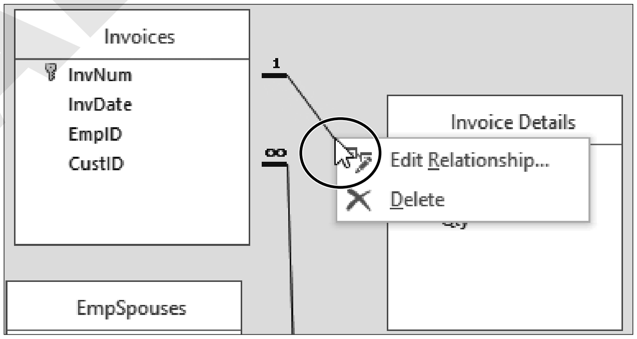
To ensure that corresponding records in related tables will all be deleted concurrently, check the Cascade Delete Related Records checkbox in the Edit Relationships window. When you set up a delete query, Access replaces the Sort row of the query grid with a Delete row. You can set criteria for specific fields in a table to identify the conditions that must be met in order to delete records, or you can set no criteria to remove all records from a table.

≡ Design→Query Type→Delete

DEVELOP YOUR SKILLS: A3-D7

In this exercise, you will create a delete query to remove the 2012 invoices from the Invoices table.

1. Open the **Invoices** table in **Datasheet View**.
2. Close the Invoices table.
3. Choose **Create→Queries→Query Design**.
4. Add the **Invoices** table to the query window and then close the Show Table box.
5. Add only the **InvDate** field to the query grid.
6. Enter `Between 1/1/2012 and 12/31/2012` as the criteria for the InvDate field.
7. Choose **Design→Query Type→Delete**.
8. **Run** ! the query and choose **Yes** when the prompt to delete 19 rows appears.
9. Choose **No** in the warning message box.
10. Choose **Database Tools→Relationships→Relationships**.
11. Right-click the **join line** between the Invoices and Invoice Details table and choose **Edit Relationship**.



12. Check the **Cascade Delete Related Records** box and click **OK**.
13. Close the Relationships window, saving the changes to the relationship layout.
14. **Run** ! the query again, choosing **Yes** when the warning prompt appears.
15. Close the query without saving it.
16. Open the **Invoices** table in **Datasheet View** and notice the 2012 invoices have been removed.
17. Close the Invoices table.
18. Choose **File→Close** to close the database.

Self-Assessment

Check your knowledge of this chapter's key concepts and skills by completing the Self-Assessment. The answers to these questions can be found at the back of this book.

1. Both tables and queries can be used as the basis of a crosstab query. *True* *False*

2. A find duplicates query could be used to locate all customers in a particular zip code. *True* *False*

3. Parameter queries are limited to one parameter per query. *True* *False*

4. An append query adds records to the end of a table. *True* *False*

5. An update query might be used to add records to the end of a table. *True* *False*

6. An action query is run whenever it is opened. *True* *False*

7. A make table query can create a new table using data from one or more existing tables. *True* *False*

8. Which function(s) is available using a crosstab query?

A. Count
B. Sum
C. Average
D. All of these options

9. Which query type might you use to ensure that a record in an invoice table has a corresponding record in a Customers table?

A. Crosstab
B. Delete
C. Update
D. Find unmatched

10. Which query type might you use to locate all customers with the same last name in a Customers table?

A. Find duplicates
B. Find unmatched
C. Crosstab
D. Update

11. Which query type might you use if you want to prompt a user for input each time the query is run?

A. Crosstab
B. Parameter
C. Update
D. Append

12. Which characters are used to enclose the desired prompt in a parameter query?

A. “ “
B. ( )
C. { }
D. [ ]

(continued)

13. Which query modifies records in an underlying table?

A. Parameter

B. Update

C. Crosstab

D. All of these options

14. Which query type(s) requires that the Cascade Delete Related Records checkbox is checked in the Edit Relationships window?

