Microsoft® Access 2019 & 365

LEVEL 2 OF 3

IAN EWELL
Davis Technical College

LABYRINTH LEARNING™

Microsoft Access 2019 & 365: Level 2

LABYRINTH LEARNING™

Labyrinth Learning
PO Box 2669
Danville, CA 94526
800.522.9746
On the web at lablearning.com

President:
Brian Favro

Product Manager:
Jason Favro

Development Manager:
Laura Popelka

Senior Editor:
Alexandra Mummery

Editor:
Alexandria Henderson

Developmental Editing:
Laura Popelka

Production Manager:
Debra Grose

Compositor:
Happenstance Type-O-Rama

Indexing:
BIM Creatives, LLC

Cover Design:
Sam Anderson Design

Interior Design:
Debra Grose

ITEM: 1-64061-163-0
ISBN-13: 978-1-64061-163-4

Manufactured in the United States of America

GPP 10 9 8 7 6 5 4 3 2 1

Contents

Access Chapter 8:
Customizing Reports

Preface

This textbook is part of our new approach to learning for introductory computer courses. We've kept the best elements of our proven, easy-to-use instructional design and added interactive elements and assessments that offer enormous potential to engage learners in a new way.

Why Did We Write This Content?

In today's digital world, knowing how to use the most common software applications is critical, and those who don't are left behind. Our goal is to simplify the entire learning experience and help every student develop the practical, real-world skills needed to be successful at work and in school. Using a combination of text, videos, interactive elements, and assessments, we begin with fundamental concepts and take learners through a systematic progression of exercises to achieve mastery.

What Key Themes Did We Follow?

We had conversations with dozens of educators at community colleges, vocational schools, and other learning environments in preparation for this textbook. We listened and adapted our learning solution to match the needs of a rapidly changing world, keeping the following common themes in mind:

Keep it about skills. Our content focus is on critical, job-ready topics and tasks, with a relentless focus on practical, real-world skills and common sense. We use step-by-step instructional design to ensure that learners stay engaged from the first chapter forward. We've retained our proven method of progressively moving learners through increasingly independent exercises to ensure mastery—an approach that has successfully developed skills for more than 25 years.

Keep it simple. Our integrated solutions create a seamless experience built on a dynamic instructional design that brings clarity to even the most challenging topics. We focus our content on the things that matter most and present it in the easiest way possible. Concise chunks of text are combined with visually engaging and interactive elements to increase understanding for all types of learners.

Keep it relevant. Fresh, original, and constantly evolving content helps educators keep pace with today's student and work environments. We reviewed every topic for relevancy and updated it where needed to offer realistic examples and projects for learners.

How Do I Use This Book?

Our comprehensive learning solution consists of a print textbook, a groundbreaking interactive ebook, and our easy-to-use eLab course management tool featuring additional learning content, such as overviews and video tutorials, and assessment content. Our interactive ebook contains learning content delivered in ways that will engage learners.

The eLab assessment solution includes Project Grader exercises for most chapters that are automatically graded by the system, in addition to clear feedback and analytics on student actions.

Included with Your Textbook Purchase

Depending on your purchase option, some or all of the following are included with your textbook:

Interactive ebook: A dynamic, engaging, and truly interactive textbook that includes elements such as videos, self-assessments, slide shows, GIFs, and other interactive features. Highlighting, taking notes, and searching for content is easy.

eLab Course Management System: A robust tool for accurate assessment, tracking of learner activity, and automated grading that includes a comprehensive set of instructor resources. eLab can be fully integrated with your LMS, making course management even easier.

Instructor resources: This course is also supported on the Labyrinth website with a comprehensive instructor support package that includes detailed lesson plans, lecture notes, PowerPoint presentations, a course syllabus, test banks, additional exercises, and more.

Student Resource Center: The exercise files that accompany this textbook can be found within eLab and in the Student Resource Center, which may be accessed from the ebook or online at: **labyrinthelab.com/office19**

We're excited to share this innovative, new approach with you, and we'd love you to share your experience with us at: lablearning.com/share

Visual Conventions

This book uses visual and typographic cues to guide students through the lessons. Some of these cues are described below:

Cue Name	What It Does
`Type this text`	Text you type at the keyboard is printed in this typeface.
Action words	The important action words in exercise steps are presented in boldface.
Ribbon	Glossary terms are highlighted with a yellow background.
Note! Tip! Warning!	Tips, notes, and warnings are called out with special icons.
	Videos, WebSims, and other ebook or online content are indicated by this icon.
Command→Command→ Command→Command	Commands to execute from the Ribbon are presented like this: Ribbon Tab→Command Group→Command→Subcommand.
Design→Themes→Themes	These notes present shortcut steps for executing certain tasks.

Display Settings

Multiple factors, including screen resolution, monitor size, and window size, can affect the appearance of the Microsoft Ribbon and its buttons. In this textbook, screen captures were taken at the native (recommended) screen resolutions in Office 2019 running Windows 10, with ClearType enabled.

Acknowledgements

Many individuals contribute to the development and completion of a textbook. We appreciate the careful attention and informed contributions of Jane Bauer of Northcentral Technical College, Deb Pheris Blencowe of Collin College, Marcio de Paula Wai of Martinez Adult School, Kimberly Duffey of North County Community College, Tracy Foreman of Huntington Beach Adult School, Lynne Kemp of North County Community College, Olivia Kerr of El Centro College, Theresa Loftis of San Bernardino Adult School, Nataliya O'Neil of North County Community College, Suzanne Wright of Francis Tuttle Technology Center, and Claudia Young of Ventura Adult School, for their assistance in the development of this book.

We are also deeply grateful to the instructors and professionals who reviewed the text and suggested improvements for this first edition.

This book has benefited significantly from the feedback and suggestions of the following reviewers:

Jessica Akers, *Tennessee College of Applied Technology*
Vickie Baldwin, *Piedmont Technical College*
Elaine Beam, *Lehigh Career & Technical Institute*
Dixie Becker, *Reading Area Community College*
Bentley Beckles, *Advantage Caribbean Institute*
John Bennett, *Denmark Technical College*
Gary Brown, *Western Technical College*
Brenda Bryant, *the gayle group*
Kristine Condon, *Kankakee Community College*
George Coss, *Macomb Community College*
Elaine Davis, *Carroll Community College*
Kevin Edwards, *CALC Institute of Technology*
Debra Farrelly, *El Paso Community College*
Bethany Flanagan, *Winston County Board of Education*
Taryn Fletcher, *Eastern Washington University*
Joyce Hill, *Mississippi Band of Choctaw Indians*
David Hoffmeier, *Gordon Cooper Technology Center*
Terri Holly, *Indian River State College*
Tony Hunnicutt, *College of the Ouachitas*
Judy Hurtt, *East Central Community College*
Dan Johnson, *Dr DJ's Tutoring*
Kyle Kelly, *Lake Michigan College*
Grace Laphan, *Rowan College at Burlington County*
Gabriele Lenga, *Truckee Meadows Community College*
Helane Littles, *Brooklyn Educational Opportunity Center*
Kim Mapes, *Lackawanna College Towanda Center*
Tamar Mosley, *Meridian Community College*
Nancy Nibley, *Simi Institute for Careers and Education*
Samuel Said, *Volunteer State Community College*
Alva Santiago, *Workforce Computer Training*
Arthur Schneider, *Portland Community College*
Albert Spencer, *North Salinas High School*
Pamela Taylor, *Los Angeles Unified School District/L. A. High School*
Karen Tuecke, *Northeast Iowa Community College*
Raji Visvanathan, *Freemont Union High School District - Adult School*
Jim West, *Computer Systems Institute*
Claudia Young, *Ventura Adult and Continuing Education*
Peter Young, *San Jose State University*

ACCESS

5 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. In this chapter, you will develop important database 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 structures
- Set field properties
- Use the Lookup Wizard

Project: Maintaining and Formatting Databases

Winchester Web Design is a website development company that specializes in building websites for small businesses. You are tasked with maintaining the company's 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 formatting to make the tables more colorful. You will then examine the relationships between tables to ensure they accurately define the database.

Creating and Modifying Relationships

As you build tables and other objects in a relational database, Access creates some of the relationships between tables based on each table's field structure. 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 in which a store pulls a product off its shelf and therefore needs to remove that product from its merchandise list, order list, inventory list, and advertising list. In most cases, you also must enforce referential integrity to ensure 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 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.

  Example: A main Customers table linked to a CustPassword table (one customer has one password)

- A **one-to-many relationship** means 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.

  Example: One employee will have many sales, and a product will be sold many times

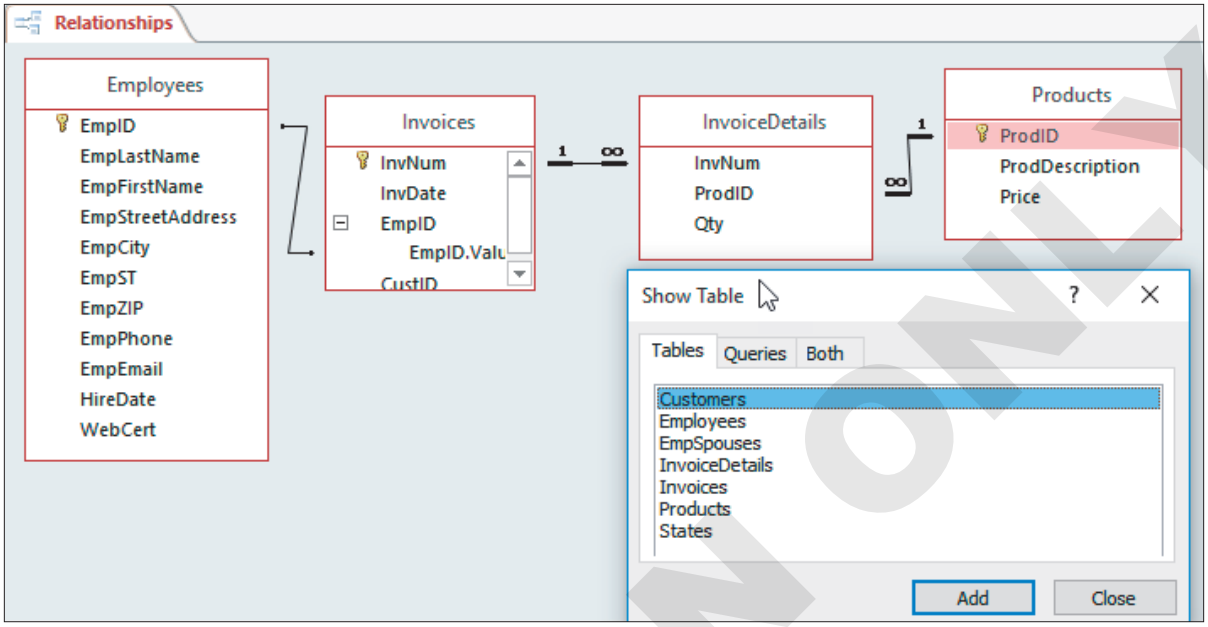
- A **many-to-many relationship** occurs when two tables may each have many matches in the other table but they do not share key fields, so they use a third junction table to connect the two tables, completing the relationship. The junction table has a one-to-many relationship to each table.

  Example: A Vendors table and a Products table, where one vendor provides many different products and one product is available from many vendors

Adding and Modifying Relationships

There are times when a database designer must add or edit a relationship. You can examine, create, edit, and delete relationships between tables in the Relationships window. The Show Table command in the Relationships window displays the Show Table window containing a list of tables and their fields. Tables are added to the Relationships window with a click of the Add button. Fields from one table

added to the Relationships window can be dragged and dropped onto a field in another table to begin creating a relationship between those fields. Conversely, relationships can be edited by selecting the join line between the relational fields and using the Edit Relationships command on the Ribbon.

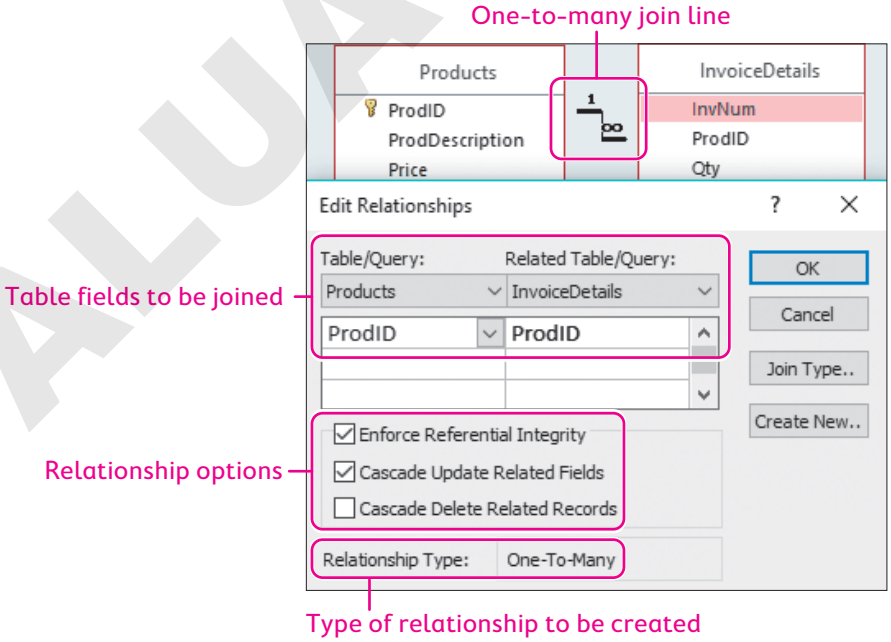


Database Tools→Relationships→Relationships

Relationship Tools→Design→Tools→Edit Relationships

The Edit Relationships Box

The Edit Relationships dialog box displays the tables and fields involved in the relationship along with options for establishing referential integrity, cascade protocols, and the relationship type being created.



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 views using real-life examples:

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

Relationship Cascade Options

Two additional relationship options are available for controlling 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.

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. Consider this option if you deleted an employee from the Employees table and want to also delete their spouse from the Spouses table. Take care, though, as any records associated with the deleted record are removed. So, if your Employees table is linked to your Invoices table, it would not be wise to delete all 2018 invoice records for an employee just because that employee retired in 2019.

View the video "Creating Relationships."

DEVELOP YOUR SKILLS: A5-D1

In this exercise, you will open the Relationships window, add a table, and create a relationship between tables with referential integrity enforced.

Before You Begin: *Download the student exercise files from your eLab course or the Student Resource Center (labyrinthelab.com/office19) and determine your file storage location before beginning this exercise.*

1. Open **A5-D1-WinDesign** from your **Access Chapter 5** folder and save it as: **A5-D1-WinDesignRev**

   When completing exercises, always choose to Enable Content.

2. Choose **Database Tools→Relationships→Relationships**.

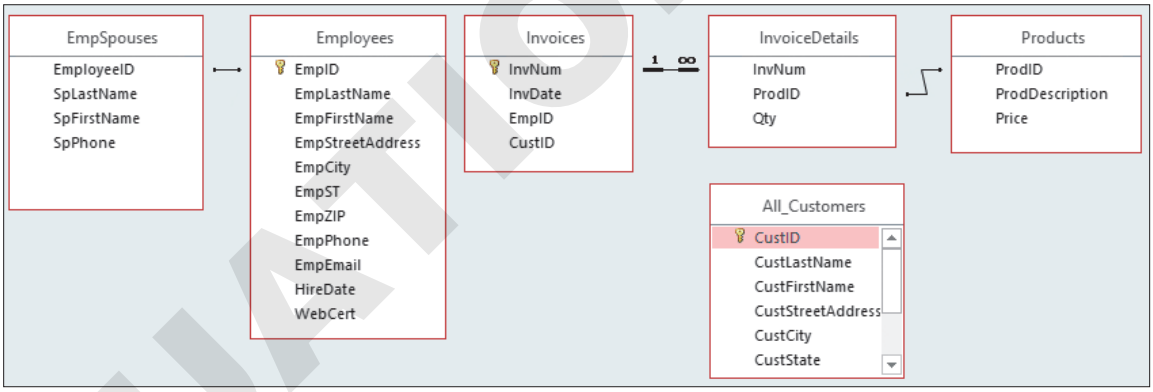
3. Choose **Relationship Tools→Design→Relationships→Show Table**.

4. Double-click the **EmpSpouses** and **All_Customers** tables to add them to the Relationships window and then close the Show Table box.

   Typically, all tables will be in the Relationships window, but sometimes tables are added later.

5. Rearrange the tables in the Relationships window by dragging their title bars so they are arranged as shown.

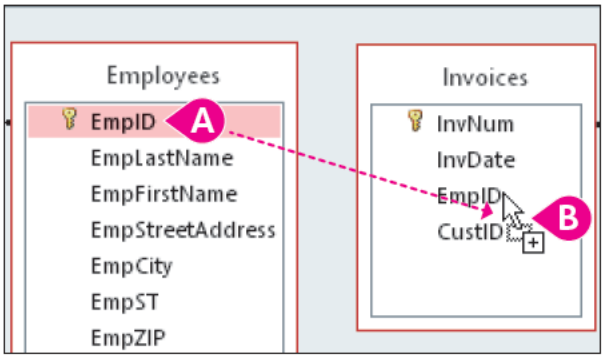
   Arranging the tables this way helps you see all the relationships, as the relationship lines are not overlapping.



Manually Set Relationships

Now you will create a one-to-many relationship between the Employees table and the Invoices table so one employee may have many invoices.

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



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

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

   The Edit Relationships dialog box opens once the mouse button is released.

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

   Access creates a join line and places symbols at each end. The Employees end of the join line displays a 1, and the Invoices end displays an infinity sign (∞).

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

   In the new relationship being created, the CustID field in the All_Customers table will be the primary key and the CustID field in the Invoices table will be the foreign key.

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.

Note! *Unless otherwise directed, keep Access and any databases or database objects being used open at the end of each exercise.*

Editing and Deleting Relationships

Deleting relationships and 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 removing a relationship or setting cascade options and then test the settings. That way you can restore the database from the backup if the changes you make 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.

Deleting Relationships

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 warn you 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.

DEVELOP YOUR SKILLS: A5-D2

In this exercise, you will edit the relationship between the Employees and Invoices tables so 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**.

   The mouse pointer must be right on the join line or you won't see the Edit Relationships option on the right-click menu.
2. Check the **Cascade Update Related Fields** checkbox and click **OK**.
3. Right-click the join line linking the **ProdID** fields in the Products and InvoiceDetails tables.

   Your mouse pointer must be directly on the line for the menu to appear.
4. Choose **Delete** and then click **Yes** to confirm the deletion.
5. Choose **File→Save** or click the **Save** button to save the relationship changes.

Documenting and Printing Relationships

After you have inspected the relationships, you may want to create a report to view a printable version of them. 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

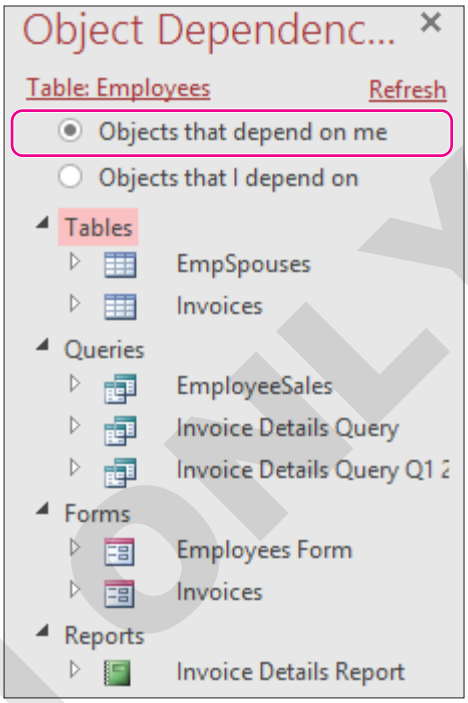
DEVELOP YOUR SKILLS: A5-D3

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

1. Choose **Relationship Tools→Design→Tools→Relationship Report**.
2. Choose **Print Preview→Page Layout→Landscape** so all tables and relationships are shown.
3. Save the report as: **`Relationships`**

   The report is added to the Report group in the Navigation pane.
4. Close the report and then close the Relationships window.

Display Object Dependencies



5. Select the **Employees** table in the Navigation pane without opening it.
6. Choose **Database Tools→Relationships→Object Dependencies**.
7. Ensure the **Objects That Depend on Me** option is selected in the Object Dependencies panel.

   The objects listed below each type are dependent on the Employees table.
8. Choose the **Objects That I Depend On** option.

   The Employees table has relationships with, or depends on, the EmpSpouses, Invoices, and States tables.
9. Close the Object Dependencies panel.

Modifying Table Structures

Database integrity and data validity are important aspects of database maintenance. Access features that enable you to modify table fields, control the data entered, and format the data to ensure consistent reporting 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 Adding, Deleting, and Editing 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 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. Take care when adding, deleting, or editing 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 change the name later without affecting its data, but note that table names are often included in other database objects that use the table's data, which means renaming a table can impact other database objects. Access should automatically update the name of the table used in other database objects. However, if you rename a table, you should make sure that every form, query, or report that uses that table still functions the same.

A good database design principle is to name all your tables in a similar fashion, for example, using the underscore between words in all tables named with more than one word (Invoice_Details). Traditionally, spaces are *not* used in table names because referencing the table in a query or entering the table name in an expression could be confusing. For instance, is *Invoice Details* two objects or one object? Fortunately, Access will enclose a table name like *Invoice Details* in [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 warning 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 Undo. 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 a zero), Access will warn you that you are about to delete data that did not conform, thus the entry made using the capital *O* will be removed after the change is confirmed. The great thing about this is that Access will allow only valid field data, which results in more accurate data.

The Yes/No Data Type

The Yes/No data type sets the field so 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: A5-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 field and any data within it have been permanently removed from the database. However, before continuing, you could use Undo to restore the field and data because you made the deletion while working in Design View.

Add New Fields

Now you will add a Yes/No field to the Customers table.

6. Right-click the **CustStreetAddress** field selection box and choose **Insert Rows**.

   A new row opens above the CustStreetAddress row.

7. Click in the empty **Field Name** box of your new field and type: **`Business`**
8. Tap Tab and set the data type to **Yes/No**.
9. Save the table and then switch to **Datasheet View**.

   Your new Yes/No Business field has all boxes unchecked, which means it is set to No.

10. Check the **Business** field boxes for the records with these CustIDs: **DavisP**, **HassanA**, and **KleinJ**.

    These records are now identified as having a business.

11. Close the Customers table.

Formatting a Table Datasheet Layout

If field values are longer than anticipated, Access displays only the portion of the data that fits within the column width, hiding some of the data. 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 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, but the data remains in the table—it's not deleted. If you want to view data in hidden columns later, unhide the column.

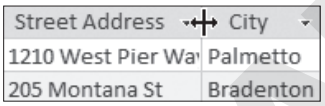
Saving a Table Layout

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.

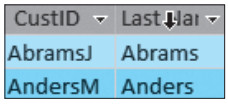
DEVELOP YOUR SKILLS: A5-D5

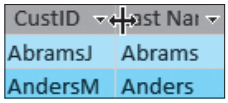
In this exercise, you will adjust the column width to allow for the best display of data in a datasheet. You will also rearrange columns and hide a column.

1. Display the **Customers** table in **Datasheet View**.
2. Position the mouse pointer over the border between the Street Address and City column headings so the adjust pointer appears as a double-headed arrow.



3. Double-click and notice that the Street Address column width adjusts to fit the widest entry in the column.
4. Position the mouse pointer on the **CustID** column heading and then drag right to select both the **CustID** and **Last Name** columns.
5. Position the mouse pointer between the column headings and double-click when the adjust pointer appears.





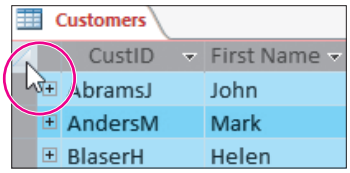
Both columns will be Best Fit.

6. Click the **First Name** column heading to select the column. Do not release your mouse button.
7. Drag the **First Name** column to the left of the Last Name column, releasing the mouse button when the black vertical bar is between the CustID and Last Name columns.

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

The First Name column should now be before the Last Name column.

8. Click the table selection button to select all data.

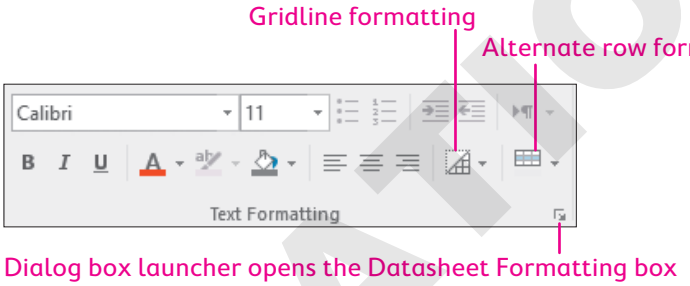


9. Double-click the border between any of the selected column headings to Best Fit all columns.
10. Save the layout changes.

Enhancing a Datasheet

Changing the datasheet layout enables you to make necessary adjustments, such as widening a field so 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. It also contains tools for formatting alternate rows and gridlines. Finally, you can find more formatting options in the Datasheet Formatting dialog box (launched from the Text Formatting group).



Home→Text Formatting

DEVELOP YOUR SKILLS: A5-D6

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

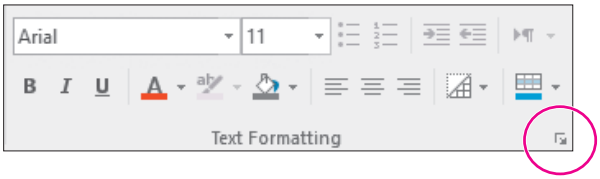
1. Choose **Home→Text Formatting→Alternate Row Color menu button** .
2. Choose **Blue, Accent 1** (the fifth color on the top row under Theme Colors).

   Alternate rows in the datasheet should now be blue.
3. Choose **Home→Text Formatting→Font menu button** and choose **Arial** from the list.

   All table data should now be formatted with the Arial font.
4. Choose **Home→Text Formatting→Gridlines menu button** and choose **Gridlines: Horizontal**.

   The datasheet will display only horizontal gridlines now.

5. Click the dialog box launcher at **Home→Text Formatting** to open the Datasheet Formatting dialog box.



6. Click the **Gridline Color menu** button ▾ and choose **Black, Text 1** (second color on the top row of Theme Colors).
7. Click **OK** to apply the black gridlines.
8. Save the changes to the Customers table.

Setting Field Properties

Use field property settings 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.

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 and 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 using varying formats. 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 warns 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.

NUMBER FIELD FORMATS

Field Size Property	Description
Byte	Stores whole numbers between 0 and 255 using 1 byte and allows no fractions or decimal points; uses the minimal 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.

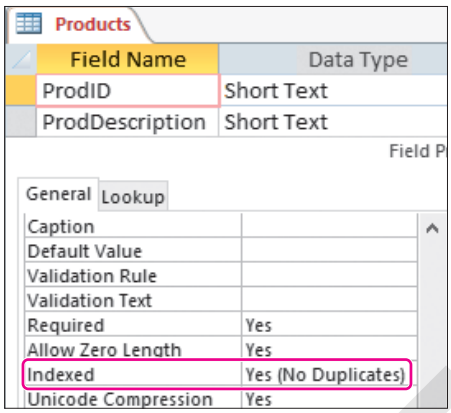
Setting Default Field Values

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.



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: A5-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 these field properties:
   - Field Size: **2**
   - Format: **>**
   - Caption: **ST**
   - Default Value: **FL**

   Entering > for the Format property converts entries to uppercase. Access places quotation marks around the Default Value field property when you click in another field property or save the table. Setting field sizes consistent with the data they hold helps the data display properly when it's included in forms and reports.
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**. For both the Some Data May Be Lost message and the Data Integrity Rules Have Been Changed message, choose **Yes**.
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 already shows* FL *because 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.

   You are prompted to enter a value for the Customers.CustLastName field because you made it a required field.
8. Click **OK**, select the **Last Name** field, and type: **Jones**
9. Tap Tab through the remaining record fields.

   You are prompted to enter a value for the Customers.CustPhone field, since it is also a required field.
10. Click **OK** and then tap the left arrow ← to select the **Telephone** field; type **9415553232** and tap Tab.

    Access automatically applies parentheses, ( ), to the area code and a hyphen, -, after 555.

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 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	This allows data entry of zero length in a field. Data is entered as open and close parentheses with no character between: (). This entry shows 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 landline, enter () in the field.
Text Format	This property sets the text in a Long Text field as Plain Text or Rich Text. Rich text fields can be formatted with different font types, sizes, and colors.
Text Align	This 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	This 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 at the right side of a field property to open the Wizard associated with that property and choose the desired settings. For example, click the Build button to open the Input Mask Wizard to format the display of text and field dates.

DEVELOP YOUR SKILLS: A5-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.

   Using the @ symbol will 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.

   These settings will require a CustFirstName to always be entered.

4. Choose **File→Save** 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.

   A message informs you that you must enter a value in the CustFirstName field. This is because you set the Allow Zero Length property to No, which requires an entry of at least one character to be made.

8. Click **OK** to dismiss the message and then click **Yes** to close the database object (Customers table) now.

   The table will close and the new record you started to enter will not be saved.

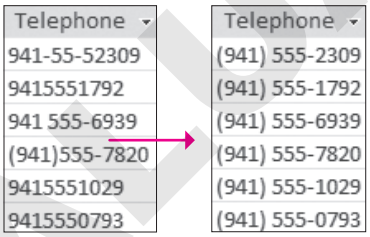
Formatting Data Using Input Masks

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.



Input masks can automatically format unformatted data.

The Build ▪▪▪ button that appears at the right end of the Input Mask box when you click the box 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 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 (letter or number) or a space	&&&& permits data entry such as a four-character ID along the lines of 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 data.
<	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 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 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 (*)	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 data consistency throughout the database.

DEVELOP YOUR SKILLS: A5-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**.

   The ProdID field consists of two numeric characters and two alphabetic characters.

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 and not the letter O for 00).

   This input mask formats the ProdID field to automatically begin with PROD- *followed by two numbers and then two letters.*

4. Choose **File→Save** 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**.

   Access displays the Update Properties dialog box, which contains a list of all objects using the field. For this field, only one object is listed.

6. Click **Yes** to update the Products Report object and then switch to **Datasheet View**.

   The ProdID field now has PROD- *preceding each product ID.*

7. Close the Products table.

Use the Input Mask Wizard

Now you will apply a standard input mask format to a field.

8. Open the **Employees** table in **Datasheet View**.

   The Telephone field contains numbers without any other symbols.

