Microsoft Office Access 2010: Part 2

Part Number: NH91120
Course Edition: 2.1

Acknowledgements

PROJECT TEAM

<table>
<thead>
<tr>
<th>Author</th>
<th>Media Designer</th>
<th>Content Editor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pamela J. Taylor</td>
<td>Alex Tong</td>
<td>Angie French</td>
</tr>
</tbody>
</table>

Notices

DISCLAIMER
While Logical Operations, Inc. takes care to ensure the accuracy and quality of these materials, we cannot guarantee their accuracy, and all materials are provided without any warranty whatsoever, including, but not limited to, the implied warranties of merchantability or fitness for a particular purpose. The name used in the data files for this course is that of a fictitious company. Any resemblance to current or future companies is purely coincidental. We do not believe we have used anyone's name in creating this course, but if we have, please notify us and we will change the name in the next revision of the course. Logical Operations is an independent provider of integrated training solutions for individuals, businesses, educational institutions, and government agencies. Use of screenshots, photographs of another entity's products; or another entity's product name or service in this book is for editorial purposes only. No such use should be construed to imply sponsorship or endorsement of the book by, nor any affiliation of such entity with Logical Operations. This courseware may contain links to sites on the internet that are owned and operated by third parties (the "External Sites"). Logical Operations is not responsible for the availability of, or the content located on or through, any External Site. Please contact Logical Operations if you have any concerns regarding such links or External Sites.

TRADEMARK NOTICES
Logical Operations and the Logical Operations logo are trademarks of Logical Operations, Inc. and its affiliates.
Microsoft® Office Excel® 2010 is a registered trademark of Microsoft Corporation in the U.S. and other countries. Microsoft® Office Outlook® and Microsoft® SharePoint® are also registered trademarks of Microsoft Corporation in the U.S. and other countries. The other Microsoft products and services discussed or described may be trademarks or registered trademarks of Microsoft Corporation. All other product and service names used may be common law or registered trademarks of their respective proprietors.
Copyright © 2012 Logical Operations, Inc. All rights reserved. Screenshots used for illustrative purposes are the property of the software proprietor. This publication, or any part thereof, may not be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, storage in an information retrieval system, or otherwise, without express written permission of Logical Operations, 500 Canal View Boulevard, Rochester, NY 14623, (800) 456-4677. Logical Operations' World Wide Web site is located at www.logicaloperations.com.
This book conveys no rights in the software or other products about which it was written; all use or licensing of such software or other products is the responsibility of the user according to terms and conditions of the owner. Do not make illegal copies of books or software. If you believe that this book, related materials, or any other Logical Operations materials are being reproduced or transmitted without permission, please call (800) 456-4677.
## Microsoft Office Access 2010: Part 2

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advanced Form Design</td>
<td>1</td>
</tr>
<tr>
<td>Adding Controls to Forms</td>
<td>2</td>
</tr>
<tr>
<td>Creating Subforms</td>
<td>14</td>
</tr>
<tr>
<td>Organizing Information with Tab Pages</td>
<td>16</td>
</tr>
<tr>
<td>Displaying a Summary of Data in a Form</td>
<td>18</td>
</tr>
<tr>
<td>Applying Conditional Formatting</td>
<td>20</td>
</tr>
<tr>
<td>Data Validation</td>
<td>23</td>
</tr>
<tr>
<td>Field and Record Validation</td>
<td>24</td>
</tr>
<tr>
<td>Form Validation</td>
<td>31</td>
</tr>
<tr>
<td>Using Macros to Improve User Interface Design</td>
<td>37</td>
</tr>
<tr>
<td>Creating a Macro</td>
<td>38</td>
</tr>
<tr>
<td>Restricting Records Using a Condition</td>
<td>44</td>
</tr>
<tr>
<td>Validating Data Using a Macro</td>
<td>47</td>
</tr>
<tr>
<td>Automating Data Entry Using a Macro</td>
<td>50</td>
</tr>
<tr>
<td>Converting a Macro to VBA</td>
<td>52</td>
</tr>
<tr>
<td>Advanced Database Management</td>
<td>55</td>
</tr>
<tr>
<td>Linking Tables to External Data Sources</td>
<td>56</td>
</tr>
<tr>
<td>Managing a Database</td>
<td>58</td>
</tr>
<tr>
<td>Determining Object Dependency</td>
<td>61</td>
</tr>
<tr>
<td>Documenting a Database</td>
<td>64</td>
</tr>
<tr>
<td>Analyzing the Performance of a Database</td>
<td>66</td>
</tr>
</tbody>
</table>
About This Course

You've covered many of the basic functions of Microsoft® Office Access® and now you're ready to learn advanced Access features such as, database management, advanced form design, packaging a database, encrypting a database, preparing a database for multi-user access and more. Knowledge of these features separate database professionals from the casual database users or occasional designers. Today's training added to that which you've gained from the previous two days round out your Access education and provide you with marketable job skills.

You can also use the course to prepare for the Microsoft Office Specialist (MOS) Certification exams for Microsoft Access 2010.

Course Description

Target Student
Students taking this course are database administrators or prospective database administrators who have experience working with Access 2010 and need to learn advanced skills.

This course is also for students who seek the Microsoft Office Specialist (MOS) certification and who want to prepare for Exam 77-885, Microsoft Office 2010.

Course Prerequisites
To ensure your success in your course you should have experience working with Microsoft Access 2010, including a working knowledge of database design and creation, form design and creation, report design and creation, a working knowledge of database querying and the various table relationships. You can obtain this level of skill and knowledge by taking the following Logical Operations courses:

- Microsoft® Office Access® 2010: Part 1
- Microsoft® Office Access® 2010: Part 2

Course Objectives
Upon successful completion of this course, students will be able to:

- Customize a form layout to improve usability and efficiency of data entry.
- Add user interface features to validate data entry.
- Use macros to improve user interface design.
- Organize data into appropriate tables to ensure data dependency and minimize redundancy.
- Lock down and prepare a database for distribution to multiple users.
- Create and modify a database switchboard and set the startup options.
The LogicalCHOICE Home Screen

The LogicalCHOICE Home screen is your entry point to the LogicalCHOICE learning experience, of which this course manual is only one part. Visit the LogicalCHOICE Course screen both during and after class to make use of the world of support and instructional resources that make up the LogicalCHOICE experience.

Log-on and access information for your LogicalCHOICE environment will be provided with your class experience. On the LogicalCHOICE Home screen, you can access the LogicalCHOICE Course screens for your specific courses.

Each LogicalCHOICE Course screen will give you access to the following resources:

• eBook: an interactive electronic version of the printed book for your course.
• LearnTOs: brief animated components that enhance and extend the classroom learning experience.

Depending on the nature of your course and the choices of your learning provider, the LogicalCHOICE Course screen may also include access to elements such as:

• The interactive eBook.
• Social media resources that enable you to collaborate with others in the learning community using professional communications sites such as LinkedIn or microblogging tools such as Twitter.
• Checklists with useful post-class reference information.
• Any course files you will download.
• The course assessment.
• Notices from the LogicalCHOICE administrator.
• Virtual labs, for remote access to the technical environment for your course.
• Your personal whiteboard for sketches and notes.
• Newsletters and other communications from your learning provider.
• Mentoring services.
• A link to the website of your training provider.
• The LogicalCHOICE store.

Visit your LogicalCHOICE Home screen often to connect, communicate, and extend your learning experience!

How to Use This Book

As You Learn

This book is divided into lessons and topics, covering a subject or a set of related subjects. In most cases, lessons are arranged in order of increasing proficiency.

The results-oriented topics include relevant and supporting information you need to master the content. Each topic has various types of activities designed to enable you to practice the guidelines and procedures as well as to solidify your understanding of the informational material presented in the course. Procedures and guidelines are presented in a concise fashion along with activities and discussions. Information is provided for reference and reflection in such a way as to facilitate understanding and practice.

Data files for various activities as well as other supporting files for the course are available by download from the LogicalCHOICE Course screen. In addition to sample data for the course exercises, the course files may contain media components to enhance your learning and additional reference materials for use both during and after the course.

At the back of the book, you will find a glossary of the definitions of the terms and concepts used throughout the course. You will also find an index to assist in locating information within the instructional components of the book.
As You Review

Any method of instruction is only as effective as the time and effort you, the student, are willing to invest in it. In addition, some of the information that you learn in class may not be important to you immediately, but it may become important later. For this reason, we encourage you to spend some time reviewing the content of the course after your time in the classroom.

As a Reference

The organization and layout of this book make it an easy-to-use resource for future reference. Taking advantage of the glossary, index, and table of contents, you can use this book as a first source of definitions, background information, and summaries.

Course Icons

Watch throughout the material for these visual cues:

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Note" /></td>
<td>A <strong>Note</strong> provides additional information, guidance, or hints about a topic or task.</td>
</tr>
<tr>
<td><img src="image" alt="Caution" /></td>
<td>A <strong>Caution</strong> helps make you aware of places where you need to be particularly careful with your actions, settings, or decisions so that you can be sure to get the desired results of an activity or task.</td>
</tr>
<tr>
<td><img src="image" alt="LearnTO" /></td>
<td><strong>LearnTO</strong> notes show you where an associated LearnTO is particularly relevant to the content. Access LearnTOs from your LogicalCHOICE Course screen.</td>
</tr>
<tr>
<td><img src="image" alt="Checklists" /></td>
<td><strong>Checklists</strong> provide job aids you can use after class as a reference to performing skills back on the job. Access checklists from your LogicalCHOICE Course screen.</td>
</tr>
<tr>
<td><img src="image" alt="Social" /></td>
<td><strong>Social</strong> notes remind you to check your LogicalCHOICE Course screen for opportunities to interact with the LogicalCHOICE community using social media.</td>
</tr>
<tr>
<td><img src="image" alt="Notes Pages" /></td>
<td><strong>Notes Pages</strong> are intentionally left blank for you to write on.</td>
</tr>
</tbody>
</table>
Advanced Form Design

Lesson Time: 1 hour, 30 minutes

Lesson Objectives

In this lesson, you will customize a form layout to improve usability and efficiency of data entry. You will:

- Add controls to forms.
- Create a subform.
- Organize information with tab pages.
- Display data in pivot tables and pivot charts.
- Apply conditional formatting.

Lesson Introduction

In parts one and two of this course, you've learned the very important building blocks that now lead you into the more advanced features and capabilities of Microsoft® Access® 2010. One of those advanced features is creating Forms that are not only professional but also engaging to the user.

The addition of controls, such as command buttons, adds extra functionality to your forms. You will edit the tab order of fields in your forms, work with the Anchoring Tool, adjust your form’s layout, create a subform, display summary data, and apply conditional formatting to your forms.
TOPIC A

Adding Controls to Forms

You add controls to forms to perform actions, to display a subset of data, to label data, to select options or to organize information. Most controls can be either bound or unbound. A bound control means that the information displayed or the selected information is directly connected to a table. In other words, the control is bound to data in a table. Unbound controls neither select data from a table nor do they insert data into a table.

Types of Controls

You can use several different types of form controls in Access 2010 that add efficiency and a professional touch to your forms. The following table lists the various form control types.

<table>
<thead>
<tr>
<th>Control</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text box</td>
<td>Bound or unbound to data.</td>
</tr>
<tr>
<td>Label</td>
<td>Unbound descriptive text.</td>
</tr>
<tr>
<td>List box</td>
<td>Bound or unbound to data.</td>
</tr>
<tr>
<td>Command button</td>
<td>Perform actions on demand.</td>
</tr>
<tr>
<td>Check box</td>
<td>Yes/No, On/Off, True/False, 1/0-Bit data type (binary). Bound or unbound to data.</td>
</tr>
<tr>
<td>Option button</td>
<td>Yes/No, On/Off, True/False, 1/0-Bit data type (binary). Bound or unbound to data.</td>
</tr>
<tr>
<td>Toggle button</td>
<td>Yes/No, On/Off, True/False, 1/0-Bit data type (binary). Bound or unbound to data.</td>
</tr>
<tr>
<td>Option group</td>
<td>A group of check boxes, option buttons, or toggle buttons that provide a limited number of options to the Form user.</td>
</tr>
<tr>
<td>Tabbed page</td>
<td>Several pages presented as a single set.</td>
</tr>
</tbody>
</table>

Command Buttons

A command button is a control that's used to cause some action, such as closing the form you're working on or running a query. You can make a command button any size that you want and label the button with text or a picture.

Quick Start

Quick Start fields are a database design element that allows you to create parts of a table faster than you would be able to on your own. Access provides you with a list of common business-oriented field categories from which to choose when you build tables. For example, if you need to create an address entry in your table, the Quick Start Field Selector contains a pre-built one that you can add with a single click. That one click inserts columns for Address, City, State/Province, ZIP/Postal code, and Country/Region.
Figure 1-1: Locating the Quick Start table fields selection.
Some of the provided fields also include combo boxes with multiple entries as examples for each. If you prefer to create your own fields, these Quick Start samples provide you with excellent guidelines and samples for which you can use as templates.

![Quick Start Field Options](image)

**Figure 1–2: Exploring the Quick Start field options.**

<table>
<thead>
<tr>
<th>Quick Start Selection</th>
<th>Provides</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
<td>ID, address, city, state/province, zip postal code, country, and region.</td>
</tr>
<tr>
<td>Category</td>
<td>Combo box list of categories.</td>
</tr>
<tr>
<td>Name</td>
<td>Last name and first name.</td>
</tr>
<tr>
<td>Payment Type</td>
<td>Combo box list of payment types.</td>
</tr>
<tr>
<td>Phone</td>
<td>Business phone, home phone, mobile phone, and fax number.</td>
</tr>
<tr>
<td>Priority</td>
<td>Combo box list of priorities.</td>
</tr>
<tr>
<td>Start and End Dates</td>
<td>Start date and end date (date picker objects).</td>
</tr>
<tr>
<td>Status</td>
<td>Combo box list of statuses.</td>
</tr>
<tr>
<td>Tags</td>
<td>Combo box list of tags with check boxes.</td>
</tr>
</tbody>
</table>

**How to Add a Command Button to a Form**

The **Command Button Wizard** guides you through the process of adding a command button to a form.

**Note:** All of the How To procedures for this lesson are available as checklists from the Checklist tile on the LogicalCHOICE Course screen.

**Add a Command Button to a Form**

To create a command button using the **Command Button Wizard**:

1. Open the form to which you wish to add a command button.
2. Right-click the form and select **Design** view.

3. Determine where the command button is to be placed on the form. If necessary, expand the form header, stretch a margin, or lengthen the form to make room for the command button.

4. From the **Form Design Tools Design** contextual tab, in the **Controls** group, select **Button**.

5. Click the form in the location where you want the button to be created.

6. In the **Command Button Wizard**, under **Categories**, select the desired control type.

7. Under **Actions**, select the desired action the command button will perform.

8. Select **Next**.

9. Label the button. Select either:
   - **Text** and then type the desired text to place on the button.
   - **Picture** and then browse for the picture to place on the button.

10. Select **Next**.

11. Type a name for the button.

12. Select **Finish**.

### Calendar/Datepicker Control

Access 2010 has introduced some new features and has removed old or outdated ones. The Calendar control is one such feature that's no longer found in Access 2010. Microsoft replaced it with the more flexible Datepicker control. The Datepicker gives you a choice of many formats or the ability to select the one that fits into your table data or that matches your preferred style. The Datepicker can be either bound or unbound. The Datepicker is actually a text box whose attributes are adjusted such that the calendar appears when selected. The user then selects or picks a date from a pop-up calendar.

### How to Add a Datepicker to a Form

Here's how you can add a Datepicker control to a form.