A. Parameter

B. Crosstab

C. Delete

D. All of these options

ACCESS 2016 4 Customizing Reports

Database reports summarize the data contained in tables or displayed in query results and enable you to provide information in a page layout suitable for printing. Although forms and reports serve two different purposes within the context of a relational database, the techniques used to customize them are similar. In this chapter, you will import reports from other databases and use features to create custom reports.

LEARNING OBJECTIVES

- Import a report into a database
- Add a subreport to a main report
- Create a report from a subreport
- Create calculated controls on a subreport
- Set page breaks in reports

CHAPTER TIMING

- Concepts/Develop Your Skills: 2 hrs
- Self-Assessment: 15 mins
- Total: 2 hrs 15 mins

PROJECT: BILLING CUSTOMERS

The company manager of Winchester Web Design, a small web page design company, wants you to improve its invoice report for customer billing. After reviewing invoices from several companies, the company manager has sketched out a design for the new invoice report layout. Your job is to create a sample of the new invoice report for the company's executive team.

Importing a Report into a Database

Access offers a variety of ways to create reports. In addition to using the Report Wizard or starting from scratch in Design View, you can also import reports from another database. Because most companies require some type of invoice to send with customer orders, locating a sample invoice report to import is not difficult.

Sometimes you have the report you want, but during its design it may have become corrupted, either due to inadvertent changes to the report itself or because of changes to an underlying query. That's when backups are invaluable. If a report becomes corrupted, you can restore it by importing database objects from a backup copy of a database. The record source should already match, and there should be no need to edit the properties or the field names.

Identifying Report Record Sources

Reports that you import retain two connections to their original database. The first is the source database table or query name, shown in the Record Source property and the second are the field names, which appear in report text boxes. As a result, when you import a report from another database, you often must establish new control sources to the destination database. You can accomplish this by:

- Editing the imported report's Record Source property to link to a table or query in the destination database.
- Editing, if necessary, the field names in the imported report's text boxes to match those shown in the new record source table or query.

≡ External Data→Import & Link→Access

DEVELOP YOUR SKILLS: A4-D1

In this exercise, you will import a report from a backup copy of a database. You will rename the report and view data from an existing table using the imported report.

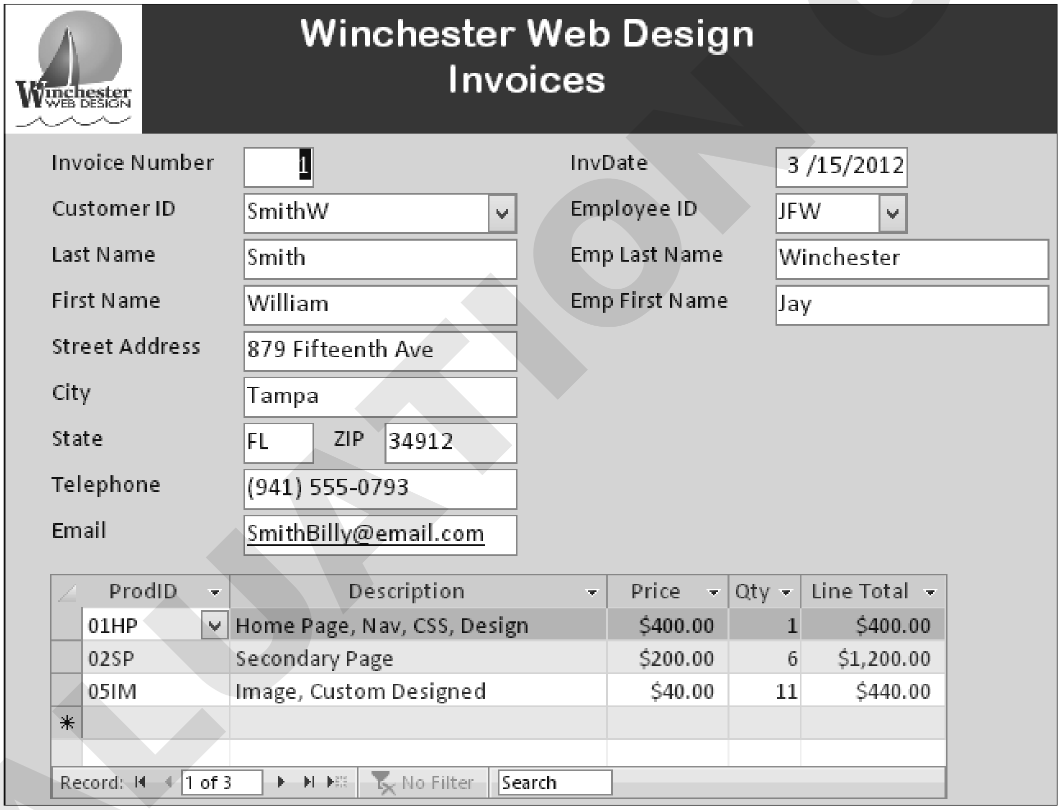
1. Open **A4-D1-WinWebDesign** from your **Access Chapter 4** folder and save it as `A4-D1-WinWebDesignRevised`.
2. Look in the Reports section in the Navigation pane and notice the database contains just two reports.
3. Choose **External Data→Import & Link→Access**.
4. Click **Browse**, navigate to your **Access Chapter 4** folder, choose **A4-D1-WinWebDesign -Backup**, and click **Open**.
5. Leave the storage setting as Import Tables, Queries, Forms, Reports, Macros, and Modules into the Current Database and click **OK**.
6. Click the **Reports** tab, choose **Customer Invoices**, and click **OK**.
7. Leave the Save Import Steps checkbox unchecked and click **Close** in the Get External Data dialog box.