9. Switch to **Design View** and click anywhere in the **EmpPhone** field.
10. Click in the **Input Mask** field property box and then click the **Build** button on the right side of the box.
11. Choose **Phone Number** as the input mask and click **Finish**.

12. Save the table and then switch to **Datasheet View**.

Notice the phone numbers are now formatted with parentheses and a hyphen because of the input mask.

13. Switch back to **Design View**.

Create Additional Settings

14. Change field properties to set the field sizes, input masks, and captions as indicated.

For the EmpLastName and EmpFirstName input masks, you're using 24 question marks; you're using 14 question marks for the EmpCity input mask. These input masks use >L< to specify that the first character should automatically be capitalized and the next 24 or 14 question marks specify that up to 24 or 14 alphabetic characters may be entered.

Field	Field Size	Input Mask	Caption
EmpID	3	>LLL	ID
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

15. Choose **File→Save** and choose **Yes** when advised that some data may be lost.

16. Switch to **Datasheet View**.

The column headings now show captions rather than field names. The Caption property lets you use descriptive headings for fields, as field names cannot contain spaces.

17. Enter these records, using Tab to move between fields and only lowercase letters in the ID, Last Name, and First Name fields:

Field Name	Field Data
ID	fdr
Last Name	roberts
First name	fred
Street Address	362 lakeview ave.
City	sarasota
State	FL
Zip	34234
Phone	9415553981
Email	fredroberts@email.com
Hire Date	9/26/2018
Web Cert	**Yes** (check)

Notice as you enter the data into the fields that some letters are automatically converted to uppercase due to the input mask rules you set.

18. Close the Employees table.

Setting Validation Rules

A validation rule is a field property that enables you to limit the values entered in the field to reduce inaccurate data entry. You could, for example, set a validation rule to limit the value typed into an HoursWorked field to 50 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, the field for which you are setting the rule must be formatted appropriately for the data type to 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's 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 whether 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 four-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: A5-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** and choose **Yes** to acknowledge that the date integrity rules have been changed.

Test the New Validation Rules

6. Switch to **Datasheet View**.

   All ProdID entries begin with PROD- *because of the input mask you set earlier.*

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.

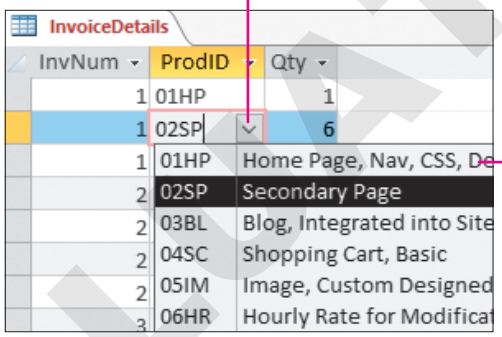
   The warning appears because of the validation rule you just set for the Price field. Notice the All prices must be at least $25 *validation message you set is displayed in the alert.*

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.



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. When a lookup field is created using values in another table, a relationship is also created between the lookup field and the field that contains the values.

Benefits of Lookup Tables

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

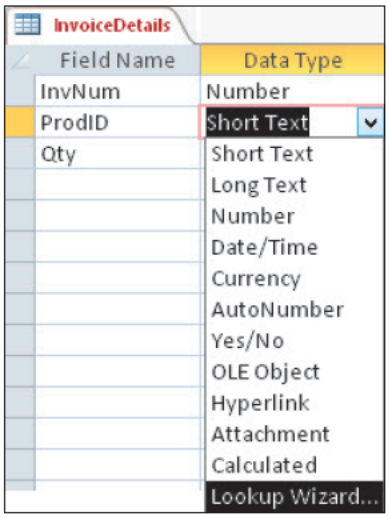
- It reduces the time required to enter the data repeatedly.
- It reduces errors associated with data entry.
- It restricts data to valid entries.

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

Performing a Lookup

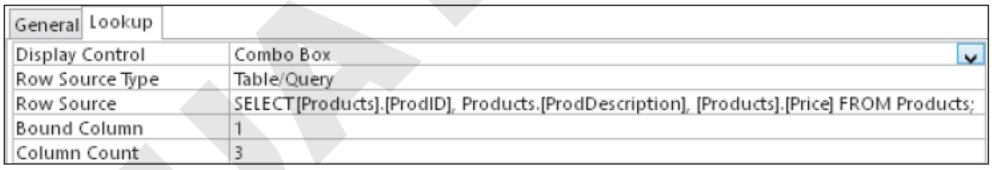
Access provides the following two ways to use the Lookup feature:

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



View the video "Using the Lookup Wizard."

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



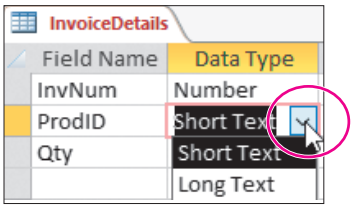
DEVELOP YOUR SKILLS: A5-D11

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 all tables are closed (you cannot make relationship changes to an open table).
2. Choose **Database Tools→Relationships→Relationships**.
3. Right-click the join line linking the **EmpID** fields in the Employees and Invoices tables.

   Your mouse pointer must be directly on the line for the menu to appear.
4. Choose **Delete** and then click **Yes** to confirm.
5. Using the same procedure, delete the relationship between **EmpID** in the Employees table and **EmployeeID** in the EmpSpouses table.
6. Close the Relationships window.
7. Display the **InvoiceDetails** table in **Design View**.

8. Click in the ProdID **Data Type** box and then click the drop-down **menu** button ▾.



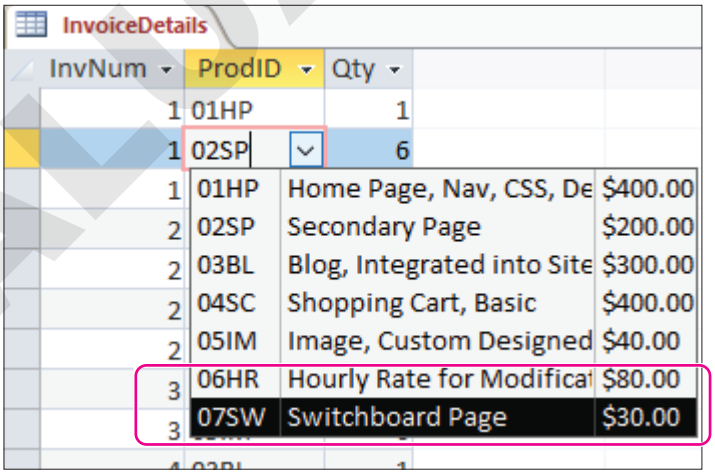
9. Choose **Lookup Wizard** from the menu.
10. Click **Next** to accept the current setting, *I Want the Lookup Field to Get the Values from Another Table or Query*.
11. Choose **Table: Products** in the next Wizard screen and then click **Next**.

    The Products table will contain the values to be looked up.
12. Move all three available 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**.

    The next Wizard screen lets you adjust the width of the lookup field columns, which determines how the columns appear when the lookup field is used.
14. Uncheck the **Hide Key Column** checkbox and then double-click the right borders of all three column headings to best fit the columns; click **Next**.
15. Click **Next** again to choose ProdID as the field that uniquely identifies the row.
16. In the next screen, leave the label set to ProdID, check the **Enable Data Integrity** box, and choose the **Cascade Delete** option.
17. Click **Finish** and then choose **Yes** in the two warning boxes.

Test the Lookup Field

18. Switch to **Datasheet View**.
19. Click in the **ProdID** field for the 02SP record (second record in the table).
20. Click the drop-down **menu** button ▾ and choose the **07SW** product.



This product ID replaces the 02SP *for that record, so invoice 1 still has three products listed on it, but one of them has changed. This lookup field makes it easy for you to view a listing of products when adding them to invoices.*

21. Close the InvoiceDetails table.

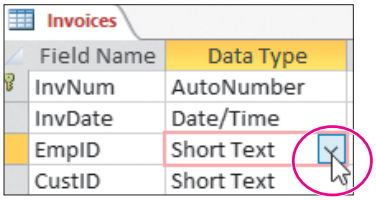
Creating Lookup Fields for Multiple Values

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: A5-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 EmpID **Data Type** box, click the **menu** button ▾, and choose **Lookup Wizard**.



3. Click **Next** to accept the current setting, *I Want the Lookup Field to Get the Values from Another Table or Query*.

Tip! *If you choose the* I Will Type the Values in That I Want *option, you can create your own list of items.*

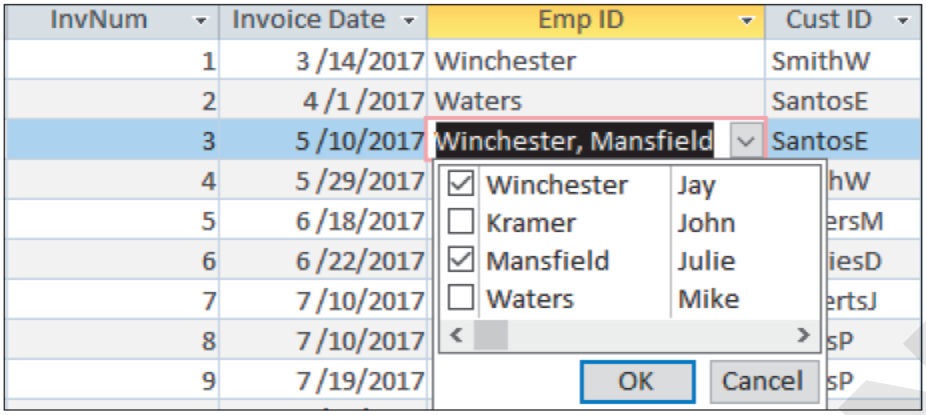
4. Choose **Table: Employees** and click **Next**.

   The Employees table will contain the values to be looked up.
5. Move the **EmpID**, **EmpLastName**, and **EmpFirstName** fields to the Selected Fields list and click **Next**.
6. Choose **EmpID** as the sort field, leave the sort order as **Ascending**, and click **Next**.
7. Click **Next** to accept the default width settings for the columns.
8. In the final Wizard screen, check the **Allow Multiple Values** box, leave the label set to **EmpID**, and click **Finish**.
9. Choose **Yes** in the message box to confirm you want to store multiple values.
10. Choose **Yes** to save the table and choose **Yes** one last time to confirm that some data may be lost.

Test the Multiple Values Lookup Field

11. Switch to **Datasheet View**.
12. Click in the **Emp ID** field for the third record (invoice 3).

13. Click the drop-down **menu** button ▾ to display the list of employees.



14. Check the box for **Winchester** (leave Mansfield checked) and click **OK**.

15. Double-click the border between the Emp ID and Cust ID columns to best fit the Emp ID column.

    Invoice 3 now has two employees assigned to it. Notice that both employee names are visible in the Emp ID field.

16. Close the database, saving the changes to the Invoices table.

Self-Assessment

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

Reinforce Your Skills

REINFORCE YOUR SKILLS: A5-R1

Create Relationships and Modify Table Structure

The president of Kids for Change, an organization that encourages young people to participate in community-based projects, has asked you to modify its database. In this exercise, you will create a relationship between tables and create a relationship report. You will rename a table and add a new field, delete and modify fields, and enhance the datasheet with color.

1. Open **A5-R1-K4C** from your **Access Chapter 5** folder and save it as: **`A5-R1-K4CRev`**
2. Choose **Database Tools→Relationships→Relationships** and then click **Show Table**.
3. Add the **Volunteers** table to the Relationships window and then close the Show Table box.
4. Expand the **Volunteers** field list by dragging down on the bottom edge of the list until all fields are shown.
5. Drag the **ActID** field from the Activities table and drop it on the **ActID** field in the Volunteers table.
6. Check the **Enforce Referential Integrity** checkbox and click **Create** to establish the relationship.

   The join line has a 1 at the Activities table end and an infinity symbol (∞) at the Volunteers table end.
7. Choose **Relationship Tools→Design→Tools→Relationship Report**.
8. Close the report, saving it as **`Relationships`** and then closing the **Relationships** window, saving the changes if prompted.

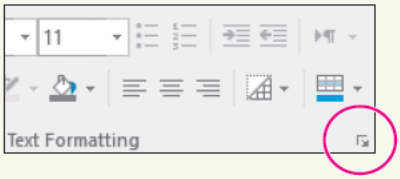
Modify the Table Structure

9. Right-click the **Staff** table and choose **Rename**; then type **`PaidStaff`** and tap Enter.
10. Display the **PaidStaff** table in **Design View**.
11. Right-click the row selector for the StaffStreet field and choose **Insert Rows**.
12. Click in the new **Field Name** box and then type **`Parent`** and tap Tab.
13. Set the Data Type of the new field to **Yes/No** and enter **`Parent of K4C child`** in the Description box.
14. Switch to **Datasheet View** and save the table.
15. Check the Parent **Yes/No** boxes for parents with the last names of **Lockwood**, **Kendall**, and **Riggs**.
16. Close the PaidStaff table and then open the **Donations** table in **Design View**.
17. Click the row selector for the Acknowledgement field and tap Delete. Choose **Yes** to confirm and then close the Donations table, saving the changes.

Format a Table Datasheet Layout

18. Display the **Children** table in **Datasheet View**.
19. Click the **First Name** column heading and then hover the mouse pointer over the heading until the white pointer appears.

20. Drag the **First Name** column to the left of the Last Name column.
21. Drag the mouse pointer over the **Address**, **City**, **ST**, and **ZIP** columns to select those columns.
22. Right-click one of the selected column headings and choose **Hide Fields**.
23. Double-click the right edges of the Mother and Father column headings to best fit those columns.
24. Choose **Home→Text Formatting→Alternate Row Color menu button** and choose any light alternate row color.
25. Choose **Home→Text Formatting→Gridlines menu button** and choose **Gridlines: Horizontal**.
26. Click the **Text Formatting** dialog box launcher to open the Datasheet Formatting dialog box.



27. Choose these settings:
    - Use the Gridline Color list to choose a dark gridline color to create a nice contrast with the light alternate row color you chose.
    - In the Border and Line Styles section, choose **Column Header Underline** as the border style.
    - Also in the Border and Line Styles section, choose **Dots** as the line style.
28. Click **OK** to apply the formats and then close the database, saving the changes.

REINFORCE YOUR SKILLS: A5-R2

Set Field Properties and Create Lookup Fields

As head of Tech Development for Kids for Change, you want to set some database field properties. In this exercise, you will set field size, convert values to uppercase, set captions and default values, make a field required, and create a custom format for an ID field. You will set a predefined telephone input mask, add a validation rule, and set a lookup field.

1. Open **A5-R2-K4C** from your **Access Chapter 5** folder and save it as: **A5-R2-K4CRev**
2. Display the **Volunteers** table in **Design View**.
3. Use this table to set the Field Size and Caption properties:

Field Name	Field Size	Caption
VolID	12	
VolLastName	25	Last Name
VolFirstName	25	First Name
VolStreet	25	Street
VolCity	25	City
VolST	2	ST
VolZIP	5	ZIP Code
VolPhone	15	Telephone
ActID	6	Act/Day

4. Set the **Required** property of the **ActID** field to **Yes**.
5. Select the **VolST** field and set the **Format** property to **>** and the **Default Value** to: **FL**

   Using > as the Format property will convert lowercase values to uppercase values.
6. Switch to **Datasheet View** and choose **Yes** to save the table; click **Yes** twice to acknowledge the two messages that appear.
7. Enter this new record:

Field Name	Value
VolID	10
LastName	Graves
FirstName	Matthew
Street	915 Beneva St
City	Sarasota
ST	FL
ZIPCode	34232
Telephone	9415556198
Act/Day	BCSat

8. Close the Volunteers table.

Set an Input Mask and Validation Rules

9. Display the **Activities** table in **Design View**.
10. Set the Input Mask property of the ActID field to: **"K4C-">LLL<LL**

    This input mask starts each ActID with the literal value K4C- followed by three uppercase (>) letters and two lowercase (<) letters; for example, K4C-DWTue for dog walking on Tuesday.
11. Switch to **Datasheet View**, saving the changes to the table.
12. Scroll to the end of the table and click in the first empty **Activity ID** field.

    Access automatically places the new prefix in the field because of the input mask you just created.
13. Close the Activities table and then display the **Volunteers** table in **Datasheet View**.

    Notice the Telephone field displays numbers with no formatting.
14. Switch to **Design View**, click anywhere in the **VolPhone** field, and then click in the **Input Mask** property box.
15. Click the **Build** ... button to start the Input Mask Wizard.
16. Choose the **Phone Number** mask and click **Finish**.
17. Switch to **Datasheet View**, saving the changes to the table.

    Notice the Telephone field is now formatted with the input mask characters.
18. Close the Volunteers table and then display the **Children** table in **Design View**.
19. Set the Validation Rule property of the BirthDate field to: **>01/01/2008**
20. Enter **Only children born after January 1, 2008, may enroll** in the Validation Text property.

21. Switch to **Datasheet View**, choose **Yes** to save the table, and then choose **Yes** again when the data integrity message appears.
22. Enter this information into a new record:

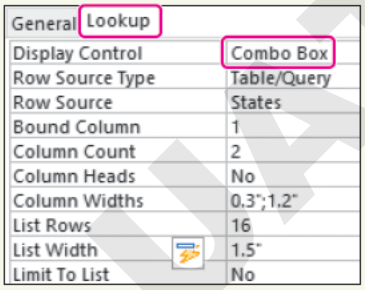
Field Name	Value
ChildID	**CasadoM**
LastName	**Casado**
FirstName	**Marty**
Address	**302 Waterside Ave**
City	**Bradenton**
ST	**FL**
ZIP	**34202**
Telephone	**9415551652**
BirthDate	**11/24/2007**

The input mask will not allow you to enter the birth date because it is prior to 01/01/2008.

23. Click **OK** and then change the Birth Date value to: **11/24/2008**
24. Complete the record by entering **Sandy** for Mother, **Javier** for Father, and **9415551653** for Emergency.

Set a Field as a Lookup Field

25. Switch to **Design View**.
26. Click anywhere in the **ChildST** field and then click the **Lookup** tab in the Field Properties box and choose **Combo Box** as the Display Control.



27. Enter these lookup properties (also shown in the preceding image):
    - Row Source: **States**
    - Column Count: **2**
    - Column Widths: **0.3**; **1.2**
    - List Width: **1.5**
28. Switch to **Datasheet View** and save the changes to the table.
29. Click in the **ST** field for *DriverJ*, which is missing the state.
30. Open the combo box of lookup values and choose **FL**.
31. Close the database, saving the changes.

REINFORCE YOUR SKILLS: A5-R3

Set Relationships and Add a Lookup Field

The Kids for Change database is performing better, but you want to improve table appearance and facilitate data entry and validation. In this exercise, you will add a field to indicate whether a staffer has a master's degree, delete an unneeded table, and rearrange fields. You will hide a field, resize columns to better display data, create a custom input mask with data validation, and apply a predefined input mask.

1. Open **A5-R3-K4C** from your **Access Chapter 5** folder and save it as: **`A5-R3-K4CRev`**
2. Choose **Database Tools→Relationships→Relationships**  .
3. Establish a relationship by dragging the **DonorID** field from the Donors table to the **DonorID** field in the Donations table.
4. Check the **Enforce Referential Integrity** checkbox and click **Create**.
5. Choose **Relationship Tools→Design→Tools→Relationship Report**  .
6. Close the report, saving it as **`Relationships`** and then closing the Relationships window.

Modify the Table Structure

7. Display the **PaidStaff** table in **Design View**.
8. Right-click the **ActID** row selector, choose **Insert Rows**, and type **`Masters`** for the new field name.
9. Set the Data Type to **Yes/No** and enter **`Master's degree or higher`** as the Description.
10. Switch to **Datasheet View** and save the table.
11. Check the **Master's** checkboxes for records with the last names of **Bryant**, **Lockwood**, and **Riggs**.
12. Switch to **Design View**.
13. Click the field selector box for the **2ndDay** field and tap Delete to remove it. Choose **Yes** to confirm the deletion.
14. Switch to **Datasheet View** and save the table.

Format a Table Datasheet Layout

15. Move the **Email Address** column to the left of the Master's column.
16. If necessary, scroll to the right until the HrlySal column is visible. Right-click the **HrlySal** column heading and choose **Hide Fields**.

    The hourly salary field is no longer displayed, but the data remains in the table.
17. If necessary, scroll to the left and then double-click the right edge of the **Email Address** column heading to Best Fit the column.

    All email addresses should now be visible.
18. Choose **Home→Text Formatting→Alternate Row Color**  **menu button** ▾ and choose any light alternate row color.
19. Choose **Home→Text Formatting→Gridlines**  **menu button** ▾ and choose **Gridlines: Horizontal**.
20. Open the Datasheet Formatting dialog box via the dialog box launcher in the Text Formatting group.

21. Open the **Gridline Color** menu, choose a dark color to complement the light row color, and click **OK**.
22. Close the PaidStaff table, saving the changes.

Set Field Properties

23. Display the **Donors** table in **Design View**.
24. Set these field sizes and captions:

Field Name	Field Size	Caption
DonorLName	25	Last Name
DonorFName	25	First Name
DonorStreet	25	Street
DonorCity	25	City
DonorST	2	ST
DonorZIP	5	ZIP
DonorPhone	15	Telephone
DonorEmail	(No change)	Email Address

25. Set the Required property of the DonorLName field to **Yes** and set the Allow Zero Length property to **No**.
26. Set the Format property of the DonorST field to **>** and the Default Value property to: **FL**
27. Switch to **Datasheet View**, saving the changes to the table, choosing **Yes** in the warning box, and choosing **Yes** again in the second warning box.
28. Enter this record, closing the Donors table when finished:

Field Name	Value
DonorID	(Automatically set)
LastName	Castro
FirstName	Lana
Street	4040 Conquistador Way
City	Bradenton
ST	FL
ZIP	34212
Telephone	9415556626
EmailAddress	MCastro@email.com
Acknowledgement	For Mina

Set Formats, Input Masks, and Allow Zero Length

29. Display the **Donations** table in **Design View**.
30. Set these field properties in the DonationType field:
    - Input Mask: **>L<LL**
    - Validation Rule: **Bus Or Pvt**
    - Validation Text: **Must be Bus or Pvt**

31. Switch to **Datasheet View**, saving the changes to the table and choosing **Yes** in both warning boxes.
32. Add this record to test the new input mask:

Field Name	Value
DonorID	`12`
DonationDate	`04152013`
Amount	`1000`
DonationType	`Pmt`

33. Tap Tab after entering the donation type.

    The validation text message appears because Pmt is not an allowable DonationType entry. This is because the validation rule you set allows only entries of Bus or Pvt.
34. Click **OK** and then type `Pvt` in the DonationType field.
35. Close the Donations table and then display the **Donors** table in **Design View**.
36. Set the Format property of the Acknowledgement field to: `@; No Comments`

    This will display the phrase No Comments *when the Acknowledgement field is left blank.*
37. Click in the **DonorPhone** field and then click in the **Input Mask** property box.
38. Click the **Build** ... button to start the Input Mask Wizard, saving the table when prompted.
39. Choose the **Phone Number** mask and click **Finish**.
40. Switch to **Datasheet View**, saving the changes to the table.

    Notice that all telephone numbers now have proper and consistent formatting. Also notice that the phrase No Comments *appears in all records in which no Acknowledgement was entered.*
41. Close the Donors table.

Set a Field as a Lookup Field

42. Display the **PaidStaff** table in **Design View**.
43. Click anywhere in the **ActID** field and then click the **Lookup** tab in the Field Properties box and choose **Combo Box** as the Display Control.
44. Enter these Lookup properties:
    - Row Source: `Activities`
    - Column Count: `2`
    - Column Widths: `0.5`; `1.5`
    - List Width: `2`
45. Switch to **Datasheet View** and save the changes to the table.
46. Right-click the **Email Address** column heading and choose **Unhide Fields**.
47. Check the **HrlySal** box and then close the Unhide Columns box.

    The hourly salary field, which had been hidden, reappears.

48. Test the new lookup field by adding this record:

Field Name	Value
StaffID	(Automatically entered)
LastName	**Francesco**
FirstName	**Dominic**
Parent	Yes (checked)
StreetAddress	**105 26th Street**
City	**Sarasota**
ST	**FL**
ZIP	**34209**
Telephone	**9415558287**
Masters	Yes (checked)

49. Type **E** in the Activity field.

Access displays the first value that begins with the letter E: EBSun.

50. Tap Tab to accept the EBSun entry.

51. Type **27** in the Hrly Sal field and **DomFrancesco@email.com** in the Email Address field.

52. Close the database, saving changes to any tables if prompted.

Apply Your Skills

APPLY YOUR SKILLS: A5-A1

Create Relationships and Modify Table Structure

The CEO of Universal Corporate Events has asked you to make some changes to the UCE database. In this exercise, you will create a relationship between two tables, set referential integrity, and create a relationship report. You will rename and add a Yes/No field to a table as well as delete, modify, and rearrange fields. Finally, you will improve the appearance of the Menus table.

1. Open **A5-A1-UCE** from your **Access Chapter 5** folder and save it as: **`A5-A1-UCERev`**
2. Establish a relationship between the **VenueID** fields in the **Venues** and **Schedules** tables, enforcing referential integrity.
3. Create a relationships report named: **`Relationships`**
4. Close the report and the Relationships window.

Modify a Table's Structure and Column Display

5. Change the name of the Contacts table to: **`VenueLiaisons`**
6. Insert a new field in the Venues table above the VenueWebSite field with these settings:

Field Name	Data Type	Description
`Kitchen`	`Yes/No`	`Does venue have a kitchen?`

7. Switch to **Datasheet View**, saving the changes, and check the **Yes/No** Kitchen boxes for **HyattS**, **ManYC**, and **SaraCC**, **SaraYC**, and **TmpCon**.
8. Close the Venues table and then open the **Schedules** table in **Design View**.
9. Delete the **VenueName** field and then close the Schedules table, saving the changes.
10. Display the **Personnel** table in **Datasheet View**.
11. Move the **Date Hired** field to the left of the Last Name column.
12. Hide the Date of Birth field so it is no longer displayed.
13. Apply **Best Fit** to the **Address** and **Email Address** columns to make the widest entries in the columns fully visible.
14. Close the Personnel table, saving the changes.

Change the Formatting of a Datasheet

15. Display the **Menus** table in **Datasheet View**.
16. Apply a light alternate row color and horizontal gridlines and change the gridline color to **Black**.
17. Close the database, saving the changes to the Menus table.

APPLY YOUR SKILLS: A5-A2

Set Field Properties, Apply Input Masks, and Create Lookup Fields

In this exercise, you will modify field properties in the UCE database tables to more accurately describe and limit the data. You will set field sizes and captions, convert field values, set a default value for easier data entry, and make a field required. You will also set a custom format for an ID field, apply a predefined input mask to a telephone field, add a validation rule, and set lookup fields.

1. Open **A5-A2-UCE** from your **Access Chapter 5** folder and save it as: **A5-A2-UCERev**
2. Display the **VenueLiaisons** table in **Design View** and set these field sizes and captions:

Field Name	Field Size	Caption
LiaisonID	12	
LiaisonLName	25	Last Name
LiaisonFName	25	First Name
LiaisonStreet	25	Street
LiaisonCity	25	City
LiaisonState	2	State
LiaisonZIP	5	ZIP Code
LiaisonPhone	15	Telephone
LiaisonEmail		Email Address

3. For the LiaisonPhone field, set the Required property to **Yes** and the Allow Zero Length property to **No**.
4. Set the Default Value property of the LiaisonState field to: **FL**
5. Switch to **Datasheet View**, saving the changes to the table and choosing **Yes** in the warning boxes, and then add this record:

Field Name	Value
LiaisonID	AntonV
LastName	Anton
FirstName	Vera
Street	44 West Florida St.
City	Bradenton
State	FL
ZIP	34205
Telephone	9415554248
EmailAddress	VAnton@email.com

6. Close the VenueLiaisons table.

Create a Custom Field Format

7. Display the **Personnel** table in **Design View**.
8. Set these properties for the **PerID** field:
   - Field Size: `10`
   - Input Mask: `"UCE-"9999`
   - Caption: `ID`

   Each PerID will start with the literal value UCE, followed by four numbers, such as UCE-1001.
9. Switch to **Datasheet View**, saving the changes to the table and responding **Yes** to the warning message.
10. Click in the empty **ID** field in the new record at the bottom of the table.

    If your input mask is set up properly, Access will automatically add the prefix UCE to the entry.
11. Close the Personnel table after verifying your input mask is set up properly.
12. Display the **VenueLiaisons** table in **Design View**.

Set an Input Mask

13. Use the Input Mask Wizard to apply the **Phone Number** mask to the LiaisonPhone field.
14. Switch to **Datasheet View**, saving the changes to the VenueLiaisons table.

    The telephone numbers should now have the input mask applied.
15. Close the VenueLiaisons table. Display the **Schedules** table in **Design View**.
16. Set these field properties for the **Guests** field:

Validation Rule	Validation Text
`>=35`	`At least 35 guests must be entered`

Set a Field as a Lookup Field

17. Click anywhere in the **EventID** field and then click the **Lookup** tab in the Field Properties box and choose **Combo Box** as the Display Control.
18. Set these Lookup properties:

Row Source	Column Count	Column Widths	List Width
`Events`	`2`	`0.6;1.5`	`2.1`

19. Click anywhere in the **Menu Code** field and then click the **Lookup** tab in the Field Properties box and choose **Combo Box** as the Display Control.
20. Set these Lookup properties:

Row Source	Column Count	Column Widths	List Width
`Menus`	`2`	`0.6;1.5`	`2.1`

21. Switch to **Datasheet View** and save the changes to the table, responding **Yes** to any warning messages.

22. Enter this record:

Field Name	Value
ScheduleID	**SEMBenson**
VenueID	**ManCtr**
EventID	**SEMNAR**
MenuCode	**DINBUF**
EventDate	**7/3/2019**
Guests	**30**

Access displays a validation text message because the number of guests entered is less than the minimum of 35.

23. Click **OK**; change the number of Guests to **40** and add **Miller** in the Liaison field.
24. Close the database, saving the changes to the table if prompted.

APPLY YOUR SKILLS: A5-A3

Improve Data Readability and Validation

The CEO of Universal Corporate Events wants you to clean up the company's database tables and improve data readability and validation. In this exercise, you will add, delete, and rearrange fields. You will hide the cost per person in the Menus table, resize columns and set field sizes, and modify a field to convert data. Finally, you will set captions, default values, and field requirements; create a custom input mask; and modify a field to look up values.

