# TABLE OF CONTENTS

**CREATING A DATABASE** .................................................................................................................................................................................. 3  
**CREATING A TABLE USING THE WIZARD** .................................................................................................................................................................. 6  
**RENAME A DATABASE OBJECT** ........................................................................................................................................................................... 12  
**REDESIGNING A TABLE** .................................................................................................................................................................................... 14  
**THE TABLE DESIGN TOOL BAR** ........................................................................................................................................................................... 15  
**FIELD NAMES** ................................................................................................................................................................................................. 16  
**FIELD TYPES** ................................................................................................................................................................................................. 16  
**FIELD DESCRIPTIONS** ...................................................................................................................................................................................... 17  
**INSERTING A FIELD** ...................................................................................................................................................................................... 17  
**DELETING A FIELD** ...................................................................................................................................................................................... 18  
**CHANGING FIELD SIZE** ................................................................................................................................................................................ 18  
**VIEWING THE CHANGES** .............................................................................................................................................................................. 19  
**CREATING A PRIMARY KEY** ............................................................................................................................................................................ 20  
**CONVERTING FIELD TYPES** ........................................................................................................................................................................ 22  
**CONVERTING TO TEXT** ................................................................................................................................................................................ 23  
**CONVERTING FROM TEXT** .......................................................................................................................................................................... 24  
**CONVERTING FROM MEMO TO TEXT** ................................................................................................................................................................. 24  
**CHANGING FIELD SIZE** .............................................................................................................................................................................. 25  
**CREATING AN INPUT MASK** ........................................................................................................................................................................ 27  
**USING THE LOOKUP WIZARD** .......................................................................................................................................................................... 33  
**ASSIGNING A CAPTION** .................................................................................................................................................................................. 39  
**SETTING DEFAULT VALUES** ........................................................................................................................................................................ 40  
**CREATING AN EXPRESSION** ......................................................................................................................................................................... 42  
**CREATING VALIDITY RULES** .......................................................................................................................................................................... 43  
**CREATING AN EXPRESSION** ......................................................................................................................................................................... 46  
**ADDING VALIDITY TEXT** .............................................................................................................................................................................. 47  
**MISCELLANEOUS PROPERTIES** .................................................................................................................................................................... 47  
**INDEXING A TABLE** .................................................................................................................................................................................... 50  
**CREATING AN INDEX** .................................................................................................................................................................................. 51  
**MODIFYING AN INDEX** ................................................................................................................................................................................ 53  
**MULTIPLE-FIELD INDEXES** .......................................................................................................................................................................... 54  
**CREATING PERMANENT RELATIONSHIPS** ..................................................................................................................................................... 56  
**VIEWING SUBDATASHEETS** ........................................................................................................................................................................ 59  
**ENFORCING REFERENTIAL INTEGRITY** ............................................................................................................................................................ 61  
**PRINTING DATABASE RELATIONSHIPS** ..................................................................................................................................................... 63
CREATING A DATABASE

USAGE: Although you will probably work with existing files, you will also want to create your own database at some point.

You can either create a blank database or base the new file on one of the built-in templates or from one of your own existing files. A template is used to determine the basic structure of the database and can contain predefined settings, such as fields.

Click on this tool to create a new database.

To have the new database based on one of Access’s templates, select one of the **Template** choices from within the “New File” task pane, as shown below:
The New File task pane contains two sections:

**New**  
This section displays a list of new documents you can create from scratch (a blank database or a new project based on one of your existing files).

**Templates**  
This section lists the templates you can base your new database on. For example, you can search for a template online, on the Microsoft Templates home page, or on your own computer.

To base the new database on a template that was added to your system when you installed Access, click on "On my computer..."

The **General** tab contains an icon for creating a blank database which you would define from scratch.

The **Databases** tab contains common database configurations which step you through a series of interactive screens. These interactive screens are referred to as wizards.

Select the type of database to create and click **OK**.
You will be taken to a second dialog box where you will assign a name to the new database. Since databases usually contain all the data for a particular company or project, use a name that will allow you to identify the purpose of the database at a later date.