8. Double-click the **Customer Invoices** report to open it in **Report View**.

9. If necessary, scroll down and then notice the empty space between the Customer Information and signature blocks.

Adding a Subreport to a Main Report

Subreports display subsets of data in reports and are derived from related database tables, similar to subforms on forms. However, a subreport can display table data by using a table, query, form, or another report as its source object. Forms are frequently created before reports and they often already display the desired data needed in a report. For these reasons, it is often best to use a subform as the basis of a subreport to streamline report design and layout.



Winchester Web Design
Invoices

Invoice Number	1	InvDate	3 /15/2012
Customer ID	SmithW	Employee ID	JFW
Last Name	Smith	Emp Last Name	Winchester
First Name	William	Emp First Name	Jay
Street Address	879 Fifteenth Ave		
City	Tampa		
State	FL ZIP 34912		
Telephone	(941) 555-0793		
Email	SmithBilly@email.com		

ProdID	Description	Price	Qty	Line Total
01HP	Home Page, Nav, CSS, Design	$400.00	1	$400.00
02SP	Secondary Page	$200.00	6	$1,200.00
05IM	Image, Custom Designed	$40.00	11	$440.00
*				

An invoice report that uses a subform to display invoice details

Adding a Subreport

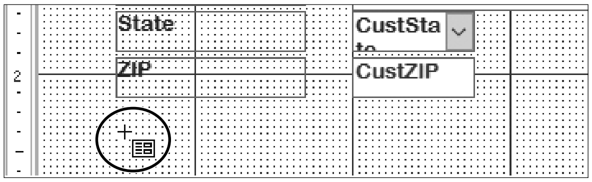
The procedures used to add a subreport to a report are basically the same as those used to add a subform to a form. You can create the subreport using the Report Wizard or add an unbound subreport control to the report. Then you identify the database object containing the fields you want to display as a subreport.

≡ Design→Controls→Subform/Subreport

DEVELOP YOUR SKILLS: A4-D2

In this exercise, you will add a subreport to the Customer Invoices report. You will use the InvoiceDetails subform as the source for the subreport.

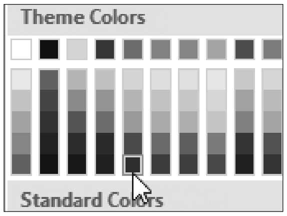
1. Display the **Customer Invoices** report in **Design View**.
2. Choose **Design→Controls→Subform/Subreport** (located at the bottom of the controls list).
3. Click just below the ZIP label to insert a control and launch the **SubReport Wizard**.