1. Open **A5-A3-UCE** from your **Access Chapter 5** folder and save it as: **A5-A3-UCERev**
2. Establish a relationship by dragging the **Grade** field in the SalaryGrades table to the SalaryGrade field in the Personnel table, enforcing referential integrity.
3. Create a relationships report named: **Relationships**
4. Review the object dependencies and then close the Object Dependencies panel.

Modify Table Structure and Column Display

5. Display the **SalaryGrades** table in **Design View**.
6. Insert a new field in the SalaryGrades table above the Salary field with these settings:

Field Name	Data Type	Description
Salaried	**Yes/No**	**Indicates salaried position**

7. Switch to **Datasheet View**, saving the changes, and check the **Yes/No** boxes for each record with data in the **Salary** field.
8. Close the SalaryGrades table and then display the **Events** table in **Design View**.
9. Delete the **MinGuests** field and then close the Events table, saving the changes.
10. Display the **VenueLiaisons** table in **Datasheet View**.
11. Select the **Telephone** and **Email Address** columns and move both to the left of the Street Address column.

12. Apply **Best Fit** to all the columns and then close the table, saving the changes.
13. Display the **Menus** table in **Datasheet View**.
14. Hide the Cost/PP field and then close the table, saving the changes.

Change the Formatting of a Datasheet and Set Field Properties

15. Display the **Venues** table in **Datasheet View**.
16. Apply a light alternate row color and horizontal gridlines and change the gridline color to **Black**.
17. Close the Venues table, saving the changes.

Set Captions, Default Values, and Field Requirements

18. Display the **Personnel** table in **Design View** and set these field sizes and captions:

Field Name	Field Size	Caption
PerLastName	25	Last Name
PerFirstName	25	First Name
PerAddr	25	Street
PerCity	25	City
PerST	2	State
PerZIP	5	ZIP Code
PerPhone	15	Telephone
SalaryGrade		Salary Grade

19. Set the Required property of the PerLastName field to **Yes**.
20. Set the Default Value property of the PerST field to: **FL**
21. Close the table, saving the changes and choosing **Yes** in the warning boxes.

Set Formats, Input Masks, Allow Zero Length, and Lookup Fields

22. Display the **Schedules** table in **Design View**.
23. Enter **>LLLL<??????** as the Input Mask field property for the ScheduleID field.

    This mask forces 4 uppercase letters followed by from 0 to 6 lowercase letters.
24. Switch to **Datasheet View**, saving the changes.
25. Scroll down and type any series of 10 lowercase letters in the ScheduleID field of a new record.

    Because of the input mask, Access converts the first 4 letters to uppercase.
26. Tap Esc to remove the data you just entered.
27. Type any series of 8 uppercase letters in the ScheduleID field.

    Because of the input mask, Access converts all but the first 4 letters to lowercase.
28. Tap Esc to exit the record without saving and then close the Schedules table, choosing **OK** and **Yes** when prompted.
29. Display the **Venues** table in **Design View**.
30. Set the Format field property of the VenueWebSite field to: **@; No Website**

    This will display No Website *if there is no data in the VenueWebSite field.*

31. Set the Allow Zero Length property of the VenueLiaison field to **No**.
32. Save the Venues table, choosing **Yes** in the warning box.
33. Use the Input Mask Wizard to apply the Phone Number mask to the VenuePhone field.
34. Click anywhere in the **VenueST** field and then click the **Lookup** tab in the Field Properties box and choose **Combo Box** as the Display Control.
35. Set these Lookup properties:

Row Source	Column Count	Column Widths	List Width
States	2	0.3;1.2	1.5

36. Switch to **Datasheet View**, saving the changes if prompted, and enter this record:

Field Name	Value
VenueID	BradCC
Name	Bradenton Country Club
Street	2903 9th Ave
City	Bradenton
ST	FL
ZIP	34205
Phone	9415550031
Kitchen	Yes
Website	bcc.com
Liaison	AntonV

37. Close the database, saving the changes to the Venues table and any other open tables.

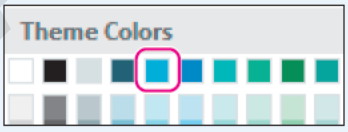
Project Grader

PROJECT GRADER: A5-P1

Taylor Games: Designing Tables and Creating Relationships

Taylor Games is getting ready to take customer orders, but before it can begin, its database tables need to be refined. You will add a relationship, modify data types, create a lookup field, set a validation rule, add a new field, and set additional field settings.

1. Download and open your Project Grader starting file.
   - *Using eLab:* Download **A5_P1_eStart** from the Assignments page. You *must* start with this file or your work cannot be automatically graded.
   - *Not using eLab:* Open **A5_P1_Start** from your **Access Chapter 5** folder.
2. In the **Order Details** table, use the **Lookup Wizard** in the Line Item field and apply the following Wizard settings to convert the field to a lookup field:
   - Get the values from the **Inventory** table.
   - Add **SKU** and then **Item** to the Selected Fields list.
   - Do not set a sort order.
   - Hide the Key Column.
   - Leave the label as *Line Item*.
   - Do not enable data integrity or allow multiple values.
3. In the **Orders** table, use these guidelines to add a new lookup field below the Order_Date field:
   - Name the field **`Service _ Rep`** and initiate the **Lookup Wizard**.
   - Get the values from the **Service Reps** table.
   - Add **Rep_ID**, **First_Name**, and **Last_Name** (in that order) to the Selected Fields list.
   - Sort by **Last_Name** in Ascending order.
   - Hide the Key Column.
   - Leave the label as *Service_Rep*.
   - Do not enable data integrity or allow multiple values.
4. Set the Caption property of the new Service_Rep field to: **`Service Rep`**
5. Use these guidelines to apply formatting to the Orders table:
   - Apply horizontal gridlines.
   - Apply alternate row color using the Turquoise, Accent 1 theme color.



6. Create a relationship using these guidelines:
   - Create a One-To-Many relationship between the **Order_ID** field in the Orders table and the **ID** field in the Order Details table.
   - Enforce referential integrity.
   - Allow cascade updating of related fields.
7. Set these field properties in the **Service Reps** table:
   - Make Position a required field.
   - In the Hire_Date field, set the Caption to: **`Hire Date`**
   - In the Hire_Date field, set this Validation Rule: **`>#12/31/2018#`**
   - In the Hire_Date field, set this Validation Text: **`2019 or later new hires`**
8. In the Service Reps table, follow these guidelines to create a new field:
   - Position the new field below the Hire_Date field.
   - Field Name: **`CSR_Training`**
   - Data Type: **Yes/No**
   - Caption: **`Trained?`**
9. With the Service Reps table in **Datasheet View**, check the boxes in the *Trained?* column for the first three service reps.
10. Save your database.
    - *Using eLab:* Save it to your **Access Chapter 5** folder as **`A5 _ P1 _ eSubmission`** and attach the file to your eLab assignment for grading.
    - *Not using eLab:* Save it to your **Access Chapter 5** folder as: **`A5 _ P1 _ Submission`**

PROJECT GRADER: A5-P2

WebVision: Refining Table Design

The WebVision database is growing, which requires additional settings and relationships. You will create a lookup field, add and delete table fields, and create relationships that connect your tables. You will then change the name of a table and set field properties.

1. Download and open your Project Grader starting file.
   - *Using eLab:* Download **A5_P2_eStart** from the Assignments page. You *must* start with this file or your work cannot be automatically graded.
   - *Not using eLab:* Open **A5_P2_Start** from your **Access Chapter 5** folder.
2. In the **Orders** table, delete the Amount field and the data in it.
3. Use these guidelines to create a new lookup field in the Orders table:
   - Position the field below the RepID field.
   - Name the field: **`Customer`**
   - Use the Lookup Wizard and get the values from the **Customers** table.
   - Add **Customer ID** and **Company Name** to the Selected Fields list.
   - Do not set a sort order.
   - Hide the Key Column.
   - Use the label: **`Customer`**
   - Do not enable data integrity or allow multiple values.

4. In the Orders table, use the **Customer** lookup field to add customers to existing orders as shown here:

Order Number	Rep ID	Customer
1	S101	**Wide World Imports**
2	S102	**Fourth Coffee**
3	S103	**Alpine Ski House**
4	S104	**Westwind Traders**
5	S101	**Coho Vineyard & Winery**

5. In the Orders table, adjust the width of the Customer column to fit the widest entry.
6. Create a relationship using these guidelines:
   - Create a One-To-Many relationship between the **OrderID** field in the Orders table and the **Order ID** field in the Order Details table.
   - Enforce referential integrity.
7. Rename the Products and Services table as: **Line Items**
8. In the **Customers** table, set these field properties for the State/Province field:
   - Field Size: **2**
   - Input Mask: **>LL**
9. In the Customers table, set these field properties for the Postal Code field:
   - Field Size: **5**
   - Input Mask (zeros): **00000**
10. In the Customers table, create an input mask on the Phone Number field using these guidelines:
    - Use the Input Mask Wizard.
    - Choose the **Phone Number** input mask.
    - Keep the placeholder character set to the default.
    - Store the data **With the symbols in the mask**.
11. Save your database.
    - *Using eLab:* Save it to your **Access Chapter 5** folder as **A5 _ P2 _ eSubmission** and attach the file to your eLab assignment for grading.
    - *Not using eLab:* Save it to your **Access Chapter 5** folder as: **A5 _ P2 _ Submission**

Extend Your Skills

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

A5-E1 That's the Way I See It

You have decided to enhance the table structure in the Blue Jean Landscaping database. Open and then save **A5-E1-BJL** as **A5-E1-BJLRev** and begin by creating a relationship between the MerchID fields in the StoreMerchandise and SalesInvoices tables. Enforce referential integrity and create a relationships report (call it **BJL Relationships**). Autofit all columns in the StoreMerchandise table, apply the input mask *"BJL"-9999* to the MerchID field, and switch the order of the Manufacturer and ItemName fields. In the Customers table, make the last name required, display the message *No Email* for customers without an email address, and use the Input Mask Wizard to apply the Phone Number input mask to the CustTelephone field.

A5-E2 Be Your Own Boss

You want to improve how your Blue Jean Landscaping database looks and behaves. Open and save **A5-E2-BJL** as **A5-E2-BJLRev** and first create relationships between the CustID fields in the Customers and SalesInvoices tables and between the CustID fields in the Customers and ServiceInvoices tables. Enforce referential integrity and create a relationships report (named **BJL Relationships**). In the ServiceInvoices table, hide the ServID field, autofit all remaining columns, and make all fields required. In the ServiceReps table, apply the input mask *"BJLRep-"9999* to the RepID field and the Phone Number input mask to the RepPhone field. Finally, set RepState as a combo box lookup field with the field properties Row Source=States, Column Count=2, Column Widths=0.5, 1.5, List Width=2.

A5-E3 Demonstrate Proficiency

Stormy BBQ is continuing to update its database as it remodels its flagship location in Key West to give it a more tropical look and feel. It's also added more employees! To begin, open **A5-E3-SBQ** and save it as: **A5-E3-SBQRev**

Apply the techniques learned in this chapter to format and dress up the Staff table. Set appropriate field sizes and captions, autofit columns as needed, and display the text *No Email Available* for staff members who don't have an email address.

ACCESS

6 Customizing Input Forms

Relational database design often requires data from more than one table or query to appear on the same form. Customizing forms and their controls while applying other form settings gives designers more control while simultaneously making forms more efficient and easier to use. In this chapter, you will add a subform to a main form, which is a handy technique used to include data from a different source. You will also explore Access features to create calculated fields, add tips to form controls, and set control properties to protect and limit data entry.

LEARNING OBJECTIVES

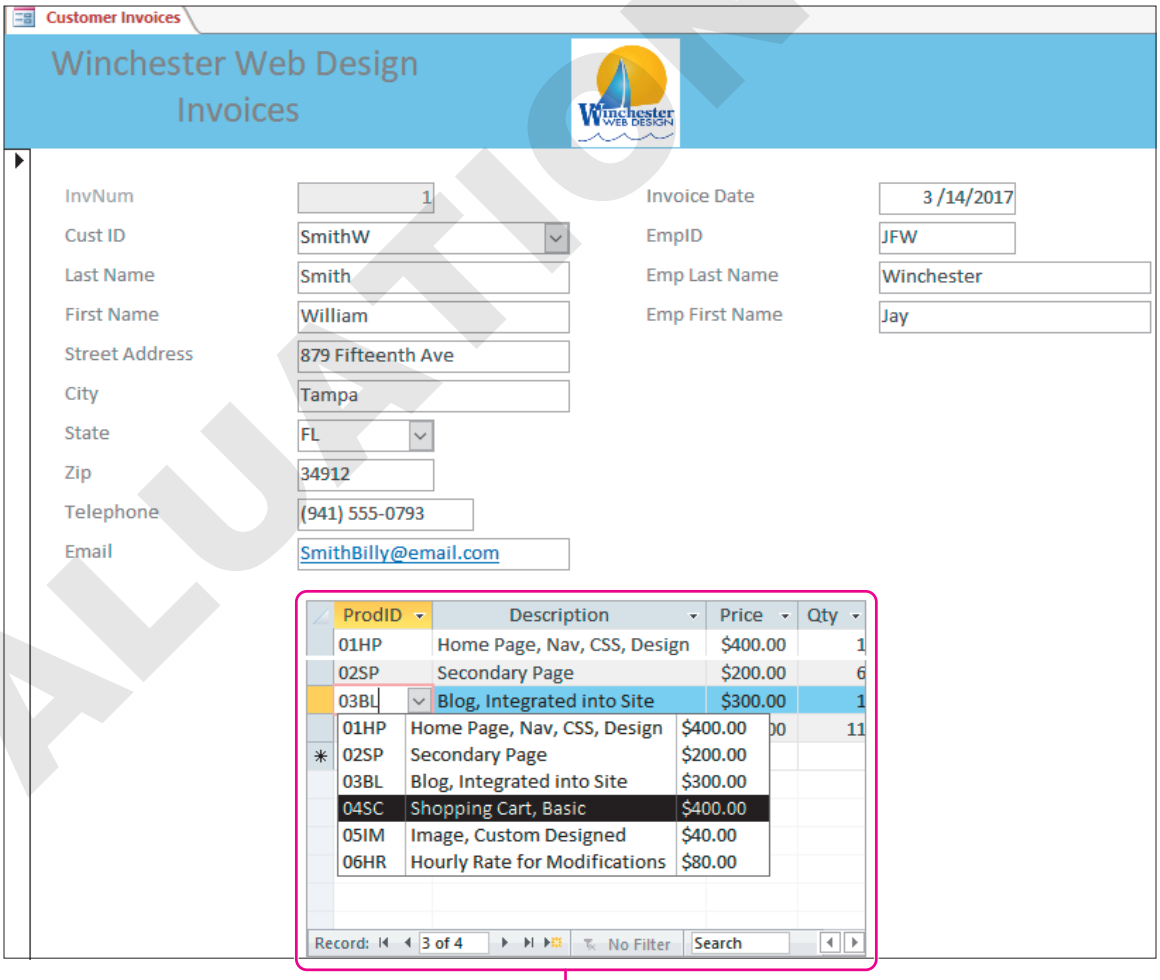
- Create a form that contains a subform
- Add a calculated control to a form
- Add a total row to a form
- Disable form fields
- Lock form fields
- Add ScreenTips to forms
- Create pop-up forms

Project: Formatting Functional Forms

Winchester Web Design has seen sales increase over recent months and wants to simplify data entry as a result. You've been asked to design advanced forms to make data entry easier for all team members 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, and you will use form features to enhance data entry. 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 a main form. Subforms work well when one-to-many relationships are set, allowing the user to work with 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.



The detail data from related tables appears in a subform.

A main form displaying customer and employee information

Creating Subforms

The easiest way to create a subform is to use the Form Wizard, which creates a new form with an option to add a subform. Subforms can also be added to existing forms by placing a Subform control on the existing form, where you can specify the subform's position and size. When the Subform control is added to the form, the Subform Wizard launches, allowing you to set subform properties.

View the Video "Creating Subforms with the Form Wizard."

Create→Forms→Form Wizard

Design→Controls→Subform

DEVELOP YOUR SKILLS: A6-D1

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

1. Open **A6-D1-WinDesign** from your **Access Chapter 6** folder and save it as: **`A6-D1-WinDesignRev`**
2. Choose **Create→Forms→Form Wizard**.
3. Choose **Table: Invoices** from the Tables/Queries list and then add the **InvNum** and **CustID** fields to the Selected Fields list.
4. Add the fields indicated to the Selected Fields list, making sure to add them in the order shown:

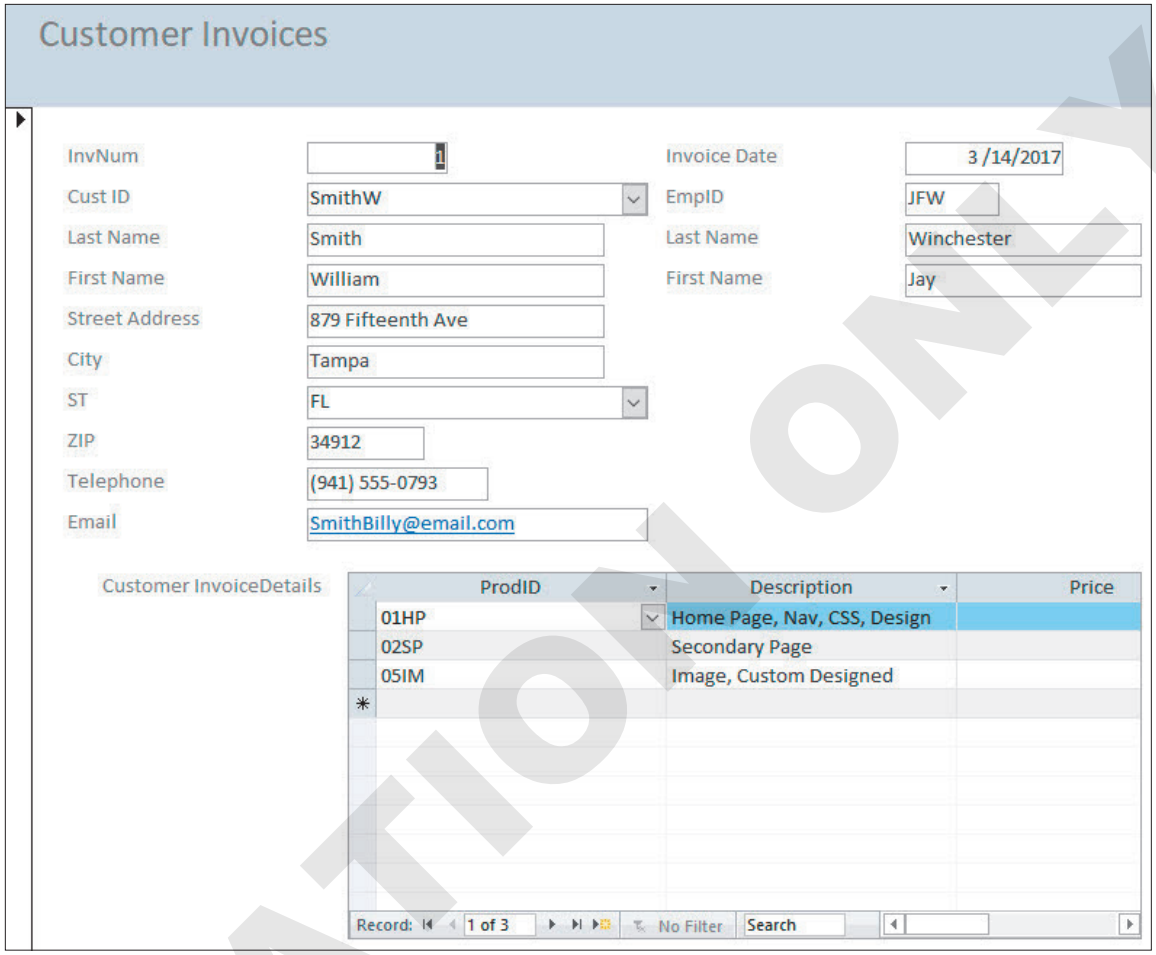
Customers	Invoices	Employees	InvoiceDetails	Products	InvoiceDetails
CustLastName	InvDate	EmpID	ProdID	ProdDescription	Qty
CustFirstName		EmpLastName		Price	
CustStreetAddress		EmpFirstName			
CustCity					
CustState					
CustZIP					
CustPhone					
CustEmail					

5. Click **Next** and notice that the data is arranged by invoice number and the **Form with Subform(s)** option is already chosen.

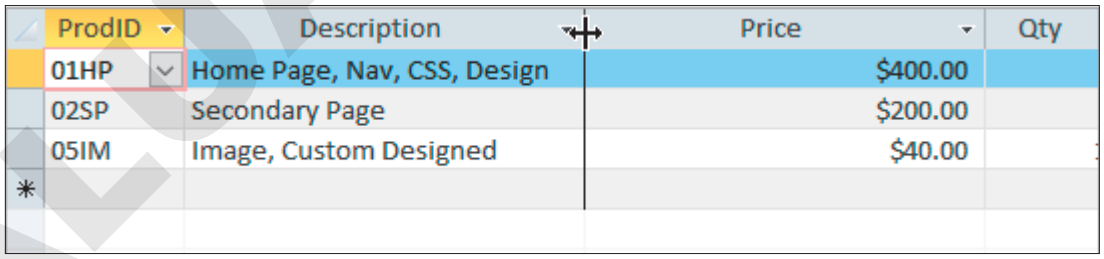
   The subform preview in the Wizard includes the ProdID, ProdDescription, Price, and Qty fields.
6. Click **Next** to accept the settings; click **Next** again to accept the Datasheet subform layout.

7. In the final Wizard screen, name the form **`Customer Invoices`** and the subform **`Customer InvoiceDetails Subform`** and then click **Finish**.

   The new form and subform display in Form View.



8. Double-click the right border of each subform column heading to autofit the columns.



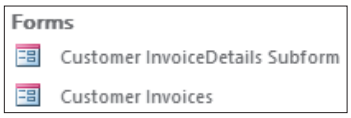
9. Click the empty **ProdID** cell in the new row at the bottom of the subform.
10. Click the drop-down **menu** button ▾ and choose **03BL**.

    It's easy to populate the first three fields of the new subform record.

11. Enter **1** in the **Qty** column and tap Tab to complete the record.

Modifying Subforms

When you use the Form Wizard to create a subform, Access creates the subform and a main form with the subform embedded within it. Both are displayed in the Navigation pane. You can then open and modify the subform by itself in Layout View or Design View, or you can open the main form and modify it and the embedded subform together.



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: A6-D2

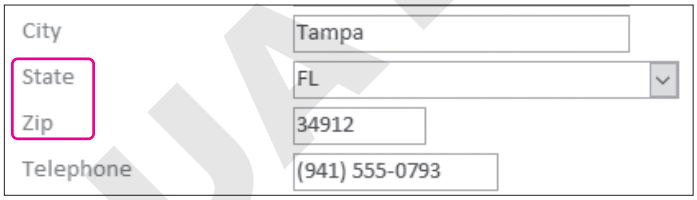
In this exercise, you will modify the Customer Invoices form and subform by deleting, moving, and sizing controls, and adding a graphic to the form.

1. Switch to **Layout View**.

   You will begin by modifying the layout and controls on the main form.

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`**



4. Locate the EmpID label and notice the Last Name and First Name labels below it.

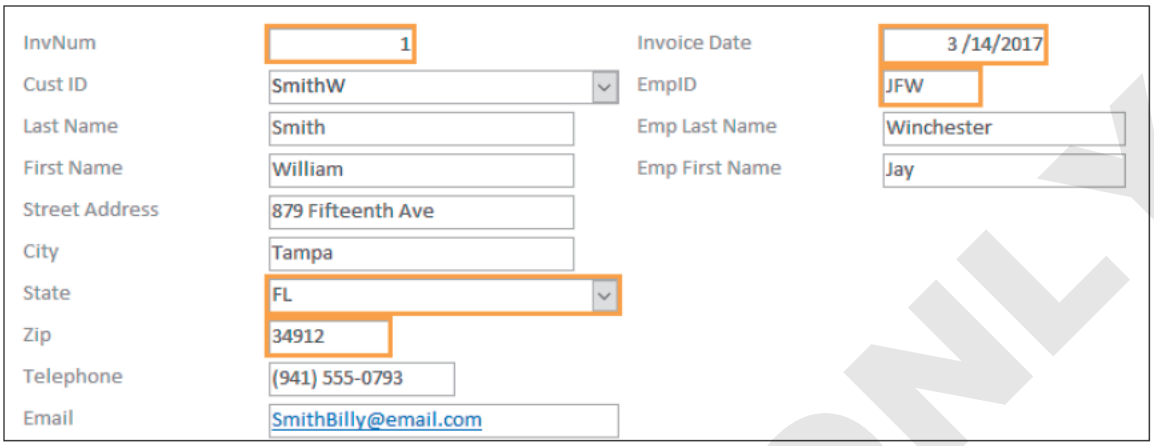
5. Change the employee **Last Name** label to **`Emp Last Name`** and the employee **First Name** label to: **`Emp First Name`**

6. If necessary, display the Property Sheet by choosing **Form Layout Tools→Design→Tools→ Property Sheet**.

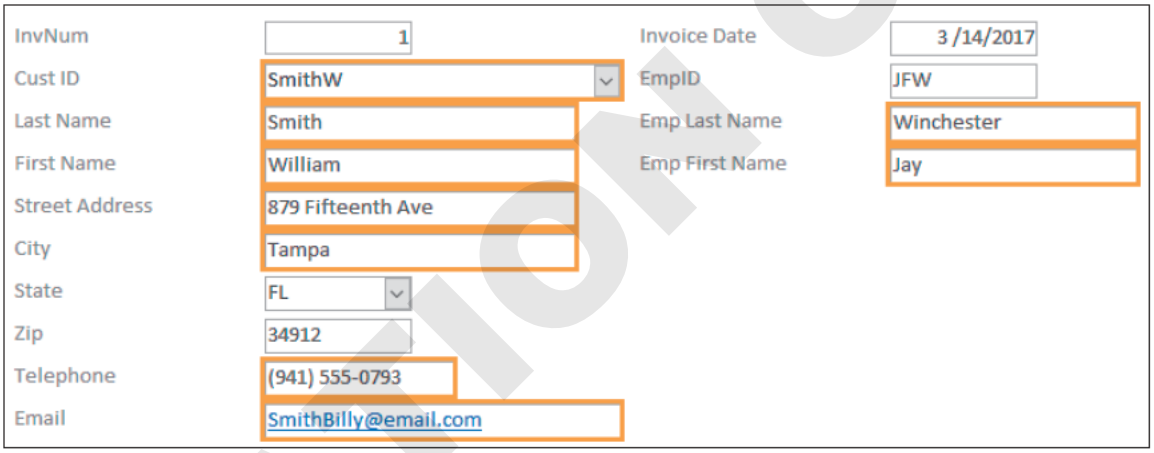
   As you select and move controls in the following steps, you may need to drag the Property Sheet out of the way or close and reopen it as needed.

Tip! *You can also tap F4 to toggle the Property Sheet open and closed.*

7. Select the text boxes shown here by holding Ctrl while clicking the boxes and then set the **Width** property to: **1**

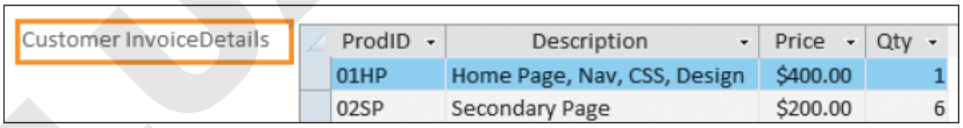


8. Select the text boxes shown here and set the Width property to: **2**

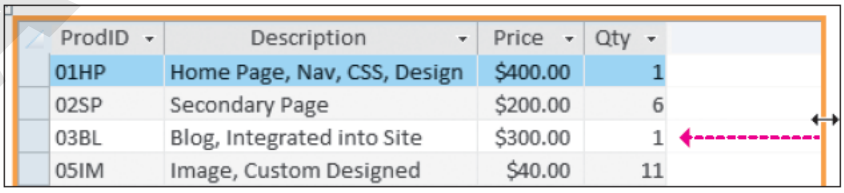


Modify, Lay Out, and Size Controls on the Subform

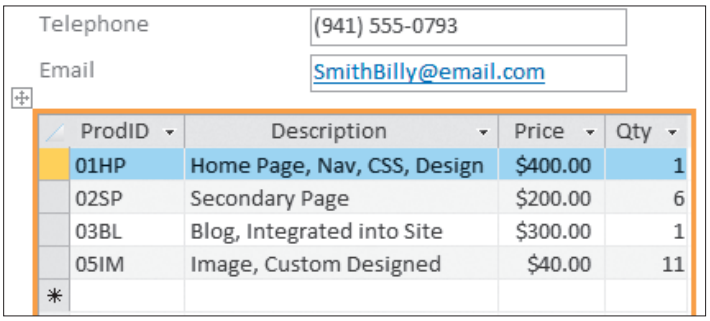
9. Select the **CustomerInvoiceDetails** label on the subform and tap Delete.



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



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



Modify the Form Header

In the next few steps, you will modify the form title and add color and a logo to the header.

12. Use these guidelines to modify the form:

- Click the title box control to select it and then click just in front of *Invoices* and press Shift+Enter to move *Invoices* to a new line.
- Set the Width property to: **3**
- Replace *Customer* with: **`Winchester Web Design`**
- Select both lines and choose **Home→Text Formatting→Center** to center the title in the box.

13. Click to the right of the title box in an empty part of the form header.

The title box will become deselected when you select the form header.