#### Add a Datepicker Control

To add a Datepicker control to a form:

1. Open the form to which you wish to add a Datepicker control.
2. Right-click the form and select **Design** view.
3. Determine where on the form you would like to place the Datepicker control, adjusting the header, margins, or size of the form as needed.
4. From the **Form Design Tools Design** contextual tab, select **Text Box**.
5. Click the form in the location where the Datepicker control is to be created.
6. Right-click the **Text Box** control you just created and select **Properties**.
7. In the **Property Sheet** in the **Name** text box, type a descriptive name.
8. From the **Format** drop-down list, select the desired format for the date field. If the control is to be bound, the format must match the field type defined in the table.
9. If the Datepicker control is to be bound, from the **Control Source** drop-down list, select the appropriate source.
10. Click the text box and change the caption to something appropriate for the Datepicker control's function.
11. If necessary, change the text box alignment and position on the form to align better with the Datepicker control.
12. Switch to **Form** view.
Form Layouts

In Access, form and report layout are quite similar. Many of the techniques, tools, and settings you have learned to use when laying out forms can be applied to report design.

When you create a new form from a query or table, the form that's created looks exactly like the table or dataset from which it was created. Sometimes this simple layout works but often you'll want to change the layout to better suit screen resolution, design aesthetic or practical functionality. For example, if you create a form from a table that has 30 fields from which your users need to supply data, they'll have to scroll down the page several times to see them all. This long page format can lead to frustrated users and skipped or incorrectly answered items. Rearranging the items on a page, shrinking or expanding fields, renaming fields and placing certain fields in a side-by-side configuration can greatly enhance the accuracy of data entry and the overall efficiency of your forms.

In the following example, you see fields that can be shrunk to more accurately represent the data contained in them, you see fields that would make more sense by being placed side-by-side and others that could be placed in different areas of a form page.

![Form Layout Example](image)

**Figure 1-3: This form needs a better layout.**

This form has a more visually appealing layout. It also would make it easier for users to enter data.
How to Adjust a Form's Layout

Use these general steps to adjust a form's layout.

Adjust Form Layout

To adjust the layout of a form:

1. Open a database that contains a form.
2. If desired, make a copy of the form before modifying the form layout.
   a. In the Access Objects pane, right-click the form and select Copy.
   b. Right-click in the Access Objects pane and select Paste.
   c. Rename the new form.
3. Open the form in Design view.
4. If you want to remove all formatting from the form:
   a. Press Ctrl+A to select all of the fields on the form.
   b. From the Form Design Tools Arrange contextual tab, select Remove Layout.
5. Modify the layout in one or more ways:
   • Drag a field to a different position on the form.
   • Shrink the size of a field.
   • Expand the size of a field.
6. Save the form.
7. View the form in Layout view to determine if you need to make additional modifications to the form layout.
8. If necessary, open the form in Design view again and arrange fields and field labels until they meet your needs.
9. Save the form when you are finished making changes.
Quick Styles
Microsoft's Access 2010 designers have the busy database designer in mind with the inclusion of Quick Styles. Quick Styles are drop-down themed designs for your Form or Report command buttons that you can apply with a single click. Quick Styles are convenient and allow you to design command buttons with color and shape schemes consistent with the rest of your database.

How to Apply Quick Styles to Form Controls
Follow these steps to apply quick styles to form controls.

Applying Quick Styles to Form Controls
To apply a Quick Style to a command button:
1. Open a form or report in Design view.
2. Select or create a command button.
3. From the Format tab in the Control Formatting group, select Quick Styles.
4. Select a style from the group to apply it to the command button.
5. Save the form.
6. Switch to Form view to observe the change.

Tab Order
When you rearrange the fields on a form, depending on how significantly you moved the fields, you might have altered the field tab order. The tab order is the order in which you navigate from field to field on a form during data entry or if you were to tab through the fields using the Tab key on your keyboard.
Change Tab Order

You can change the tab order of the fields to make the form easier and more intuitive when performing data entry operations. Generally, top to bottom, left to right is the standard order. However, this might not be true for all applications, therefore Access allows you to alter tab order based upon your needs.

How to Change Form Tab Order

Here are general steps for changing tab order to make a form more usable.

Change Tab Order

To change the tab order on a form:

1. Open the form in Form view.
2. Press Tab repeatedly to note the order in which the cursor moves from field to field each time you press Tab.
3. Switch to Design view.
4. From the Form Design Tools Design contextual tab, select Tab Order.
5. In the Tab Order dialog box, either:
   • Select a field and drag it into the desired order, repeating as needed until the fields are in the desired order.
   • Select Auto Order
6. Select OK.
7. Save the form.
8. Switch to Form view and press Tab to move through the fields and verify that they are in the desired order.

The Anchoring Tool

Form anchoring allows you to set form fields at a desired screen location. Access 2010 offers nine options for anchoring forms. When you adjust the form layout, you remove the anchoring from the form. You will want to set the anchoring again after you have adjusted the size and position of the fields on the form.

Note: Be careful with the Anchoring tool as it can drastically change the look of your forms and the ability for users to use them. Top Left is usually the best position for most applications.

Some of the Anchoring options result in fields overlapping or being repositioned so that they would be difficult to use. In most cases, you will want to use the Top Left Anchoring position.

How to Use the Anchoring Tool to Set a Form's Screen Position

Here's how to use the anchoring tool to make rapid updates to form layouts.

Anchor the Screen Position of a Form

To set the position of a form with the Anchoring Tool:

Note: Be careful to check the layout of your form when you use the Anchoring Tool. It can drastically change the look of your forms and make them difficult to use. Only save a form layout when you have checked the form in Layout view and are satisfied with the layout of the fields on the form. The Top Left option is usually best for most applications.

1. Open the form in Layout view.
2. Press Ctrl+A to select all of the fields on the form.
3. Select the Form Design Tools Design tab.
4. In the Position group, select Anchoring.
5. Select the desired layout scheme.
6. Save the form.
ACTIVITY 1–1
Changing the Tab Order on a Form

Data Files
Inventory Database.accdb

Scenario
Users have complained that after a recent form redesign that they can't efficiently work with the form due to the odd tab order. Many of the users have made mistakes that require hours of corrections and data re-entry. The request is for you to fix the form so that users can enter data by simply pressing Enter after their current entry to move to the next field.

1. Remove the layout from the Inventory Database.
   a) From the C:\091003Data\Advanced Form Design folder, open Inventory Database.accdb.
   b) Save the file as My Inventory Database.accdb.
   c) Open frmInventory in Design view.
   d) Press Ctrl+A to select all of the fields in the form.
   e) On the Form Design Tools Arrange contextual tab, in the Table group, select Remove Layout.

2. Arrange the fields to match the layout in the following graphic.
a) Resize fields appropriate to the data they contain.
b) Move fields to match the layout shown in the graphic.
c) Switch to **Layout** view to verify that the size and position of the fields is appropriate.

In **Layout** view data in the fields will help you determine if the field size and layout will work well. **Design** view doesn't show you the data, so it can be difficult to know whether the size and placement of fields will work for the data in your database.

3. Change the tab order.
   a) On the **Form Design Tools Design** contextual tab, in the **Tools** group, select **Tab Order**.
   b) In the **Tab Order** dialog box, select **Auto Order** to change the tab order on the form to the standard top-to-bottom, left-to-right order.
   c) Select the grey box to the left of **TargetInventory**. This grab point is where you select a field so you can move it up or down in the **Custom Order** list.
d) Drag **TargetInventory** up so it is between **Rack** and **UnitsInStock**.
e) Select **Cancel**.

4. Save and test the form.
   a) Save the form.
   b) Switch to **Form** view.
   c) Press **Tab** several times to verify that the navigation between fields is top-to-bottom, left-to-right.

   **Note:** To explore naming conventions for Access databases and related objects, see the LearnTO **Set Up and Use a Naming Convention** presentation from the LearnTO tile on the LogicalCHOICE Course screen.
TOPIC B

Creating Subforms

In Access, you can include subforms that link to the main form to report new or supplementary data that supports the main form. In this topic, you will learn how to create subforms.

Subforms

Subforms are independent forms that you add to existing forms that hold special data or additional data that you need to work with but that you need displayed differently on the body of the main form. For example, in your `frmInventory`, the fields, `OurUnitCost` and `RetailPrice` might be fields that you want shown on the main form but simple, single value fields aren't adequate. Subforms provide this visual and practical functionality.

How to Create a Subform

Subforms help organize information in a form in the Classic Datasheet style.

Create a Subform

To create a subform:

1. Open a form in Design view.
2. Select the Form Design Tools Design contextual tab.
3. Make room on your form. For example, you might need to drag the form footer down.
4. In the Controls group, select Subform/Subreport.
   a. Click the form where you want to place the subform or subreport.
5. On the first page of the Subform Wizard, select the desired data source and then select Next.
6. Select the table or query from which you want to bind fields on the Subform and then select Next.
   a. Select fields from the available fields, then select Next.
   b. Select the appropriate field link.
7. Type a name for the new form and then select Finish.
**ACTIVITY 1–2**  
Creating a Subform

**Before You Begin**  
My Inventory Database.accdb is open.

**Scenario**  
Your manager has asked that you create a subform on the Inventory form so that users can see more details about the suppliers your company uses.

1. Place a subform on a form.
   a) Open the frmInventory form from My Inventory Database in Design view.
   b) Grab the Form Footer bar and drag it down approximately one inch to make room for the Detail section of the Inventory form.
   c) In the Controls group, select Subform/Subreport. You might need to select the More button to see this control.
   d) Click in the form where the Subform is to be placed. The Subform Wizard opens.

2. Create the subform.
   a) If necessary, select Use existing Tables and Queries and then select Next.
   b) From the Tables/Queries drop-down list, select tblSuppliers.
   c) Double-click SupplierID to add it to the Selected Fields list box.
   d) Add Company and ContactPhone to the Selected Fields list box.
   e) Select Next.
   f) Select Next to accept the default fields link.
   g) Select Finish to accept the default name and close the wizard.

3. Save and test the subform.
   a) Save frmInventory.
   b) Switch to Form view.
   c) Scroll through the records using the navigation bar of the main form, frmInventory, and observe the changes.
TOPIC C

Organizing Information with Tab Pages

It's easier for users to enter required information if it's in manageable chunks. Tab pages provide that functionality, and also give the user the ability to flip back or forward during the data entry process. In this topic, you'll learn how to create tab pages.

Tab Pages

Tab pages or tabbed pages are an organizational unit useful for collecting and aggregating similar data onto separate forms, while enjoying the ability to switch easily among open forms via tabs. Tab pages also have the effect of making your forms look cleaner and simpler to the user. For example, if you want to create a medical history form, the first tab would be titled, Patient Information, and would hold general contact information such as name, address and phone number. Subsequent forms could be titled, Insurance Information, Allergies, Family History and Personal History. This separation of information makes it easier for the user to enter required information in manageable chunks rather than a page that's effectively three feet long.

How to Use the Tab Control

Follow these steps to organize information using tab controls.

Creating Tab Controls
To create Tab controls:
1. Create a new form and switch to Design view.
2. On the Design tab, select the Tab control.
3. Click in the form at the location where the Tab control will be placed.
4. If necessary, resize the Tab control.
5. Select Property Sheet to open the Form Properties.
6. In the Property Sheet, in the Name field, enter a descriptive name.

Assigning Tab Names
To assign tab names:
1. Right-click the Page # tab and select Properties.
2. In the Property Sheet pane, from the Selection type: Page drop-down list, select Page #.
3. On the All tab, in the Name field, type the name that should appear on the tab.
4. Repeat these steps for additional tabs on the form.

Adding Fields to a Tab Page
To add fields to the Tab Page:
1. Right-click the tab and select Form Properties.
2. On the All tab, from the Record Source field drop-down list, select the table or query to use as the source.
3. On the Form Design Tools Design contextual tab, in the Tools group, select Add Existing Fields.
4. From the Fields List pane, select the desired fields and drag them onto the form. You can select multiple fields using Shift+click or Ctrl+click and drag them all onto the form at once.
5. Arrange the fields as desired.
ACTIVITY 1–3
Creating Tab Pages

Before You Begin
My Inventory Database.accdb is open.

Scenario
Your manager has decided that she wants a tabbed form to display supplier information and some limited Inventory information. She’s thinking about a tab for Supplier Info and a tab for Inventory.

1. Prepare to set Tab control on a new form.
   a) Select Create→Forms→Blank Form to create a new form.
   b) Switch to Design view.
   c) On the Design tab, select the Tab control.
   d) Click in the form at the intersection of 1" and 1".
   e) Drag the resize handle down to the intersection of 5" and 5" to resize the tab control.

2. Assign tab names.
   a) Right-click the Page 1 tab and select Properties.
   b) In the Property Sheet pane, select the Other tab.
   c) In the Name field, type Suppliers.
   d) In the Property Sheet pane, from the Selection type: Page drop-down list, select Page2.
   e) In the Name field, type Address.

3. Add fields to the Suppliers page.
   a) On the form, right-click Suppliers and select Form Properties.
   b) In the Property Sheet pane, on the Data tab, from the Record Source drop-down list, select tblSuppliers.
   c) Select Tools→Add Existing Fields.
   d) In the Field List pane, select Company, ContactPhone, FirstName, LastName, and ContactEmail.
      Remember, you can select multiple fields if you hold Ctrl and then click each of the fields.
   e) Drag the selected fields on to the Suppliers page in the form.

4. Add fields to the Address page.
   a) Select the Address tab.
   b) In the Field List pane, select City, StateProvince, Country and PostalCode.
   c) Drag the selected fields on to the Address page in the form.

5. Save and test the form.
   a) Save the form as frmSuppliers.
   b) Switch to Form view.
   c) Scroll through the records to see that the fields on the form are populated with information from tblSuppliers.
TOPIC D

Displaying a Summary of Data in a Form

Users find it easier to view numerical data in graphical form, especially when comparing numbers or looking for trends in data. Therefore, Access 2010 provides you with the ability to display data using summary tools: PivotTables, PivotCharts, PivotTable View and PivotChart View.

PivotTables

A PivotTable is a data summary tool used to summarize columns and rows of data. It provides you with a matrix view of your data that you can manipulate and customize. PivotTable data appears in the form as table style data in rows and in columns.

How to Create a PivotTable in a Form

Do the following to create a PivotTable.

Creating a PivotTable

To create a PivotTable in a form:

1. From the Navigation pane, open a form or query.
2. Select Create→Forms→More Forms→PivotTable.
3. Click in the new form you just created in the previous step to display the PivotTable Field List.
4. Drag the desired fields on to the form.
5. Save the form.

PivotCharts

PivotChart is a data analysis tool that allows you to visualize PivotTable data in a graphical format. Although the Bar type graphical data representation is the default one, you may select from an array of choices such as a Pie, Bubble, Star, Stock and others with suboptions within those such as 3-D graphs, stacked graphs, exploded, and so forth.

How to Create a PivotChart in a Form

Do the following to create a PivotChart.

Creating a PivotChart

To create a PivotChart in a form:

1. From the Navigation pane, open a form or query.
2. Select Create→Forms→More Forms→PivotChart.
3. Click in the new form you just created in the previous step to display the Chart Field List.
4. Drag the desired fields on to the form.
5. On the PivotChart Tools Design contextual tab, in the Show/Hide group, select Legend to display a legend for the graphical data.
6. Save the form.
ACTIVITY 1–4
Creating a PivotChart in a Form

Before You Begin
My Inventory Database.accdb is open.

Scenario
Your manager has requested that you supply a Pivot Chart so that she can view Units in Stock, Target Inventory and Reorder Levels for your entire inventory. She also wants to see the data grouped by order date. Additionally, she wants to be able to select these numbers for the various departments for which inventory is kept.

1. Create a PivotChart to graph units in stock, target inventory, and reorder level.
   a) From the Navigation pane, open tblInventory.
   b) Select Create→Forms→More Forms→PivotChart.
   c) Click in the new form you just created in the previous step to display the Chart Field List.
   d) From the Chart Field List pane, drag UnitsInStock onto the tblInventory chart.
   e) Add TargetInventory and ReorderLevels to the tblInventory chart.

2. Graph the data by the date it was last ordered.
   a) From the Chart Field List pane, drag LastOrdered to the Drop Category Fields Here box at the bottom of the graph.
   b) Observe the change to the graph. Instead of just one bar for each field, there are now bars for each of the dates items were last ordered.

3. Filter the data to show everything only items in the Bathroom department.
   a) From the Chart Field List pane, drag Dept to the Drop Filter Fields Here box at the top of the graph.
   b) Select the Dept drop-down arrow.
   c) Deselect All.
   d) Select Bathroom.
   e) Select OK.
   Notice that the PivotChart graph changed now that only Bathroom items are being graphed instead of all items.