4. Choose **InvoiceDetails Subform** from the Use an Existing Report or Form option and click **Next**.
5. Click **Next** again to accept Choose from a List as the linking method.
6. Click **Finish** to accept InvoiceDetails Subform as the name.
7. Display the Property Sheet, if necessary, and click the **Data** tab.
8. Click the **Format** tab in the Property Sheet and set the following properties:

Property	Setting
Width	**5.6"**
Height	**1.5"**
Top	**2.9"**
Left	**0.5"**

9. Click the **InvoiceDetails Subform** label, which should be just below the ZIP label, and set the following properties:

Label	Property
Caption	**Invoice Details**
Width	**1.2"**
Height	**0.25"**
Top	**2.5"**
Left	**0.5"**
Font Name	**Arial Rounded MT Bold**
Font Size	**10**
Fore Color	**Blue, Accent 1, Darker 50%**
	

10. Switch to **Report View** to see your completed report.
11. Use the record Navigation bar and the scroll bar to review the database records.
12. Close the report, saving the changes.

Creating a Report from a Subreport

Using a subform as the record source for a subreport is convenient because the subform already includes all of the needed data. However, when a subreport uses a form as the record source, any changes made to the subreport layout are reflected in the source form. If you don't want the source form to be changed, you can save the subreport as a separate report in the database, change the main report's Record Source property to the new report object, and then edit the subreport.

Access allows you to save an existing form as a new form, and an existing report as a new report. When a subform is used as the record source for a subreport, you can open the subreport in a separate window and save it as a separate report.

☰ File→Save As→Save Object As

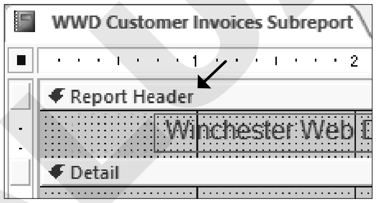
DEVELOP YOUR SKILLS: A4-D3

In this exercise, you will create and save a new report based on the subreport from the Customer Invoice Report. You will then edit the source object property in the main report to display the new subreport.

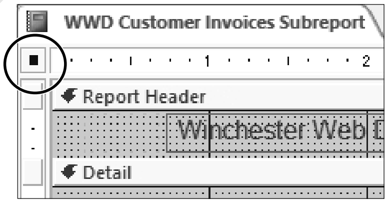
1. Open **InvoiceDetails Subform** from the Forms section of the Navigation pane.
2. Choose **File→Save As→Save Object As→Save As**.
3. Type `WWD Customer Invoices Subreport` as the name, choose **Report** from the **As** drop-down list, and click **OK**.
4. Close InvoiceDetails Subform.
5. Open the new **WWD Customer Invoices Subreport** in Design View.
6. Display the Property Sheet, if necessary.
7. Click the **All** tab and type `Customer Invoices Subreport` as the caption.
8. Choose **Design→Header/Footer→Title**.
9. Type `Winchester Web Design Invoice Details` in the title control and tap Enter.
10. Set the following property values for the new title control:

Property	Setting
Width	**3"**
Height	**0.25"**
Left	**1"**
Font Name	**Arial**
Font Size	**12**
Text Align	**Center**

11. Click the **Report Header** section bar and set the Height property to `0.3"`.



12. Click the **Report Selector** button and set the Width property to `5.6"`.



13. Close the subreport, saving the changes.
14. Open the **Customer Invoices** report in **Design View**.
15. If necessary, display the Property Sheet and click the **Data** tab.
16. Click on the **subreport** to select it.
17. Click in the **Source Object** property box and choose **Report.WWD Customer Invoices Subreport** from the drop-down menu.
18. Switch to **Report View** to see how the new subreport looks.
19. Scroll through the report, seeing how the subreport always shows the correct invoice details.
20. Close the Customer Invoices report, saving the changes.

Numbering Items in a Report

As the number of records in a table grows, the length and number of records in a report or subreport also grows. You can number the records in a report to help track the items listed. If a report is grouped, you can set the count to restart numbering at the beginning of each group.