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

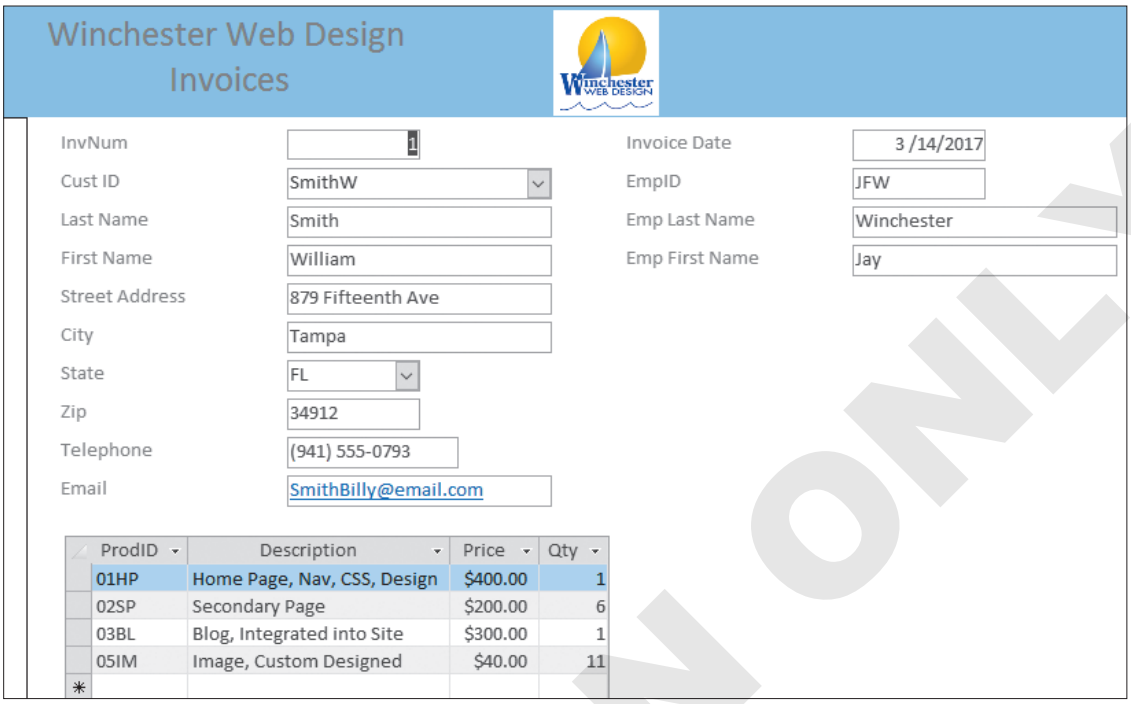
15. Click the **Build** ... button and choose a light blue color or whatever color you feel looks best in the form header.

16. Choose **Form Layout Tools→Design→Header/Footer→Logo**.

17. Navigate to your **Access Chapter 6** folder, choose **WWD-Logo.bmp**, 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.



Adding Calculations to Forms

There are several ways to add calculations to forms, and 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, you will need to create a calculated control by adding an unbound control to the form and placing a formula within the control source property.

Applying Totals to Forms in Datasheet Layout

The Form Wizard has an option for creating 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 an aggregate function such as count, sum, or average to create totals for numeric fields in the datasheet.

Home→Records→Totals Σ

DEVELOP YOUR SKILLS: A6-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 click **Next**.
3. Choose **Datasheet** as the layout and click **Next**.
4. Name the new form **`Invoices Query Form`** and click **Finish**.

   The form looks like a table when displayed in Datasheet View.
5. Choose **Home→Records→Totals** Σ.

   A Total line appears at the bottom of the datasheet layout.

	18	12/4 /2017	Secondary Page
	19	12/9 /2017	Secondary Page
	Total		

6. Click in the **Qty** cell on the Total row.

   A menu button appears at the left end of the cell.
7. Click the **menu** button ▾ and choose **Count**, then click in the **LineTotal** cell and choose **Sum** from the menu.

   The Count function simply counts the number of rows containing a quantity while the Sum function adds all line totals in the column.

	18	12/4 /2017	Secondary Page	$200.00	4	$800.00
	19	12/9 /2017	Secondary Page	$200.00	3	$600.00
	Total				125	$62,920.00

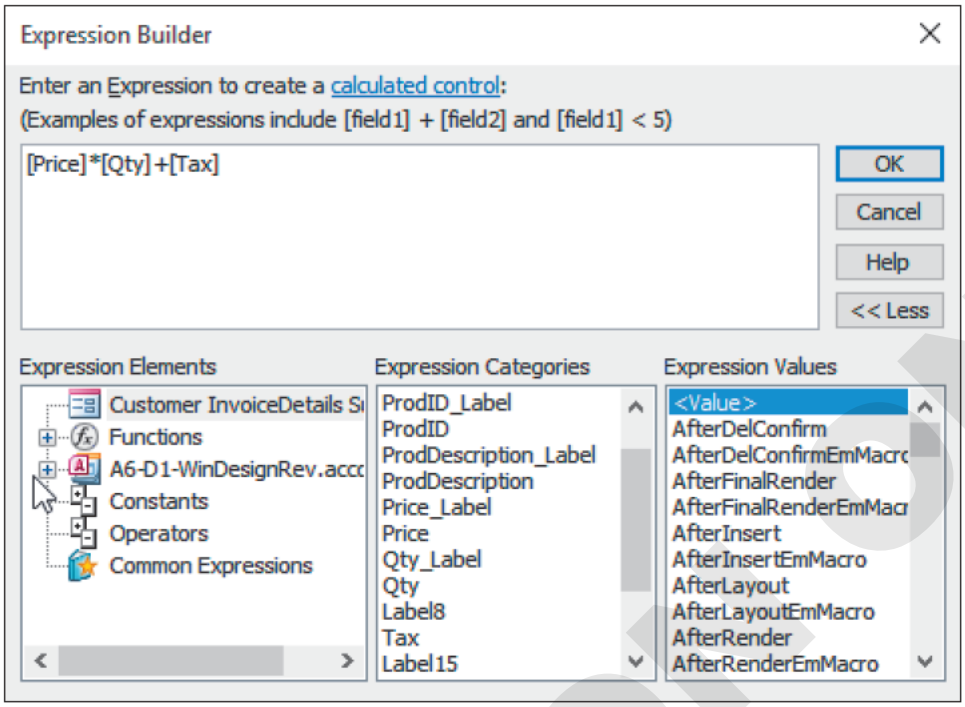
8. Scroll through the datasheet's 125 rows and notice that the Total row remains fixed at the bottom of the window.

   The Totals feature is an easy way to use aggregate functions and is available in forms and in tables and queries.
9. 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's necessary to display calculations in Form View, which displays just a single record at a time. You can do this by inserting a calculated control. A calculated control is an unbound text box control placed on the form with a formula inserted in the Control Source property. Usually, 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 fields shown on the form. Calculated controls can also contain aggregate functions or an array of built-in functions and expressions made available by using the Expression Builder.

The Expression Builder dialog box contains a list of available fields in the current object and a selection of built-in operators, functions, and expressions that can be leveraged by both novice and advanced users.



View the Video "Calculated Controls in Forms."

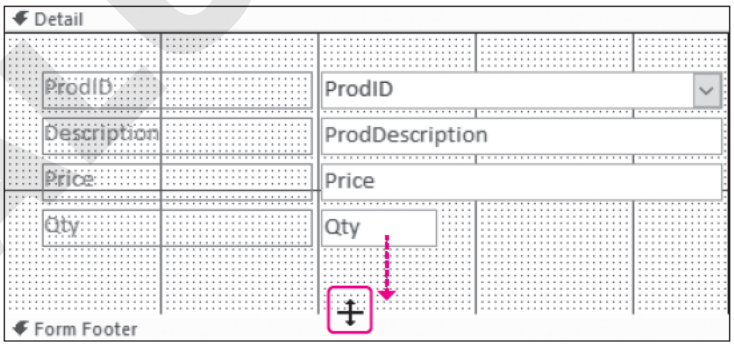
DEVELOP YOUR SKILLS: A6-D4

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

1. Display the **Customer InvoiceDetails** subform in **Design View**.

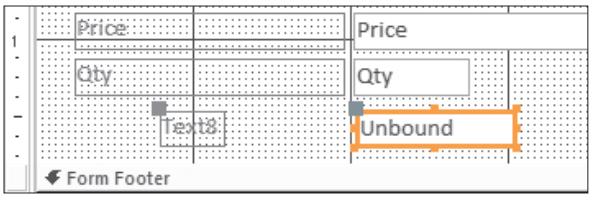
   The first step is to insert a new control.

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 two new text boxes.



3. Choose **Format Design Tools→Design→Controls→Text Box** ab|.

4. Click just below the Qty control text box to insert a new control.

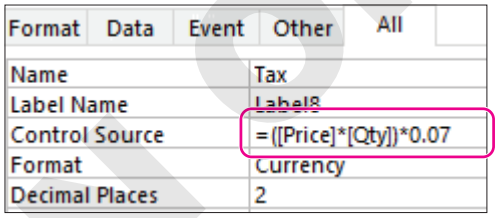


5. Use the arrow keys as needed to nudge the control so it is aligned with the Qty control.

Create Formulas

6. If necessary, choose **Format Design Tools→Design→Tools→Property Sheet** to display the Property Sheet.
7. Click the **All** tab in the Property Sheet box and set these properties:

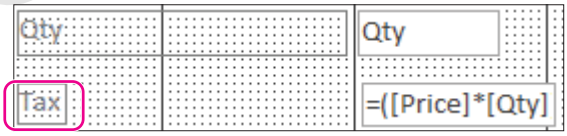
Property	Value
Name	**Tax**
Control Source	**=(Price*Qty)*.07**
Format	**Currency**
Decimal Places	**2**



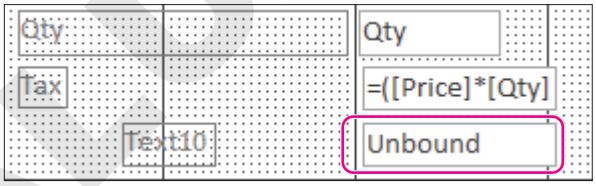
Notice that the text for Price and Qty are now surrounded by brackets, which indicates that these are fields.

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

Property	Value
Caption	**Tax**
Width	**0.25**
Left	**0.25**



9. Choose **Format Design Tools→Design→Controls→Text Box** ab|.
10. Click just below the Tax control text box to insert a new control.



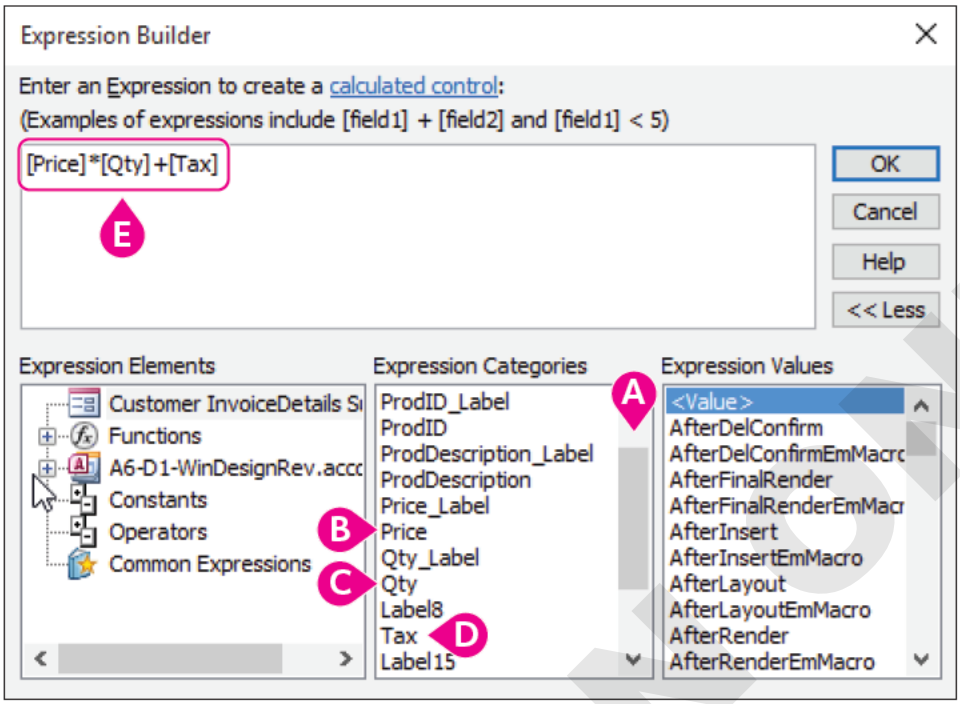
11. Use the arrow keys as needed to align the control with the Tax and Qty controls above it.
12. Click the **All** tab in the Property Sheet box and set these properties:

Property	Value
Name	**Line Total**
Format	**Currency**
Decimal Places	**2**

13. Click the **Control Source** property and then click the **Build** ... button to the far right.

The Expression Builder dialog box appears.

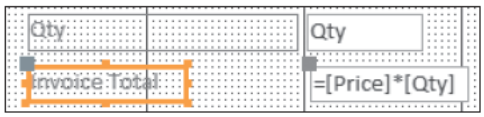
14. Follow these steps to create a calculated control with the Expression Builder:



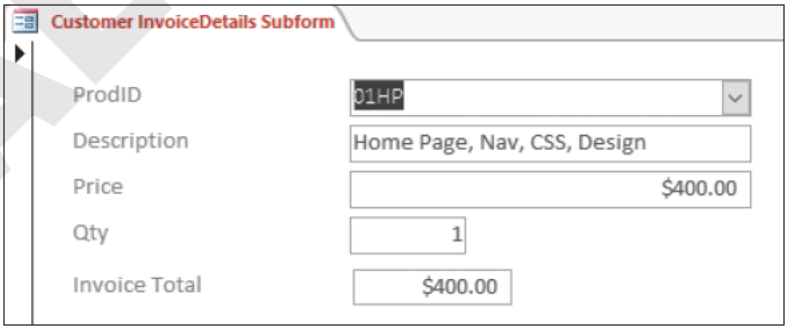
- **A** Scroll the Expression Categories pane until the Tax control is visible.
- **B** Double-click the **Price** control to enter the field into the formula.
- **C** Tap [*] and then double-click the **Qty** control.
- **D** Tap [+] and then double-click the **Tax** control.
- **E** Ensure your formula matches the example and then click **OK**.

15. Click the **Text10** label on the subform and set these properties:

Property	Value
Caption	`Invoice Total`
Width	`1`
Left	`0.25`



16. Switch to **Form View** to see your new calculated controls in action.



17. Navigate to record 16 and others for which the quantity is greater than 1 and notice that your calculated controls always perform the correct calculations.

18. Close and save the form.

Setting Properties to Assist and Control Data Entry

Some properties are meant to assist with data entry, while others help control or limit it. These properties are easily set using the Property Sheet.

Disabling Form Fields

Sometimes a form may include data that users should not enter themselves or even access, for example, setting up passwords or entering Social Security numbers. Many forms also contain settings that enter default values, such as states, or values that correspond to data contained in another field, such as city and ZIP code fields. To protect a field on a form from being edited during data entry, you can disable the field in the Property Sheet. Disabled fields are unavailable for data entry; they're still visible, but they appear grayed out and are not accessible to the user. During data entry, Access automatically skips a disabled field and moves directly to the next enabled and accessible field. To disable a field, just set the Enabled property to No.

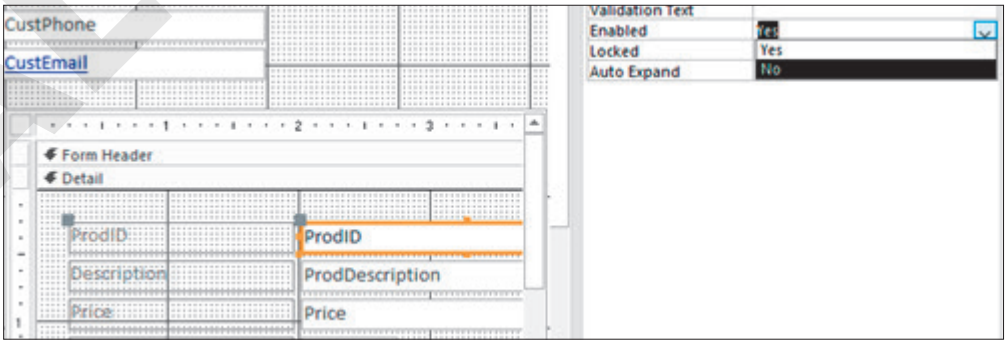
Locking Form Fields

Another way to protect fields from being edited is to lock them. Locked fields remain available on the form and appear as normal; this allows users to click in the field, but they cannot change the data. Locked fields improve readability when a form is printed versus disabled fields, which appear grayed out and print faintly. To lock a field, set the Locked property to Yes.

DEVELOP YOUR SKILLS: A6-D5

In this exercise, you will disable the InvNum field in the Customer Invoices form and the LineTotal field in the Customer InvoiceDetails subform. In addition, you will lock the Price field in the subform so it cannot be edited.

1. Display the **Customer Invoices** form in **Design View**.
2. Click the **InvNum** text box on the main form and then click the **Data** tab on the Property Sheet.
3. Disable the text box by changing the Enabled property to **No**.
4. Click anywhere on the subform to make it active and then click the **ProdID** text box.



5. Disable the ProdID text box.
6. Click the **Price** text box in the subform and set the Locked property to **Yes**.

Test the Property Settings

7. Switch to **Form View** and try clicking in the disabled **InvNum** field.

   The field is grayed out, and you can't click in it.

8. Try clicking in any of the cells of the disabled ProdID field in the subform.

   In the subform, only the data appears grayed out, but you still cannot click in the field.

9. Click in any **Price** field in the subform and try changing the number.

   Locking allows users to select a field, but the data cannot be changed.

Adding Control Tips

When you create a table and define fields, you can enter field details in the Description column. These field descriptions appear in the status bar when the fields are active during data entry. They also appear in the status bar when a field appears on a form. Although forms identify most fields with control labels, sometimes labels for specific fields, such as State and ZIP, are removed from a form when the controls are grouped under a more general label such as *Address*.

To help data entry personnel determine what data to type in a field, you can add descriptive messages to display onscreen by setting the ControlTip Text property for a control. ControlTips appear when the user points to the control. Setting control tips helps to provide explanations for controls.

DEVELOP YOUR SKILLS: A6-D6

In this exercise, you will create control tips for the disabled and locked fields on the Customer Invoices form to explain why they are inaccessible.

1. Display the **Customer Invoices** form in **Design View**.
2. Click the **InvNum** text box and then click the **Other** tab on the Property Sheet.
3. Click in the **ControlTip Text** property text box and type: **`Invoice numbers are assigned automatically and cannot be changed.`**
4. Enter these control tips for the subform controls:

ProdID text box	**`Product IDs are assigned by supervisors and cannot be edited.`**
Price text box	**`Product prices cannot be changed.`**

5. Save the changes to the form.
6. Switch to **Form View** and point to the InvNum field to display the control tip.
7. Point to the ProdID and Price fields in the subform and notice the control tips do not appear.

   Control tips display only in Form View. The subform is currently displayed in Datasheet View, so the tips don't show.

8. Open the **Customer InvoiceDetails Subform** in **Form View** and point to the ProdID and Price controls to display the control tips.
9. Close the Customer InvoiceDetails **Subform**.

   Leave the Customer Invoices form open.

Creating a Pop-Up Form

Forms and reports can be set to open in pop-up windows that stay on top of other open database objects. Pop-up forms can prompt a user for information or display a window containing supplemental data. Such forms and reports can help data entry personnel look up values while processing orders or looking up item prices. You can apply different formats to pop-up forms.

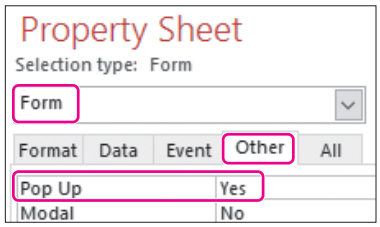
POP-UP WINDOW MODES

Mode	Description
Modal Pop Up	Displays a custom dialog box that prevents access to other database objects until the dialog box is closed or its required actions are taken Example: If you choose to print a report and have the Print dialog box open, you cannot make changes to the report until you click OK or Cancel in the dialog box.
Modeless Pop Up	Creates a pop-up window that sits on top of other open windows in such a way that users cannot work in the database while it is open Example: When processing orders, you could set the Inventory List to open as a modeless pop-up form to ensure you have the correct inventory number.

DEVELOP YOUR SKILLS: A6-D7

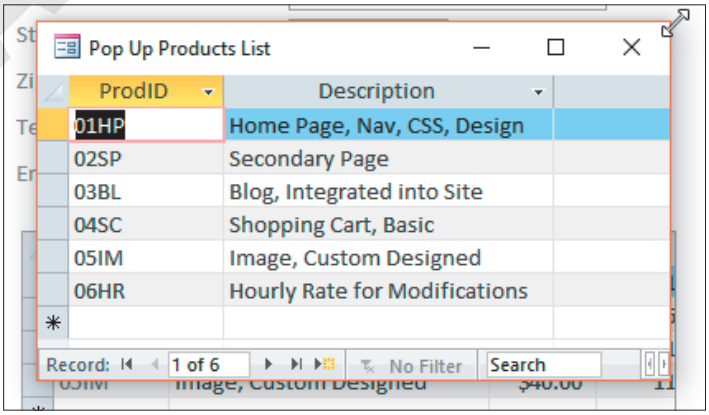
In this exercise, you will create and test a pop-up form using the Winchester Web Design Products table.

1. Select the **Pop Up Products** table from the Navigation pane.
2. Choose **Create→Forms→Form Wizard**.
3. Add all three fields to the Selected Fields list, choose **Datasheet Layout**, and name the form: **`Pop Up Products List`**
4. Switch to **Design View** and choose **Form Design Tools→Design→Tools→Property Sheet**.
5. If necessary, choose **Form** from the Selection Type list and then click the **Other** tab and set the Pop Up property to **Yes**.



6. Switch to **Datasheet View** and double-click the right border of each column heading to autofit the columns.
7. Resize the window using the sizing handle until it just encloses the datasheet.

   You may need to reposition the form first by dragging the title bar until you can see the frame borders.



8. Position the pop-up form in a location where all fields in the Customer Invoices form remain visible.
9. Use the Navigation bar at the bottom of the Customer Invoices form to navigate through the records.

   Notice the pop-up form remains available and in place.
10. Close the database, saving the changes to any unsaved forms.

Self-Assessment

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

Reinforce Your Skills

REINFORCE YOUR SKILLS: A6-R1

Create and Modify a Form with a Subform

In this exercise, you will create a new Donors form for Kids for Change. You will delete, move, and size controls; modify the title; and add a logo to the form.

1. Open **A6-R1-K4C** from your **Access Chapter 6** folder and save it as: **A6-R1-K4CRev**
2. Select the **Donors** table in the Navigation pane and choose **Create→Forms→Form Wizard**.
3. Add all fields from the Donors table to the Selected Fields list, except the first field, DonorID.

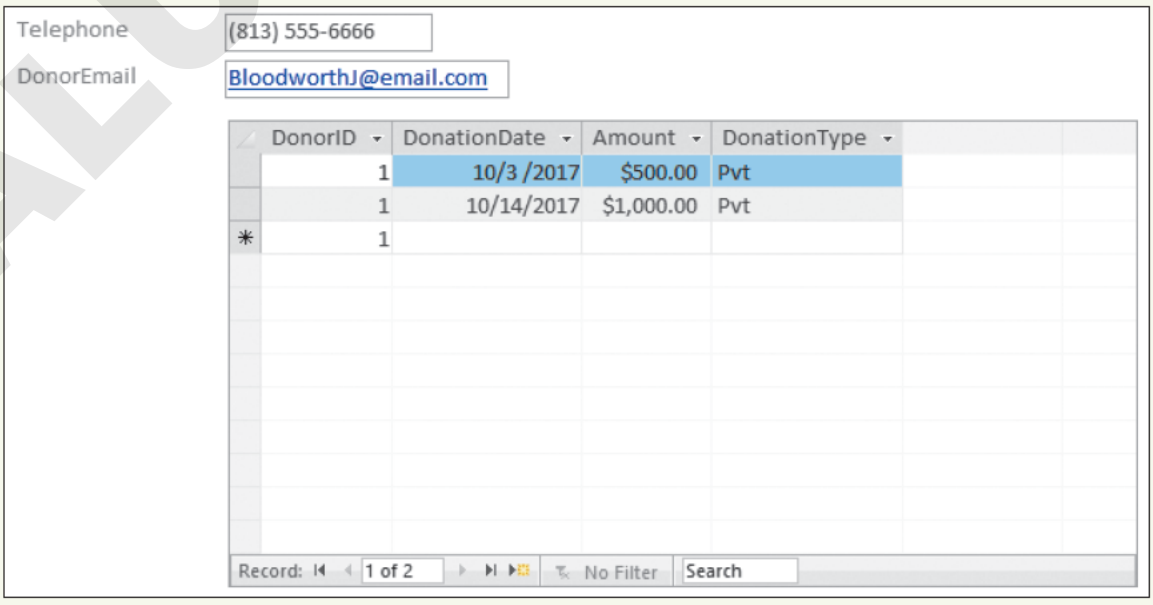
   Hint: Use the Move All Fields [>>] button and Remove [<] button on the DonorID field.
4. Click the last field in the Selected Fields list.

   This ensures the fields you add in the next step are placed at the bottom of the list.
5. Add all four fields from the Donations table to the Selected Fields list.
6. Click **Next** and click **Next** again to leave the viewing options set to By Donors and a Form with Subform(s).
7. Leave the layout set to Datasheet and click **Next**.
8. On the final Wizard screen, name the form **Donors Form** and the subform **Donations Subform** and click **Finish**.

   The new form and subform display in Form View.

Adjust the Subform Layout

9. Switch to **Layout View** then click the **Donations** label to the left of the subform and tap [Delete].
10. Double-click the right edge of each subform column heading to autofit the columns.
11. Size the subform by dragging its borders and then use the arrow keys to position the subform as shown here:



Set Form Field Properties

12. Use the Navigation bar at the bottom of the main form window to move to the third record (McGovern).

    Notice the email address text box is a bit too narrow for the email address.

13. Click the email address to select it and then display the Property Sheet.
14. Click the **Format** tab and then set the Width property to: **`1.8`**
15. Use the Navigation bar to scroll through the database records.

    The Acknowledgements text box is larger than it needs to be.

16. Click the **Acknowledgements** text box and set these properties:

Property	Value
Width	**2**
Height	**0.5**

17. Set the widths of the State and DonorZIP text boxes to: **`0.8`**

Modify the Form Header

18. Click the **Donors Form** title in the Form Header to select the title box and set the Width property to: **`3.5`**
19. Click in the title box and change the title to: **`Kids for Change Donor Form`**
20. Choose **Format Layout Tools→Design→Header/Footer→Logo** .
21. Navigate to your **Access Chapter 6** folder, choose **K4C-Logo.bmp**, and click **OK**.
22. Set the Width and Height properties to **`0.8`** and the Left property to: **`4`**
23. Switch to **Form View** to view your completed form.
24. Close the database, saving the changes to both the form and subform.

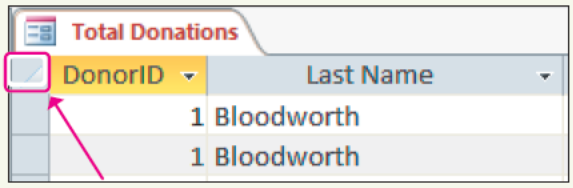
REINFORCE YOUR SKILLS: A6-R2

Add a Totals Row to a Form

In this exercise, you will add a Totals row to a form that counts the number of individual donations and totals the amount of the donations.

1. Open **A6-R2-K4C** from your **Access Chapter 6** folder and save it as: **`A6-R2-K4CRev`**
2. Select the **Donations Query** in the Navigation pane and launch the **Form Wizard**.
3. Move all fields except Acknowledgement, ScholarFund, and NetAmt to the Selected Fields list and click **Next**.
4. Accept the By Donations view and click **Next**.

5. Choose the **Datasheet** layout and click **Next**.
6. Name the new form **`Total Donations`** and click **Finish**.
7. Click the **Select All** box at the top-left corner of the datasheet to select all columns.



8. Double-click the right edge of any column heading to autofit all columns.
9. Choose **Home→Records→Totals** Σ.

   A Total row is added to the bottom of the datasheet.
10. Click in the **Last Name** cell on the Total row.
11. Click the **menu** button ▾ on the left and choose **Count** from the drop-down menu.

    There are 16 individual donations.
12. Click in the **Amount** cell on the Total row and choose **Sum** from the menu.
13. Autofit the Amount column so the total is visible.

    The donations total $11,150, all in even dollar amounts so decimal places aren't needed.
14. Switch to **Layout View** and then click the **Amount** text box and display the Property Sheet.
15. On the **Format** tab, set the Decimal Places property to **0** and then close the Property Sheet.
16. Switch to **Datasheet View** and autofit the Amount column again.
17. Close the database, saving the changes to any unsaved forms.

REINFORCE YOUR SKILLS: A6-R3

Disable and Lock Fields, Calculated Controls, and Pop-Up Forms

In this exercise, you will disable and lock subform fields to prevent information from being altered by those without permission to do so. You will also add a calculated control to a form and create a pop-up form for easy access to donor information.

1. Open **A6-R3-K4C** from your **Access Chapter 6** folder and save it as: **`A6-R3-K4CRev`**
2. Display the **Donations** subform in **Design View** and then click the **DonorID** text box and display the Property Sheet.
3. Click the **Data** tab and change the Enabled property to **No**.
4. Click the **Other** tab and then click in the **ControlTip** text box and type: **`Donor IDs are set by the administrator and cannot be edited.`**
5. Click in the **Amount** text box on the form and then click the **Data** tab and set the Locked property to **Yes**.
6. Click the **Other** tab and then click in the **ControlTip Text** box and type: **`Donor totals are calculated by the program and cannot be edited.`**
7. Switch to **Form View** and point to the DonorID and Amount controls to see the control tips you just created.
8. Close the form, saving the changes.

Create a Calculated Control

9. Choose the **ScholarFund Donations** table in the Navigation pane and launch the **Form Wizard**.
10. Add all fields to the form and click **Next**.
11. Choose the **Datasheet** layout option and click **Next**.
12. Leave the name as *ScholarFund Donations* and click **Finish**.
13. Switch to **Design View** and then drag the top edge of the **Form Footer** section bar down to make room for a new text box.
14. Choose **Format Design Tools→Design→Controls→Text Box** and then click just below the ScholarFund control to insert a new text box there.
15. Use the arrow keys as needed to align the control with the ScholarFund control.
16. Make sure the new control is still selected and, if necessary, display the Property Sheet.
17. Click the **All** tab and set these properties:

Property	Value
Name	`Total Donation`
Control Source	`=Amount+ScholarFund`
Format	`Currency`
Decimal Places	`0`

18. Click the text box label and set these properties:

Property	Value
Caption	`Total Donation`
Width	`1`
Left	`0.25`

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

    Notice the fields are of different widths, creating a poor form layout.
20. Switch back to **Design View** and set the Width property of all text boxes to: **1**
21. Switch back to **Form View**.

    The field widths are now consistent, but the left alignment of the DonationType field needs to be changed.
22. Switch back to **Design View** and set the Text Align property of the DonationType text box to **Right**.
23. Switch back to **Form View** to view your completed form.

Create a Pop-Up Form

24. Choose the **Donors** table in the Navigation pane and launch the **Form Wizard**.
25. Add the **DonorID**, **DonorLName**, and **DonorFName** fields to the Selected Fields list and click **Next**.
26. Choose **Datasheet** layout and click **Next**.