4. On the Design tab, in the Show/Hide group, click Legend to display a legend for your graphical data.

5. Save the form as frmInventoryPivotChart and close the form and table.
Applying Conditional Formatting

One way to make information stand out is to apply conditional formatting when specific criteria are met. In this topic, you will apply conditional formatting to your data to make it stand out in forms and reports.

Conditional Formatting

Conditional formatting is a feature of Microsoft Access that allows you to apply special formatting to data when a particular criterion is met. The criterion is manually set by the database administrator to highlight or to differentiate data in a form or report.

To create a criterion or conditional for special formatting, you use the Conditional Formatting Rules Manager. Using the Rules Manager, you can define a range or build an expression to set up your conditions. For example, you might want to change the font color to red when an item's inventory falls below a certain number.

How to Apply Conditional Formatting

Do the following to apply conditional formatting.

Apply Conditional Formatting

To apply conditional formatting to data:
1. Open a form in Design view.
2. Select the field to which you want to apply conditional formatting.
3. Right-click the selected field and select Conditional Formatting.
4. In the Conditional Formatting Rules Manager, select New Rule.
5. Select the desired rules for conditional formatting.
6. Select OK to finish editing the conditional rules.
7. Select OK to accept the new conditional formatting rule.
ACTIVITY 1–5
Changing the Display of Data Conditionally

Before You Begin
My Inventory Database.accdb is open.

Scenario
The Sales Manager wants you to change the Inventory form to show when Units in Stock have fallen below Reorder Levels by changing the UnitsInStock field to red.

1. Apply conditional formatting to the UnitsInStock field.
   a) Open frmInventory in Design view.
   b) Right-click UnitsInStock and select Conditional Formatting.
      The Conditional Formatting Rules Manager dialog box opens.
   c) Select New Rule.
   d) Verify that Select the rule type is set to Check values in the current record or use an expression.
   e) In the Edit the rule description section set the first box to Field Value is.
   f) Set the second box to less than or equal to.
   g) In the third box, type [ReorderLevel]
   h) From the Font Color drop-down list, select Red.
   i) Select OK.
   j) Select OK to close the Edit Formatting Rule dialog box.

2. Save and test the form.
   a) Save the form.
   b) Switch to Form view.
   c) Navigate through the records.
      Any Units in Stock numbers that are less than the Reorder Level show up in red in the Form.
   d) Close the form and database.
Summary

In this module, you learned how to add controls to forms. Controls are essential to creating an easy-to-use and professional database application. The idea behind controls is to make the application pleasant for the user. Buttons, graphs, tab order and subforms make the user feel as if he or she is using a piece of commercial-quality software. You also organized forms with tabbed pages, displayed data using PivotTables and PivotCharts, and applied conditional formatting to data.

Why is tab order important in form design?

What is the purpose of adding command buttons to forms?

Note: Through the social networking tools provided on the LogicalCHOICE Course screen, you can follow up with your peers after completing the course for further discussion and supporting resources.

http://www.lo-choice.com
Data Validation

Lesson Time: 1 hour, 10 minutes

Lesson Objectives

In this lesson, you will add user interface features to validate data entry. You will:

• Constrain data with field validation.
• Perform form validation.

Lesson Introduction

Being a database user is very different from being a database designer. Users don't know what you had in mind when you created the database nor should they. A user should be able to open your database, without any knowledge of its inner workings and productively enter data into it and query it without a great deal of detail about which type of data goes where. That's your job as a designer. If you've ever worked with a database that someone else designed, you probably noticed that some of the data was out of place or incorrect for the field. For example, you might have seen a city name where a street address belonged.

Data inconsistency can lead to wasted time and wasted money for you if you have to find and correct data entries on a regular basis. The answer to the problem is to implement a data validation plan. By using data validation techniques such as input masks and lookup lists and Form controls, database users won't be able to enter inconsistent data. One example is zip code information. Zip codes are numbers only. What if you had no data validation rules in place and a user enters "abcede" into a zip code field rather than "12345?" The results can be disastrous, if that information is tied to billing or customer information. You can see how important consistent data is to a database and to its users.
TOPIC A

Field and Record Validation

Field validation refers to using the fields in the database tables to constrain the data entered into them. To use these field constraints, you must use the Field properties, use input masks or use lookup lists. Each validation method uses a different approach to data constraint. Which method you use is a matter of appropriateness for the field data. For example, to limit a zip code to five numbers or five numbers, a hyphen and four numbers (for extended zip codes) works for numeric data. However, constraining a Last Name field to letters only works for text data.

Field Properties

Every field in a table has associated properties. To view the properties for a field, switch to Design view and select a field from the Field Name list.
<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProductCode</td>
<td>Text</td>
</tr>
<tr>
<td>Dept</td>
<td>Text</td>
</tr>
<tr>
<td>SupplierID</td>
<td>Text</td>
</tr>
<tr>
<td>ItemDescription</td>
<td>Text</td>
</tr>
<tr>
<td>UnitsInStock</td>
<td>Number</td>
</tr>
<tr>
<td>TargetInventory</td>
<td>Number</td>
</tr>
<tr>
<td>ReorderLevel</td>
<td>Number</td>
</tr>
<tr>
<td>LastOrdered</td>
<td>Date/Time</td>
</tr>
<tr>
<td>Location</td>
<td>Text</td>
</tr>
<tr>
<td>Rack</td>
<td>Text</td>
</tr>
<tr>
<td>Origin</td>
<td>Text</td>
</tr>
<tr>
<td>OurUnitCost</td>
<td>Currency</td>
</tr>
<tr>
<td>RetailPrice</td>
<td>Currency</td>
</tr>
</tbody>
</table>

![Figure 2-1: Field Properties for a Number data type field.](image)

You can manually edit field properties or work through wizards for some of the options to guide you.
How to Set Field Validation Rules

Follow these steps to set field validation rules.

Set Field Validation Rules

To set a field validation rule:
1. Open a table in Datasheet view.
2. Select the field to which you want to apply the validation rule.
3. Select Table Tools Fields→Field Validation→Validation→Field Validation Rule.
4. Either:
   • Enter a validation expression
   • Use the Expression Elements to assist you in building a validation rule.
5. Select OK to accept the expression and apply the rule.

Input Masks

An input mask is a format for data entry. Its use dictates how data can be entered into a table. For example, if you want your data entry clerks to enter a phone number into a table, you could use a mask such as: (___) ___-____. This mask forces the person entering data to enter a ten-digit phone number into the field. Input masks only work with text and date/time data types. Although a phone number is a number, it isn't a proper number and so a phone number field should be defined as text in your tables. Input masks may be entered manually or with the Input Mask Wizard.

The Input Mask Wizard

The Input Mask Wizard steps you through the process of creating an input mask for your field data. Some of the more common input masks are provided for you in the wizard. Phone number is one of them. You can even try out the pre-built input masks for yourself in the wizard before you commit to one.

Input Mask Characters

How would you prevent a data entry clerk from entering letters into your phone number field, since it is a text data type? Nothing will prevent it unless you define your input mask characters such that the clerk may enter numbers only.

Access includes some pre-built input masks. The pre-built input mask for phone number is (!999) "000\-0000;;_). It uses the number 9 for the area code portion of the phone number. This means that the data entry person could choose to enter nothing for the area code. If you want to require that users enter the area code, you will have to manually replace the 9s with 0s to ensure that numbers are entered for the area code.

Each of the characters in an input mask has a specific purpose. The following table lists the characters and the function they provide in the input mask.

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>User must enter a digit (0 to 9).</td>
</tr>
<tr>
<td>9</td>
<td>User can enter a digit (0 to 9), but is not required to do so.</td>
</tr>
<tr>
<td>#</td>
<td>User can enter a digit, space, plus or minus sign. If skipped, Access enters a blank space.</td>
</tr>
<tr>
<td>Character</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>L</td>
<td>User must enter a letter.</td>
</tr>
<tr>
<td>?</td>
<td>User can enter a letter, but is not required to do so.</td>
</tr>
<tr>
<td>A</td>
<td>User must enter a letter or a digit.</td>
</tr>
<tr>
<td>a</td>
<td>User can enter a letter or a digit, but is not required to do so.</td>
</tr>
<tr>
<td>&amp;</td>
<td>User must enter either a character or a space.</td>
</tr>
<tr>
<td>C</td>
<td>User can enter a character or a space, but is not required to do so.</td>
</tr>
<tr>
<td>. , : : /</td>
<td>Decimal and thousands placeholders, date and time separators. The character you select depends on your Microsoft Windows regional settings.</td>
</tr>
<tr>
<td>&gt;</td>
<td>Converts all characters that follow to uppercase.</td>
</tr>
<tr>
<td>&lt;</td>
<td>Converts all characters that follow to lowercase.</td>
</tr>
<tr>
<td>!</td>
<td>Causes the input mask to fill from left to right instead of from right to left.</td>
</tr>
<tr>
<td>\</td>
<td>Characters immediately following will be displayed literally.</td>
</tr>
<tr>
<td>&quot;&quot;</td>
<td>Characters enclosed in double quotation marks will be displayed literally.</td>
</tr>
</tbody>
</table>

### Lookup Lists

A lookup list is a collection of values that restrict the choices for a particular field. The list source can be a query, a table or a manually created custom list. The list can display more than one column of data but only one column is used when the user selects a value. A good example is the list of states and their abbreviations. In your form, you might want to display the full state name and the abbreviation but only keep the abbreviation as the data.

#### The Lookup Wizard

The Lookup Wizard assists you in selecting a table, query or in creating your own custom list for use in data entry. If you are creating the list based on a table or query, you can specify the order in which the list is sorted. If you are creating a list from within the wizard, the items will appear in the order you enter them.

List items from a table or query can be composed of one or more columns. You can resize the default column width to a width appropriate for the data being displayed.

You can allow users to make a single selection from the list or allow them to select multiple list items. If users are allowed to select multiple items, instead of just a text list, the items will be displayed as a check box list.

### How to Use the Lookup Wizard

Use the Lookup Wizard to create lists for data entry.

#### Building a Lookup List from a Table or Query

To use the Lookup Wizard to build a list from a table or query:

1. Open a table in Design view.
2. Click in the Data Type column of the field for which you want to create a Lookup List.
3. From the Data Type drop-down list, select Lookup Wizard.
4. In the Lookup Wizard, select I want the lookup field to get the values from another table or query.
5. Select Next.
6. Select the desired table or query and then select **Next**.
7. Select the desired field (or fields) and add to the **Selected Fields** box.
8. Select **Next**.
9. If desired, specify the sort order for list box items and then select **Next**.
10. If desired, double-click the right edge of the column heading to adjust the column width to the best fit for the data and then select **Next**.
11. Type the desired label in the text box or accept the default label.
12. If you only want values from this field that are in this list and don't want users typing in other values, check **Limit To List**.
13. If users can select more than one value in the list, check **Allow Multiple Values**.
14. Select **Finish**.
15. Save the table.

**Building a Lookup List from a List you Type**

To use the Lookup Wizard to build a list from data you manually enter:
1. Open a table in **Design** view.
2. Click in the **Data Type** column of the field for which you want to create a Lookup List.
3. From the **Data Type** drop-down list, select **Lookup Wizard**.
4. In the **Lookup Wizard**, select **I will type in the values I want**.
5. Select **Next**.
6. Type the desired value in the text box under **Col1** and then press **Tab** to create a blank row.
7. Continue typing values and pressing **Tab** to create the list.
8. When the list is complete, select **Next**.
9. Type the desired label in the text box or accept the default label.
10. If you only want values from this field that are in this list and don't want users typing in other values, check **Limit To List**.
11. If users can select more than one value in the list, check **Allow Multiple Values**.
12. Select **Finish**.
13. Save the table.

**Expression Builder**

The **Expression Builder** is not a wizard. It is an open field where you manually create expressions to carry out a comparison, enforce a validation rule, calculate a value or check data integrity. Using a combination of Expression Elements, Table fields, Expression Categories, and Expression Values. You might use any or all of those components in an expression.

Expressions can be very simple or quite complex. The following list shows a very simple expression and a very complex expression:
- > 1
ACTIVITY 2-1
Validating Data with a Field Property

Data Files
C:\091003Data\Data Validation\Inventory Database.accdb

Scenario
When entering new products into the Inventory database, users often make the mistake of entering a reorder value less than five, which is the minimum reorder number for any product. Your manager has requested that you fix it on the database side so that no one may enter a number less than five for any product reorder level.

1. Display the Expression Builder dialog box to set validation on the ReorderLevel field.
   a) Open Inventory Database.accdb.
   b) Save the file as My Inventory Database.accdb
   c) Open tblInventory in Datasheet view.
   d) Click in the ReorderLevel field.
   e) Select Table Tools Fields→Field Validation→Validation.
   f) Select Field Validation Rule.

2. In the Enter an Expression to validate the date in this field text box, type >4 and then select OK.
3. Test the validation.
   a) Save and close the table.
   b) Open *frmInventory*.
   c) Create a new blank record.
   d) In the *ReorderLevel* field, type 3 and then press **Tab**.
   e) In the *ReorderLevel* field, type 5 and then press **Tab**.
      No error is displayed because 5 is a valid value for this field.

4. Close the form.
   a) If you receive an error message about not saving the form, select **Yes**.
TOPIC B

Form Validation

Form validation is similar to field and record validation in that you can manually enter expressions or build them with the Expression Builder. The difference is that you can further restrict data entry with form validation. Designers who want to add more data integrity validation may do so by using Control wizards for combo boxes, list boxes, and option groups.

Control Wizards

Control wizards are wizards that step you through the process of creating controls and restricting data along the way. You have to enable Control wizards to appear when you select a control that has an associated creation wizard. You must open a form in Design view to use the Control wizards.

The Combo Box Control

The Combo Box is a special form control that allows you to supply users with a list of choices when entering data but also allows the user to type in a value if a suitable isn't found in the list. You can select Limit to List in the Combo Box properties, which will prevent any foreign entries. Combo Boxes may be bound or unbound to data.

![Combo Box Wizard](image)

*Figure 2–2: Selecting the type of combo box for a form.*

Remember or Store Values

In the Combo Box Wizard you specify whether to remember the value for later use or store the value in a field. If you are going to be using the value the user selects in the combo box to perform a task, you will want to select the option for Access to remember the value. If you want to store the value in your database, you need to select a field in which to store the value.
How to Add a Combo Box Control

Follow these steps to add a combo box control to a form.

Adding a Combo Box Control

To add a Combo Box control to a form:

1. Open a form in Design view.
2. In the Controls group, select Combo Box.
3. Click in the form where the combo box is to be placed. The Control Box wizard opens.
4. Select I want the combo box to get the values from another table or query and then select Next.
5. Select the desired table or query and then select Next.
6. Select the field (or fields) that contain the values to include in the combo box and then select Next.
7. If desired, sort the order the list box items are listed in and then select Next.
8. If desired, adjust the width of the columns in the combo box and then select Next.
9. Select either:
   - Remember the value for later use if the value will be used later to perform a task.
   - Store that value in this field and then select the desired field if the value is to be stored in a database.
10. Select Next.
11. Type a label for the combo box or accept the default name and then select Finish.
12. Save the form.

The List Box Control

A List Box control is a control that’s similar to a combo box but does not allow values other than those displayed in the list. List boxes also display multiple rows, so their placement on a form is more critical than that of a combo box. A list box can be bound or unbound to data.

The Option Group Control

The Option Group control is a form control that is composed of option buttons, toggle buttons or check boxes. Users may only select one choice from an Option Group control. One of the options can be set as a default option. For example, if you offer customers three shipping methods and only one option is available, an option group control would be appropriate.

How to Add an Option Group Control

Follow these steps to add an Option Group control.