Setting Properties to Number Items

By adding a text box to the Detail section and setting its Control Source property to *=1*, you can automatically number items in a report. In addition, you can set the Running Sum property to identify the portion of a report for which you want to count items. For example, suppose you have an invoice report that groups services by invoice number. You can set the Running Sum property to count the items in each group and then start counting again with the next group.

Numbering Subreports Separately

Access does not permit numbering items in a subreport control on a main report. However, because you saved the subreport as a separate report, you can add the numbering controls directly to the subreport by opening it in a separate window. Any edits you make when it is open as a separate item are reflected in the main report the next time you open it.

DEVELOP YOUR SKILLS: A4-D4

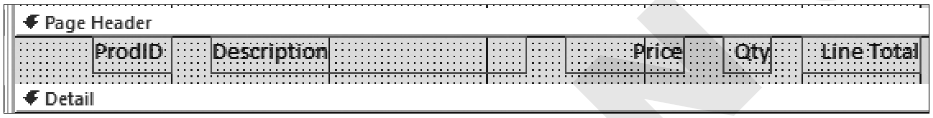
In this exercise, you will reposition the controls in the Page Header of the WWD Customer Invoices Subreport and then add a text box control to count the number of line items.

1. Display the **WWD Customer Invoices Subreport** in **Design View**.
2. Right-click the **Detail** section bar and choose **Page Header/Footer**.
3. Click the **Page Header** section bar and change the Height property to **`0.3"`**.

4. Select the **ProdID** label in the Detail section and use Ctrl+X to cut the control.
5. Click in the **Page Header** section and use Ctrl+V to paste the label.
6. With the **ProdID** label still selected, set both the Width and Left properties to `0.5"`.
7. Use the procedure in steps 4–6 to move the **Description**, **Price**, **Qty**, and **LineTotal** labels one at a time into the Page Header section, setting the Width and Left properties as follows:

Property	Width Setting	Left Setting
Description	2"	1.25"
Price	0.75"	3.5"
Qty	0.3"	4.5"
LineTotal	0.75"	5"

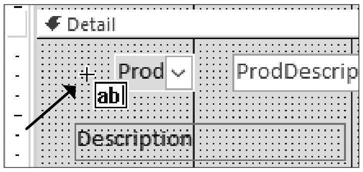
8. If necessary, click on the labels and use the up arrow ↑ key to nudge them up until they are vertically aligned with the ProdID label as shown here.



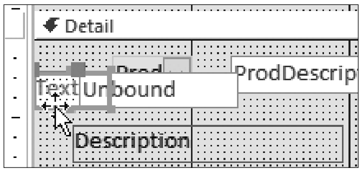
9. Set these property values for the text boxes in the **Detail** section:

Text Box Control	Width	Top	Left
ProdID	0.5"	0.1"	0.5"
ProdDescription	2"	0.1"	1.25"
Price	0.75"	0.1"	3.5"
Qty	0.3"	0.1"	4.5"
LineTotal	0.75"	0.1"	5"

10. Click the **Detail** section bar and set the Height property to `0.5"`.
11. Click the **Selection Type menu** button ▼ at the top of the Property Sheet and choose **Report**.
12. Set the Width property to `6"`.
13. Switch to **Report View**.
14. Switch to **Design View** and choose **Design→Controls→Text Box** ab|.
15. Click to the left of the *ProdID* text box in the Detail section to position the new box in that area.



16. Click the label of the new text box and tap Delete to remove it.



17. Click the new text box control and set these property values using the **All** tab in the Property Sheet:

Property	Setting
Name	**txtCount**
Width	**0.3"**
Top	**0.1"**
Left	**0.1"**

18. Click the **Data** tab and set the Control Source property to **`=1`** and the Running Sum property to **`Over Group`**.
19. Switch to **Report View** and scroll through the report.
20. Close the WWD Customer Invoices Subreport, saving the changes.
21. Display the **Customer Invoices** report in **Design View**.
22. Click the **InvoiceDetails Subform** to select it and set the Width property to **`6"`**.
23. Switch to **Print Preview** and use the Navigation bar to review the various report pages.
24. Close Print Preview and then close the Customer Invoices report, saving the changes.