27. Enter **`Donor Popup`** as the name, choose the option to **Modify the Form's Design**, and click **Finish**.

    The form displays in Design View.

28. Click the **Other** tab and set the Pop Up property to **Yes**.
29. Save the change to the form and then switch to **Datasheet View**.
30. Autofit the three columns in the pop-up form.
31. Adjust the height and width of the pop-up form by dragging the frame borders until the datasheet fits nicely within the frame.

    You may need to drag the form slightly by the title bar before sizing it.

32. Drag the pop-up form to the right of the ScholarFund Donations fields.
33. Navigate through the records in the ScholarFund Donations form while the pop-up form remains in place, giving you access to all donor information at a glance.
34. Close the database, saving changes to any unsaved forms.

Apply Your Skills

APPLY YOUR SKILLS: A6-A1

Create a Form and Subform and Add a Totals Row

Universal Corporate Events would like you to help them track venues, employees, and their pay. In this exercise, you will create a form with a subform, as well as a quick form that counts salaried employees and totals and averages the salaries.

1. Open **A6-A1-UCE** from your **Access Chapter 6** folder and save it as: **A6-A1-UCERev**
2. Select the **Venues** table and start the **Form Wizard**.
3. Move all fields from the Venues table to the Selected Fields list except VenueLiaison.
4. Choose the **Schedules** table in the Tables/Queries list.
5. Move all fields from the Schedules table to the Selected Fields list except Schedules.VenueID.
6. Accept viewing your data By Venues and the Form with Subform(s) option.
7. Choose **Datasheet** as the layout option.
8. Use **Venue Events** for the main form's name and **Venue Events Subform** for the subform's name.
9. Switch to **Layout View** and then modify the subform layout as described:
   - Remove the Venue Events label.
   - Widen the frame enough so all columns are visible.
   - Autofit all columns.
   - Reduce the frame width until it is just wide enough to contain the datasheet.
   - Nudge the frame to align it with the text boxes on the main form.
10. Close the form, saving the changes to the form and subform.
11. Choose **Salaried Personnel Query** in the Navigation pane and start the **Form Wizard**.
12. Add all fields to the Selected Fields list, choose the **Datasheet** layout, and name the new form: **Salaried Personnel Totals**
13. Switch to **Design View** and then select the **Salary** label and the **SalaryAmt** text box controls.
14. Press Ctrl+C and then Ctrl+V to copy and paste a duplicate salary field under the existing label and text box.

    You may need to drag the Form Footer section down slightly to make room for the salary field.
15. Change the name of the first Salary label to: **Total Salaries**
16. Change the name of the second Salary label to: **Average Salary**
17. Switch to **Datasheet View** and autofit all columns.
18. Add a **Totals** row and insert these functions:
    - **Count** in the Salaried field
    - **Sum** in the Total Salaries field
    - **Average** in the Average Salary field

    Notice the Count function displays the 11 records where the box in the Salaried field was checked.

19. Uncheck several **Salaried** boxes and notice that the function updates the changes.
20. Close the database, saving the changes to any unsaved forms.

APPLY YOUR SKILLS: A6-A2

Add a Calculated Control to a Form and Change the Form's Layout

In this exercise, you will add a calculated control to a main form by first copying an existing field, which will retain the formatting of the existing field. You will also adjust the size and position of the controls to change the form's appearance.

1. Open **A6-A2-UCE** from your **Access Chapter 6** folder and save it as: **A6-A2-UCERev**
2. Open the **Event Pricing Entry** form in **Form View**.

   Notice that the labels have a raised effect. You will create a calculated control while retaining the formatting of these fields by copying and pasting an existing field.
3. Switch to **Design View** and then select the **Chg/PP** label and the **ChgPP** text box.

   In the next step, you may need to drag the Form Footer bar down slightly to make room for the new controls.
4. Press Ctrl+C to copy the controls and Ctrl+V to paste them.

   An identical text box and label appear below the existing text box and label.
5. Change the *Chg/PP* label text for the new label to: **Total**
6. On the **Data** tab of the Property Sheet, change the Control Source property of the new text box to: **=Guests*ChgPP**
7. Change the Width property of all eight control labels to **1** and the Left property to: **1.5**
8. Switch to **Form View** and navigate through several records to verify the calculated control is working and the text boxes are wide enough to accommodate all records.
9. Close the database, saving the changes to the form.

APPLY YOUR SKILLS: A6-A3

Create a Pop-Up Form

In this exercise, you will create a pop-up form to help facilitate data entry.

1. Open **A6-A3-UCE** from your **Access Chapter 6** folder and save it as: **A6-A3-UCERev**
2. Select the **Events** table and launch the **Form Wizard**.
3. Move both the **EventID** and **EventName** fields to the Selected Fields list, choose the **Tabular** layout, and name the form: **Events Pop Up**
4. Switch to **Design View** and delete the *Events Pop Up* title in the Form Header section.
5. Set the Top property to **0.1** for the **Event ID** label and the **Event Name** text box in the Form Header.
6. Set the Form Header Height property to: **0.4**
7. Choose **Form** from the Selection Type drop-down list at the top of the Property Sheet.
8. On the **Other** tab, set the Pop Up property to **Yes**.

9. Switch to **Form View** and display the **Venue Events** form in **Form View**.
10. Move the pop-up form so you can view the data in the Venue Events form and subform.

    You can see what each Event ID on the Venue Events subform means by looking on the Events Pop Up form.
11. Close the database, saving the changes to the form.

Project Grader

PROJECT GRADER: A6-P1

Taylor Games: Creating and Customizing an Input Order Form

Taylor Games is getting ready to take customer orders, but before it can begin, its service reps need a new order form. In this exercise, you will create a form that contains a subform and modify the design for visual appeal. Then you will lock a form field and add a calculated control.

1. Download and open your Project Grader starting file.
   - *Using eLab:* Download **A6_P1_eStart** from the Assignments page. You *must* start with this file or your work cannot be automatically graded.
   - *Not using eLab:* Open **A6_P1_Start** from your **Access Chapter 6** folder.
2. Use the Form Wizard and these guidelines to create a new form and subform:
   - Add all available fields from the **Orders** table.
   - Add (in order) **Line Item** and **Quantity** from the **Order Details** table.
   - Add the **Price** field from the **Inventory** table.
   - View the data **by Orders** as a **Form with Subform(s)**.
   - Use the **Datasheet** layout.
   - Use **`Order Form`** as the form title and **`Order Subform`** as the subform title.
   - You will modify the form's design in the next steps.
3. Set the Width property of the Order_ID text box to: **`0.5`**
4. Set the Width property of the Service_Rep text box to: **`1.05`**
5. Set the ControlTip Text property of the Customer_State text box to: **Use two letter abbreviation**
6. Set these properties for the **Order Subform_Label** (contains the text *Order*):

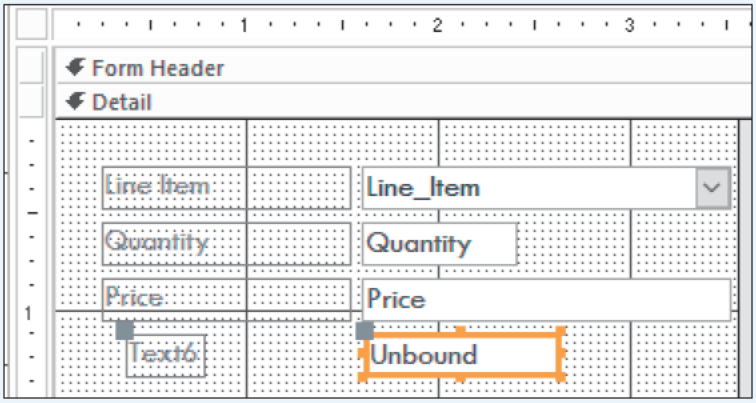
Property	Value
Top	`2.85`
Left	`0.25`

7. Set these properties for the **Order Subform**:

Property	Value
Height	`2.5`
Top	`3.1`
Left	`0.25`

8. In the Order Subform, set the Locked property of the Price control to **Yes**.

9. In the Order Subform, insert an unbound text box control directly below the Price control.



10. Set these properties for the unbound text box control:

Property	Value
Name	**LineTotal**
Format	**Currency**

11. Create a formula in the unbound text box control that multiplies the Quantity field by the Price field.

12. Set these properties for the Unbound label control (contains the text *Text6*):

Property	Value
Name	**LineTotal**
Caption	**Line Total**
Width	**1**
Left	**0.25**

13. In the order form, use these guidelines to create a new Title control:
    - Delete the current Title control from the Form Header (contains the title *Order Form*).
    - Insert a new Title control.
    - Set these properties for the new Title control:

Property	Value
Width	**2**
Height	**0.5**
Font Size	**26**
Text Align	**Center**
Font Weight	**Bold**

14. Insert a Logo control in the Form Header and set these properties for it:

Property	Value
Picture	Insert the **Taylor Games Logo.png** picture from your **Access Chapter 6** folder.
Width	**1**
Height	**0.6**

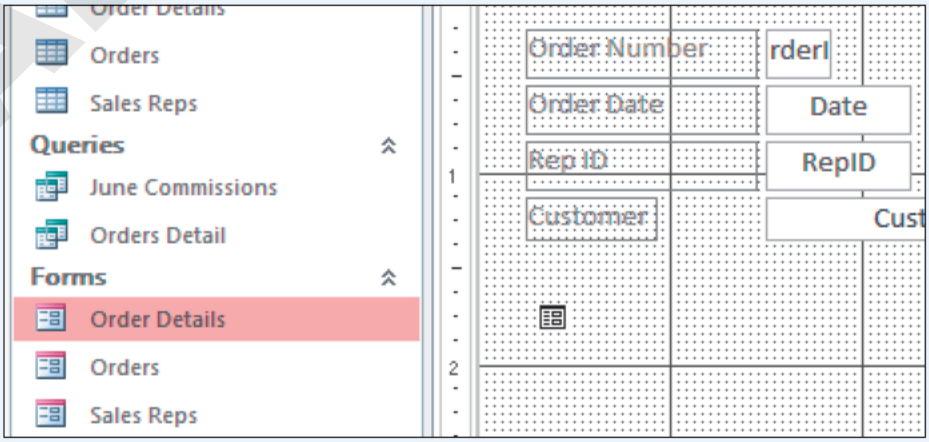
15. Set the Back Color property for the FormHeader section to: **Background Light Header**
16. Save your database.
    - *Using eLab:* Save it to your **Access Chapter 6** folder as **A6 _ P1 _ eSubmission** and attach the file to your eLab assignment for grading.
    - *Not using eLab:* Save it to your **Access Chapter 6** folder as: **A6 _ P1 _ Submission**

PROJECT GRADER: A6-P2

WebVision: Create and Add a Subform

WebVision has improved its database normalization and would like to capitalize on these changes in the current order form. In this exercise, you will start by adding a calculated field to an existing query. Then, you will create a datasheet form, add a total row, and add the datasheet form as a subform to the existing order form. You will then disable form fields and create a pop-up form. Last, you will enter a new order using the form you created.

1. Download and open your Project Grader starting file.
    - *Using eLab:* Download **A6_P2_eStart** from the Assignments page. You *must* start with this file or your work cannot be automatically graded.
    - *Not using eLab:* Open **A6_P2_Start** from your **Access Chapter 6** folder.
2. In the **OrderDetails Query**, add a calculated field named **Line Total** that multiplies the Quantity field by the Cost field.
3. Create a datasheet form that is based on the OrderDetails Query.
4. Add a Totals row to the new form and then SUM the data in the Line Total column.
5. Save the form with the name: **OrderDetails Subform**
6. Insert the OrderDetails Subform into the Orders form (by dragging it from the Navigation pane into the detail section).

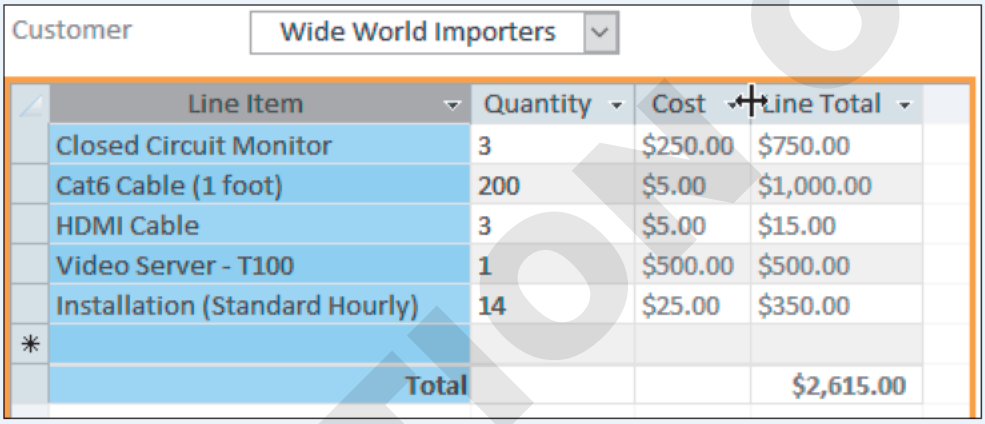


7. Delete the subform label (named *OrderDetails Subform*).

8. Set the OrderDetails Subform properties using the following guidelines:

Property	Value
Link Master Fields	`OrderID`
Link Child Fields	`OrderID`
Width	`5`
Height	`3`
Top	`1.5`
Left	`0.25`

9. Delete the Order ID textbox and label controls from the subform.
10. Disable the Cost and Line Total text box controls by setting the Enabled property to **No**.
11. In **Layout View**, resize the columns for each of the subform columns so all field data is displayed.



12. Set these properties in the **Sales Reps** form:

Property	Value
Pop Up	**Yes**
Modal	**No**

13. Close all open objects, saving any changes, and then save your database.
    - *Using eLab:* Save it to your **Access Chapter 6** folder as **`A6 _ P2 _ eSubmission`** and attach the file to your eLab assignment for grading.
    - *Not using eLab:* Save it to your **Access Chapter 6** folder as: **`A6 _ P2 _ Submission`**

Extend Your Skills

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

A6-E1 That's the Way I See It

You want to create a form with a subform in the Blue Jean Landscaping database. Open **A6-E1-BJL**. Save it as **A6-E1-BJLRev** and start the Form Wizard. Create a Blue Jean Landscaping Customer Sales form with a CustSales Details subform in datasheet layout that includes:

- SalesNum and SalesDate from the MerchSales table
- CustLastName from the Customers table
- ItemName, Manufacturer, and Price from the StoreMerchandise table
- QtySold from the MerchSalesDetails table

Enhance the appearance of the forms and add any features you feel will help facilitate effective data entry.

A6-E2 Be Your Own Boss

Blue Jean Landscaping would like you to add calculated controls to the newly created Sales Invoices form so it may show order totals. They have also asked that several field properties be modified to assist and control data entry. Open **A6-E2-BJL** and save it as: **A6-E2-BJLRev**

Open the SalesInvoices form and add a calculated control that multiplies Cost by Qty Sold to produce a line total. Disable the Invoice number field, lock the Cost field, and add a control tip that will notify the user they cannot change the cost. Enhance the appearance of the forms and add any features you feel will help facilitate effective data entry.

A6-E3 Demonstrate Proficiency

Stormy BBQ is continuing to update its database and now wants to have a more exact record of its merchandising sales. Open **A6-E3-SBQ**. Save it as **A6-E3-SBQRev** and then create a new form using the default form and subform names in datasheet layout that includes:

- SalesID and SalesDate from the MerchSales table
- SKU from the MerchSalesDetails table
- Manufacturer, ItemName, and ListPrice from the Merchandise table
- QtySold from the MerchSalesDetails table

In the subform, add a calculated control that multiplies ListPrice by QtySold to produce a line total. Create a pop-up tabular form using the Merchandise table that displays SKU, ItemName, and ListPrice. Enhance the appearance of the forms as desired.

ACCESS

7 Creating Complex Queries

As your database grows, so will the need to quickly retrieve and modify exact data. Complex queries help with this because they allow you to further refine search results and perform actions that modify records. 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. You will also 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

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 both data entry and updates, as well as to better analyze data. The tools you will use include crosstab queries for data analysis, parameter queries that will 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 (rows) of the datasheet, and it arranges the fields to be summarized across the top (columns) so you can calculate sums, averages, counts, or totals by both 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.

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

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 crosstab queries. You can create a crosstab query while working with an existing query in Design View using the Crosstab option; alternatively, use the Query Wizard.

Design→Query Type→Crosstab

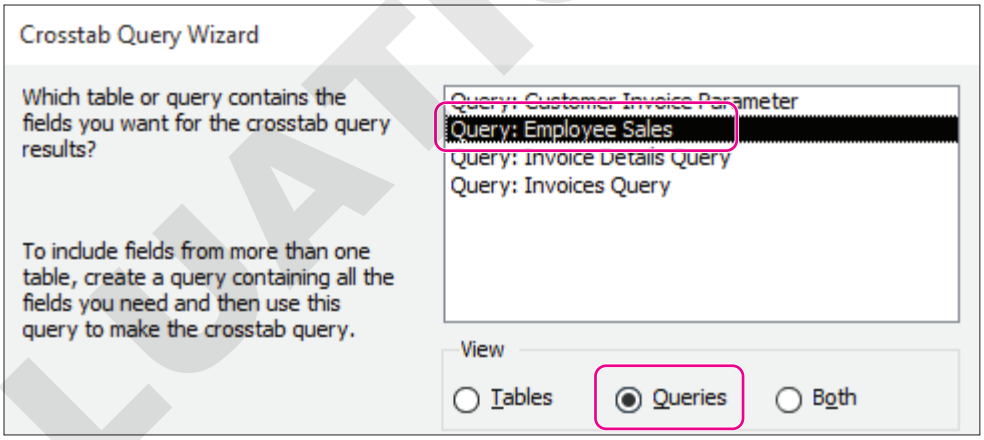
Watch the video "Creating Crosstab Queries."

DEVELOP YOUR SKILLS: A7-D1

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

1. Open **A7-D1-WinDesign** from your **Access Chapter 7** folder and save it as: **A7-D1-WinDesignRev**
2. Double-click the **Employee Sales** query to run it and display the resulting datasheet.

   The query contains line item sales data. Each employee has multiple transactions, and each transaction contains the product description, price, and quantity. The LineTotal field is a calculated field that multiplies price by quantity. You will use this query as the basis for your crosstab query.
3. Close the Employee Sales query.
4. Choose **Create→Queries→Query Wizard**.
5. Choose **Crosstab Query Wizard** and click **OK**.
6. Choose the **Queries** view option and 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.

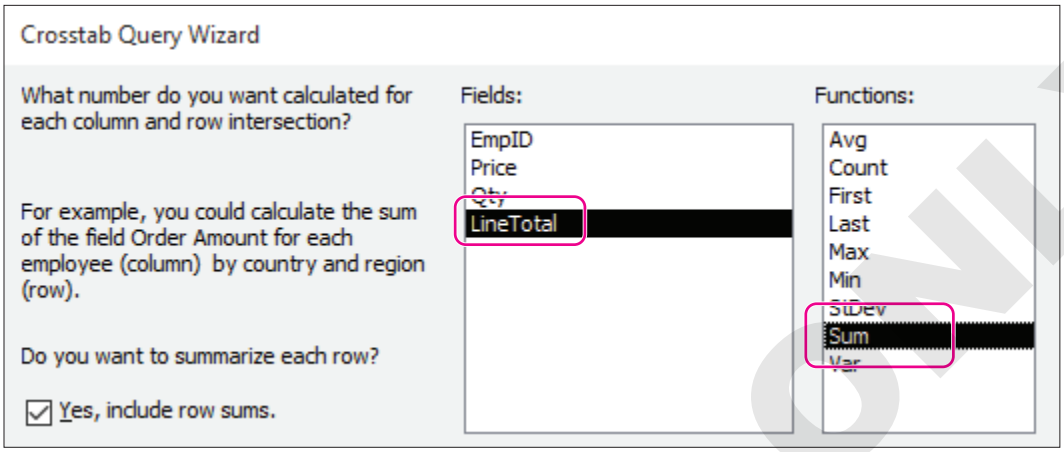
   Your crosstab query will display employee last names as row headings in the query results datasheet. Each employee will have a single row, with their last name displayed in the first cell of the row and their sales information displayed in the other row cells.
9. Click **Next** and choose **ProdDescription** for the column headings.

   The various product descriptions (Blog, Home Page, Web Page) will appear as column headings.

Tip! *The sample query at the bottom of the Wizard can be a useful guide for deciding which fields to use as the row and column headings.*

10. Click **Next**, choose **LineTotal** from the Fields list, and then choose **Sum** from the Functions list.

    The crosstab query will examine all transactions in the underlying Employee Sales query and sum the line totals for each product description. For example, a grand total of all line totals will be created where the product description is Blog and the employee is Kramer.



11. Leave the **Yes, Include Row Sums** option checked and click **Next**.

    This option creates one additional column in the datasheet to hold a total for each employee. The total will be the sum of all cells in the crosstab query datasheet for that employee.

12. Leave the default query name as *Employee Sales_Crosstab* and click **Finish**.

13. Take a moment to examine the query results.

Last Name	Total Of Lin	Blog	Home Page	Hourly Billir	Image	Shopping Ca	Web Page
Kramer	$13,680.00	$600.00	$800.00	$2,160.00	$2,520.00		$7,600.00
Mansfield	$10,520.00	$600.00	$400.00	$1,840.00	$1,680.00	$1,200.00	$4,800.00
Waters	$20,120.00	$1,200.00	$1,600.00	$4,000.00	$2,120.00	$1,200.00	$10,000.00
Winchester	$16,940.00	$300.00	$2,000.00	$2,160.00	$2,880.00	$800.00	$8,800.00

    Complete the Query Wizard steps again, if necessary, so you fully understand how the query options produce the resulting datasheet.

14. Close the query.

Find Queries

Database tables often contain common fields that link, or relate, the tables. For example, the Product ID field from a Products table also appears in an Invoices table, so invoices can show only existing products. Thus, it's important that records entered in one table have a matching record in the related table. That is, an invoice should never show a product that is not in the Products table.

Sometimes databases are poorly designed and incorrect data is allowed. For example, a user may enter a product that does not exist into a new invoice. Data is also sometimes imported from other data sources, which might result 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 matching records in another table. For example, you could create a find unmatched query to ensure 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: A7-D2

In this exercise, you will create a query to locate records in the Customers table that do not have a matching customer ID in the Invoices table. Then you will 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.

   When the query is created, records from the Customers table will appear in the query results.
3. Choose the **Invoices** table and click **Next**.

   You will set up the query to find records in the Invoices table that do not have a matching customer record in the Customers table. Notice that the CustID fields are chosen as the matching fields in the Wizard screen. They're automatically chosen because a one-to-many relationship is already set up between these fields in the Customers and Invoices tables.
4. Click **Next** to accept CustID as the matching field.
5. Add **CustLastName**, **CustFirstName**, **CustPhone**, and **CustEmail** to the Selected Fields list.

   These are the fields from the Customers table that will be displayed in the query results, creating a contact list of customers who don't have invoices in the system. In other words, these are customers who haven't purchased anything in a while, making them potential prospects for new sales.
6. Click **Next** and then click **Finish** to accept the default query name *Customers Without Matching Invoices*.

   Your query should produce a data set of three records only.

Customers Without Matching Invoices

Last Name	First Name	Telephone	Email
Abrams	John	(941) 555-9902	JPAbrams@email.com
Fleetwood	Candace	(941) 555-9256	CandyWin@email.com
Winkler	Samuel	(941) 555-2054	SamWinkler45@email.com

7. Close the datasheet after reviewing it.

Create a Find Duplicates Query

8. Choose **Create→Queries→Query Wizard**, choose **Find Duplicates Query Wizard**, and click **OK**.

9. Click **Next** to choose the Customers table as the query to search for duplicate field values.
10. Add **CustLastName** to the Duplicate Value Fields list and click **Next**.

    You are looking only for the records of customers who have the same last name.
11. Add **CustFirstName** and **CustPhone** to the Additional Query Fields list and click **Next**.

    These fields will appear in the resulting datasheet but aren't used in the duplicate values test. Only CustLastName is being checked for duplicates.
12. Name the query **`Customers with the Same Last Name`** and click **Finish**.

    The query results show just two customers with the same last name (Roberts).

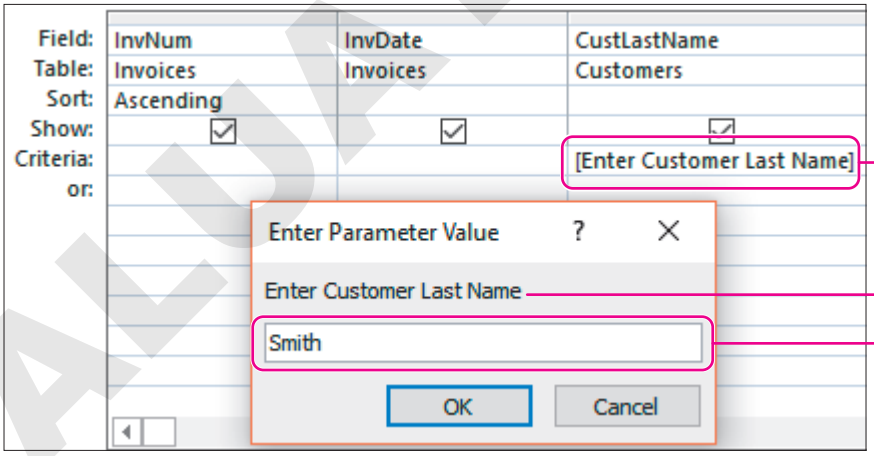
Customers with the Same Last Name

Last Name	First Name	Telephone
Roberts	John	(941) 555-7820
Roberts	Ilsa	(941) 555-7821

13. Close the query after reviewing the results.

Parameter Queries

A parameter query is a select query that prompts users 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.



Surround the parameter query prompt text in square brackets in a Criteria cell.

The prompt text appears here when the query is run.

The user enters the desired parameter value here.

Customer Invoice Parameter

InvNum	Invoice Date	Last Nam	Description	Price	Qty	LineTotal
1	3 /14/2017	Smith	Home Page	$400	1	$400
1	3 /14/2017	Smith	Web Page	$200	6	$1,200
1	3 /14/2017	Smith	Image	$40	11	$440

In this example, only records in which the customer last name is *Smith* are returned.

Complex Parameter Queries

Suppose you want to see all items purchased by a particular customer and those equal to or greater than a particular price; for example, all items purchased by Smith with a price greater than or equal to $300. You can do this by creating an AND condition using parameters in the CustLastName 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:					

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 (=).

EXAMPLES OF PARAMETER QUERY CRITERIA FOR A SINGLE FIELD

Parameter Criteria	Result
Between [What is the start date?] And [What is the end date?]	These criteria prompt the user to enter start and end dates. Access recognizes the Between and And expressions and returns dates within the range entered.
>=[Enter minimum price]	This displays the prompt *Enter minimum price* and returns only records greater than or equal to the price entered.

DEVELOP YOUR SKILLS: A7-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 Last Name]`**

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

3. Run the query, type **`Roberts`** in the parameter box that appears, and click **OK**.

   All line items from invoices 34 and 7, where the customer last name is Roberts, *are returned.*

Create an AND Parameter Condition

4. Switch to **Design View** and enter this parameter in the Price criteria box: **`>=[Enter Minimum Price]`**

5. Run the query and enter **Roberts** in the first parameter box and **300** in the second.

   Now the record set has records only with Roberts *in the Last Name field in which the price is greater than or equal to 300.*

Customer Invoice Parameter

InvNum	Invoice Date	Last Nam	Description	Price	Qty	LineTotal
34	8 /4 /2018	Roberts	Home Page	$400	1	$400
34	8 /4 /2018	Roberts	Blog	$300	1	$300
7	7 /10/2017	Roberts	Home Page	$400	1	$400
7	7 /10/2017	Roberts	Shopping Cart	$400	1	$400

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. They 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% 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 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. It's good practice *to not* save action queries after running them or to delete action queries if they are saved. 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.

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 to archive them when they become obsolete, such as when a product is no longer available. When you rerun a make table query, Access will replace the table created with the previous running of the query. To retain the previously created table, you must first rename it; this way it won't be replaced.

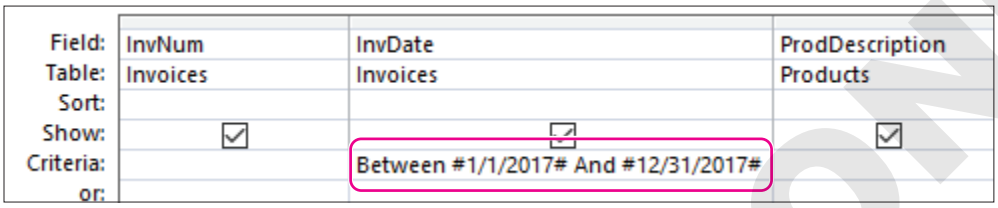
Design→Query Type→Make Table

DEVELOP YOUR SKILLS: A7-D4

In this exercise, you will create a make table action query to save all of the 2017 invoice records in a new table.

1. Open **Invoices Query** in **Design View**.
2. Enter **`Between 1/1/2017 And 12/31/2017`** as the criterion for the **InvDate** field.

   This criterion will produce a datasheet with old invoices no longer needed in the database. You'll then use the make table query feature to move the records to a new table.



3. Choose **Query Tools→Design→Query Type→Make Table**.
4. Enter **`2017 Invoices`** as the table name and click **OK**.
5. **Run** the query and choose **Yes** to paste 61 rows into a new table.

   The new table named 2017 Invoices appears at the top of the Tables section in the Navigation pane.
6. Open the new **2017 Invoices** table in **Datasheet View**.

   Notice that all of the line items listed have a 2017 invoice date.
7. Close the 2017 Invoices table and then close the invoices query without saving the changes.

   It's important not to save the query because it's used for other purposes and because you want to preserve the new table without risking an overwrite of it.
8. Open the **Invoices** table in **Datasheet View**.

   The Invoices table still contains the 2017 records.

Note! *Make table queries don't remove data from underlying tables; they simply copy the data to new tables.*

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: A7-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 six records.
2. Open the **NewProducts** table to see the five records that will be appended to the Products table.

   The tables have the same field structure, which includes field order and matching field names and field data types.
3. Close both tables and then choose **Create→Queries→Query Design** to create a new query.
4. Add the **New Products** table to the query window and then close the Show Table dialog box.
5. Add all fields from the New Products table to the query grid in the same order they appear in the New Products list.
6. Choose **Query Tools→Design→Query Type→Append**.
7. Click the **Table Name menu** button, choose **Products**, and click **OK**.