Adding an Option Group Control

To add an Option Group control to a form:

1. Open a form in Design view.
2. In the Controls group, select Option Group Control.
3. Click in the form where the Option Group control is to be placed. The Option Control Wizard opens.
4. Type labels for each of the options. Use Tab or the mouse to move to the next option label.
5. Select Next.
6. If you want a default option selected, select **Yes, the default choice is** and then select the option that the field will default to. If you don't want a default option selected, select **No, I don't want a default.**

7. Select **Next.**

8. Type the values for each option or accept the default values and then select **Next.**

9. Select either:
   - **Save the value for later use** if the value will be used later to perform a task.
   - **Store the value in this field** and then select the desired field if the value is to be stored in a database.

10. Select **Next.**

11. Select the type of controls to use. The options are:
   - **Option buttons**
   - **Check boxes**
   - **Toggle buttons**

12. Select the style to use. The options are:
   - **Etched**
   - **Flat**
   - **Raised**
   - **Shadowed**
   - **Sunken**

13. Select **Next.**

14. Type a caption for the option group and then select **Finish** and then save the form.

---

**Record Validation**

Record validation is similar to field validation in that you're using expressions to check data before it's saved to the database to ensure its consistency with corresponding data and to maintain reasonable conditions. Record validation occurs between fields in the same table. For example, if your Orders table contains a **ShippedDate** field and an **OrderDate** field, you want to be sure that the **OrderDate** is before the **ShippedDate**. The way to perform this check is to use record validation. Such an expression might look like: `[OrderDate] <= [ShippedDate]`. If a user accidentally enters a date older than the order date, the user will receive an error.

**How to Create a Record Validation Rule**

Follow these steps to create a record validation rule.

**Creating a Record Validation Rule**

To create a record validation rule:

1. Open a table in **Datasheet** view.
2. Select the field to which you want to apply the validation rule.
3. From the **Table Tools Fields** menu, in the **Field Validation** group, select **Validation.**
4. Select **Record Validation Rule** to open the **Expression Builder.**
5. Either:
   - Enter a validation expression.
   - Use the **Expression Elements** and **Expression Categories** to build a validation rule.
6. Select **OK** to accept the expression and apply the rule.
ACTIVITY 2–2
Using a Combo Box Control to Limit Option Values

Before You Begin
The My Inventory Database frmInventory form is open in Design View.

Scenario
When users enter new items into My Inventory Database, they often don't enter the correct storage or display location for the item. Create a list of all possible locations so that the users may select the correct one. You want to enable wizard controls to assist you when you add controls to forms.

1. Enable Control wizards.
   a) On the Form Design Tools Design tab, in the Controls group, select the More button.
   b) If necessary, select Use Control Wizards.

2. Add a Combo Box control to the form.
   a) Delete the Location field from the form.
   b) In the Controls group, select Combo Box.
   c) Click the form where you deleted the Location field to add the Combo Box control to the same location in the form.

3. Use the Combo Box Wizard to add values to the Location field.
   a) In the Combo Box Wizard select I will type in the values that I want and then select Next.
   b) In first text box, type Showroom and then press Tab.
   c) Add additional values: Basement, Mill, Shed A, Shed B, Shed C and then select Next.
   d) Select Store that value in this field and then from the drop-down list select Location.
   e) Select Next.
   f) In the text box, type Location as the label name and then select Finish.

4. Save and test the form.
   a) Save the form.
   b) Switch to Form view.
   c) Select the Location field drop-down arrow to verify the options you added are listed.
Summary

In this module, you learned various data validation techniques to increase the integrity of your data and to minimize mistakes and errors in data entry. Data validation rules might slow the data entry process but it will ensure accuracy.

Why should you apply data validation rules to every data point in your databases?

Is it reasonable to use form validation and record validation on the same table of data? If so, why?

Note: Through the social networking tools provided on the LogicalCHOICE Course screen, you can follow up with your peers after completing the course for further discussion and supporting resources.
Using Macros to Improve User Interface Design

Lesson Time: 1 hour

Lesson Objectives

In this lesson, you will use macros to improve user interface design. You will:

- Create macros.
- Restrict records using a condition.
- Validate data using a macro.
- Automate data entry using a macro.
- Convert a macro to VBA.

Lesson Introduction

You have seen how you can make data entry easier for users with things like lookup lists, combo box controls, and validation. Another useful feature in Microsoft® Access® is to use macros to perform repetitive tasks or actions that are difficult for users to perform.
TOPIC A

Creating a Macro

In this topic, you will learn to create macros to automate actions, to trigger events, to restrict records, to validate data, and to automate data entry. You will also create standalone macros and embedded macros.

Macros

Macros are pseudo-programs that perform a list of actions based on a set of instructions. Programmers prefer Visual Basic for Applications (VBA) and see macros as a simple alternative or a shortcut to real programming. This opinion has persisted since macros first appeared in Access 1.0. Alternatively, macros are a good option to learning a complex programming language to perform relatively simple tasks. You can use macros to perform repetitive tasks or tasks that require an action that isn't easily duplicated by a user. Some examples of useful macros are opening and closing database objects, filtering forms and reports, starting wizards and dialogs and to perform some action on startup.

There are two general types of macros: event (embedded) macros and named (standalone) macros. Event macros are those that are triggered by an action. Named macros are macros that are called by name.

The Macro Builder Window

The Access 2010 Macro Builder or macro designer uses formal program construction, known as top-down flow. Top-down refers to the concept that a program begins at the top of the page and parses down the page in normal left-to-right reading fashion.
When you open the Macro Builder, you're ready to begin adding actions. There are two ways to add actions to your macros. The first is to use the Add New Action drop-down list, select an action from the list and then enter any required conditions or arguments into provided fields. The second is to use the action catalog to select macro actions to add to your macros. Expand the Action categories until you find the action you want and drag it into the Macro pane or double-click the action to add it into the Macro pane.

When you first open the Macro Builder, you might not see Add New Action in the drop-down list, but when you click into that pane, the words will display. You can reposition actions by dragging them to the new location once they are placed into the Macro pane. Alternatively, you can use the arrows to move actions up or down in the Macro pane.

*Figure 3-1: The Macro Builder window and Action Catalog.*
To remove an action from the Macro pane, click the Close (X) control. You edit a macro action by giving it focus. Once you change focus to another action, any arguments or conditions are shown for easy reference.

**Macro Builder Enhancements**

Microsoft developers redesigned and enhanced the Access 2010 Macro Builder. The redesign gives database designers the ability to create robust applications, to reduce coding errors and to be more productive. The purpose behind the redesign is to make it easier to translate macro actions into VBA.

Some of the redesign includes:

- Intellisense for expressions
- Action catalog
- Readable program flow and logic
- Ability to create more complex logic
- Macro reuse
- XML tie-ins

The design of the Macro editor has changed from a three-column: Condition, Action, Arguments template into a more open text editor look and feel. You now have an action catalog from which to pull standard programming elements. Don't worry though, macros are still easy to create and use.

**Macro Actions**

The purpose of a macro is to cause something to happen—an action. Therefore, macros are made up of actions. You can have as few as one action and a maximum of 999 actions in a single macro.

**Types of Macro Actions**

There are eight major macro action types. Those eight types are action groups whose members have similar functions. These macro actions are found in the action catalog. The following table identifies the eight macro types and their general functionality.

<table>
<thead>
<tr>
<th>Macro Action</th>
<th>Functionality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Entry Operations</td>
<td>Delete, edit, and save records.</td>
</tr>
</tbody>
</table>
Macro Action | Functionality
--- | ---
Data Import/Export | Outlook and Word import and export options.
Database Objects | GoTo, Open, Print, Select.
Filter/Query/Search | Filers, searches, and shows records.
Macro Commands | Cancel, run, set, start, or stop macros.
System Commands | Beep, close, houseglass, and quit.
User Interface Commands | MessageBox, BrowseTo, NavigateTo, Undo, SetMenuItem.
Window Management | Close, maximize, minimize, move, or restore a window.

Action Arguments

*Argument* is a programming term that refers to a bit of information that directs an action. When you give someone directions to the mall, you might say something like, “Go two miles East and then go one mile South.” The action is “Go” and the arguments are two miles, East, one mile, and South. The arguments for the Go action are distance and direction.

Macro action arguments are the parameters an action uses to perform that action. For example, if you use the OpenForm action, one of the arguments must be the name of the form to open. For the GoToRecord Action, one of the arguments is the name of the form, table or query you want to use as a record source. Another argument for the GoToRecord action is which record you want to go to: previous, next, first, last, new or go to a specific record expressed as an offset from the current record.

Object Events

A macro is a collection of code that performs some action or actions for you. You can use a Macro in a number of ways to perform those actions. An object event is an action triggered by a user on a database object—objects such as forms, reports and controls.

![Event options for a form field.](image-url)
### Event Group Description

<table>
<thead>
<tr>
<th>Event Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Window Events</td>
<td>Opening, closing, or resizing a window.</td>
</tr>
<tr>
<td>Data Events</td>
<td>Updating or deleting records or any data.</td>
</tr>
<tr>
<td>Focus Events</td>
<td>Activating, entering, or exiting a field.</td>
</tr>
<tr>
<td>Keyboard Events</td>
<td>Pressing or releasing keys.</td>
</tr>
<tr>
<td>Mouse Events</td>
<td>Clicking or double-clicking the mouse.</td>
</tr>
<tr>
<td>Print Events</td>
<td>Format and print.</td>
</tr>
<tr>
<td>Error and Timing Events</td>
<td>An error or a lapse of time between records.</td>
</tr>
</tbody>
</table>

### Events and Macros

When you attach a macro to an event, you're associating macro actions with some user action. For example, when a user tabs from one field to another, the act of pressing the Tab key on the keyboard to move from one field to another is an event. Attaching a macro to that event can perform some automated action, such as closing the form, opening another form or setting the Recordset to New so that you can enter a new record automatically.

### How to Attach a Macro to an Event

Follow these steps to attach a macro to an event.

**Note:** All of the How To procedures for this lesson are available as checklists from the Checklist tile on the LogicalCHOICE Course screen.

#### Attaching a Macro to an Event

To attach a macro to an event:

1. Open a form in **Design** view.
2. Select the field that you want to use for the macro event.
3. Select the **Event** tab on the field **Property Sheet**.
4. Select an event.
5. From the drop-down list, select the macro.
6. Save the form.
ACTIVITY 3–1
Creating a Macro to Open a New Record on a Form

Data Files
C:\091003Data\Data Validation\Using Macros to Improve User Interface Design.accdb

Scenario
Users are having difficulty creating a new record in the Inventory form. They don't know about the New Record button near the bottom of the screen, so the Data Entry Manager wants you to create a Macro that opens the Inventory form, frmInventory, ready to accept a new record.

1. Create a macro for the frmInventory form to create a new blank record.
   a) Open Inventory Database.accdb.
   b) Save the file as My Inventory Database.accdb.
   c) Select Create→Macros & Code→Macro.
   d) From the Add New Action drop-down list, select the action OpenForm.
   e) In the Form Name field drop-down list, select frmInventory.
   f) Set the View property to Form.
   g) Set the Data Mode to Edit.
   h) Set Window Mode to Normal.
   i) From the Add New Action drop-down list, select GoToRecord.
   j) In the GoToRecord action, for the Object Type, select Form.
   k) For the Object Name, select frmInventory.
   l) For the Record, select New.

2. Note: When you save standalone macros, they appear in the newly created Macro group in the All Access Objects navigation pane with your other objects such as forms, reports, queries, and tables.

Save and test the macro.
   a) Save the macro as mcrInventoryNew.
   b) Select Database Tools→Run to test the macro. It should create a new blank record.
   c) Close the form and the macro.
TOPIC B

Restricting Records Using a Condition

At the beginning of this lesson, you learned that the Access 2010 Macro Builder's design is very different from previous versions. In the old versions, the Macro Builder had three columns: Actions, Conditions and Arguments. We've covered actions and arguments so now it's time to look at conditions. Conditions are usually the result of an If statement. If this condition exists, then perform some action based on that condition. If the condition does not exist, then either take no action or take a different action. Conditions can be further qualified with a Where condition to filter results.

Macro Conditions

A macro condition is an expression that enables a macro to perform a task if a specific set of circumstances exist. The macro looks for True/False conditions to take an action. If a condition is True, then the macro takes an action. If False, the macro takes a different action. A single condition can control more than one action.

For example, if you receive orders from customers by phone and you have a particular customer that's behind in payment, then you would like to know that without having to search through records. As soon as you enter the customer name, you want a warning to appear that tells you that this account is frozen until a payment is made. This is a very good example of how a macro could work based on conditions.

If one of your tables includes a column that contains Accounts Payable information and your customer's status show >90 (more than 90 days without payment), then you could create a MessageBox that appears when that customer's account is activated with an order in progress.

The Where Condition

The Where condition filters and selects records in forms and reports. You use the Where condition in the OpenForm and OpenReport macro actions. The Where condition can be entered manually or built using the Expression Builder.

![Figure 3-4: The OpenForm macro action with Where Condition.](image)

How to Use a Where Condition in a Macro

Follow these steps to use a Where condition in a macro.
Using a Where Condition in a Macro

To use a Where condition in a macro:

1. Create a new macro.
2. From the Add New Action drop-down list, select OpenForm.
3. In the Where Condition text box, enter the condition to evaluate. Alternatively, you can use the Expression Builder to create the condition.
4. When the Where condition has been created, select OK.
5. Save the macro.
ACTIVITY 3–2
Using the Where Condition to Restrict Data

Before You Begin
My Inventory Database.accdb is open.

Scenario
Your manager wants to be able to quickly view all Inventory items that are running low (fewer than nine items in stock). You determine that a good solution for this request is to create a Macro that filters the records in the frmInventory form.

1. Create a macro that opens frmInventory.
   a) Select Create→Macros & Code→Macro.
   b) From the Add New Action drop-down list, select OpenForm.
   c) From the Form Name drop-down list, select frmInventory.

2. Use Expression Builder to set the condition to alert the user if stock is at or below 9 units.
   a) To the right of the Where Condition field, select the Expression Builder button.
   b) In Expression Builder, in the Expression Elements pane, expand My Inventory Database.accdb.
   c) Expand Tables to view all of the tables in the database.
   d) Select tblInventory.
   e) In the Expression Categories pane, double-click UnitsInStock.
   f) In the Enter an expression to carry out an action or execute logic text box, at the end of the expression that was built with your selections, type <=9 and then select OK.

3. Save and test the macro.
   a) Save the macro as mcrInventoryCheck
   b) Run the macro.
   c) Verify that the macro displayed the frmInventory form and shows 14 records that met the conditions set in the macro.
   d) Close the form and the macro.
TOPIC C

Validating Data Using a Macro

As previously discussed, data integrity is extremely important. Without accurate data, your database loses value and has to be cleaned up, which consumes time and is quite costly. Creating a macro to prompt the user to enter correct information ensures data integrity and maintains your database's value.

Event Properties for Data Validation

There are four comment event properties used to trigger data validation.

<table>
<thead>
<tr>
<th>Event Property</th>
<th>When Execution Will Occur</th>
</tr>
</thead>
<tbody>
<tr>
<td>Before Update</td>
<td>Before the entered data is updated.</td>
</tr>
<tr>
<td>After Update</td>
<td>After the entered data is updated.</td>
</tr>
<tr>
<td>Before Insert</td>
<td>After a new record is typed in.</td>
</tr>
<tr>
<td>On Delete</td>
<td>On delete request, but before deletion.</td>
</tr>
</tbody>
</table>

Macro Actions for Data Validation

When validating data, macros are likely to use certain actions. The following table lists macro actions for data validation.

<table>
<thead>
<tr>
<th>Action</th>
<th>Action Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cancel Event</td>
<td>Prevents the user from creating a new record unless certain conditions are met.</td>
</tr>
<tr>
<td>GoToControl</td>
<td>Specifies the data insertion point on a form.</td>
</tr>
<tr>
<td>MessageBox</td>
<td>Displays a custom message to the user.</td>
</tr>
</tbody>
</table>

Embedded Macros

Embedded macros are macros that are part of an event property. They are not standalone database objects. They can only be accessed from the event property of the object they are attached to. You can embed macros in event properties of forms, reports, and controls.
ACTIVITY 3–3
Using a Macro to Validate Data

Before You Begin
My Inventory Database.accdb is open with the frmInventory form open in Design View.