![Image of database creation dialog box]

If you want the database to be stored in a drive/folder other than the one shown, make the appropriate changes.

When finished, click on **Create**.

A blank database window will appear. You can now create a table.
CREATING A TABLE USING THE WIZARD

**USAGE:**

Before creating a new table, you should sit down and plan it out carefully, deciding on how many fields you need and what type of field each should be. This pre-planning can make your life easier once all data has been entered. When creating tables, you will be asked to name the fields to be included within the table as well as the type of field being created and the allowable size when entering data into that field.

If you should create a table and then decide the design is incorrect, Access does allow you to go back and redesign it. However, if you change the design after data has been entered, you can run into problems (such as losing information).

To create a new table, follow these steps:

1. Click on this icon to activate the table option.
2. Click on .

Access will display the following dialog box:

There are several options for creating a new table. You can create one from scratch using the **Design View**, **Import** a table from another application/database, or create a **Link** to an external table.

The easiest way to create a table is to use the **Table Wizard**. If you use the wizard, you can select from a variety of business and personal table examples that have all the fields already defined.
USING THE TABLE WIZARD

The easiest way to create a table is to be guided through the process by the Table Wizard, which is a built-in utility.

When you are ready to continue, select the Table Wizard.

The following dialog box will be displayed:

Start by selecting a table that will be the closest match to what you want to do from the list of Sample Tables. You can choose from a list of Business or Personal applications.

To add all the sample fields to your new table, click on ▶️. If you only want some of the fields, highlight each field and click on ▶️ to add one field at a time. You can also double-click a field to add it to your list. As you add each field, you will see the list of fields displayed in the last section titled Fields in my new table.

If you accidentally added a wrong field, or decide to remove fields from your new table, use ◀️ to remove all of the fields, or use ◀️ to remove one field at a time.
If you want to change the name of a field, select the field name (from the right-most column) and click on Rename Field...

Replace the current field name with your own and click OK.

Once you have returned to the Wizard and have finished selecting your fields, click on Next >.

Now you can name your table and specify a primary key. A primary key is a type of field used by Access to uniquely identify each record in case you have duplicate entries. Tables will also display the data in the order of the primary key field.

If you do not care about which field will be used to differentiate the records, let Access set the primary key. Otherwise, choose to set the primary key yourself. However, make sure you select a field that will be unique for each entry (e.g., SSN).

When you are ready to continue, click on Next >.
If you elected to choose your own primary key, you will see the following dialog box:

Use the pull-down list at the top of the dialog box to pick the field that should be used for the primary key. You should pick a field that will not contain duplicates such as a customer ID, social security number, license number, etc. Access will use the field you choose as the default sort order when you view the data.

If you want, you can have Access automatically number your fields by choosing the first radio button option **Consecutive numbers**.

If you want to have control over the numbers yourself, choose the second option, which allows you to enter your own control numbers.

If your primary key field will contain a combination of numbers and letters, select the last option.

When you select one of the last two options, keep in mind that you must keep track of the fields so that no duplicates are entered.

When you are ready to continue, click on **Next >**.
The next two screens are displayed if your database contains more than this one table. The first screen specifies a relationship:

Is your new table related to any other tables in your database? Related tables have matching records. Usually, your new table is related to at least one other table in the current database.

In some cases, the wizard will create table relationships for you. The list below shows how your new table is related to existing tables. To change how a table is related, select a table in the list and click Relationships.

For example, clients may have many orders. Instead of keeping multiple records for each order with the clients billing information (thereby duplicating the name and address multiple times), you would keep the orders in a separate table. The two tables would be related since each client may have at least one order stored in the order table. To relate this table, click on Relationships...

Select the type of relationship and, click on Next >.
The last screen allows you to now enter data into the new table:

That's all the information the wizard needs to create your table.

After the wizard creates the table, what do you want to do?

- Modify the table design.
- Enter data directly into the table.
- Enter data into the table using a form the wizard creates for me.

Display Help on working with the table.

The first option will be discussed later. It allows you to go in and modify the table design in case you forgot to add a field or want to remove one. You would use this feature if you want to add some custom features.