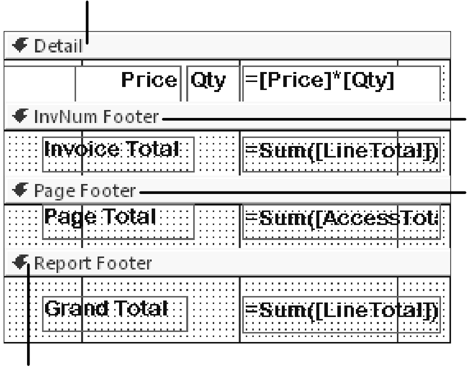
Creating Calculated Controls on a Subreport

Reports summarize data contained in tables and queries to present useful, organized information. This typically means that calculated fields must be added to a report for subtotals, grand totals, and averages.

Positioning Calculated Controls

Calculated controls are built in reports by using the Control Source property of an unbound text box control to which you add a formula. The placement of the calculated control determines how Access performs the calculation.

Calculated control in a Detail section performs a calculation for each detail line.



Calculated control in a Group Footer calculates the total for the group.

Calculated control in a Page Footer calculates the total for the page.

Calculated control in a Report Footer calculates the total for the entire report.

DEVELOP YOUR SKILLS: A4-D5

In this exercise, you will add a calculated control to the Report Footer section of the WWD Customer Invoices Subreport.

1. Display the **WWD Customer Invoices Subreport** in **Design View**.
2. Click the **Report Footer** section bar and set the Height property to `0.3"`.
3. Choose **Design→Controls→Text Box** ab|.
4. Click anywhere in the **Report Footer** section.
5. Click the **All** tab in the Property Sheet.
6. Set the following properties for the new control:

Property	Setting
Name	**CustomerTotal**
Control Source	**=Sum([Price]*[Qty])**
Format	**Currency**
Height	**0.25"**
Top	**0"**
Left	**4.75"**

7. Click the new text box label and set the following property values:

Property	Setting
Caption	**Invoice Total**
Width	**1.2"**
Height	**0.25"**
Top	**0"**
Left	**3"**
Fore Color	**Text Dark**

8. Close and save the changes to the subreport.
9. Display the **Customer Invoices** report in **Report View**.

Growing a Subreport

When the number of records or amount of data displayed in a subreport varies, you can set the Can Grow property setting to allow the subreport space to expand so more data displays vertically. You can also change the orientation of the print layout to allow more horizontal space on each report page.

DEVELOP YOUR SKILLS: A4-D6

In this exercise, you will adjust the margins of the Customer Invoices report and set the Can Grow and Can Shrink properties. These properties will adjust the size of the subreport to fit the contents.

1. Display the **Customer Invoices** report in **Design View**.
2. Choose **Page Setup→Page Size→Margins→Narrow**.
3. Click on the **InvoiceDetails Subreport** to select it.
4. Click the **Format** tab on the Property Sheet and, if necessary, set both the Can Grow and Can Shrink properties to **Yes**.
5. Choose **File→Save** to save the report.
6. Switch to **Report View** and scroll through the report.

Setting Page Breaks and Customizing Controls

As you view the Winchester Web Design Customer Invoices report in Report View, you may notice that the number of invoice records displayed on each screen varies depending on the number of items ordered. To ensure that each customer invoice starts on a new page, you can add a page break control. By default, when you view a report in Print Preview, data for each customer/record automatically appears on a separate page; however, multiple records appear on the same page when the report is displayed in Report View.

To add a title or general company information to an invoice, place title controls in the Page Header section rather than the Report Header, which only prints on the first page.

Positioning the Page Break Control

To ensure that each invoice record prints on a separate sheet of paper, you can set page breaks. When you set page breaks, add the break at the end of the Detail section so Access knows to start a new page before printing the next page header.