Scenario
Your manager is happy with your progress on My Inventory Database but there are still a lot of informational fields that are being left blank when data entry clerks get in a hurry. Your manager wants to ensure that the SupplierID field isn't left blank. Each supplier must have an ID before the record is saved.

1. Open the Macro Builder to begin creating a macro for the SupplierID field that runs On Exit.
   a) In the frmInventory form, select SupplierID.
   b) Display the Property Sheet pane.
   c) In the Property Sheet pane, select the Event tab.
   d) In the On Exit field, select the Builder button.
   e) In the Choose Builder dialog box, with Macro Builder selected, select OK.

2. Create a macro to test whether a supplier has been selected.
   a) From the Add New Action drop-down list, select If.
   b) In the If field, type IsNull([SupplierID])=True
3. Create a custom message to display if the user doesn't select a supplier.
   a) From the Add New Action drop-down list within the If statement, select CancelEvent.
   b) From the Add New Action drop-down list within the If statement, select MessageBox.
   c) In the Message field of the MessageBox action, type *You must select a Supplier ID from the list*

4. Add actions to the macro to force the user to select a supplier.
   a) From the Add New Action drop-down list within the If statement, select GoToControl.
   b) In the ControlName field of the GoToControl action, type *SupplierID*

5. Save and test the macro.
   a) Save and close the macro.
   b) Verify that Embedded Macro is now listed for the On Exit property.
   c) Save the form.
   d) Display frmInventory in Form view.
   e) In a new record, in the Product Code field, type *bathhw-90* then press Tab to move through the rest of the fields.
   f) When you attempt to press Tab to move past the SupplierID field, you should receive the message you created in the macro.
   g) Select any supplier from the list and finish creating the new record and save it.
TOPIC D

Automating Data Entry Using a Macro

Making data entry easy for your users through automation not only increases the accuracy of your data but it also increases the speed at which your users can enter data. For example, if you have your users enter part numbers, the price can be automatically entered based on the part number. You can employ macros to do this work for you by associating them with events.

Event Properties for Automating Data Entry

The Event property determines when the event launches the macro.

<table>
<thead>
<tr>
<th>Event Property</th>
<th>Execution Timing</th>
</tr>
</thead>
<tbody>
<tr>
<td>On Enter</td>
<td>At arrival time onto the control, when the user tabs into the field.</td>
</tr>
<tr>
<td>Before Update</td>
<td>Before control data is updated.</td>
</tr>
<tr>
<td>After Update</td>
<td>After control data is updated.</td>
</tr>
<tr>
<td>On Exit</td>
<td>Upon leaving a control, when the user tabs away from the field.</td>
</tr>
</tbody>
</table>

**Note:** For validating data in data entry forms, it is often best to use the On Exit event so that when the user tabs away from the field, they receive immediate feedback on whether their entry is incorrect.

Macro Actions for Automating Data Entry

Macro actions set values and locations for values on forms for automated data entry.

<table>
<thead>
<tr>
<th>Action</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>SetValue</td>
<td>Automatically enter a value into a field. Field name and data value are entered as arguments for SetValue.</td>
</tr>
<tr>
<td>GoToControl</td>
<td>An action that moves the cursor to the field for data entry.</td>
</tr>
</tbody>
</table>
ACTIVITY 3–4
Automating Data Entry with a Macro

Before You Begin
Have My Inventory Database.accdb open and the frmInventory form in Design View.

Scenario
Your manager wants a further tweak of the data entry system to streamline some of the operations and to ease the number of data points users have to enter manually. She wants you to have the database automatically set a value for the Department (Dept) as “Bathroom” for each item that has a product code that begins with “bathhw”.

1. Select the ProductCode field and make sure that the Property Sheet is visible.
2. Click the Builder icon in the On Exit event field of the Property Sheet and launch the Macro Builder.
3. In the Macro Builder, select the If action from the Add New Action drop-down list.
4. Enter [ProductCode] Like “[bathhw]***” into the If field.
5. From the Add New Action drop-down list above the EndIf select the SetValue action. If you don't see SetValue in your list of actions, in the Show/Hide group, select the Show All Actions button and it will appear in the list.
6. For the Item argument, enter [Dept] and for the Expression enter “Bathroom”.
7. Save and close the macro.
8. Save the form.
9. Test the form. Switch to Form View and test the form by creating a new record and entering bathhw-95 as the ProductCode.
   a) Switch to Form View and click to deselect the first field and create a new record.
   b) Create a new record.
   c) In the ProductCode field, enter bathhw-95 and view the results.
10. Save the record and close the form.
TOPIC E

Converting a Macro to VBA

As you learn to use macros in Access, you will also want to know how to convert a macro to VBA format.

Macros and VBA

To view the code for a macro, the only programmatic element of your databases thus far, you must first convert the macro to Visual Basic for Applications (VBA) code. Visual Basic is the preferred procedural language for all of the Microsoft Office products. But, for simplicity, you can continue to use macros and then convert the macro code to VBA.

How to Convert a Macro to VBA

Follow these steps to convert a macro to VBA.

Converting a Macro to VBA

To convert a macro to VBA:

1. Open the Macro that you want to convert in Design view.
2. From the Design tab, select Convert Macros to Visual Basic.
3. In the Convert Macro dialog box, select the Convert button.
4. Leave the default selections checked.
5. Select OK in the Conversion Finished dialog box.
7. Close the Code view and close the Microsoft VBA window to return to the database.
ACTIVITY 3–5
Converting a Macro to VBA

Before You Begin
My Inventory Database.accdb is open.

Scenario
Your manager realizes that macros are very handy but are not 100-percent efficient. Additionally, she realizes that your department requires a certain amount of error checking to satisfy corporate standards for development. She wants you to convert your macros to VBA code.

1. Open the mcrInventoryCheckMacro in Design view.

2. From the Design tab, select Convert Macros to Visual Basic. Then select the Convert button on the Convert Macro dialog box. Then select OK in the Conversion Finished dialog box.

   **Note:** Leave the default selections checked.

3. In the Microsoft Visual Basic for Applications window, select View→Code.

4. Close the Code view and close the Microsoft VBA window to return to the database.

5. Close the macro and the database.
Summary

In this module, you used macros to assist users in their data entry. You created a macro to open a new record in a form. You used a macro to display records that met a specific criteria. You also validated whether a user entered required data and automatically filled in a field based on data entered in a different field. All of these macros help users enter data more easily and accurately.

Why would you use a standalone macro versus an embedded macro?

Why would you use a Where Condition?

Note: Through the social networking tools provided on the LogicalCHOICE Course screen, you can follow up with your peers after completing the course for further discussion and supporting resources.
Lesson Objectives

In this lesson, you will organize data into appropriate tables to ensure data dependency and minimize redundancy. You will:

- Link tables to external data sources.
- Manage a database.
- Determine object dependency.
- Use Database Documenter to document your database.
- Analyze database performance.

Lesson Introduction

Creating tables, queries, forms, reports and macros is only part of the job of managing a database or multiple databases. As a database designer and administrator, you must perform many non-creative, non-design tasks. Linking your databases to external data sources, managing the links to those external data sources, performing database backups, performing compact and repair on your databases, determining object dependency, database documentation and analyzing database performance. As the database administrator, you have the responsibility of guaranteeing database integrity through good design, clean data and a healthy database. This lesson guides you through those processes.
TOPIC A

Linking Tables to External Data Sources

Access includes a feature that allows you to link your tables to external data sources. One major advantage of this capability is that when those external files are updated, Microsoft® Access® automatically sees the updates without intervention. This removes responsibility of third party data from the local Access administrator and from its users. It also expands the possibilities for accessing third-party data that would consume valuable data entry time and valuable space within the Access database.

External Data Sources

Linking your Access tables to external data sources might seem esoteric at first, but it is a very important and advanced function. One of the most popular external data sources is an Excel spreadsheet. Unfortunately, Excel spreadsheet use is epidemic in companies; therefore, it makes sense for Microsoft to allow Access database designers to connect to these ubiquitous entities as an external source of data.

Excel isn't the only choice for external data sources. You may also attach to other external sources including other Access databases, SharePoint lists, delimited or fixed-width text files, ODBC databases, HTML documents, Outlook folders, dBASE files and Paradox files.

The Linked Table Manager

The Linked Table Manager is a utility that refreshes your link to external data. This functionality is useful if an external data source has been updated. The Linked Table Manager displays a list of all linked tables, and you can select the ones you wish to refresh.
ACTIVITY 4-1
Using an External Data Source

Data Files
C:\091003Data\Advanced Database Management\Inventory Database.accdb
C:\091003Data\Advanced Database Management\Faucets.xlsx

Scenario
One of your bathroom faucet suppliers created an Excel spreadsheet catalog for its new line of faucets. They want to supply their retailers with this new catalog. Your manager requests that you use the catalog in your database. You decide that it would be more efficient to use the spreadsheet as an external data source rather than import the data into a table, since suppliers often change their offerings on a seasonal basis.

1. Open the Inventory Database.accdb.
   a) Save the file as My Inventory Database.accdb

2. Link the database to the Excel spreadsheet Faucets.xlsx.
   a) Select External Data→Import & Link→Excel.
      b) Select Browse.
      c) Navigate to C:\091003Data\Advanced Database Management\Faucets.xlsx, select Faucets.xlsx and then select Open.
      d) Select Link to the data source by creating a linked table.
      e) Select OK.
      f) In the Link Spreadsheet Wizard select Sheet1 and then select Next.
      g) Check First Row Contains Column Headings and then select Next.
      h) Name the linked table tblFaucets and then select Finish.
      i) Select OK to acknowledge that your linked table has been created.

3. View the contents of the linked table.
   a) In the Navigation pane, open tblFaucets to view the table contents.
   b) Close the tblFaucets table and the database.
Managing a Database

A database is a tool and like any other tool, you must perform some regular maintenance and housekeeping on your database to maintain data integrity and overall database health. In this topic, you'll learn two basic Access database management tasks: database backup and how to compact and repair a database.

Open a Database in Exclusive Mode

Any time that you perform actions or maintenance on a database, you need to open it in exclusive mode. Exclusive mode means that no one else can access the database but you. This is very important in networked, multi-user databases. Exclusive mode prevents other users from using the database when performing maintenance to prevent database or record corruption. When your maintenance is complete, remember to close the database.

How to Open a Database in Exclusive Mode

Before you perform maintenance or other exclusive activities, you need to open the database in exclusive mode.

Opening a Database in Exclusive Mode

To open a database in exclusive mode:
1. Select File→Open.
2. Select the desired database.
3. From the Open drop-down list, select Open Exclusive.

Database Backup

If you've worked with computers for very long, you understand the need for good backups. A backup is a copy of a production database. It is a snapshot of the data and the objects at the time of the backup. Any data entered or any changes made since that backup won't be reflected in the previous backup. This is one reason why database managers perform daily backups. And, the really paranoid database managers perform multiple backups during a day.

There are at least two backup methods for Access databases. One is a simple copy of the database.accdb file to a new location, which is an exact copy of a point in time or snapshot. You create this copy as you would any other file.

The second, preferred method is accomplished through Access itself. Access creates a copy of the current database with a time stamp of your current backup. The backup copy receives its name from the current date. For example, if today's date is December 1, 2012, then your backup for the Inventory Database receives the name, Inventory Database_2012-12-01.accdb.

How to Create a Database Backup

Follow these steps to create a database backup.
Creating a Database Backup

To create a database backup:

1. Open the database to be backed up.
2. Select File→Save & Publish.
3. In the Save Database As pane, select Backup Database.
4. Select Save As.
5. Browse to the location where you want to save the backup copy and then select Save.

The Compact and Repair Database Option

As a database is used, records are inserted, changed, deleted; tables are ever changing, new forms are being created, reports are generated, macros are created, edited and deleted. A database is a dynamic entity. During the course of its use, the database can become fragmented and possibly corrupted. Many of the problems associated with working with databases can be mitigated by performing regular backups and performing simple maintenance known as compact and repair.

The reasons for accidental database corruption are many but common ones are that during a data entry session, network connectivity is lost to a shared database or there is a problem on the physical disk where the database resides. **Compact and Repair Database** resolves many of these problems by reducing the file size of your database by stitching it back together physically and by checking its file integrity. **Compact and Repair Database** is not a substitute for backups; it is a separate part of maintenance that should be performed on a regular basis. Very busy databases should have **Compact and Repair Database** performed on a daily basis.

If your database is in serious need of maintenance, you might notice status changes on the status bar such as **Removing temporary objects**. However, there are no other prompts or dialog boxes for you to interact with. Everything occurs in the background.
ACTIVITY 4–2
Compacting and Repairing a Database

Data Files
C:\091003Data\Advanced Database Management\Inventory Database.accdb

Scenario
After taking the Access 2010 course, you realize how important it is to backup and perform a compact and repair database maintenance on your databases. To that end, you decide to perform both on a regular basis.

1. View My Inventory Database in exclusive mode.
   a) Select File→Open.
   b) Select My Inventory Database.
   c) From the Open drop-down list, select Open Exclusive.

2. Create a backup of My Inventory Database.
   a) Select File→Save & Publish.
   b) Select Backup Database.
   c) Select Save As.
   d) Select Save.

3. Compact and repair My Inventory database.
   a) In Windows Explorer navigate to C:\091003Data\Advanced Database Management.
   b) Record the size of the My Inventory Database.accdb file. ____________________
   c) Switch back to Access and select Database Tools→Compact and Repair Database.
      Because this is a relatively small database, the process runs very quickly.
   d) Switch to File Explorer and record the size of the My Inventory Database.accdb file.
      ____________________
      The file should be smaller now that it has been compacted and repaired.
TOPIC C

Determining Object Dependency

All objects in Access have dependencies. Sometimes an object depends on the object you are referring to and sometimes the object is depended upon by other objects. In this topic, you will examine both types of dependencies.

Object Dependency

Object dependency in a database can become quite complex but it's necessary to understand object dependency in cases of advanced troubleshooting. Tables have relationships with each other, forms have dependencies on queries and on tables. Reports have dependencies on tables and queries and macros have their dependencies. All database objects can have dependencies.

Access defines two types of dependencies for objects: **Objects that depend on me** and **Objects that I depend on**. Where me or I is the object you select.

Object Dependencies Task Pane

You do not have to open an object to check its dependencies. You can check them through the database tool **Object Dependencies**.

![Figure 4-1: The Object Dependencies pane showing Objects that depend on me for tblInventory.](image_url)
How to Check Object Dependencies

Follow these steps to check an object's dependencies.

Checking an Object's Dependencies

To check an object's dependencies:

1. In the Navigation pane, select an object for which you want to check the dependencies.
2. Select Database Tools→Object Dependencies.
3. Select either:
   - Objects that depend on me
   - Objects that I depend on
4. In the Object Dependencies pane, view the dependencies.
5. If necessary, expand items in the list to see the dependencies.
6. If you select a different object to view, select Refresh.
ACTIVITY 4–3
Viewing Object Dependency

Before You Begin
My Inventory Database.accdb is open.

Scenario
One of your users has complained that when she opens the Inventory form, it looks as if the Query underlying the SupplierID field is coming from another table and feels that something is wrong. Use the Object Dependencies task pane to view dependencies and put her mind at ease.

1. View dependencies for frmInventory.
   a) In the Navigation pane, select, but do not open frmInventory.
   b) Select Database Tools→Relationships→Object Dependencies.
      Notice that no tables depend on this form.
   c) Select Objects that I depend on.
      Notice that the form does rely on the tblSuppliers and tblInventory tables.
   d) Expand tblInventory and tblSuppliers.
   e) Close the Object Dependencies pane.

2. Open frmInventory and view the Property Sheet to see the relationship to SupplierID.
   a) Open frmInventory in Design view.
   b) Select the SupplierID field.
   c) In the Property Sheet pane select the Data tab.
   d) Notice that the Row Source is the table tblSuppliers.
   e) Close the form.
TOPIC D

Documenting a Database

You understand what your Inventory Database does for you, since you designed it. You understand what all of the embedded macros were for and you certainly recall all of your validation rules down to the last expression. The problem is that sometimes database designers don't maintain a database throughout its entire life cycle. Those who inherit your masterpiece might not understand the purpose of the validation Macro or the other nuances of your work. Therefore, it's necessary to document these items. Documentation isn't as exciting as creating a form that queries inventory information, makes automated updates to fields and then calculates profit and loss but it's very important. In this topic, you'll learn to use the Database Documenter tool.