   An Append To row is added to the query. When you run the query, it will copy all records from the underlying New Products table to the Products table. The tables have identical field structures, so the data will drop right into the existing table.

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

8. **Run** the query and choose **Yes** to append the five rows to the Products table.

   Nothing appears to happen when you run the query—but don't run it again! You will see the changes only after you open the destination table to which the records were appended. If you do run the query again, Access will add the same records to the destination table again, creating duplicate data.
9. Display the **Products** table in **Datasheet View** to verify that the new records were appended.

   The Products table should now contain 11 records.
10. Close the Products table and then 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 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: A7-D6

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

1. Open the **Products** table in **Datasheet View** and notice the Home Page price is $400.

   The update query will increase this and all other prices by 10%.

2. Close the Products table and then choose **Create→Queries→Query Design**.
3. Add the **Products** table to the query window and then close the Show Table dialog box.
4. Add all fields from the Products table to the query grid in the same order they appear in the Products list.
5. Choose **Query Tools→Design→Query Type→Update**.

   An Update To row is added to the query grid.

6. Click in the **Update To** cell for the Price field and enter: **`[Price]*1.1`**

   Be sure to include the square brackets, [ ], when entering this formula so Access will recognize Price as a field. Multiplying by 1.1 increases the price by 10%.

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

7. **Run** the query and choose **Yes** when the warning prompt appears to update 11 rows.
8. Close the query without saving it.

   Once again, it's good practice not to save action queries, such as update queries. Running a query by accident can corrupt data and recovering corrupted data is often difficult or impossible to do.

9. Open the **Products** table in **Datasheet View** and notice the Home Page price went from $400 to $440 (an increase of 10%).
10. Close the Products table.

Delete Queries

A delete query removes 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 corresponding records in related tables will 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: A7-D7

In this exercise, you will create a delete query to remove the 2017 invoices from the Invoices table.

1. Open the **Invoices** table in **Datasheet View**.

   The table still has invoices dated from 2017. These invoices were copied to the 2017 Invoices table using a make table query in a previous exercise. Because make table queries do not delete records, you'll take care of this using a delete query.

2. Close the Invoices table and then choose **Create→Queries→Query Design**.
3. Add the **Invoices** table to the query window and then close the Show Table dialog box.
4. Add only the **InvDate** field to the query grid.
5. Enter **`Between 1/1/2017 And 12/31/2017`** as the criterion for the InvDate field.

   Access may view typing 2017 as an attempt to insert the 2017 Invoices table. To avoid adding this table to the criteria, tap Spacebar at the end of the criteria and then press Enter.

6. Choose **Query Tools→Design→Query Type→Delete**.

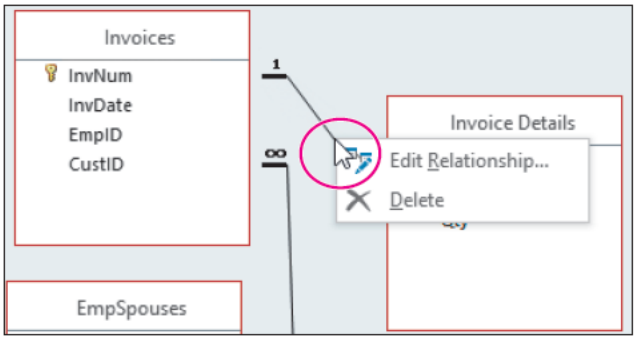
   A Delete row is added to the query grid.

7. **Run** the query and choose **Yes** when the prompt to delete 19 rows appears.

   A second warning appears, notifying you that the records cannot be deleted because there is a key violation. This is because the Cascade Deleted Records option is not activated for a relationship between the Invoices and Invoice Details table. This option must be activated for you to run delete queries.

8. Choose **No** in the warning message box and then choose **Database Tools→Relationships→Relationships**.

9. Right-click the join line between the Invoices and Invoice Details table and choose **Edit Relationship**.



10. Check the **Cascade Delete Related Records** box and click **OK**.
11. Close the Relationships window, saving the changes to the relationship layout if prompted.
12. **Run** ! the query again, choosing **Yes** when the warning prompt appears.

    This time, the delete query runs, removing the 2017 records from the Invoices table.
13. Close the query without saving it.

    Remember, it's good practice not to save action queries, especially when they are relatively easy to re-create, as in this example.
14. Open the **Invoices** table in **Datasheet View** and notice that the 2017 invoices have been removed.
15. Close the Invoices table and then close the database.

Self-Assessment

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

Reinforce Your Skills

REINFORCE YOUR SKILLS: A7-R1

Create Crosstab and Find Queries

Kids for Change is fine-tuning its database. In this exercise, you will create a crosstab query to track donations and find queries to locate problem records.

1. Open the **A7-R1-K4C** database from your **Access Chapter 7** folder and save it as: **A7-R1-K4CRev**
2. Choose **Create→Queries→Query Wizard**, choose **Crosstab Query Wizard** in the first screen, and click **OK**.
3. Choose the **Queries** view, choose **Donations Query** as the query that contains the fields you want in the results, and click **Next**.
4. In the next Wizard screen, move **DonorLName** to the Selected Fields list and click **Next**.

   Donor last names will become your row headings and the field where the results are grouped.
5. In the next Wizard screen, choose only **DonationDate** as the field to appear in the column headings and click **Next**.

   Because DonationDate is a date field, the Wizard asks you to choose an interval, such as day, month, or year.
6. Choose **Month** as the interval and click **Next**.
7. Choose **Amount** in the Fields list and **Sum** in the Functions list to identify the field that contains values and the function you want to use.
8. Click **Next**, leave the query name unchanged, and then finish the query.

   The query returns the total donations for each donor, organized by month.
9. Close the query.

Create a Find Unmatched Records Query

10. Launch the **Query Wizard**, choose **Find Unmatched Query Wizard**, and click **OK**.
11. Choose the **Activities** table and click **Next**.

    The Activities table will display in the query results.
12. Choose **Volunteers** as the table with related records and click **Next**.
13. Choose **Day** in the Activities table field list and **VolDay** in the Volunteers table field list and then click **Next**.
14. Add **Activity**, **Day**, and **MeetTime** to the Selected fields list and then click **Next**.
15. Click **Finish** to accept the default query name.

    Your query should return four records of activities that do not have a matching volunteer assigned to them.
16. Close the query when you have finished viewing the results.

Create a Find Duplicates Query

17. Launch the **Query Wizard**, choose **Find Duplicates Query Wizard**, and click **OK**.
18. Choose **Donors** as the table to check for duplicates and click **Next**.
19. Add **DonorLName** to the Duplicate-Value Fields list and click **Next**.

    You are looking only for records of donors with the same last name.
20. Add **DonorFName** and **DonorPhone** to the Additional Query Fields list and click **Next**.
21. Accept *Find Duplicates for Donors* as the default query name and click **Finish**.

    The query should return records for Clay Boltwood and Nancy Boltwood.
22. Close the database, saving the changes to any unsaved queries.

REINFORCE YOUR SKILLS: A7-R2

Create a Parameter Query

In this exercise, you will make a copy of an existing query. You will modify the copied query, turning it into a parameter query to return donor records by state.

1. Open the **A7-R2-K4C** database from your **Access Chapter 7** folder and save it as: **A7-R2-K4CRev**
2. Click **Donations Query** in the Navigation pane to select it.
3. Press Ctrl+C to copy and Ctrl+V to paste the copy.
4. Enter **Donations by State** as the new query name and click **OK**.
5. Display the **Donations by State** query in **Design View**.
6. Type **[Enter State Abbreviation]** in the Criteria cell of the DonorST field and tap Enter to complete the entry.

Field:	DonorID	DonorLName	DonorFName	DonorStreet	DonorCity	DonorST
Table:	Donations	Donors	Donors	Donors	Donors	Donors
Sort:	Ascending					
Show:	☑	☑	☑	☑	☑	☑
Criteria:						[Enter State Abbreviation]
or:						

7. **Run** ! the query.
8. Type **MA** in the Parameter Value prompt box and click **OK**.

   Only donations from Massachusetts donors are returned.
9. Close the query, saving the changes.
10. Choose **File→Close** to close the database; if you see a message regarding emptying the Clipboard, click **Yes**.

REINFORCE YOUR SKILLS: A7-R3

Create Action Queries

Kids for Change is improving and updating its records. In this exercise, you will create a make table query that produces a table to archive the 2017 donation records, a query that appends new records to the Children table, an update query that reduces the duration of each activity by half, and a query that deletes old donations from the Donations table.

1. Open the **A7-R3-K4C** database from your **Access Chapter 7** folder and save it as: **`A7-R3-K4CRev`**

   Remember, you must enable a database if you will be running action queries, so click the Enable Content button that appears after saving the file.

   To begin, you will create the make table query.
2. Open **Donations Query** and switch to **Design View**.
3. Scroll the query grid to the right and type **`Between 1/1/2017 And 12/31/2017`** in the Criteria cell of the DonationDate field.
4. Choose **Query Tools→Design→Query Type→Make Table**.
5. Type **`2017 Donations`** as the name of the new table and click **OK**.
6. **Run** the query and choose **Yes** in the warning box.
7. Close the Donations Query without saving the changes.

Create an Append Query

8. Open the **Children** table and notice that it contains 17 records; close the table.
9. Choose **Create→Queries→Query Design** to create a new query.
10. Add the **NewChildren** table to the query window and then close the Show Table dialog box.
11. Add all fields from the NewChildren table to the query grid in the same order they appear in the NewChildren list.
12. Choose **Query Tools→Design→Query Type→Append**.
13. Click the **Table Name menu** button ▾, choose **Children**, and click **OK**.

    The Append To row is added to the query grid.
14. **Run** the query and choose **Yes** to add the 10 rows to the Children table.
15. Close the query, saving it as: **`Append New Children`**

Create an Update Query

16. Display the **Activities** table in **Datasheet View** to see the current Hrs values (*2 or 4 hours*) and then close the table.
17. Choose **Create→Queries→Query Design**.
18. Add the **Activities** table and close the Show Table dialog box.
19. Add the **Hours** field to the query grid.
20. Choose **Query Tools→Design→Query Type→Update**.

    An Update To row is added to the query.

21. Type **`[Hours]/2`** in the Update To cell of the Hours field.

    It's important to include the square brackets so that Access can perform the correct calculation. This calculation will divide the current activity hours value in half.

22. **Run** ! the query and choose **Yes** to update 25 rows; close the query without saving it.
23. Open the **Activities** table and note the activities that were listed as 2 and 4 hours each are now 1 and 2 hours; close the table.

Create a Delete Query

24. Open the **Donations** table and notice it contains donations from the year 2017; close the Donations table.
25. Choose **Create→Queries→Query Design**  .
26. Add the **Donations** table to the query and then close the Show Table dialog box.
27. Double-click the **DonationDate** field to add it to the query grid.
28. Type **`Between 1/1/2017 And 12/31/2017`** in the Criteria row for the DonationDate field.
29. Choose **Query Tools→Design→Query Type→Delete**  .

    A Delete row is added to the query grid.

30. **Run** ! the query and choose **Yes** in the warning box.
31. Close the query without saving it and then open the **Donations** table and confirm that the 2017 records have been removed.
32. Close the database.

Apply Your Skills

APPLY YOUR SKILLS: A7-A1

Create Crosstab and Find Queries

Universal Corporate Events has asked you to create queries to analyze the company's data and identify unmatched and duplicate database records. In this exercise, you will respond to this request by creating crosstab, find unmatched records, and find duplicate records queries.

1. Open the **A7-A1-UCE** database from your **Access Chapter 7** folder and save it as: **A7-A1-UCERev**
2. Use the **Query Wizard** to create a crosstab query using these parameters:

View	**Query: Event Revenue**
Row Heading(s)	**VenueID**
Column Heading(s)	**ContactID**
Field(s)	**TotalRev**
Function(s)	**Sum**
Name	**Contact Revenue by Venue**

3. Finish the query.

   Seven data rows should be returned.
4. Close the query and then use the **Query Wizard** to create a find unmatched query using these parameters:

View	**Table: Venues**
Related Records	**Table: Schedules**
Fields in Venues	**VenueID**
Fields in Schedules	**VenueID**
Fields to see in query results	**VenueName, VenueStreet, VenueCity, VenueST, VenueZIP, VenuePhone, VenueWebSite**
Name	**Venues Without Event Scheduled**

5. Finish the query.

   Three data rows should be returned.
6. Close the query and then use the **Query Wizard** to create a find duplicates query using these parameters:

View	**Query: Event List**
Duplicate-Value Field	**EventDate**
Additional Fields	**VenueID, ContactID, MenuPlan, Guests**
Name	**Find Double-Booked Dates**

7. Finish the query.

   Two data rows should be returned.
8. Close the database.

APPLY YOUR SKILLS: A7-A2

Create a Parameter Query

In this exercise, you will create a parameter query to return personnel records by city.

1. Open the **A7-A2-UCE** database from your **Access Chapter 7** folder and save it as: **A7-A2-UCERev**
2. Create a new query using **Query Design** and add the **Personnel** table to the query.
3. Add the **PerLastName**, **PerFirstName**, **PerAddr**, **PerCity**, **PerPhone**, and **PerEmail** fields to the query.
4. Make **[Enter City]** a criterion for the PerCity field.
5. Run the query using **Sarasota** as the parameter value.

   The query should return five records in which the city is Sarasota.
6. Close the database, saving the query as: **Personnel City**

APPLY YOUR SKILLS: A7-A3

Create Action Queries

Universal Corporate Events is updating and consolidating its events data. In this exercise, you will create a make table query to archive the records for older events, an append query to add new records to the Schedules table, and an update query to change personnel salaries. Finally, you will create a query to delete older events from the main Schedules table.

1. Open the **A7-A3-UCE** database from your **Access Chapter 7** folder and save it as: **A7-A3-UCERev**

   You will start by creating a make table query.
2. Display **Schedules Query** in **Design View** and add the criterion **<01/01/2019** to the EventDate field.
3. Use the **Make Table** query type to create a new table with the name: **Older Events**
4. Run the query.

   The new table should contain six records.
5. Close the Schedules Query without saving the changes.

Create an Append Query

6. Create a new query, adding all fields from the **New Schedules** table.
7. Use the **Append** query type to convert the query to an append query using **Schedules** as the table name to append to.
8. Run the query, choosing **Yes** when asked if you wish to add the 30 rows to the Schedules table.
9. Close the query, saving it as: **Append Schedules**
10. Open the **Schedules** table to verify it contains 72 records; close the table.

Create an Update Query

11. Create a new query using the **SalaryGrades** table and adding only **SalaryAmt** to the query grid.
12. Use the **Update** query type to add an Update row to the query.
13. Use an update criterion that multiplies the SalaryAmt field by: **`1.07`**

    This will produce a 7% increase to the numbers in the SalaryAmt field.
14. Run the query, choosing **Yes** to update 21 records.
15. Close the query, saving it as: **`Salary Updates`**
16. Close the SalaryGrades table.

Create a Delete Query

17. Create a new query using only the **EventDate** field from the Schedules table.
18. Use the criterion **`<1/1/2019`** in the EventDate field.
19. Use the **Delete** query type and then run the query, choosing **Yes** to delete six records.
20. Close the new query without saving it.
21. Open the **Schedules** table to verify all records with an event date prior to 1/1/2019 have been deleted and then close the database.

Project Grader

PROJECT GRADER: A7-P1

Taylor Games: Updating Inventory

Taylor Games has new inventory it would like to add to its database. It has noticed that the selling prices for some of the new items are above or below the desired profit margin and need to be updated. In this exercise, you will start by using an append query to merge data with new inventory into the inventory table. Then, you will create a calculated field and a complex parameter query with criteria to identify which items are below and above the desired margin and make a new table using the results. Last, you will make an update query and modify the sales price so all items are within the desired margin.

1. Download and open your Project Grader starting file.
   - *Using eLab:* Download **A7_P1_eStart** from the Assignments page. You *must* start with this file or your work cannot be automatically graded.
   - *Not using eLab:* Open **A7_P1_Start** from your **Access Chapter 7** folder.
2. Use these guidelines to create an Append query:
   - Create a new query adding all fields from the **New Inventory** table.
   - Append to the **Inventory** table in the current database.
   - Run the query to complete the append action, then save the query with the name: **`Inventory Append`**
3. Create a new query named **`Margin Parameter`** that uses all fields from the Inventory table.
4. In the **Margin Parameter** query, add a calculated field named **`Margin`** that calculates: **`Cost/Price`**
5. Add this parameter to the Margin Parameter query: **`Between [Enter Minimum Margin] And [Enter Maximum Margin]`**
6. In the Margin Parameter query, use the **Make Table** action to make a table in the current database with the Table Name: **`Items Within Margin`**
7. Run the query to create the Items Within Margin table using these minimum and maximum margin numbers:
   - Enter Minimum Margin: **`0.5`**
   - Enter Maximum Margin: **`0.75`**
8. Use these guidelines to create an Update query:
   - Add only the **Price** field from the Inventory table to the query grid.
   - Set the Update To criteria as: **`[cost]*1.5`**
   - Run the query and save it as: **`Margin Update`**
9. Save your database.
   - *Using eLab:* Save it to your **Access Chapter 7** folder as **`A7 _ P1 _ eSubmission`** and attach the file to your eLab assignment for grading.
   - *Not using eLab:* Save it to your **Access Chapter 7** folder as: **`A7 _ P1 _ Submission`**

PROJECT GRADER: A7-P2

WebVision: Create a Crosstab Query and Update Orders

WebVision has noticed some discrepancies in recent orders and customer records. It would also like to see the total sales for each sales rep for the customers they service. In this exercise, you will start by creating a Crosstab query. Then, you will create a Find Unmatched query to track down orders that do not contain any details and a Delete query to remove the incomplete orders. Last, you will create a Find Duplicates query to find duplicate customer records.

1. Download and open your Project Grader starting file.
   - *Using eLab:* Download **A7_P2_eStart** from the Assignments page. You *must* start with this file or your work cannot be automatically graded.
   - *Not using eLab:* Open **A7_P2_Start** from your **Access Chapter 7** folder.
2. Use these guidelines to create a Crosstab query using the **Crosstab Query Wizard**:
   - Contains fields from the **OrderDetails** query.
   - Use the **RepID** field for row headings.
   - Use the **Company Name** field for column headings.
   - Calculate each column and row intersection using the Sum function on the **Line Total** field.
   - Name the query: **`Rep Sales by Customer`**
3. Make these changes to the **Rep Sales by Customer** query:
   - Change the name of the Total of Line Total field to: **`Total`**
   - Set the Format field property to **Currency** for both the **Line Total** and **Total** fields.
4. Use these guidelines to create a Find Unmatched query:
   - The **Orders** table contains the records in the query results.
   - The **Order Details** table contains the related records.
   - Use the **OrderID** field in the Orders table and the **Order ID** field in the Order Details table as the matching fields.
   - Show all available fields in the query results.
   - Name the query: **`Orders Without Details`**
5. Use these guidelines to create a Delete query:
   - Base the query on the **Order Details** table.
   - Add **Order ID** to the query grid.
   - Set Criteria to delete records where the **Order ID** is greater than **5**.
   - Run the query and click **Yes** when prompted.

   If performed correctly, you will see a prompt notifying you that you are about to delete two rows.
   - Save the query with the name: **`Incomplete Order Delete`**
6. Use these guidelines to create a Find Duplicates query:
   - Search the **Customers** table for duplicate values.
   - Use **Billing Address**, **City**, **State/Province**, and **Postal Code** for fields that may contain duplicate information.
   - Add all remaining available fields to the Additional Query Fields list.
   - Name the query: **`Duplicate Customers`**

7. Save your database.
   - *Using eLab:* Save it to your **Access Chapter 7** folder as **A7 _ P2 _ eSubmission** and attach the file to your eLab assignment for grading.
   - *Not using eLab:* Save it to your **Access Chapter 7** folder as: **A7 _ P2 _ Submission**

Extend Your Skills

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

A7-E1 That's the Way I See It

You would like to make several enhancements to the Blue Jean Landscaping database to ensure more accurate data in query results. Open **A7-E1-BJL** and save it as: **A7-E1-BJLRev**

Make a copy of the Service Invoices Query, naming it *Acre Rate Range*, and then modify it to add a parameter that prompts the user to enter the rate per acre. The query should return only records that match the acre rate entered by the user. The second update is to the StoreMerchandise table. Create a *New Merchandise* append query that appends all records from the NewMerchandise table to the StoreMerchandise table. Verify that your queries function properly.

A7-E2 Be Your Own Boss

Business has picked up at Blue Jean Landscaping! You're modifying the database to ensure efficiency and to cope with unexpected situations. Open **A7-E2-BJL** and save it as: **A7-E2-BJLRev**

Use the Store Inventory query as the basis for a new query named *Manufacturer Item Inventory* that prompts the user to enter a manufacturer and returns all records for that manufacturer. Create a *2019 Sales* query that creates a new table containing all 2019 sales records from the Merch Sales query. Finally, create a *New Customers* append query that appends all records from the NewCustomers table to the Customers table.

A7-E3 Demonstrate Proficiency

Stormy BBQ has asked you to refine the merchandising section of its database to reflect recent price changes and move older records. Open **A7-E3-SBQ** and save it as: **A7-E3-SBQRev**

Create a query to copy the 2018 merchandise sales records from the MerchSales table into a new table and then create a delete query to remove those records from the MerchSales query. Decide what names to assign to the new table and to either of the two queries you choose to save. Create a query that increases the list prices of all items in the Merchandise table by 5%. You decide whether to save the query after running it (and which name to use, if you save it).

ACCESS

8 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 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
- Add a chart to a report

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 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 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 in the new record source table or query.

External Data→Import & Link→New Data Source→From Database→Access

DEVELOP YOUR SKILLS: A8-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 **A8-D1-WinDesign** from your **Access Chapter 8** folder and save it as: **`A8-D1-WinDesignRev`**
2. Look in the Reports section in the Navigation pane and notice the database contains three reports.
3. Choose **External Data→Import & Link→New Data Source→From Database→Access**.

   The Get External Data – Access Database dialog box appears.
4. Click the **Browse** button and navigate to your **Access Chapter 8** folder.

5. Choose the file **A8-D1-WinDesign-Backup** and click **Open**.

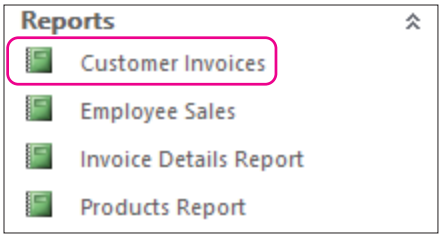
   This database is a backup copy of the WinDesign database.

6. Leave the **Import tables, queries, forms, reports** storage setting selected and click **OK**.

   Access opens the Import Objects dialog box and displays object names contained in the backup database in tabbed groups.

7. Click the **Reports** tab, choose **Customer Invoices**, and click **OK**.
8. Leave the Save Import Steps box unchecked and click **Close**.

   The Customer Invoices report is added to the Reports section of the Navigation pane.

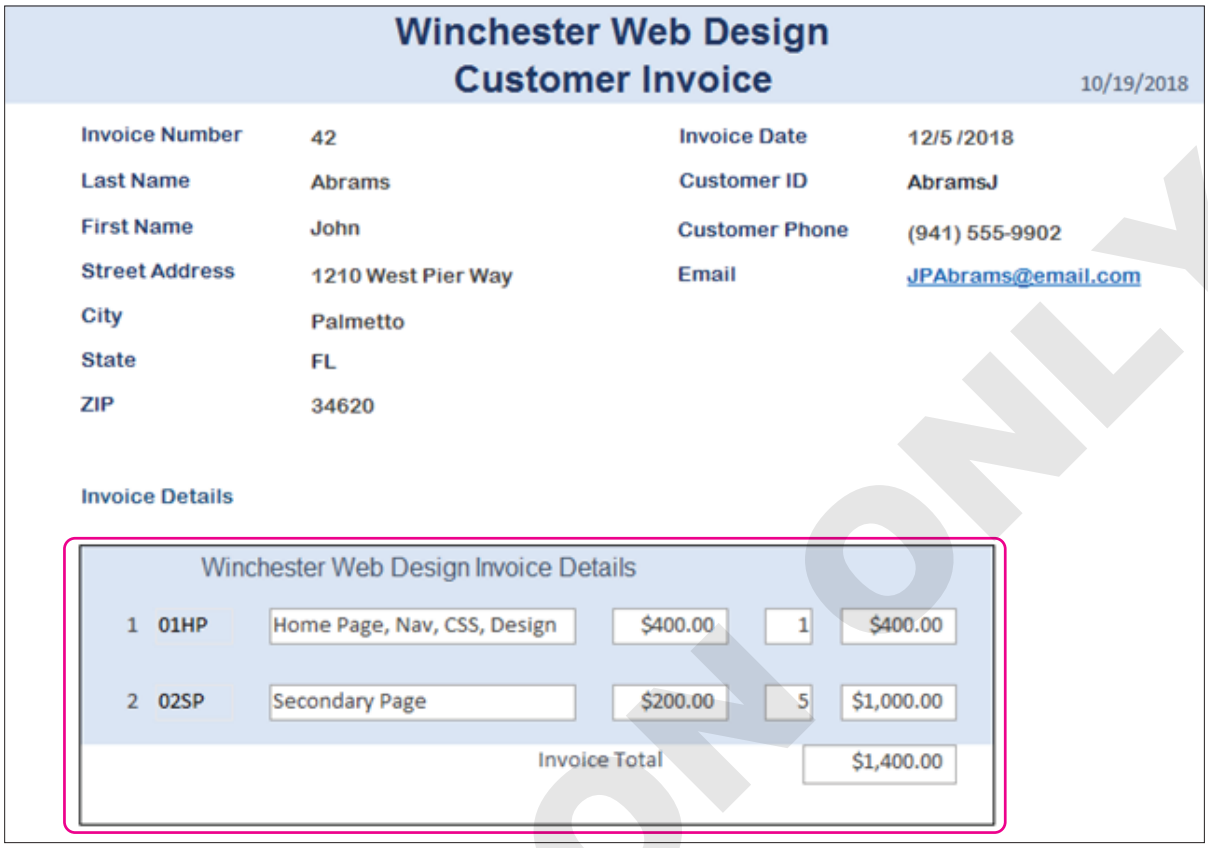


9. Double-click the **Customer Invoices** report to open it in **Report View**.
10. Scroll down as necessary to see the empty space between the customer information and signature blocks.

    This is a great spot to insert a subform/subreport that includes the invoice detail lines.

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 often already display the required report data. To streamline the report design and layout, it's best to use a subform as the basis of a subreport.



This invoice report 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.

View the video "Adding a Subreport Using the Wizard."

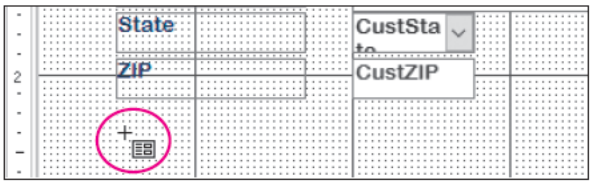
Report Design Tools→Design→Controls→Subform/Subreport

DEVELOP YOUR SKILLS: A8-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 **Report Design Tools→Design→Controls→Subform/Subreport** (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 the **Use an Existing Report or Form** option.
5. Choose **InvoiceDetails Subform** from the list and click **Next**.
6. Click **Next** again to accept the *Choose from a list* linking method.
7. Click **Finish** to accept *InvoiceDetails Subform* as the name.

   The subform is inserted in the report.

Set Properties

8. Display the Property Sheet, if necessary, and click the **Data** tab.

   Notice the Source Object is set to Form.InvoiceDetails Subform.

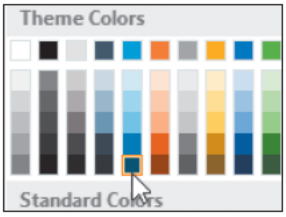
9. Switch to the **Format** tab in the Property Sheet and set these properties:

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

10. Click the **InvoiceDetails Subform** label, which is just below the ZIP label, and set these properties:

Property	Setting
Caption	**Invoice Details**
Width	**1.2**
Height	**0.30**
Top	**2.5**

Property	Setting
Left	**0.5**
Font Name	**Arial Rounded MT Bold**
Font Size	**10**
Fore Color	**Blue, Accent 1, Darker 50%**



11. Switch to **Report View** to see your completed report.
12. Use the Navigation bar and the scroll bar to review the database records.

    The information in both the report and the subform changes each time a new record is displayed.
13. 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 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→Save As

DEVELOP YOUR SKILLS: A8-D3

In this exercise, you will create and save a new report based on the subreport from the Customer Invoices 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**, choose **Save Object As** from the File Types list, and then click the **Save As** button in the right pane.
3. Type **`WWD Customer Invoices Subreport`** as the name, choose **Report** from the As drop-down list, and click **OK**.

   A new report is added to the Reports section of the Navigation pane.
4. Close the InvoiceDetails Subform.
5. Open the new **WWD Customer Invoices Subreport** in **Design View**.
6. Display the Property Sheet, if necessary, and then click the **All** tab and type **`Customer Invoices Subreport`** as the caption.

Create a Title for the Subreport

7. Choose **Report Design Tools→Design→Header/Footer→Title**.

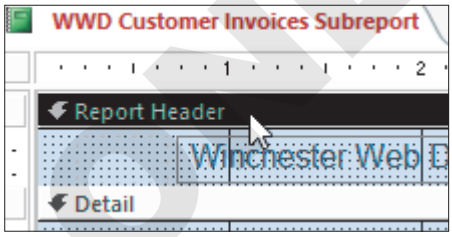
   Access places a title and empty placeholder controls in the Report Header.
8. Type **`Winchester Web Design Invoice Details`** in the title control and tap Enter.

9. Set these property values for the new title control:

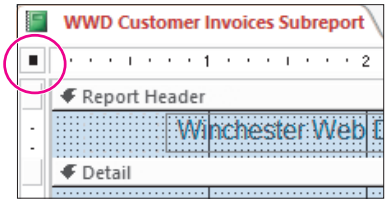
Property	Setting
Width	**3**
Height	**0.25**
Left	**1**
Font Name	**Arial**
Font Size	**12**
Text Align	**Center**

10. Click the **Report Header** section bar and set the Height property to: **0.3069**

    Inserting the title in the header widens the subreport, so you will now reset the width.