☰ Design→Controls→Insert Page Break

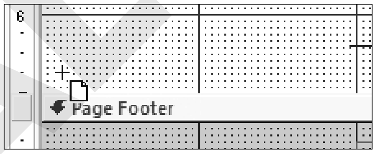
DEVELOP YOUR SKILLS: A4-D7

In this exercise, you will modify the Winchester Web Design Customer Invoice Report. You will add a title and the current date and set page breaks to print each invoice on a separate page.

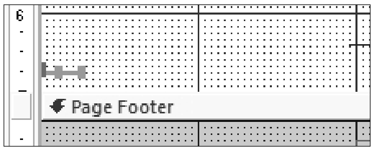
1. Display the **Customer Invoices** report in **Design View**.
2. Click the **Customer Invoice** title in the Page Header and then click again just in front of *Customer* to position the insertion point there.
3. Type `Winchester Web Design` and then press Shift+Enter to force *Customer Invoice* to a second line.
4. Click in an empty part of the Page Header and then click the title box to select it.
5. Set the Width property to `4"` and the Left property to `2"`.
6. Choose **Design→Header/Footer→Date and Time**.
7. Choose the **MM/DD/YYYY** date format, remove the check from the **Include Time** box, and then click **OK**.
8. Select the **date** control and press Ctrl+X to cut it from the Report Header.
9. Click the **Page Header** section bar and press Ctrl+V to paste the date into the Page Header.
10. Set the following properties for the date control:

Property	Setting
Width	**2"**
Height	**0.2"**
Top	**0.4"**
Left	**5.8"**

11. Scroll down to the bottom of the Detail section.
12. Choose **Design→Controls→Insert Page Break**.
13. Place the page break in the Detail section just above the Page Footer section bar.



Before



After

14. Switch to **Print Preview** and use the Navigation bar to browse the various pages.
15. Close Print Preview.
16. Choose **File→Close** to close the database and save the changes to the **Customer Invoices** report.

Self-Assessment

Check your knowledge of this chapter's key concepts and skills by completing the Self-Assessment. The answers to these questions can be found at the back of this book.

1. Reports can be imported from other databases. *True* *False*

2. Reports can be created from subreports. *True* *False*

3. The Can Grow property allows a subreport to expand to accommodate a varying number of records. *True* *False*

4. Page breaks are typically added to the Page Header section of a report. *True* *False*

5. Which property is used to link an imported report to a table or query in the destination database?

A. Control Source

B. Caption

C. Record Source

D. Name

6. Which of these can a subreport display data from?

A. Table

B. Query

C. Form

D. All of these options

7. Which of these is used in the Control Source property to automatically number items in a report?

A. =1

B. =Num

C. Control Source

D. -Num

8. Which property is used to created calculated controls in reports?

A. Name

B. Control Source

C. Caption

D. Pop Up

Self-Assessment Answer Key

CHAPTER 1: REFINING TABLE DESIGN

Item	Answer	Page Number
1	True	2
2	True	2
3	True	3
4	True	2
5	False	9
6	False	18
7	True	22
8	True	20
9	D	8
10	A	17
11	D	17, 20, 22
12	A	6
13	B	22

CHAPTER 2: CUSTOMIZING INPUT FORMS

Item	Answer	Page Number
1	True	30
2	True	30
3	False	32
4	False	32
5	True	35
6	True	35
7	True	36
8	False	30
9	True	30
10	True	35
11	D	35
12	A	36

CHAPTER 3: CREATING COMPLEX QUERIES

Item	Answer	Page Number
1	True	42
2	True	44
3	False	45
4	True	48
5	False	49
6	True	46
7	True	47
8	D	42
9	D	43
10	A	44
11	B	44
12	D	45
13	B	49
14	C	50

CHAPTER 4: CUSTOMIZING REPORTS

Item	Answer	Page Number
1	True	56
2	True	59
3	True	65
4	False	65
5	C	56
6	D	57
7	A	61
8	B	63

Index

NOTES

NOTES

NOTES

NOTES