The Database Documenter

The Database Documenter is an automated tool to document every detail of your database schema. This tool, like the Backup tool, is a snapshot in time. If you change one thing in your database, such as add a form, you need to update the documentation for it.

Access creates a very detailed documentation which can take several minutes to complete if you have a lot of components in the database. When Database Documenter is finished, the documentation is displayed on screen. The documentation is not automatically saved in your database or as an external file. You can save the documentation as a PDF or XPS file.

How to Document a Database with Database Documenter

Follow these steps to document a database with Database Documenter.

Documenting a Database with Database Documenter

To document a database with Database Documenter:

1. Select Database Tools→Analyze→Database Documenter.
2. In the Documenter dialog box, on the All Object Types tab, select Select All.
3. Select the object to document.
4. Select Options to set the documentation detail.
5. Select OK to close the Print Table Definition dialog box.
6. Select OK to create the documentation.

Saving a Database Documenter Report

To save the documentation after you run Database Documenter:

1. Right-click the documentation and select Export.
2. Select the desired export option.
3. Browse to the location where you would like to save the documentation.
4. Type a name for the document and then select Publish to save it.
5. Select Save export steps.
ACTIVITY 4–4
Using the Database Documenter

Before You Begin
My Inventory Database.accdb is open.

Scenario
You've been promoted to Database Designer at your company and now all of your maintenance, daily housekeeping and minor changes have been delegated to other employees. Your manager has advised you to prepare the appropriate documentation for all of your databases so that there can be a smooth handoff and so that you won't be bothered with a lot of questions about functionality and purpose of databases that you've created and maintained. You realize that you have created too many databases for it to be reasonable to go back and create documentation for each one. Fortunately, you remember the Database Documenter in Access.

1. Create documentation for My Inventory Database.
   a) Select Database Tools→Analyze→Database Documenter.
   b) Select the All Object Types tab.
   c) Select Select All to select all database objects.
   d) Click tblInventory.
   e) Select Options to see the documentation details.
   f) Select OK to leave the options at their defaults.
   g) Select OK to create the documentation.

2. Save the documentation as a PDF file.
   a) Right-click the documentation and select Export.
   b) Browse to C:\091003Data\Advanced Database Management.
   c) Type Inventory Documentation and then select Publish to save the file.
   d) Select Save export steps.
   e) Select Save Exports.
   f) Close the Object Definition window.
TOPIC E

Analyzing the Performance of a Database

One of the things you never want to hear your users say is, “The database is slow.” First, because that information doesn’t help you pinpoint a problem. Second, because there’s a problem somewhere and you have to locate it and fix it. Just as regular backups can prevent disasters and regular compact and repair can prevent corruption, performance analysis can help you identify and fix problem before you get those dreaded calls about database slowness. Access has a built-in tool, the Performance Analyzer, that’s designed to help you do just that: Analyze performance.

Performance Analyzer

More than the basic function of analyzing performance, the Access Performance Analyzer tool also recommends improvements to you.

How to Use Performance Analyzer

Follow these steps to use the Performance Analyzer.

Using Performance Analyzer

To use the Performance Analyzer:

1. Open a database for analysis.
2. Select Database Tools→Analyze→Analyze Performance.
3. In the Performance Analyzer dialog box, select objects to analyze and then select OK.
4. Review the analysis results and the suggestions for increasing performance.
ACTIVITY 4–5
Using the Performance Analyzer

Before You Begin
My Inventory Database.accdb is open.

Scenario
Your manager overheard two of the data entry clerks discussing how they think something might be wrong with My Inventory Database because, as they put it, “It’s getting slow.” Your manager has asked you to stay after work and to see if you can do anything to speed it up.

1. Analyze the performance of My Inventory Database.
   a) With My Inventory Database open, select Database Tools→Analyze→Analyze Performance.

   ![Image of Database Tools ribbon with Analyze Performance selected]

   b) In the Performance Analyzer dialog box, select the All Object Types tab.
   c) Select Select All to select all of the database objects for analysis.

   ![Image of Performance Analyzer dialog box with tables selected]

   d) Select OK to begin the analysis.

2. Review the analysis.
   a) Read the recommendations, ideas, and suggestions.
   b) Select one of the results and review the Analysis Notes for the selected item.
c) Select **Close**.
d) Close the database.
Summary

In this module, you performed a variety of advanced database management tasks. You started by linking tables to external data sources. Next you managed databases by opening them in exclusive mode, backing them up, and compacting and repairing them. Next you examined object dependencies, both objects that depend on the selected object and those objects that the selected object depends on. Then you used the built-in tool to document a database. Finally, you analyzed database performance.

Why it is important to open the database in exclusive mode prior to performing any maintenance?

How often should you perform a Compact and Repair on the databases you use?

Note: Through the social networking tools provided on the LogicalCHOICE Course screen, you can follow up with your peers after completing the course for further discussion and supporting resources.
Distributing and Securing a Database

Lesson Time: 1 hour, 10 minutes

Lesson Objectives

In this lesson, you will lock down and prepare a database for distribution to multiple users. You will:

• Prepare a database for multiple user access.
• Implement security.
• Set a database password.
• Convert an Access database to an ACCDE database.
• Package a database with a digital signature.

Lesson Introduction

Microsoft® Access® provides database administrators with tools to move from a single user database to a multi-user networked database application. Microsoft has done a great job in leveraging Access to do a lot of heavy database application work before moving up to Microsoft’s SQL Server®. And, if you need to move up to SQL Server, Microsoft Access allows you to do so seamlessly. This lesson guides you in preparing your database for multi-user access by splitting your databases into front end and into backend components. You're also given tools to implement security for your databases including trusted locations, passwords, conversion to ACCDE format and digital signatures.
TOPIC A

Preparing a Database for Multiple User Access

The purpose of a database is to store and consume information. Databases can remain private or single user but many users can also use them. Multi-user databases sometimes encounter performance problems over time. You learned in the previous lesson that you can analyze performance and make changes to enhance your overall database experience. This topic introduces you to a method of increasing database performance for multi-user access: splitting the database. Performance isn’t the only reason why database administrators split Access databases but it certainly tops the list of reasons.

Splitting a Database

Splitting a database is a simple process that divides your single ACCDB file into two files. One file contains the presentation components consisting of forms, queries, reports and macros. The second file contains the data in tables.

You don’t have to do anything special when opening a database that has been split. Open the ACCDB file as usual and your tables will be there. Essentially, what the database splitter does is move your tables to a separate database and then attaches the tables as external data sources like those covered in the topic Linking Tables to External Data Sources. You can see this by examining the tables in your split database. Note the external data link symbol associated with them. Compare it to the tblFaucets table as shown in the following figure.

![Figure 5-1: Viewing tables in a split database.](image)

Front-end and Back-end Databases

For discussions of these two files, the one that contains the presentation components is the Front-End database file with the standard ACCDB name, My Inventory Database.accdb, for example. The second file (Tables) uses the file name, ACCDB_BE, to designate it as the Back-End database.

The Database Splitter

The Database Splitter is the tool that splits your database into the two parts. It’s a very simple process to perform. You will need to open the database in Exclusive Mode prior to splitting the database. Once your database is split into the ACCDB and the ACCDB_BE files, you can open the Back-End database to see for yourself that it only contains tables.

How to Split a Database

Follow these steps to split a database.
Splitting a Database

To split a database:

1. Open the database in exclusive mode.
2. Select **Database Tools→Move Data→Access Database**.
3. Select **Split Database**.
4. Browse to the location where you want to save the back-end database and then select **Split**.
5. After the database has been split, in the **Database Splitter** information box, select **OK**.

**Note:** All of the How To procedures for this lesson are available as checklists from the **Checklist** tile on the LogicalCHOICE Course screen.
ACTIVITY 5–1
Splitting a Database

Data Files
C:\091003Data\Distributing and Securing a Database\Inventory Database.accdb

Scenario
You've decided to split your database due to the increased number of over-the-network users in your organization. You have your manager's permission to make the change. You've already made a backup of your database prior to this activity.

1. Open the Inventory Database in exclusive mode.
   a) Save the file as My Inventory Database.accdb

2. Split the database.
   a) Select Database Tools→Move Data→Access Database.
   b) Read the information in the Database Splitter dialog box and then select Split Database.
   c) Browse to the C:\091003Data\Distributing and Securing a Database folder.
   d) Select Split to save My Inventory Database_be.accdb to the selected folder.
   e) In the informational message box that is displayed stating that your database successfully split, select OK.
   f) Observe the tables listed in the Tables section of the Navigation pane. The arrow to the left of the tables indicates that the table is located in an external database, in this case the My Inventory Database.accdb_be file.
Note: To learn how to normalize the databases you create, see the LearnTO Normalize a Database presentation from the LearnTO tile on the LogicalCHOICE Course screen.
TOPIC B

Implementing Security

It seems that everyone in the computing business is obsessed with security. But, the paranoia is not altogether unfounded. Hardly a day goes by without hearing about some website, bank or company that's been hacked and its data security compromised. It's an unfortunate side effect of our technology dependence. Fortunately, Microsoft knows this and provides you with adequate security measures in Access. Of course, no security strategy is 100-percent foolproof. Security is an ongoing battle that, at best, ends in a draw.

Security Strategies in Access 2010

Access 2010 offers several different security strategies.

<table>
<thead>
<tr>
<th>Strategy</th>
<th>Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database Password</td>
<td>Encrypt and password protect the database. This is a basic security strategy.</td>
</tr>
<tr>
<td>Secure VBA code</td>
<td>Password protect VBA code to prevent copying and editing.</td>
</tr>
<tr>
<td>Startup Options</td>
<td>Automatically open a Switchboard or other form. Setting other options to protect parts of the database.</td>
</tr>
<tr>
<td>Hide Database Objects</td>
<td>Prevent users from seeing objects in the Navigation pane.</td>
</tr>
<tr>
<td>Trust Center Options</td>
<td>Set Trusted Locations and options for databases.</td>
</tr>
<tr>
<td>Digital Signatures</td>
<td>A signed certificate verifies that the database is from a trusted or reliable source.</td>
</tr>
</tbody>
</table>

The Message Bar

The Message Bar appears just below the Ribbon in Access and displays important security information to the user. The types of messages that appear in the message bar include security alerts, policy messages, workflow tasks, and server information. If the message bar has been hidden, select Database Tools→Show/Hide and then check Message Bar.

Figure 5-2: The message bar appears to inform users of security information.
The Trust Center Dialog Box

The Trust Center dialog box allows you to set security options for Access. The Trust Center dialog box options determine how much or how little security is set on your database from a user experience perspective. That is to say, if you change settings here, your user’s experience with database security will change. For example, you can disable the Message Bar notices from appearing, which will never show any messages about blocked content.

![The Trust Center dialog box.](image)

Trusted Locations

A Trusted Location is a folder that is known to Microsoft Access as a location that should not be checked by The Trust Center. The purpose of Trusted Locations is to prevent any warnings about macros, ActiveX controls or data connections that Trust Center ordinarily would see as potentially harmful.

How to Add a Trusted Location

Follow these steps to add a trusted location.

Adding a Trusted Location

To add a trusted location:

1. Select File→Options.
2. Select the Trust Center tab.
3. Select Trust Center Settings.
4. Select Trusted Locations.
5. Select Add new location.
6. In the Path text box, type the path for the trusted location.
7. If necessary, select Subfolders of this location are also trusted.
8. Select OK to add the trusted location.
9. Select OK to accept the changes and close Trust Center.
10. Select OK to accept the changes and close Access Options.
ACTIVITY 5–2
Adding a Trusted Location

Scenario
Since you've split your database, you've decided to begin implementing security for your databases. One of those security methods is to place backend database files into a Trusted Location on a network drive.

Add a Trusted Location.
  a) Select File→Options.
  b) Select the Trust Center tab.
  c) Select Trust Center Settings.
  d) Select Trusted Locations.
  e) Select Add new location.
  f) In the Path text box, type C:\Temp for the path for the trusted location.
  g) Select Subfolders of this location are also trusted.
  h) Select OK three times.
TOPIC C

Setting Passwords

Just as you use passwords to access your computer, your network, and other resources, you can use passwords on Access databases. Passwords help you keep your data secure. In this topic, you will set a password on a database.

Password Protection

Setting a password on a database is a first line of defense security measure. However, a password is only one measure that you should take in your total security strategy. Follow standard password rules when assigning passwords to databases. Using a password prevents unauthorized access to a database that might contain sensitive data. It also sends an implied warning message that this database's information is private.

<table>
<thead>
<tr>
<th>Guideline</th>
<th>Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>Use at least eight characters. More characters adds strength.</td>
</tr>
<tr>
<td>Numbers</td>
<td>Add at least one numeric character.</td>
</tr>
<tr>
<td>Capitalization</td>
<td>Use a mixture of upper and lower case letters.</td>
</tr>
<tr>
<td>Alternative characters</td>
<td>Use at least one of these characters: ! @ # $ % ^ &amp; * ().</td>
</tr>
<tr>
<td>Letter substitution</td>
<td>Be careful to avoid using standard character substitutions such as @ for a, $ for s, and 3 for e.</td>
</tr>
</tbody>
</table>

Database Access Modes

Access has four basic access modes that you use when opening a database.

<table>
<thead>
<tr>
<th>Access Mode</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open</td>
<td>Standard or default mode. Multi-user.</td>
</tr>
<tr>
<td>Open Read-Only</td>
<td>Cannot write to the database, but can read data.</td>
</tr>
<tr>
<td>Exclusive</td>
<td>Only the exclusive user may open the database.</td>
</tr>
<tr>
<td>Exclusive Read-Only</td>
<td>Single user, read-only access.</td>
</tr>
</tbody>
</table>

How to Set a Database Password

Follow these steps to set a database password.

Setting a Database Password

To set a password on a database:

1. Open the database in exclusive mode.
2. Select File→Info.
3. Select Encrypt with Password.
4. Enter and confirm the desired password and then select OK.
5. In the warning box, select OK.
ACTIVITY 5–3
Setting a Database Password

Before You Begin
My Inventory Database.accdb is open.

Scenario
Since your database is used by several different departments, some unauthorized, your manager has suggested that you password protect it so that only authorized users may access it. You already sent users an email letting them know the database will be unavailable during the time you are setting the password.

1. Ensure that My Inventory Database is open in Exclusive mode.

2. Set a password for the database.
   a) Select File→Info.
   b) Select Encrypt with Password.
   c) In the Password text box, type Password
   d) In the Verify text box, type Password and then select OK.
   e) In the message box, select OK to acknowledge the message about row level locking.

3. Test the password.
   a) Close the database.
   b) Open the My Inventory Database file.
   c) When prompted for the password, type Password
TOPIC D

Converting an Access Database to an ACCDE File

If you want to prevent anyone from creating or modifying forms, reports, or code in your Access database, you can convert the database to an Access Execute Only Database, or ACCDE database. Converting the database to the ACCDE file format is another way to secure your database. In this topic, you will convert your database to an ACCDE database.

The ACCDE File Format

Note: When you save the ACCDE file and return to your database, you return to the original ACCDB file, not the newly saved ACCDE version.

The ACCDE file format is an Access database that has been converted from the standard ACCDB file to the Execute Only file type. The ACCDE is a locked-down version of your database. This means that users cannot make design changes to Forms, Reports, Macros or VBA code. If you're distributing your database to users outside of your design group, then it's wise to convert them first so that you don't have users changing your design elements.

Figure 5-4: Creating an ACCDE file from an open database.
User Templates

For creating entire specific-use databases, Access supplies you with five templates from which to choose: Comments, Contacts, Issues, Tasks, and Users. You'll find these templates on the Ribbon under Create→Application Parts→Quick Start.

![Quick Start Database Templates menu](image)

**Figure 5-5: The Quick Start Database Templates menu.**

The templates provide you with a complete database that may include tables, queries, forms and reports. The template databases contain no data.