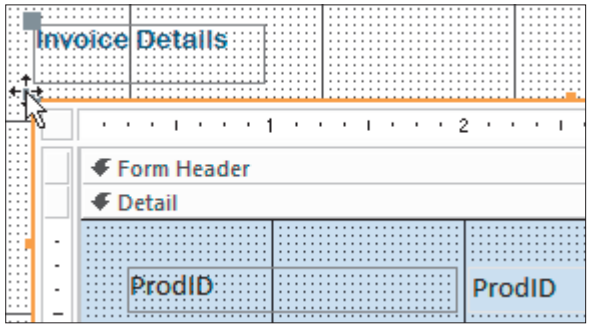


11. Click the report selector button and set the Width property to: **5.6**



12. Close the subreport, saving the changes, and then open the **Customer Invoices** report in **Design View**.
13. If necessary, display the Property Sheet and click the **Data** tab.
14. Click the subreport to select it.
15. Click in the **Source Object** property box.

    A drop-down menu button appears at the far right of the property box.



16. Choose **Report.WWD Customer Invoices Subreport** from the drop-down menu.
17. Switch to **Report View** to see how the new subreport looks.

    Notice the change in appearance compared to the subform used previously.

18. Scroll through the report, seeing how the subreport always shows the correct invoice details.
19. 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: A8-D4

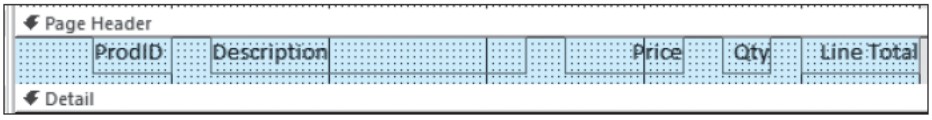
In this exercise, you will reposition the controls in the WWD Customer Invoices Subreport page header and add a text box control to count the number of line items.

1. Display the **WWD Customer Invoices Subreport** in **Design View**.
2. Right-click the **Detail** section bar and choose **Page Header/Footer**.

   You will add labels to the Page Header section.
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 bar 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	Left
Description	**2**	**1.25**
Price	**0.75**	**3.5**
Qty	**0.3**	**4.5**
LineTotal	**0.75**	**5**

8. If necessary, click the **Description**, **Price**, **Qty**, and **LineTotal** labels and use the up arrow ↑ key to nudge them so they're vertically aligned with the ProdID label.

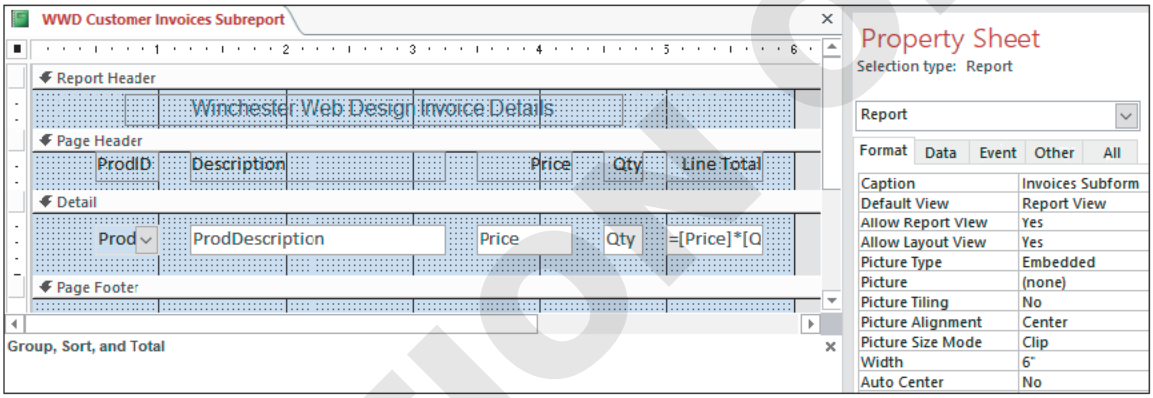


Move and Resize Report Text Box Controls

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**



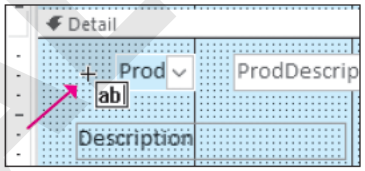
13. Switch to **Report View**.

There is room for a small field to the left of the ProdID field, which is where you will insert numbering.

Add and Format a Text Box

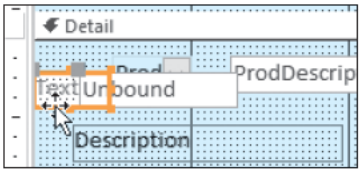
14. Switch to **Design View** and choose **Report Design Tools→Design→Controls→Text Box** ab.
15. Click to the left of the ProdID text box in the Detail section to position the new box in that area.

The precise location is not important, as you will set the position, using specific properties, in a moment.



16. Click the new label control and tap Delete to remove it.

You may need to move your new text box if it is obscuring the label.



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**
Back Style	**Transparent**
Border Style	**Transparent**

Set Control Properties to Sum

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.

    Notice that the numbering continues sequentially throughout 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** and set the Width property to: **6**
23. Switch to **Print Preview** and use the navigation bar to review the various report pages.

    Each invoice begins on a new page and the invoice detail line items begin at number 1 for each invoice.
24. Close Print Preview and then close the Customer Invoices report, saving the changes.

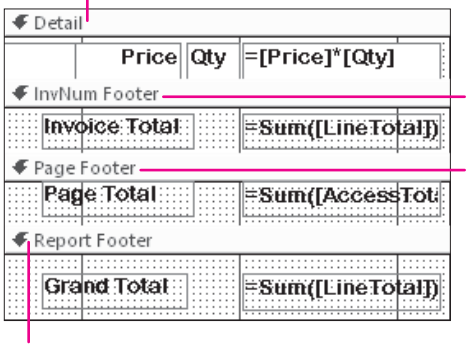
Creating Calculated Controls on a Subreport

Reports summarize data in tables and queries to present useful, organized information. This means that a report would typically show subtotals, grand totals, and averages to summarize a specified group of data. You can do this by adding calculated controls.

Positioning Calculated Controls

Calculated controls are built-in reports using the Control Source property of an unbound text box control to which you add a formula. The section or group where a calculated control is placed determines how Access performs the calculation.

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



A calculated control in a Group Footer section calculates the total for the group.

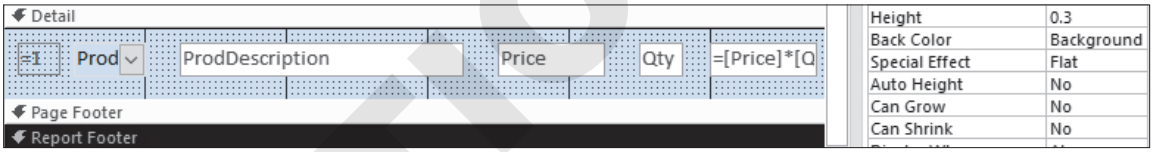
A calculated control in a Page Footer section calculates the total for the page.

A calculated control in a Report Footer section calculates the total for the entire report.

DEVELOP YOUR SKILLS: A8-D5

In this exercise, you will add a calculated control to the report footer section in 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**



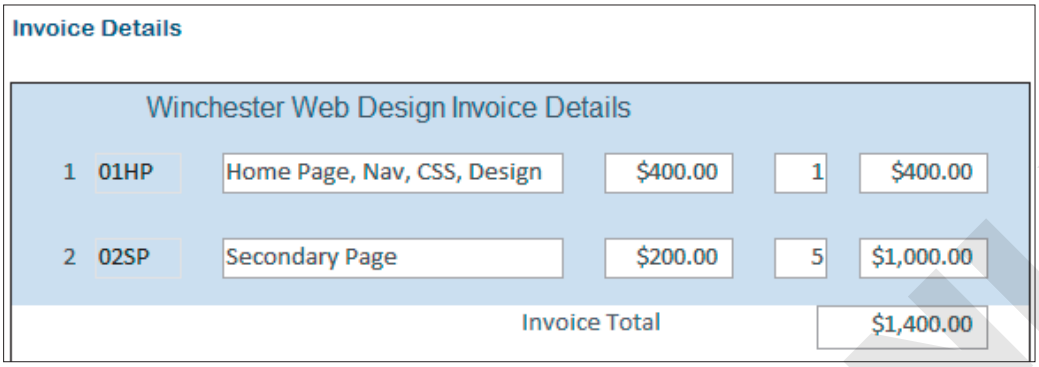
3. Choose **Report Design Tools→Design→Controls→Text Box** ab|
4. Click anywhere in the **Report Footer** section.
5. Switch to the **All** tab in the Property Sheet and set these properties for the new text box control:

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

6. Click the new label control and set these property values:

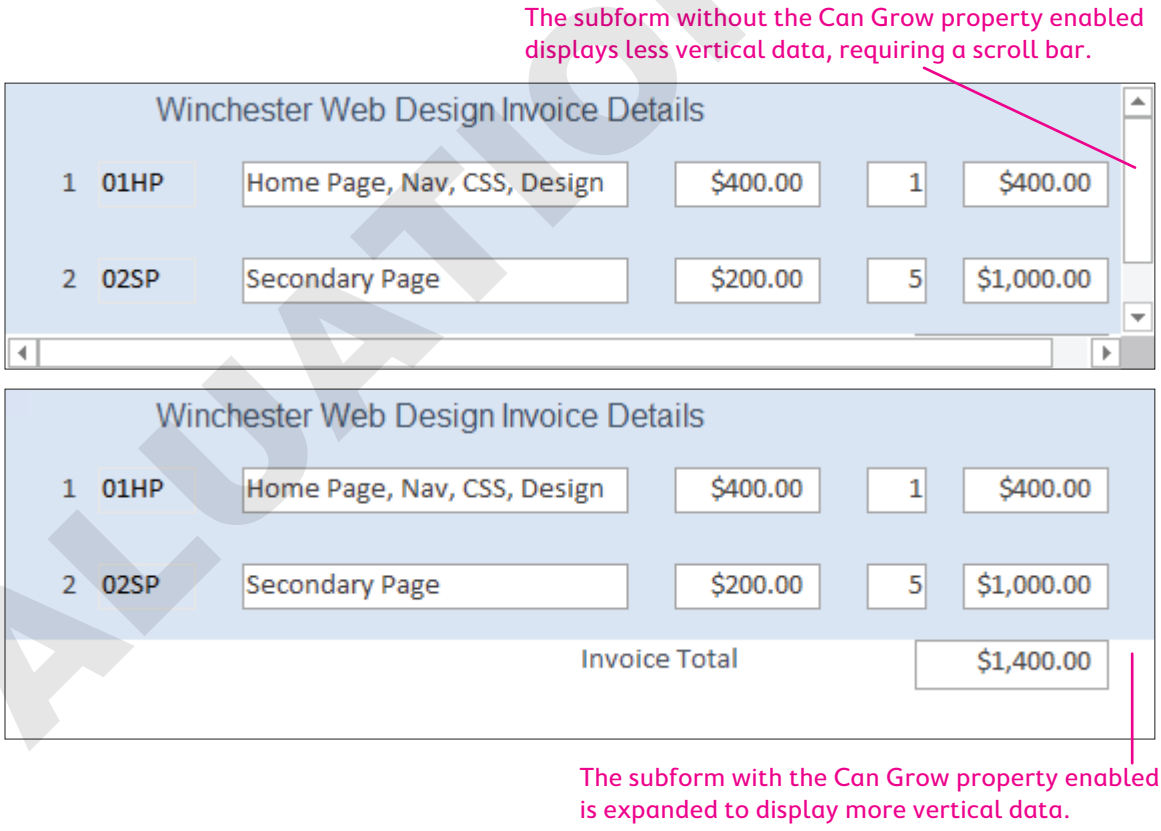
Property	Setting
Caption	**Invoice Total**
Width	**1.2**
Height	**0.25**
Top	**0**
Left	**3**
Fore Color	**Text Dark**

7. Close the subreport, saving the changes.
8. Display the **Customer Invoices** report in **Report View**.



Growing and Shrinking a Subreport

When the number of records or amount of data displayed in a subreport varies, you can set the Can Grow and Can Shrink properties to allow the subreport space to expand or shrink 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: A8-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 **Report Design Tools→Page Setup→Page Size→Margins→Narrow**.
3. Click the **InvoiceDetails Subreport**.
4. If necessary, display the Property Sheet; on the **Format** tab, set the Can Grow property to **No**.
5. Switch to **Report View** and scroll through the report.

   The subform now displays a vertical scroll bar in order to view additional records.
6. Switch to **Design View** and click the **InvoiceDetails Subreport**.
7. Click the **Format** tab on the Property Sheet and, if necessary, set the Can Grow and Can Shrink properties to **Yes**.
8. Switch to **Report View** and scroll through the report.

   The subreport grows and shrinks to best fit the contents.
9. Save 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 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 rather than the Report Header, which prints only on the first page.

Tip! *Add the page break at the end of the Detail section so Access knows to start a new page before printing the next page header.*

Report Design Tools→Design→Controls→Insert Page Break

DEVELOP YOUR SKILLS: A8-D7

In this exercise, you will modify the Customer Invoices report by adding a title and the current date as well as setting up page breaks to print each invoice on a separate page.

1. Display the **Customer Invoices** report in **Design View**.

   To begin, you will modify the report title.
2. Click the **Customer Invoice** title control in the Page Header section and then click 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 an empty area in the Page Header section and then click the title control once more to select it.
5. Set the Width property to **4** and the Left property to **2**.

Add a Date Control

In the next few steps, you will insert a date that will display in the report header.

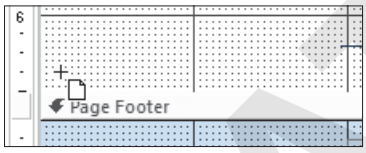
6. Choose **Report Design Tools→Design→Header/Footer→Date and Time**.
7. Choose the third date option **MM/DD/YYYY** format, remove the check from the **Include Time** box, and click **OK**.

   Access places the new date control in the Report Header section. Next, you will move the date to the Page Header section so it appears on every page rather than just the first page of the report.
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 these properties for the date control:

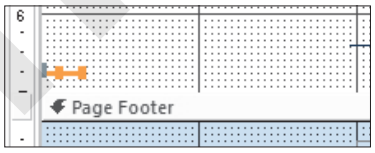
Property	Setting
Width	**2**
Height	**0.2**
Top	**0.4**
Left	**5.8**

Add a Page Break Control

11. Scroll to the bottom of the Detail section.
12. Choose **Report Design Tools→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 pages.

    The date appears on each invoice because you added it in the Page Header section. Each invoice begins on a new page. Sometimes it's not necessary to add a page break to a report and doing so may add an unneeded page. If this occurs in your report, remove the page break.
15. Close Print Preview and then close the report, saving the changes.

Adding Charts

Summarized and numerical data is often displayed visually using a chart. By adding a chart to a report, the information and relationships between report data is more easily interpreted. While you can add charts to reports and forms, they're more commonly used in reports. Access offers multiple chart types, including column, bar, line, and pie, among others.

You can insert charts using the Chart control or the Insert Chart command.

Report Design Tools→Design→Controls→Insert Chart

Formatting Modern Charts

You can format and modify common settings using either the Property Sheet or the Chart Settings task pane, which means formatting charts in Access is similar to formatting charts in other Microsoft Office applications, like Excel and PowerPoint.

View the video "Adding Modern Charts."

Report Design Tools→Design→Tools→Chart Settings

Refreshing Chart Data

Like many other Access objects, charts use the contents in a table or query as the data source. When the data source of a chart is changed, the chart may not automatically reflect those changes. The Refresh All command manually updates a chart to match record source data.

Home→Records→Refresh All

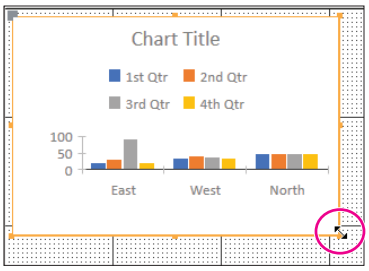
DEVELOP YOUR SKILLS: A8-D8

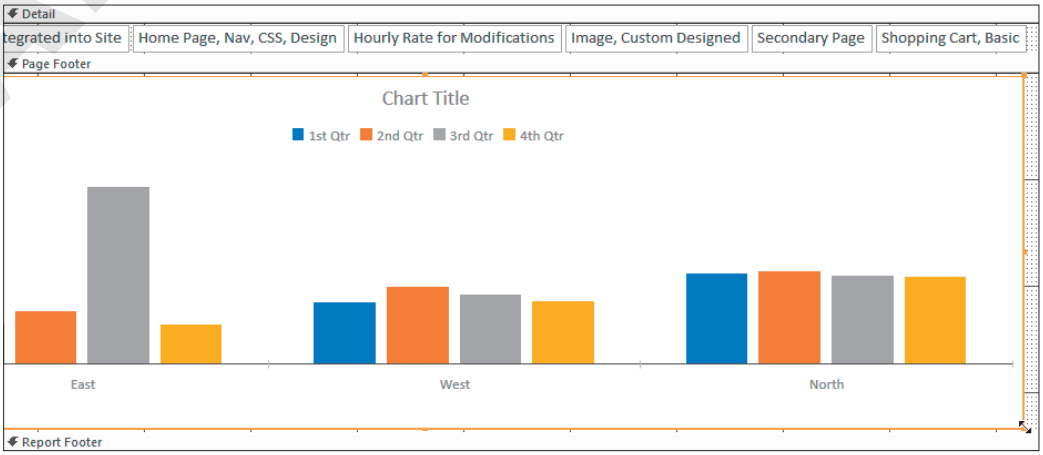
In this exercise, you will modify Winchester's Employee Sales report by adding a chart. You will then format the chart and manually update it after making changes to the record source.

1. Display the **Employee Sales** report in **Design View**.
2. Choose **Report Design Tools→Design→Controls→Insert Chart**.

   A menu appears allowing you to select the chart type.
3. Choose **Column→Clustered Column**.

   Your mouse pointer changes to a chart icon with a plus sign.
4. Click in the upper-left corner of the **Page Footer** section to insert the chart control.
5. Close the Property Sheet and the Chart Settings pane, if necessary.
6. Hover your mouse pointer over the bottom right of the new chart control until a diagonal double-pointed arrow appears.
7. Resize the control by dragging to the bottom right until your chart is as tall as the Page Footer section and as wide as all text box controls in the Detail section.





Format the Chart

8. Choose **Report Design Tools→Design→Tools→Chart Settings**.

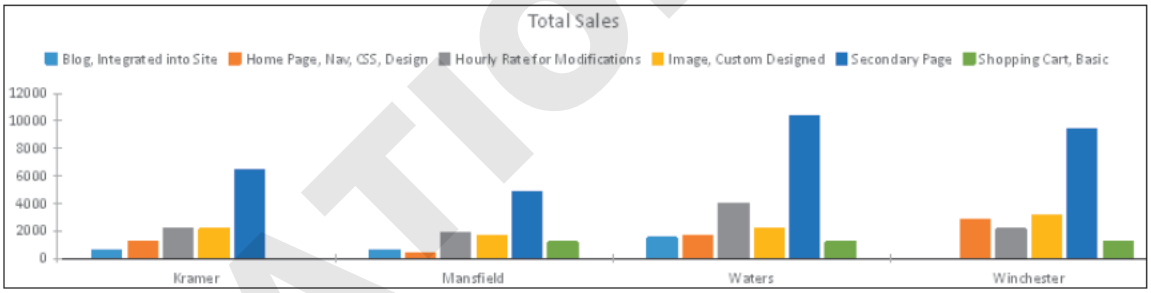
   The Chart Settings pane appears.

9. Choose **EmployeeSales** from the Data Source drop-down list.
10. Choose **EmpLastName** for the Axis.
11. Choose **ProdDescription** as the Legend.
12. Choose **LineTotal (Sum)** for the Values.
13. Open the Property Sheet and set these chart properties:

Property	Setting
Has Legend	**Yes**
Chart Title	**Total Sales**
Category Axis Font Size	**14**
Primary Values Axis Format	**Currency**

Tip! *There are many chart properties and settings available. If you will use charts frequently, you may want to explore each of these settings.*

14. Switch to **Report View** to see the newly created chart.



Refresh the Chart

15. Open the **InvoiceDetails** table and change the quantity for the first record to: **100**
16. Switch to the **Employee Sales** report and choose **Home→Records→Refresh All**.

    The Home Page column for Winchester rises dramatically.

17. Switch to the **InvoiceDetails** table and change the quantity for the first record back to: **1**
18. Switch to the **Employee Sales** report and refresh the chart using the **Refresh All** command.

    The chart data has normalized.

19. Close the Employee Sales report, saving any changes, and then close the database.

Self-Assessment

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

Reinforce Your Skills

REINFORCE YOUR SKILLS: A8-R1

Import a Report and Add a Subreport

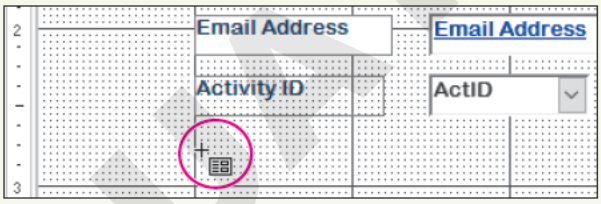
In this exercise, you will import a report from the Kids for Change database and then add a subform with details on the activities provided by the staff.

1. Open **A8-R1-K4C** from your **Access Chapter 8** folder and save it as: **A8-R1-K4CRev**
2. Choose **External Data→Import & Link→New Data Source→From Database→Access**.
3. Click **Browse** and navigate to your **Access Chapter 8** folder, choose **A8-R1-K4C-Backup**, and click **Open**.
4. Leave the **import** storage setting chosen and click **OK**.
5. Click the **Reports** tab, choose **Staff Report**, and click **OK**.
6. Leave the Save Import Steps checkbox unchecked and click **Close** in the Get External Data dialog box.
7. Display the newly imported **Staff Report** in **Report View**.

   The empty space between the staff records is where you will insert the subreport.

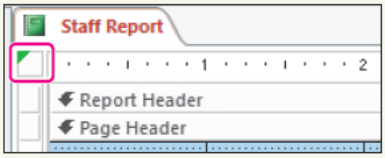
Add and Format a Subreport

8. Switch to **Design View** and choose **Report Design Tools→Design→Controls→ Subform/Subreport** (scroll to the bottom of the controls list).
9. Click just below the Activity ID label in the Detail section to insert a control and launch the Subreport Wizard.

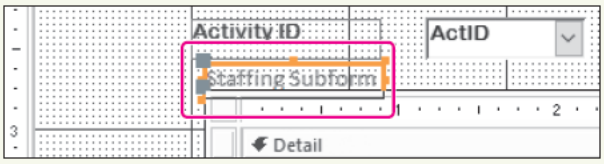


10. Choose **Staffing Subform** from the Use an Existing Report or Form option and click **Next**.
11. Click **Next** again to accept *Choose from a List* as the linking method.
12. Click **Finish** to accept *Staffing Subform* as the name.

    A small green triangle in the report selector indicates a possible error. Clicking the triangle displays a smart tag that shows the error and possible solutions. In this case, the report is wider than the page. You will correct this in the next few steps.



13. Click the **Staffing Subform** label and tap Delete.

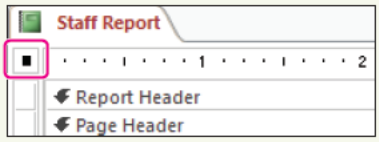


14. Click the subreport and set these properties:

Property	Setting
Width	**7.7**
Height	**0.5**
Top	**3**
Left	**0**
Border Style	**Transparent**

15. Click the report selector and set the Width property to: **7.8**

The green smart tag indicator should be gone.



16. Save Staff Report and then switch to **Report View**.
17. Scroll through the report to see that the activity details now appear under the staff information.
18. Close the database.

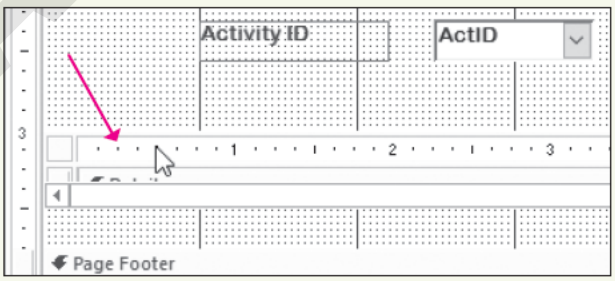
REINFORCE YOUR SKILLS: A8-R2

Create a Report from a Subreport and Number Report Items

In this exercise, you will save a subreport as a separate report. You will also update a report of donations to Kids for Change, numbering them in order to add a count of the donations per donor.

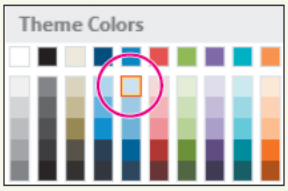
1. Open **A8-R2-K4C** from your **Access Chapter 8** folder and save it as: **A8-R2-K4CRev**
2. Open **Staff Report** and switch to **Design View**.
3. Right-click the subreport control and choose **Subreport in New Window**.

The Staffing Subform subreport opens in a separate window.

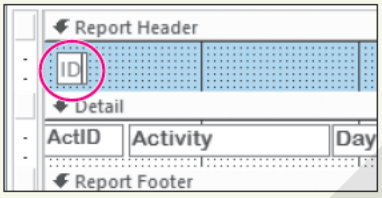


4. Choose **File→Save As→Save Object As→Save As**.
5. Type **`K4C Staffing Subreport`** as the name, choose **Report** from the Save As drop-down list, and click **OK**.
6. Close the subform and the report.
7. Display the new **K4C Staffing Subreport** in **Design View**.
8. Click the **Detail** section bar and set the Height property to: **`0.3`**
9. Right-click the **Detail** section bar and choose **Report Header/Footer**.

   The Report Header section appears.
10. Click the **Report Header** section bar.
11. Set the Back Color property to **Blue, Accent 1, Lighter 80%**.

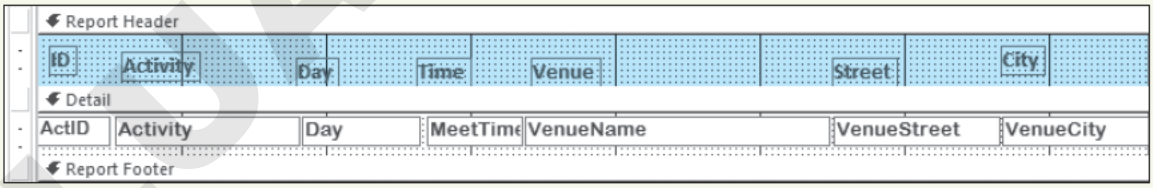


12. Choose **Report Design Tools→Design→Controls→Label** Aa and then click in the **Report Header** section above the ActID control and type: **`ID`**



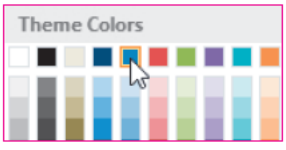
13. Add six more labels: **`Activity`**, **`Day`**, **`Time`**, **`Venue`**, **`Street`**, and **`City`**

    You don't need to be precise with the label positions, as you will modify their properties in the next step.

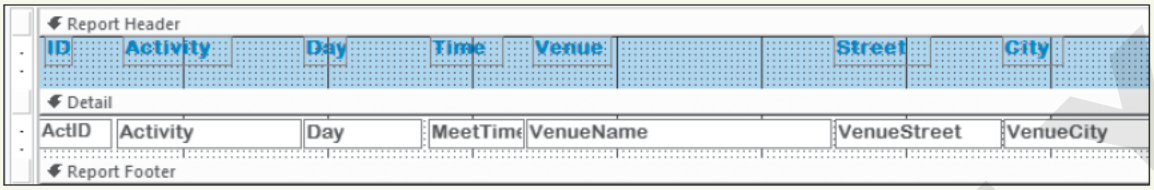


14. Press Ctrl and click all seven labels to select them and then set these properties:

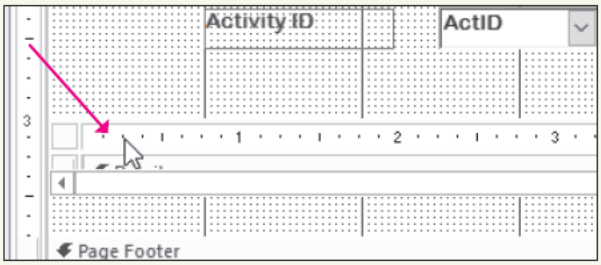
Property	Setting
Height	**0.2**
Top	**0**
Font Name	**Arial Rounded MT Bold**
Font Size	**10**
Fore Color	**Blue, Accent 1**



15. Click in an empty spot in the Report Header section to deselect the labels.
16. If necessary, click the individual labels and widen them so the text is fully visible and then align them with the text boxes in the Detail section.



17. Close and save K4C Staffing Subreport.
18. Display **Staff Report** in **Design View** and click the subreport control.



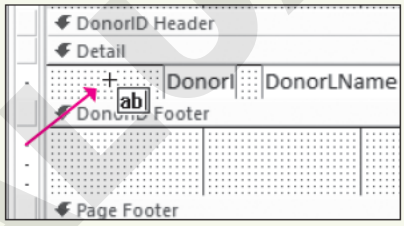
19. Set these properties on the **All** tab of the Property Sheet:

Property	Value
Name	`Staffing Subreport`
Source Object	`Report.K4C Staffing Subreport`

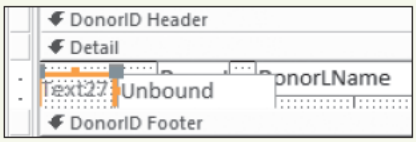
20. Switch to **Report View** to see the changes and then close and save Staff Report.

Number Report Items

21. Display **Donations Report** in **Design View**.
22. Choose **Report Design Tools→Design→Controls→Text Box** and click in the **Detail** section to the left of the DonorID text box.



23. Click the label control for the new text box (to the left of the text box—it may be difficult to see) and tap Delete.



24. Select the new text box control and then set these property values on the **All** tab of the Property Sheet:

Property	Setting
Name	**txtCounter**
Control Source	**=1**
Width	**0.3**
Top	**0**

Property	Setting
Left	**0.3**
Border Style	**Transparent**
Font Name	**Arial**
Font Weight	**Semi-Bold**

25. Click the **Data** tab in the Property Sheet and set the Running Sum property to **Over All**.
26. Switch to **Report View**.

    Access numbers the detail lines consecutively for each individual donation.
27. Close the database, saving the changes to the Donations Report.