You, as a database creator and designer, can also save a database as a template. The file format for an Access database template is ACCDT. For example, if you create a payments database as Payments.accdb, then you can save that database as Payments.accdt. As part of the distribution mechanism for your databases, Access gives you the opportunity to customize your template prior to saving it. Once you've customized your template using the dialog box shown in the following figure, it will appear in the list with the other templates in the Ribbon.
The following are the components of the Create New Template from This Database dialog box.

- **Name**: A required field that identifies the template. Access displays this name with the template.
- **Description**: The purpose of the template. This description appears in the ToolTip for the template.
- **Category**: Select User Templates to have the template show up under User Templates on the Ribbon. You can also add a category by typing it and you can then assign this and future templates to that category. The new category will show up in the Ribbon.
- **Icon**: You can specify an icon to display for your template.
- **Preview**: Provide a larger image to display for the template in Backstage view. When you browse for a template, this image is what you will see for it.
- **Primary Table**: Select a table that will be the primary table for the template. When someone uses the template, Access starts a wizard to help create a relationship between this table and others. Although the primary table is used by default, you can select a different table when the wizard runs.
- **Instantiation Form**: Select a form that will open by default when databases made from this template are first opened.
- **Application Part**: Select this check box to save the database as an application part. Clear this check box to save the database as a database template. You must select this check box before you can specify a value for the primary table.

![Create New Template from This Database](image)

**Figure 5–6: Customizing a template for distribution.**

Learn more about advanced template properties.
Share your template with the community.
How to Convert an Access Database to an ACCDE File

Follow these steps to convert an Access database to an ACCDE file.

Converting an Access Database to ACCDE
To convert an Access database to the ACCDE file format:

1. Open the database you want to convert.
2. Select **File → Save & Publish**.
3. In the **Save Database As** pane, select **Make ACCDE**.
4. Select **Save As**.
ACTIVITY 5–4
Converting a Database to ACCDE Format

Before You Begin
My Inventory Database.accdb is open.

Scenario
Upper management has decided that this database needs wider distribution in the company but an effort needs to be made that no changes can be made to the structure of the database design. They want users to add records but no other changes. You know that if you convert the database to an ACCDE format, it prevents those changes. You have already made a backup of your database.

1. Select File→Save & Publish.
2. Select Make ACCDE.
3. Select Save As and browse to C:\091003\Data\Distributing and Securing a Database.
4. Select Save.
TOPIC E

Packaging a Database with a Digital Signature

A database contains code in the form of macros, VBA code and actions that users are apprehensive about. When you package your database for distribution to users outside of your workgroup or trust zone, you want those users to trust your content; code included, and use your database with confidence. Digitally signing your database with a certificate provides this confidence by informing the user that the content is safe and signed by you as such.

Digital Signatures

A digital signature is an authentication tool that verifies the content in a file. It tells the user that the content can be trusted. It's used to validate the source of the content and identify the content as safe to use. A digital signature is used by Access to verify the source of a database against its list of trusted publishers.

Digital Certificates

A digital certificate is an identification tool. It verifies the identity of the data creator and sender. The certificate contains the name of the file publisher, expiration date, a serial number and a public key. Access 2010 allows you to create a self-signed digital certificate.

The Package and Sign Feature

The Access 2010 Package and Sign feature creates an Access deployment file, which is known as an Access Signed Package. It uses a new file type: ACCDC to denote that it contains a digital signature. The Package and Sign feature also compresses the database so that it is easily distributed via email or file transfer methods.

How to Digitally Sign a Database

Follow these steps to create and sign a digital certificate.

Creating a Digital Certificate

To create a digital certificate:

1. In Windows, select the Start button.
2. In the Search text box, type Digital Certificate for VBA Projects
3. In the Create Digital Certificate dialog box, in the Your certificate's name text box, type the name for your certificate.
4. Select OK.
5. In the SelfCert Success message box, select OK.

Signing a Database

To sign a database:

1. In Access, open the database to be signed.
2. Select File→Save & Publish.
3. In the Save As pane, select Package and Sign and then select Save As.
4. In the Windows Security dialog box, select the desired certificate and then select OK.
5. Browse to the location where you want to save the signed package and then select Create.
ACTIVITY 5–5
Digitally Sign a Database

Scenario
Since you're going to be packaging your database for wider use, you need to assign it a Digital Certificate to guarantee authenticity to those who receive it.

1. Create a digital certificate.

   Note: You only have to type as much of the Digital Certificate for VBA Projects as needed to bring it up in the list of programs.

   a) In Windows, select the Start button.
   b) In the Search text box, type Digital Certificate for VBA Projects.
   c) Select Digital Certificate for VBA Projects.
   d) In the Create Digital Certificate dialog box, in the Your certificate's name text box, type My Inventory Certificate.
   e) Select OK twice.

2. Sign the database with your certificate.

   a) Switch back to Access.
   b) Select File→Save & Publish.
   c) In the Save As pane, select Package and Sign and then select Save As.
   d) In the Windows Security dialog box, select My Inventory Certificate and then select OK.
   e) Browse to C:\091003Data\Distributing and Securing a Database and then select Create.
      The file is saved with the .accdc file extension indicating that the file has been signed with a certificate.
   f) Close the database.
Summary

In this module, you secured your database in several ways. You split the database to create a front-end and a back-end database to ready the database for use by multiple users. You set up a trusted location and added a password to the database. You converted the database to an ACCDE file format so that users cannot change or create forms, reports, or code in your database. Finally, you packaged and digitally signed the database. All of these measures combined help secure the integrity of your database.

Which security measures will you implement for your databases?

Will you be splitting your databases? Why or why not?

Note: Through the social networking tools provided on the LogicalCHOICE Course screen, you can follow up with your peers after completing the course for further discussion and supporting resources.

Note: To explore using an Access database as a web database, see the LearnTO Use Access as a Web Database presentation from the LearnTO tile on the LogicalCHOICE Course screen.
Managing Switchboards

Lesson Time: 30 minutes

Lesson Objectives

In this lesson, you will create and modify a database switchboard and set the startup options. You will:

• Create a switchboard.
• Modify a switchboard.
• Set startup options.

Lesson Introduction

In Microsoft® Access®, a switchboard is a form that usually opens when you open a database. It is the master form. You can place anything you want on a switchboard; links to other forms, links to reports and control buttons to close the database, print or exit Access. You should design your switchboard with the idea that it is the user's primary database interface. Chances are good that you'll hide the other components of your database, so a well-designed switchboard is essential to a smooth user experience.
TOPIC A

Creating a Database Switchboard

Think of your switchboard and its components as you would if you were designing web pages to navigate around your website. The switchboard is similar to the Index page of a website. If you're not familiar with website construction, think of your switchboard as being similar to the Table of Contents in a book. But, instead of page numbers, you have links to forms or buttons that perform some action for you.

Switchboard Manager

Before you create a switchboard, you'll have to make the Switchboard Manager visible to you. Once it is visible, it will stay on the Ribbon and be available when you need to create or modify a Switchboard form. The difference in using the Switchboard Manager and simply creating a form as a switchboard is that you have the advantage of the Switchboard Manager, which is designed specifically to help create a complex form as a switchboard.

Switchboard Manager Dialog Box

When you first open the Switchboard Manager, you receive a message that the Switchboard Manager was unable to find a valid switchboard in this database and asks if you would like to create one.

![Figure 6-1: The initial message from the Switchboard Manager.](image)

After you select Yes, the Switchboard Manager dialog box opens to assist creation of your switchboard.

![Figure 6-2: The Switchboard Manager dialog box.](image)
How to Create a Switchboard

Follow these steps to create a switchboard.

Adding the Switchboard Manager to the Ribbon

To add the Switchboard Manager to the Ribbon:

1. Select File→Options.
2. Select Customize Ribbon.
3. Under Main Tabs select Create and then select New Group.
4. Select Rename and rename the item Switchboard.
5. From the Choose commands from drop-down list, select Commands Not in the Ribbon.
6. Scroll down and select Switchboard Manager.
7. With the new Switchboard group selected, select Add.
8. Select OK.

Creating a Switchboard

To create a switchboard:

1. Select Create→Switchboard→Switchboard Manager.
2. Select Yes to confirm you would like to create a new switchboard.
3. In the Switchboard Manager dialog box, select Edit.
4. In the Edit Switchboard Page dialog box, select New.
5. In the Edit Switchboard Item dialog box, in the Text text box, type the name that will be displayed in the switchboard for the item.
6. From the Command drop-down list, select the action to be performed by the selection.
7. A third drop-down list changes based on the command selected. For example, if you selected Open Form in Edit Mode, the third drop-down list would allow you to select the form to open. If you selected Run Macro, it would allow you to select the macro to run.
8. Select OK to add the form to the switchboard.
9. Select Close twice. The switchboard is listed with the other forms for the database.
ACTIVITY 6–1
Creating a Simple Switchboard

Data Files
C:\091003Data\Managing Switchboards\Inventory Database.accdb

Scenario
Now that your database is more complex and widely used, users are complaining that it's too difficult to use and they don't know which item to open for a particular task. You decide to create a simple navigation screen or switchboard for your users. First, you have to add the Switchboard Manager to your Ribbon so that you'll have easy access to it now and in the future.

1. Open the Inventory Database.accdb.
   a) Save the file as My Inventory Database.accdb

2. Add Switchboard Manager to the Ribbon.
   a) Select File→Options.
   b) Select Customize Ribbon.
   c) Under Main Tabs, select Create, and then select New Group.
   d) Select Rename and rename the item Switchboard.
   e) From the Choose commands from drop-down list, select Commands Not in the Ribbon.
   f) Scroll down and select Switchboard Manager.
   g) With the new Switchboard group selected, select Add.
   h) Select OK.

3. Create a switchboard with an item to open the frmInventory form in Edit mode.
   a) Select Create→Switchboard→Switchboard Manager.
   b) Select Yes to confirm you would like to create a new switchboard.
   c) In the Switchboard Manager dialog box, select Edit.
   d) In the Edit Switchboard Page dialog box, select New.
   e) In the Edit Switchboard Item dialog box, in the Text text box, type Inventory Form
   f) From the Command drop-down list, select Open Form in Edit Mode.
   g) From the Form drop-down list, select frmInventory.
   h) Select OK to add the form to the switchboard.
   i) Select Close twice.

4. Test the switchboard.
   a) Open Switchboard form in Form view.
   b) Select Inventory Form to open the frmInventory form.
   c) View the results, then close the frmInventory form.
TOPIC B

Modifying a Database Switchboard

You used Switchboard Manager to create a basic switchboard form. You can use that same tool to make some edits to the switchboard, but changes to color and layout need to be done in the Design or Layout view of the form. In this topic, you will modify the switchboard.

Switchboard Modification

You can modify the Switchboard form in Design View or by using the Switchboard Manager. However, you can only modify the switchboard’s colors and layout in Design View or in Layout View. The Switchboard Manager allows you to create objects easily but has no design elements.

To make your Switchboard more visually appealing, you can change colors, extend the Form's size, change fonts and much more. Use your imagination but remember that visually less is often more. Don’t use too many colors, fonts or design elements because it makes the form too busy and will have the opposite effect that you're looking for in a primary application interface, which is an efficient design with attractive color and layout.

How to Apply a Background Image to a Form

To further enhance the look of your switchboard, or any form, by adding a background image. A background image can be as simple as a small graphic that you create with Microsoft Paint or something as elaborate as a photograph.

Applying a Background Image to a Form

To add a background image to a form:

1. Open the form in Design view.
2. Select the section (header, detail or footer) of the form where you want your image to be located.
3. Open the form's proper Format property sheet, if it's not open.
4. Select the tab of the property sheet.
5. Find the Picture field and either select the drop-down list to display any images known to the database or select the ellipsis button to browse for an image.
6. Select the image from the drop-down list or from your browse window.
7. Select OK in the browse window.
8. Save the form.
9. Open the form in Form view.
10. To adjust the image size or positioning (Center is the default position), return to Design View and change the Picture Alignment property on the Format tab of the property sheet.
11. Save the form.
ACTIVITY 6–2
Modifying a Switchboard

Before You Begin
My Inventory Database.accdb is open with frmI.

Scenario
Your company has strong preferences that as many visual elements as possible adhere to the company's official colors. You decide that the switchboard you created should use the company colors Blue and Red.

1. Change the background colors on the switchboard.
   a) View the Switchboard in Design view.
   b) Right-click one of the colored areas of the form and select Fill/Back Color.
   c) For the top section of the form, select Red.
   d) Repeat steps b and c to apply Red to the left section of the form and for the section of the form with the menu item, select Blue.

2. Save and test the form.
   a) Save the form.
   b) Open the form in Form view.
      Notice that it is difficult to read the black text on the blue background.

3. Change the font colors on the switchboard.
   a) Open the form in Design view.
   b) Right-click the menu item text and select Front/Fore Color.
   c) Select White.

4. Save and test the form again.
   The text is more legible now that it is in a more contrasting color.

5. Close the form.
TOPIC C

Setting Startup Options
An additional modification you might want to make to your database is to configure startup options. Startup options can help protect your database and make it easier for users to access forms, reports, and queries. In this topic, you will configure startup options for a database.

Database Startup Options
Setting Startup Options helps to further protect your database. In the Startup Options, you can title your Database application, you can select a form to launch on Database startup, you can also hide the Navigation pane that contains your Database Objects (tables, forms, reports, queries and macros).

Current Database Options
Startup Options are changed in Access Options. Select Current Database in the left pane and then configure the options to suit your needs.

Note: If you need to make changes after you have set the startup options, hold Shift while opening the database.

Figure 6–3: Startup Options.
Error Checking Options

You can set error checking options under the Object Designers selection of the Access Options dialog. These error checking options apply to all databases. Error checking is important for troubleshooting your database objects. The default is to have all error checking options selected.

Figure 6-4: Error checking options for all databases.
ACTIVITY 6–3
Setting Startup Options

Before You Begin
C:\091003Data\Managing Switchboards\Inventory Database.accdb

Scenario
You want to set further customizations on your database that make it more professional and hide certain elements that make up your database. General users don't need to tamper with or even know about many aspects of the database. You have decided to make some refinements to the Startup Options available to you.

1. Save the file as My Inventory Database.accdb
2. Configure startup options for the My Inventory Database file.
   a) Select File→Options.
   b) Select Current Database.
   c) In the Application Title text box, type My Inventory Database
   d) From the Display Form drop-down list, select Switchboard.
   e) Uncheck Display Status Bar.
   f) Uncheck Display Navigation Pane.
   g) Select OK to accept the changes to the startup options.
   h) In the message box, select OK to acknowledge that you need to close and reopen the database.

3. Test the changes.
   a) Close My Inventory Database.
   b) Reopen My Inventory Database.
   c) Observe the database window.
      There is no navigation pane or status bar. The database opened directly to the switchboard form you created.

Note: To learn how you can use your new Access skills, see the LearnTO Identify Ways to Apply Your Access Skills presentation from the LearnTO tile on the LogicalCHOICE Course screen.
Summary

In this module, you learned how to manage switchboards. You created a switchboard and then modified the design of the switchboard. You also configured the startup options for a database so that when a user opens the database, they see the switchboard. Using the switchboard can make it easier for users to find the database components they need to work with and it also helps make the database more secure because users can’t accidentally move or delete form, query, or report items.

Why should you create a switchboard for your database?

Why is switchboard design important?

Note: Through the social networking tools provided on the LogicalCHOICE Course screen, you can follow up with your peers after completing the course for further discussion and supporting resources.

Note: To learn how you can sell the databases you create, see the LearnTO Go Commercial with Your Databases presentation from the LearnTO tile on the LogicalCHOICE Course screen.
Selected Logical Operations courseware addresses Microsoft Office Specialist (MOS) certification skills for Microsoft Office 2010. The following table indicates where Access 2010 skills that are tested on Exam 77-885 are covered in the Logical Operations Microsoft Office Access 2010 series of courses.