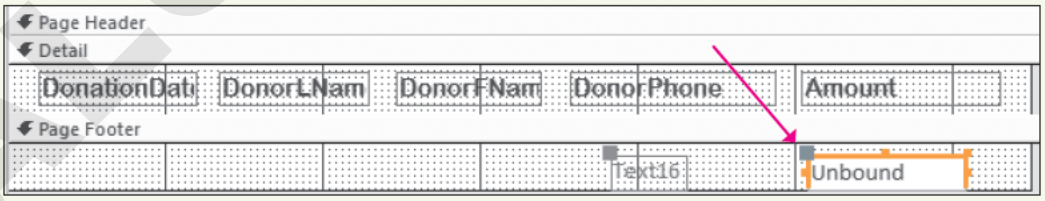
REINFORCE YOUR SKILLS: A8-R3

Insert Calculated Fields and Page Breaks

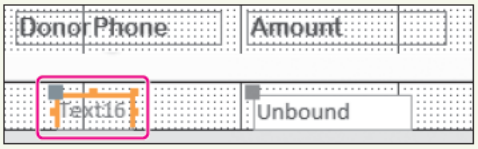
Kids for Change wants to improve its donations reporting. In this exercise, you will add a field to calculate the total monthly donations for each donor. You will set the subreport to grow and shrink, and you will add custom controls and a page break. You will then add a column chart that shows total donations by donor with data labels.

1. Open **A8-R3-K4C** from your **Access Chapter 8** folder and save it as: **A8-R3-K4CRev**

   You will begin by adding a calculated field to a report.
2. Display **K4C Donors Subreport** in **Design View**.
3. Choose **Report Design Tools→Design→Controls→Text Box** and click in the Page Footer section under the Amount text box to place a new text box there.



4. Click the text box label and tap Delete.



5. Click the new text box and set these properties on the **All** tab of the Property Sheet:

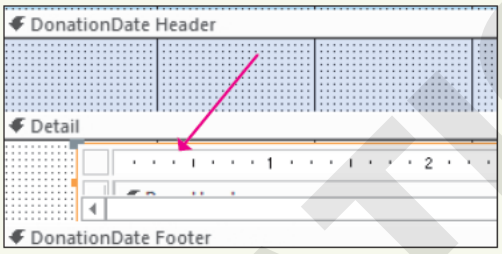
Property	Setting
Name	**MonthTotal**
Control Source	**=Sum([Amount])**
Format	**Currency**
Width	**1.3**
Height	**0.2**

Property	Setting
Top	**0**
Left	**5**
Font Name	**Arial Rounded MT Bold**
Text Align	**Right**

6. Close and save K4C Donors Subreport.

Grow and Shrink a Subreport

7. Display **Monthly Donations Report** in **Design View**.
8. Click the subform to select it.



9. Click the **All** tab in the Property Sheet and set the Can Grow and Can Shrink properties to **Yes**.
10. Switch to **Report View** and scroll through the report.

    Notice how the number of donations per month changes and the report shrinks and grows to accommodate the number of donations each month.

Modify the Title

11. Switch to **Design View**.
12. Click the **Monthly Donations Report** title control in the Page Header section and then click just in front of *Monthly* to position the insertion point.
13. Type **`Kids for Change`** and press Shift+Enter to force *Monthly Donations Report* to a second line.
14. Click a blank part of the header and then click the title control again to select it.

15. Set these properties for the title control:

Property	Setting
Width	**4**
Left	**2**
Text Align	**Center**

16. Switch to **Print Preview** and scroll through the first page of the report.

    More than one month appears per page.

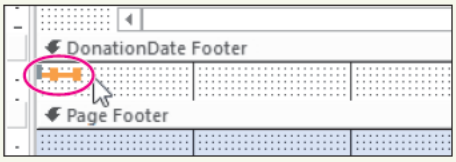
17. Use the navigation bar to navigate to page 2.

    The April donations spill over from the previous page. In the next steps, you will insert a page break so each month begins on a new page.

Insert a Page Break

18. Close Print Preview and then switch to **Design View** and choose **Report Design Tools→Design→Controls→Insert Page Break**.

19. Click in the left side of the DonationDate Footer section to place the page break.



20. Set the Top property of the page break control to: **0**

21. Click the **DonationDate Footer** section bar and set the Height property to: **0.001**

    This will make the DonationDate Footer section as short as possible so the page break doesn't push the DonationDate header to the next page.

22. Switch to **Print Preview** and use the navigation bar to scroll through the report pages.

    Now the donations for each month start on new pages.

Add a Column Chart

23. Close the Monthly Donations report, saving the changes, and then display the **Total Donations Chart** report in **Design View**.

24. Choose **Report Design Tools→Design→Controls→Insert Chart**.

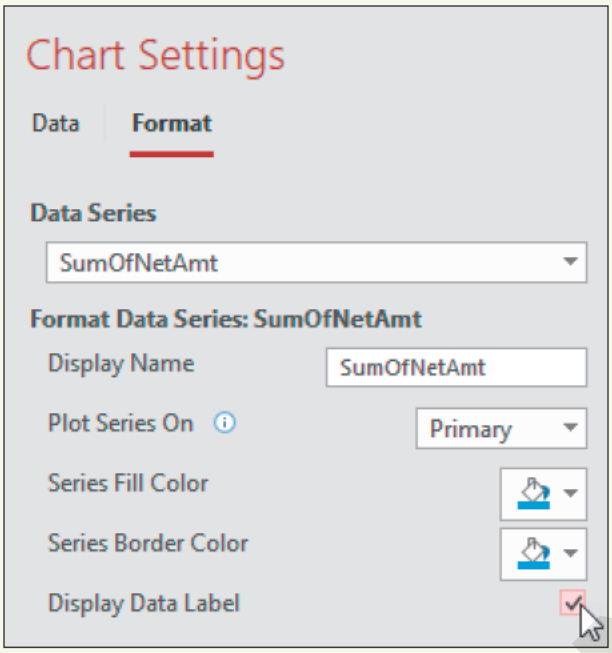
25. Choose **Column→Clustered Column**.

26. Click in the upper-left corner of the **Detail** section to insert the chart control.

27. Choose **Donations Query** from the Data Source drop-down list.

28. Choose **DonorLName** for the Axis, **(None)** as the Legend, and **NetAmt (Sum)** for the Values.

29. Click the **Format** tab at the top of the Chart Settings pane and check the box for **Display Data Label**.



30. Resize your control by dragging to the bottom right until your chart fills the entire Detail section.
31. Open the Property Sheet and set these chart properties:

Property	Setting
Has Legend	**No**
Has Title	**No**
Primary Values Axis Format	**Currency**

32. Switch to **Report View** and view the report; close the database, saving the changes.

Apply Your Skills

APPLY YOUR SKILLS: A8-A1

Work with Reports and Subreports

In this exercise, you will help Universal Corporate Events create a report that breaks down the company's revenue by venue. You will add a subreport to a main report.

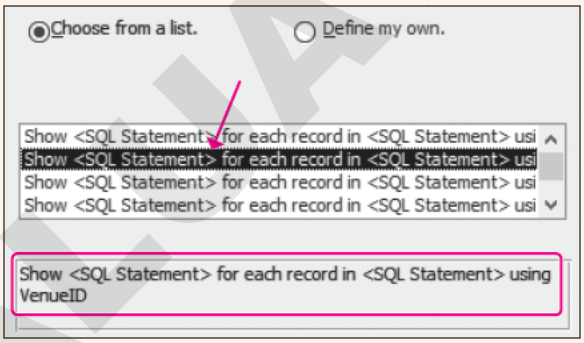
1. Open **A8-A1-UCE** from your **Access Chapter 8** folder and save it as: **A8-A1-UCERev**

   The first step is to import the needed report.
2. Choose **External Data→Import & Link→New Data Source→From Database→Access**.
3. Choose **A8-A1-UCE–Backup** from your **Access Chapter 8** folder.
4. Leave the **Import** storage setting chosen and click **OK**.
5. Click the **Reports** tab, choose **Venue Revenue Report**, and click **OK**.
6. Choose to not save the import steps and click **Close**.
7. Display the newly imported **Venue Revenue Report** in **Report View**.

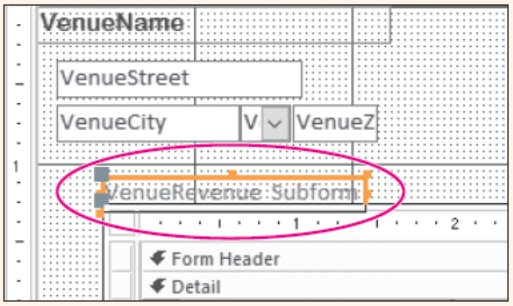
   The empty space between the locations is where you will insert the subreport.

Add a Subreport

8. Switch to **Design View** and choose **Report Design Tools→Design→Controls→Subform/Subreport**.
9. Click in the **VenueID** header section below the other controls and launch the Subreport Wizard.
10. Choose **VenueRevenue Subform** from the Use an Existing Report or Form option and click **Next**.
11. Choose the second linking option (link by VenueID) and click **Next** again.



12. Click **Finish** to accept *VenueRevenue Subform* as the name.
13. Delete the VenueRevenue Subform label located just above the subform.



14. Select the new subform and set these properties:

Property	Setting
Width	**7**
Height	**1.5**
Top	**1.3**
Left	**0.25**
Border Style	**Transparent**

15. Click the **VenueID Header** section bar and set the Height property to: **3**
16. Switch to **Report View** and scroll through the report.

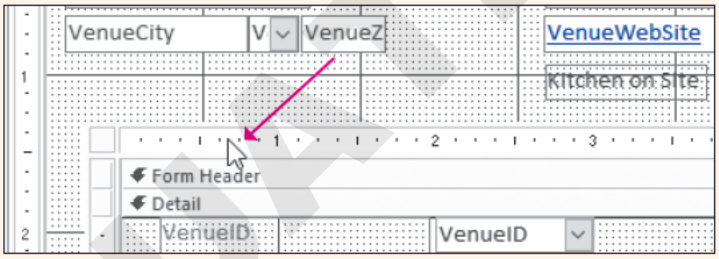
    Each venue now has an event data subform associated with it that contains all the revenue data for the events.
17. Close the database, saving the changes to the Venue Revenue Report.

APPLY YOUR SKILLS: A8-A2

Create a Report from a Subreport and Number Report Items

In this exercise, you will continue to help Universal Corporate Events create an effective venue revenue report. You will create a report from a subreport and number items on a subreport.

1. Open **A8-A2-UCE** from your **Access Chapter 8** folder and save it as: **`A8-A2-UCERev`**
2. Display **Venue Revenue Report** in **Design View**.
3. Right-click the subreport and choose **Subreport in New Window**.



4. Save the object as a report named: **`VenueRevenue Subreport`**
5. Close VenueRevenue Subreport.
6. Make sure the subform/subreport is selected and the Property Sheet is displayed.
7. Set the Source Object property on the **All** tab to **Report.VenueRevenue Subreport**.
8. Save the Venue Revenue Report and then switch to **Print Preview**.

   The subreport will be displayed.
9. Close Print Preview and close Venue Revenue Report.

Format the Report

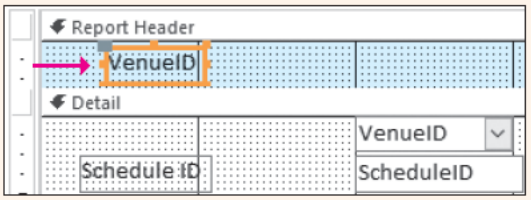
Now you will resize the labels and move them from the Detail section to the Report Header section.

10. Display the **VenueRevenue SubReport** in **Design View** and set the Width property to: **6**

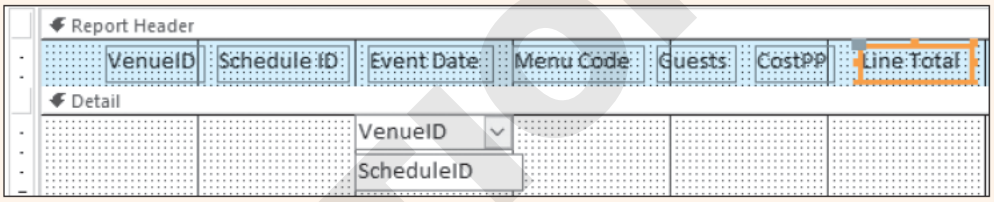
    This widens the subreport enough to accommodate the changes you are about to make.

11. Use Ctrl and the mouse to select all labels and text boxes in the Detail and Report Footer sections.
12. Click the **Format** tab in the Property Sheet and set the Border Style property to **Transparent** and the Fore Color property to **Black, Text 1**.

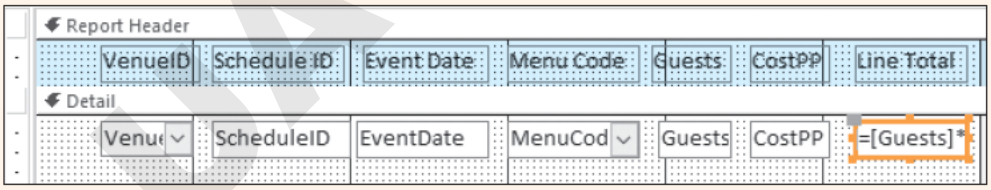
13. Click an empty part of the form to deselect all controls and then click each label individually and reduce the size to just fit the caption (label text).
14. Click the **Report Header** section bar and set the Height property to: **0.3**
15. Click the **VenueID** label and press Ctrl+X.
16. Click in the **Report Header** section, press Ctrl+V to paste the label, and then use the arrow keys to nudge it to about 0.25 in from the left edge.



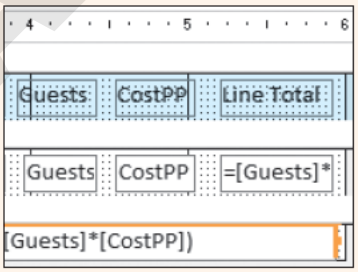
17. Cut and paste the remaining labels from the Detail section into the Report Header and then use the arrow keys to align them approximately as shown:



18. Rearrange the Detail section text boxes by dragging them with the mouse, nudging them with the arrow keys, and reducing their widths to align them with the Report Header labels.



19. Click the **Detail** section bar and set the Height property to: **0.3**
20. Use the right arrow → key to right-align the calculated control in the Report Footer section with the Line Total label and the Detail section calculated control.



21. Switch to **Layout View** and, if necessary, resize and reposition the controls so they are aligned nicely and all data is visible.

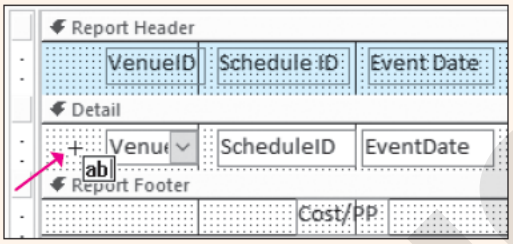
VenueRevenue Subreport

VenueID	Schedule ID	Event Date	Menu Code	Guests	CostPP	Line Total
PalmCt	BRTLuna	10/11/2019	BARSNK	50	$7.50	$375.00
WMinst	HOLMiller	12/31/2019	CHFBRK	100	$16.00	$1,600.00
Meadow	HOLMiller	7/1/2019	DESSRT	25	$13.00	$325.00

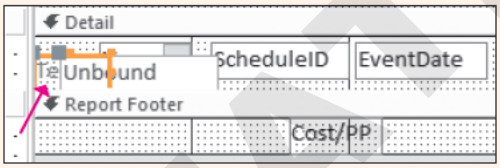
22. Save the subreport.

Number Report Items

23. Switch to **Design View** and insert a new text box in the Detail section to the left of the VenueID text box.



24. Delete the associated label control.



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

Property	Setting
Name	**txtCount**
Control Source	**=1**
Width	**0.2**
Top	**0**
Left	**0**
Border Style	**Transparent**

26. Click the **Data** tab in the Property Sheet and set the Running Sum property to **Over All**.
27. Select all text boxes in the Detail section and set the Top property to: **0**

28. Save the **VenueRevenue SubReport** and then display the **Venue Revenue Report** in **Print Preview**.

    Navigate through the report and notice that the detail lines for each venue are now numbered sequentially.

29. Close the database, saving changes to any reports.

APPLY YOUR SKILLS: A8-A3

Add a Calculated Field and a Page Break

Universal Corporate Events continues to refine its database reports. In this exercise, you will first add a field to calculate the total of all the venues' revenues. You will also add a page break to another report to allow each month to begin on a new page.

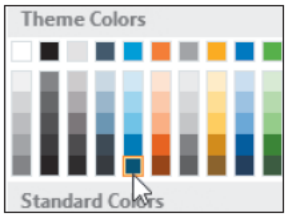
1. Open **A8-A3-UCE** from your **Access Chapter 8** folder and save it as **`A8-A3-UCERev`**
2. Display **Venue Revenue Report** in **Report View**.
3. Scroll through the report and notice that each venue is associated with Line Total calculations and a total of all line totals per venue.

   The one thing missing is a grand total at the bottom of the report that sums all totals for the venues. You will add a calculated control that produces a grand total.

4. Switch to **Design View**.
5. Click the **Report Footer** section bar and set the Height property to: **`0.4`**
6. Insert a text box anywhere in the Report Footer section and set these properties for the label control:

   Hint: Use the Format tab on the Property Sheet.

Property	Setting
Caption	**Grand Total for All Venues**
Width	**2**
Top	**0.1**
Left	**2**
Font Weight	**Semi-Bold**
Fore Color	**Blue, Accent 1, Darker 50%**



7. Set these property values for the new text box:

   Hint: Use the All tab on the Property Sheet.

Property	Setting
Name	**ActivityCost**
Control Source	**=Sum([Guests]*[ChgPP])**
Format	**Currency**
Width	**1.5**
Top	**0.1**

Property	Setting
Left	**5**
Border Style	**Transparent**
Font Weight	**Semi-Bold**
Fore Color	**Blue, Accent 1, Darker 50%**

8. Select the **VenueRevenue Subform** and set the Can Grow and Can Shrink properties on the Format tab of the Property Sheet to **Yes**.

   This ensures that all venue detail lines will be displayed.

9. Switch to **Report View** and scroll to the end of the report.

   Notice the Grand Total for All Venues *label and the grand total calculated control.*

10. Close Venue Revenue Report, saving the changes.

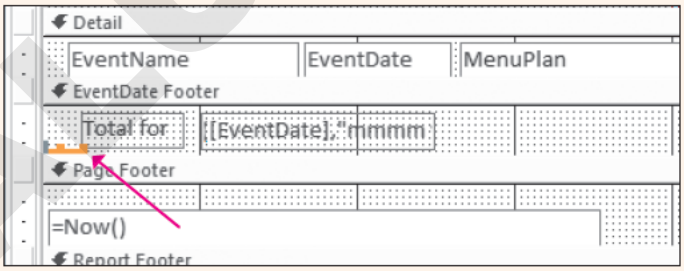
Insert a Page Break

11. Display the **Event Revenue Report** in **Print Preview**.
12. Scroll through the first page and notice that several months are included.

    The page break you are about to insert will position each month on a separate page.

13. Close Print Preview and then switch to **Design View** and insert a **Page Break** control on the left side of the EventDate Footer section, just below the *Total for* label.

    Be sure to position the break below the label or portions of the report will be cut off.



14. Switch to **Print Preview** and notice that just one month appears on the first page.
15. Use the navigation bar to go to the next page, and, once again, just one month will be displayed because of the page break.
16. Close the report, saving your changes.

Insert a Pie Chart

17. Display the **Revenue by Venue** report in **Design View**.
18. Use the **Insert Chart** command to create a new pie chart.
19. Click in the upper-left corner of the **Detail** section to insert the chart control and use these chart settings:
    - Data source: **Event Revenue**
    - Legend: **(None)**
    - Values: **TotalRev (Sum)**
20. On the **Format** tab, check the box for the **Display Data Label** option.
21. Resize your chart control to fill the entire Detail section.
22. On the Property Sheet, set these properties for the chart:

Property	Setting
Legend Position	**Bottom**
Has Title	**No**
Primary Values Axis Format	**Currency**

23. Close the database, saving the changes to the report.

Project Grader

PROJECT GRADER: A8-P1

Taylor Games: Updating Inventory

Taylor Games is ready to take sales orders. Before it begins, it needs to create a printable sales receipt for its customers. In this exercise, you will start by adding a subreport to a main report. Next, you will add calculated controls to provide order subtotals, taxes, and totals. You will then set page breaks for printing individual receipts.

1. Download and open your Project Grader starting file.
   - *Using eLab:* Download **A8_P1_eStart** from the Assignments page. You *must* start with this file or your work cannot be automatically graded.
   - *Not using eLab:* Open **A8_P1_Start** from your **Access Chapter 8** folder.
2. Add a subreport to the **Sales Receipt** report using these guidelines:
   - Position the subreport in the detail section below the text boxes.
   - Use the **OrderDetails** report for the data.
   - Leave all linked field options set to the default values.
   - Leave the Name as *OrderDetails*.
3. Delete the OrderDetails subreport label and set the following subreport properties:

Property	Value
Width	**6**
Height	**3**
Top	**0.45**
Left	**0**
Border Style	**Transparent**
Can Shrink	**Yes**

4. In the *Detail* section of the subreport, set the Border Style property for all text box controls to **Transparent**.
5. In the *Report Footer* section of the subreport, insert a new text box control and set these properties for it:

Property	Value
Name	**Subtotal**
Control Source	**=Sum([Line Total])**
Format	**Currency**
Top	**0.10**
Left	**4.2**
Border Style	**Transparent**

6. Set these properties for the new unbound label (contains the text *Text40*):

Property	Value
Name	**Sub Total**
Caption	**Sub Total**
Width	**1**
Top	**0.10**
Left	**3.5**

7. Insert a new text box control below the Subtotal text box and set these properties for it:

Property	Value
Name	**Taxes**
Control Source	**=[Subtotal]*.075**
Format	**Currency**
Top	**0.4**
Left	**4.2**
Border Style	**Transparent**

8. Set the following properties for the new unbound label (contains the text *Text42*):

Property	Value
Name	**Tax**
Caption	**Tax**
Width	**1**
Top	**0.4**
Left	**3.5**

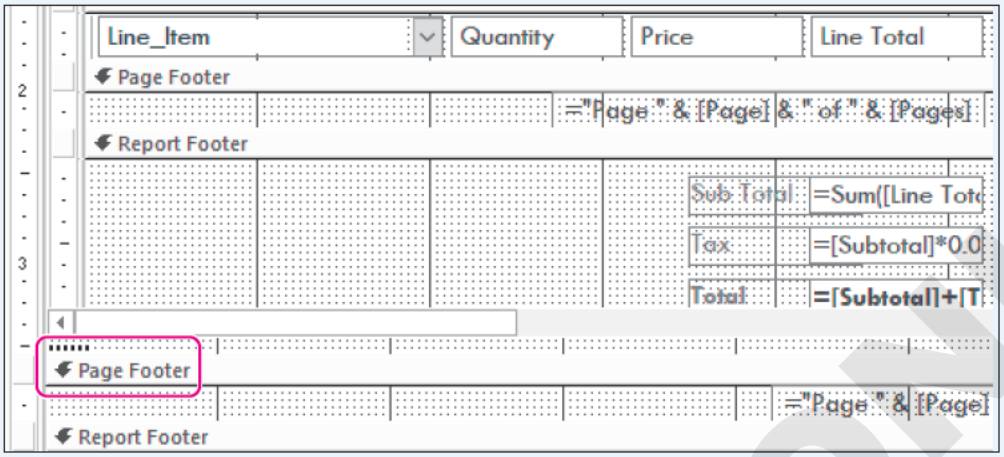
9. Insert a new text box control below the Taxes text box and set these properties for it:

Property	Value
Name	**Order Total**
Control Source	**=[Subtotal]+[Taxes]**
Format	**Currency**
Top	**0.7**
Left	**4.2**
Border Style	**Transparent**
Font Weight	**Bold**

10. Set the following properties for the new unbound label (contains the text *Text44*):

Property	Value
Name	**Total**
Caption	**Total**
Width	**1**
Top	**0.7**
Left	**3.5**
Font Weight	**Bold**

11. Insert a page break control below the subreport and just above the Sales Receipt form's page footer.



12. Save your database.
    - *Using eLab:* Save it to your **Access Chapter 8** folder as **`A8 _ P1 _ eSubmission`** and attach the file to your eLab assignment for grading.
    - *Not using eLab:* Save it to your **Access Chapter 8** folder as: **`A8 _ P1 _ Submission`**

PROJECT GRADER: A8-P2

WebVision: Create a Chart and Back Up the Database

WebVision would like you to make some database improvements. In this exercise, you will create a chart displaying the total sales for each sales rep. You will also import an invoice report, located in a backup database, and make a few changes to improve visual design.

1. Download and open your Project Grader starting file.
    - *Using eLab:* Download **A8_P2_eStart** from the Assignments page. You *must* start with this file or your work cannot be automatically graded.
    - *Not using eLab:* Open **A8_P2_Start** from your **Access Chapter 8** folder.
2. Use the **Report Design** tools to create a new report with the name: **`Annual Sales by Rep`**
3. Insert a date and time control using these guidelines and properties:
    - Use the second date option.
    - Do not include a time.

Property	Value
Width	**1**
Text Align	**Center**

4. Insert the **WebVision Logo.jpg** image from your **Access Chapter 8** folder and set the Width property to: **`1.25`**
5. Set the ReportHeader section Back Color property to **Background Form**.

6. Add a pie chart to the Detail section using these chart settings:

Setting	Value
Data Source	**Rep Sales by Customer query**
Axis (Category)	**Last Name**
Legend (Series)	**(None)**
Values (Y axis)	**Total (Sum)**
In the Format list	**Check the Display Data Label box**

7. Set these chart properties:

Property	Value
Top	**0**
Left	**0**
Width	**6.5**
Height	**5**
Legend Position	**Bottom**
Legend Text Font Size	**14**
Chart Title	**Annual Sales by Rep**
Chart Title Font Size	**20**
Primary Values Axis Format	**Currency**

8. Set the Report Width property to: **6.5**
9. Use these guidelines to import a report from another database:
   - Choose the **A8_P2_Backup** Access database located in your **Access Chapter 8** folder as the data source.
   - Choose the **Invoices** report as the object to import.
10. Save your database.
    - *Using eLab:* Save it to your **Access Chapter 8** folder as **A8 _ P2 _ eSubmission** and attach the file to your eLab assignment for grading.
    - *Not using eLab:* Save it to your **Access Chapter 8** folder as: **A8 _ P2 _ Submission**

Extend Your Skills

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

A8-E1 That's the Way I See It

You would like to modify a database report to show sales details for the Blue Jean Landscaping database. Open **A8-E1-BJL** and save it as: **A8-E1-BJLRev**

Import the Customer Sales Report from **A8-E1-BJL-Backup**. Then add the CustomerSales Subform to the Detail section of the Customer Sales Report.

A8-E2 Be Your Own Boss

As the owner of Blue Jean Landscaping, you pride yourself on creating organized and easy-to-use reports. You want to add more calculations to your report and to add page breaks for more organized viewing of the data. Open **A8-E2-BJL** and save it as: **A8-E2-BJLRev**

Open Customer Sales Report and create a new subreport from the CustomerSales Subform. Calculate the line totals for each subreport line using the formula *Price*QtySold*. Insert a page break so each customer appears on a separate page when you view the report in Print Preview.

A8-E3 Demonstrate Proficiency

The Stormy BBQ Key West store and restaurant is enjoying increased sales, so you must make some changes to the database reports to produce more useful sales results. Open **A8-E3-SBQ** and save it as: **A8-E3-SBQRev**

Import the Merchandise Sales Report from **A8-E3-SBQ-Backup**. Insert the MerchandiseSales Subform into the Detail section of the Merchandise Sales Report to add individual sale line items (choose Show each record in Merchandise using SKU). Save the subform as a report, being sure to close the subform, and then open the new subreport so the changes you make are not reflected in the subform. Calculate each line total (*ListPrice*QtySold*) in the Detail section of the subreport. Create a new report that includes a pie chart in the Detail section. Ensure the chart uses the MerchandiseSalesQuery as the data source and displays the total sales by Item Name.

Glossary

action query Performs one of four actions on a group of records (delete, update, append, or create a new table)

aggregate function Function that performs a calculation on a set of values, returning a single value

append query Query that adds a group of records from one or more sources to the end of one or more tables

calculated control Unbound text box control with a formula inserted in the control source property that references other controls within a form or report

Cascade Delete Relationship that records in a related table whenever related records in the primary table are deleted

Cascade Update Relationship that updates the value in the key field of a related table when the primary key value in the primary table is changed

chart Graphical display of data in a visual layout; create charts to better interpret the relationships between report data (can also be used with forms)

crosstab query Query that lists fields to be grouped in rows and fields to be summarized in columns so you can calculate sums, averages, counts, or totals by group and subgroup

Database Documenter Documents objects in the database so you can track changes to database design and relationships; builds an Object Definition document that provides a detailed description of each database object

delete query Query that deletes a group of records from one or more tables

Detail section Main section of a form or report that contains the text boxes that display data from underlying database tables; detail content varies from record to record

expression Combination of field names and arithmetic and logical operators required to perform the calculation; an Access formula

Expression Builder Feature that assists users in building functions via a dialog box containing a list of available fields in the current object and an array of built-in operators, functions, and expressions

find duplicates query Locates records containing duplicate field values in a single table or query datasheet

find unmatched query Locates records in one table that have no matching records in another table

group Collection of controls or records with at least one feature in common; quick forms tie all automatically inserted text boxes and corresponding labels into one group, allowing you to move the entire group but not individual controls; or, if you want to display all vendors with offices in the same state, you could group on the State field

index (database) Like a book index; its main function is to speed up database operations; an index set on key fields uses one or more hidden columns in a table for faster data retrieval

input mask Controls data formats by setting the required characters to display as users enter data, such as slashes (/) for a date field

lookup field Enables users to select a field value in one table by looking up values from another table or by selecting values from a list they create.

make table query Query that creates a new table from the selected data in one or more tables

Object Dependencies panel Allows the display of database objects that either use or are used by other objects

parameter query Query that filters records and returns only a subset that matches the value entered, delivering on-the-fly results

referential integrity Relationship protocol that maintains the validity of related data; requires that the data types of related primary and foreign key fields are the same or compatible

smart tags Indicate common actions that may be taken if certain conditions are encountered or if a control has a problem; clicking a smart tag displays a list of possible actions

subform A secondary (child) form placed on a main (parent) form, allowing the user to view and complete data entries for multiple tables through one form

subreports Display subsets of data derived from related database tables, similar to subforms

update query Query that makes global changes to a group of records in one or more tables

validation rule Field property that enables you to limit values entered in the field to reduce errors associated with data entry (e.g., limit the value typed into an Hours Worked field to less than 60)

validation text Contains instructions or valid data values to help guide the data entry personnel

Index