<table>
<thead>
<tr>
<th>Objective Domain</th>
<th>Covered In</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Managing the Access Environment</td>
<td></td>
</tr>
<tr>
<td>1. Create and manage a database</td>
<td></td>
</tr>
<tr>
<td>1.1. Use Save Object As</td>
<td>Part 1</td>
</tr>
<tr>
<td>1.1.2. Use Open</td>
<td>Part 1</td>
</tr>
<tr>
<td>1.1.3. Use Save and Publish</td>
<td>Part 1</td>
</tr>
<tr>
<td>1.1.4. Use Compact &amp; Repair Database</td>
<td>Part 3, Topic 4-B</td>
</tr>
<tr>
<td>1.1.5. Use Encrypt with Password commands</td>
<td>Part 3, Topic 5-C</td>
</tr>
<tr>
<td>1.1.6. Create a database from a template</td>
<td>Part 1</td>
</tr>
<tr>
<td>1.1.7. Set Access options</td>
<td>Part 1</td>
</tr>
<tr>
<td>1.2. Configure the Navigation Pane</td>
<td></td>
</tr>
<tr>
<td>1.2.1. Rename objects</td>
<td>Part 1</td>
</tr>
<tr>
<td>1.2.2. Delete objects</td>
<td>Part 1</td>
</tr>
<tr>
<td>1.2.3. Set Navigation options</td>
<td>Part 3, Topic 6-C</td>
</tr>
<tr>
<td>1.3. Apply Application Parts</td>
<td></td>
</tr>
<tr>
<td>1.3.1. Use Blank Forms</td>
<td>Part 1; Part 3, Topic 1-C</td>
</tr>
<tr>
<td>1.3.2. Use Quick Start</td>
<td>Part 3, Topic 1-A</td>
</tr>
<tr>
<td>1.3.3. Use user templates</td>
<td>Part 3, Topic 5-D</td>
</tr>
<tr>
<td>2. Building Tables</td>
<td></td>
</tr>
<tr>
<td>2.1. Create tables</td>
<td></td>
</tr>
<tr>
<td>2.1.1. Create tables in Design View</td>
<td>Part 1</td>
</tr>
<tr>
<td>2.2. Create and Modify Fields</td>
<td></td>
</tr>
<tr>
<td>2.2.1. Insert a field</td>
<td>Part 1</td>
</tr>
<tr>
<td>2.2.2. Delete a field</td>
<td>Part 1</td>
</tr>
<tr>
<td><strong>Objective Domain</strong></td>
<td><strong>Covered In</strong></td>
</tr>
<tr>
<td>----------------------------------------------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>2.2.3. Rename a field</td>
<td>Part 1</td>
</tr>
<tr>
<td>2.2.4. Hide or Unhide fields</td>
<td>Part 1</td>
</tr>
<tr>
<td>2.2.5. Freeze or Unfreeze fields</td>
<td>Part 1</td>
</tr>
<tr>
<td>2.2.6. Modify data types</td>
<td>Part 1</td>
</tr>
<tr>
<td>2.2.7. Modify the field description</td>
<td>Part 2</td>
</tr>
<tr>
<td>2.2.8. Modify field properties</td>
<td>Part 1</td>
</tr>
<tr>
<td>2.3. Sort and filter records</td>
<td></td>
</tr>
<tr>
<td>2.1.3. Use Find</td>
<td>Part 1</td>
</tr>
<tr>
<td>2.3.2. Use Sort</td>
<td>Part 1</td>
</tr>
<tr>
<td>2.3.3. Use Filter commands</td>
<td>Part 1</td>
</tr>
<tr>
<td>2.4. Set relationships</td>
<td></td>
</tr>
<tr>
<td>2.4.1. Define Primary Keys</td>
<td>Part 1, Part 2</td>
</tr>
<tr>
<td>2.4.2. Use Primary Keys to create Relationships</td>
<td>Part 1, Part 2</td>
</tr>
<tr>
<td>2.4.3. Edit Relationships</td>
<td>Part 2</td>
</tr>
<tr>
<td>2.5. Import data from a single data file</td>
<td></td>
</tr>
<tr>
<td>2.5.1. Import source data into a new table</td>
<td>Part 2</td>
</tr>
<tr>
<td>2.5.2. Append records to an existing table</td>
<td>Part 2</td>
</tr>
<tr>
<td>2.5.3. Import data as a linked table</td>
<td>Part 2</td>
</tr>
<tr>
<td>3. Building Forms</td>
<td></td>
</tr>
<tr>
<td>3.1.1. Use the Form Wizard</td>
<td>Part 1</td>
</tr>
<tr>
<td>3.1.2. Create a Blank Form</td>
<td>Part 1</td>
</tr>
<tr>
<td>3.1.3. Use Form Design Tools</td>
<td>Part 1</td>
</tr>
<tr>
<td>3.1.4. Create Navigation forms</td>
<td>Part 3, Topic 1-A</td>
</tr>
<tr>
<td>3.2. Apply Form Design options</td>
<td></td>
</tr>
<tr>
<td>3.2.1. Apply a Theme</td>
<td>Part 1</td>
</tr>
<tr>
<td>3.2.2. Add bound controls</td>
<td>Part 3, Topic 2-B</td>
</tr>
<tr>
<td>3.2.2.1. Text box</td>
<td>Part 1</td>
</tr>
<tr>
<td>3.2.2.2. Drop down</td>
<td>Part 1</td>
</tr>
<tr>
<td>3.2.3. Format Header/Footer</td>
<td>Part 1</td>
</tr>
<tr>
<td>3.2.4. View Code</td>
<td>Part 3, Topic 3-B</td>
</tr>
<tr>
<td>3.2.5. Convert Macros to Visual Basic</td>
<td>Part 3, Topic 3-B</td>
</tr>
<tr>
<td>3.2.6. View Property Sheet</td>
<td>Part 1</td>
</tr>
<tr>
<td>3.2.7. Add Existing Fields</td>
<td>Part 1</td>
</tr>
<tr>
<td>3.3. Apply Form Arrange options</td>
<td></td>
</tr>
<tr>
<td>3.3.1. Use the Table functions</td>
<td>Part 2</td>
</tr>
<tr>
<td>3.3.1.1. Insert</td>
<td>Part 2</td>
</tr>
</tbody>
</table>
### Objective Domain

<table>
<thead>
<tr>
<th>Objective Domain</th>
<th>Covered In</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3.1.2. Merge</td>
<td>Part 2</td>
</tr>
<tr>
<td>3.3.1.3. Split</td>
<td>Part 2</td>
</tr>
<tr>
<td>3.3.2. Move table</td>
<td>Part 2</td>
</tr>
<tr>
<td>3.3.3. Reposition / Format controls</td>
<td>Part 2</td>
</tr>
<tr>
<td>3.3.3.1. Anchor</td>
<td>Part 3, Topic 1-A</td>
</tr>
<tr>
<td>3.3.3.2. Padding</td>
<td>Part 2</td>
</tr>
<tr>
<td>3.3.3.3. Margins</td>
<td>Part 2</td>
</tr>
</tbody>
</table>

### 3.4. Apply Form Format options

<table>
<thead>
<tr>
<th>Objective Domain</th>
<th>Covered In</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.4.1. Reformat Font in form</td>
<td>Part 3, Topic 6-B</td>
</tr>
<tr>
<td>3.4.2. Apply background image to form</td>
<td>Part 3, Topic 6-B</td>
</tr>
<tr>
<td>3.4.3. Apply Quick Styles to controls in form</td>
<td>Part 3, Topic 1-A</td>
</tr>
<tr>
<td>3.4.4. Apply conditional formatting in form</td>
<td>Part 3, Topic 1-E</td>
</tr>
</tbody>
</table>

### 4. Creating and Managing Queries

#### 4.1. Construct queries

<table>
<thead>
<tr>
<th>Objective Domain</th>
<th>Covered In</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1.1. Create Select query</td>
<td>Part 1</td>
</tr>
<tr>
<td>4.1.2. Create Make Table query</td>
<td>Part 1</td>
</tr>
<tr>
<td>4.1.3. Create Append query</td>
<td>Part 1</td>
</tr>
<tr>
<td>4.1.4. Create Crosstab query</td>
<td>Part 1</td>
</tr>
</tbody>
</table>

#### 4.2. Manage source tables and relationships

<table>
<thead>
<tr>
<th>Objective Domain</th>
<th>Covered In</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2.1. Use the Show Table command</td>
<td>Part 1</td>
</tr>
<tr>
<td>4.2.2. Use Remove Table command</td>
<td>Part 1</td>
</tr>
<tr>
<td>4.2.3. Create ad hoc relationships</td>
<td>Part 1</td>
</tr>
</tbody>
</table>

#### 4.3. Manipulate fields

<table>
<thead>
<tr>
<th>Objective Domain</th>
<th>Covered In</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.3.1. Add field</td>
<td>Part 1</td>
</tr>
<tr>
<td>4.3.2. Remove field</td>
<td>Part 1</td>
</tr>
<tr>
<td>4.3.3. Rearrange fields</td>
<td>Part 1</td>
</tr>
<tr>
<td>4.3.4. Use Sort and Show options</td>
<td>Part 1</td>
</tr>
</tbody>
</table>

#### 4.4. Calculate totals

<table>
<thead>
<tr>
<th>Objective Domain</th>
<th>Covered In</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.4.1. Use the Total row</td>
<td>Part 1</td>
</tr>
<tr>
<td>4.4.2. Use Group By</td>
<td>Part 1</td>
</tr>
</tbody>
</table>

#### 4.5. Generate calculated fields

<table>
<thead>
<tr>
<th>Objective Domain</th>
<th>Covered In</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.5.1. Perform calculations</td>
<td>Part 1</td>
</tr>
<tr>
<td>4.5.2. Use the Zoom box</td>
<td>Part 1</td>
</tr>
<tr>
<td>4.5.3. Use Expression Builder</td>
<td>Part 1</td>
</tr>
</tbody>
</table>

### 5. Designing Reports

#### 5.1. Create reports
<table>
<thead>
<tr>
<th>Objective Domain</th>
<th>Covered In</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1.1. Create a Blank Report</td>
<td>Part 1</td>
</tr>
<tr>
<td>5.1.2. Use Report Design Tools</td>
<td>Part 1</td>
</tr>
<tr>
<td>5.1.3. Use the Report Wizard</td>
<td>Part 1</td>
</tr>
<tr>
<td><strong>5.2. Apply Report Design options</strong></td>
<td></td>
</tr>
<tr>
<td>5.2.1. Apply a Theme</td>
<td>Part 1</td>
</tr>
<tr>
<td>5.2.2. Add calculated controls</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.2.2.1. Total report records</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.2.2.2. Group report records</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.2.3. Add bound/unbound controls</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.2.3.1. Text box</td>
<td>Part 1</td>
</tr>
<tr>
<td>5.2.3.2. Hyperlink</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.2.3.3. Drop down</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.2.3.4. Graph</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.2.3.5. Insert page break</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.2.4. Header/Footer</td>
<td>Part 1</td>
</tr>
<tr>
<td>5.2.4.1. Insert page number</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.2.4.2. Insert logo</td>
<td>Part 1</td>
</tr>
<tr>
<td>5.2.5. Reorder tab function</td>
<td>Part 2</td>
</tr>
<tr>
<td><strong>5.3. Apply Report Arrange options</strong></td>
<td></td>
</tr>
<tr>
<td>5.3.1. Use the Table functions</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.3.1.1. Insert</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.3.1.2. Merge</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.3.1.3. Split</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.3.2. Move table</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.3.3. Reposition / Format records</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.3.3.1. Padding</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.3.3.2. Margins</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.3.4. Align report outputs to grid</td>
<td>Part 2</td>
</tr>
<tr>
<td><strong>5.4. Apply Report Format options</strong></td>
<td></td>
</tr>
<tr>
<td>5.4.1. Rename label in a report</td>
<td>Part 1</td>
</tr>
<tr>
<td>5.4.2. Apply background image to report</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.4.3. Change shape in report</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.4.4. Apply conditional formatting in report</td>
<td>Part 1</td>
</tr>
<tr>
<td><strong>5.5. Apply Report Page Setup options</strong></td>
<td></td>
</tr>
<tr>
<td>5.5.1. Change page Size</td>
<td>Part 2</td>
</tr>
<tr>
<td>5.5.2. Change page orientation</td>
<td>Part 2</td>
</tr>
<tr>
<td><strong>Objective Domain</strong></td>
<td><strong>Covered In</strong></td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>5.6. Sort and filter records for reporting</td>
<td></td>
</tr>
<tr>
<td>5.6.1. Use the Find command</td>
<td>Part 1</td>
</tr>
<tr>
<td>5.6.2. Use Sort command</td>
<td>Part 1</td>
</tr>
<tr>
<td>5.6.3. Use Filter commands</td>
<td>Part 1</td>
</tr>
<tr>
<td>5.6.4. Use view types</td>
<td>Part 1</td>
</tr>
</tbody>
</table>
action
Programmatic steps performed by a macro. For example: Open a form.

arguments
A parameter given to a macro to perform an action upon. For example: Inventory form.

conditions
A filter in a Macro. Example: [Field1] > [Field2].

control
A database object on a form or report that performs some task. For example: a Command button that closes a form.

data validation
Restrictions that limit user choices and restrict opportunities to introduce incorrect entries into a database. For example: Limiting choices to a drop-down list of options.

Datepicker
The Access 2010 replacement for the Calendar control. The Datepicker is an active control that provides an interactive calendar object from which a user may pick a date to be used on a form or report.

digital certificate
To create a digital signature, you have to have a signing certificate, which proves identity.

digital signature
An electronic, encrypted stamp of authentication. A signature confirms that the information originated from the signer and has not been altered.

event
An action performed by a user on a form that triggers a corresponding action by the database. For example, tabbing from one field to another triggers an event to check data validity.

Exclusive Mode
A method of opening an Access database such that only one person has access to the database for maintenance purposes.

Expression Builder
The part of the Macro Builder where the Database Designer enters equations and field names usually associated with data validation.

external data source
Any data that does not reside inside the Access database. For example: An Excel file.

input mask
A data validation technique setup to guide the user in entering the correct data for a field.

macro
A program within Access that performs a series of defined actions.
**Macro Builder**
The editor where a Database Designer constructs Macro Actions into a functioning macro.

**object dependency**
Database entities (object) rely on one another for data. These relationships are known as dependencies.

**subform**
A form that is placed inside another form usually to display a limited amount of data in a datasheet style format.

**Switchboard**
A special Access form that uses built-in controls to open other forms, reports or to close the application.

**tab order**
The top-to-bottom, left-to-right sequence of moving from one form field to another.

**Tab Pages**
A special form that separates information into multiple pages in a tabbed format.

**Trust Center**
The location within Access where a Database Designer manages security and privacy of the database and its components.

**Trusted Location**
A folder on a local or network disk that is recognized by Access as being safe to load content from without passing through Trust Center checks.
Index

A
ACCDE file format 82
action 40
Anchoring Tool 9
arguments 41

C
Calendar control 5
Combo Box control 31
command button 2
Compact and Repair 59
Conditional formatting 20
types 2
Control wizard 31

D
database
   Error Checking Options 99
       split 72
   Startup Options 98
database access modes 80
database backup 58
Database Documenter tool 64
database options
   Compact and Repair 59
data entry
   actions for automating 50
   event properties for automating 50
data sources
   external 56
data validation
   actions 47
   properties 47
   Datepicker control 5
digital signature 87

E
event 41
   exclusive mode
       opening a data base 58
Expression Builder 28

F
Field properties 24
form layouts 6

I
input mask 26
input mask characters 26
Input Mask Wizard 26

L
Linked Table Manager 56
List Box control 32
Lookup list 27
Lookup Wizard 27

M
macro actions
   types 40
Macro Builder
   enhancements 40
macro conditions 44
macros
   and VBA 52
embedded 47
Message Bar 76

O
object dependence 61
Object events 41
Option Group control 32

P
Package and Sign feature 87
password protection 80
PivotChart 18
PivotTable 18

Q
Quick Start fields
  options 4
Quick Styles 8

R
Record validation 33

S
security strategies 76
subform 14
switchboard
  modification 95
Switchboard Manager 92

T
tab order
  change 9
Tab pages 16
Trust Center 77
Trusted Locations 77

U
user templates 83

V
VBA 52
Visual Basic for Applications, See VBA

W
Where condition 44