The second option places you in the datasheet view and allows you to begin entering data into the table.

The third item creates a generic form and then allows you to begin entering data into the table.

The last option offers some help when you are modifying the table design or entering data.

When you are done, click on Finish.
RENAMING A DATABASE OBJECT

If you decide to change the name of a table (or any other database object), you can easily do so from within the database window.

To do so, follow the steps outlined below:

1. Select the table’s icon (or the icon representing the object you want to rename).
2. Click once on the object’s name - to select it.
3. Click a second time on the object’s name or press @.
4. The current name of the object will be selected. Begin editing or changing the name, as needed.
5. When done, click away or press E.

NOTE: Be sure to close the object (e.g., table) to be renamed before following the steps above.

NOTE: Be careful in step 3 (shown above) not to click the mouse a second time too quickly! If you double-click on an object, you will be opening it.
**PRACTICE EXERCISE**

<table>
<thead>
<tr>
<th>Instructions:</th>
<th></th>
<th>Using the Wizard, create a new table based on the <strong>Products</strong> sample table.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Use the following fields:</td>
</tr>
<tr>
<td><strong>Field Name</strong></td>
<td></td>
<td><strong>ProductID</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>ProductName</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>UnitPrice</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Name the new table <strong>Items</strong> and make the <strong>ProductID</strong> field the Primary Index based on numbers you enter.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Add the following records to the table:</td>
</tr>
<tr>
<td><strong>ProductID</strong></td>
<td><strong>ProductName</strong></td>
<td><strong>UnitPrice</strong></td>
</tr>
<tr>
<td>1</td>
<td>Chia Pet</td>
<td>12</td>
</tr>
<tr>
<td>2</td>
<td>Pet Rock</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>Ginsu Knife Set</td>
<td>22</td>
</tr>
<tr>
<td>4</td>
<td>PEZ Dispenser</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>Lava Lamp</td>
<td>13.65</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Print/Preview the record listing for the <strong>Items</strong> table.</td>
</tr>
</tbody>
</table>
REDESIGNING A TABLE

USAGE:

It is unrealistic to believe you will never make a mistake when creating tables. At some point you will probably need to go back and alter the design of an existing table. After having worked with the table, you may realize that you missed a field, made a field too short, or come across some other problem.

Although good planning can help avoid many of these problems, real life situations almost always point out deficiencies in our designs. The good news is that you can easily redesign and customize your tables even after data has been entered.

To modify a table's design, open the table in **design** view. There are two easy ways of switching into design view. If in the database window, highlight the table's name to be modified and click on ![Design](image). If you are in Datasheet view, click on ![Design](image).

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CustomerID</td>
<td>AutoNumber</td>
<td></td>
</tr>
<tr>
<td>CompanyName</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>ContactFirstName</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>ContactLastName</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>City</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>State</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>PhoneNumber</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>FaxNumber</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>EmailAddress</td>
<td>Text</td>
<td></td>
</tr>
</tbody>
</table>

Field Properties

<table>
<thead>
<tr>
<th>General</th>
<th>Lookup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field Size</td>
<td>Long Integer</td>
</tr>
<tr>
<td>New Values</td>
<td>Increment</td>
</tr>
<tr>
<td>Format</td>
<td>Caption</td>
</tr>
<tr>
<td>Indexed</td>
<td>Yes [No Duplicates]</td>
</tr>
</tbody>
</table>

The top section of the window will be used to identify the name, type of data and an optional description for each field.

**NOTE:** The field description can be used to display more information on the status line. For example, you could display instructions for filling out the field info.
THE TABLE DESIGN TOOL BAR

While in design mode, you should notice that the tool bar has changed to include several new icons, as discussed below:

- Use this button to switch between design mode and the datasheet.
- Saves the current table design.
- Sets current field as the primary key.
- Used to set additional indexes.
- Inserts a blank row for a new field at the current location.
- Deletes the current row/field.
- Sets properties for the current field.
- Click on this button to access the Field Builder which allows you to add fields from a list of categories.
- Click on this button to display the database window.
- Click on this button to create a new object (such as a table, form or report).
- Click on this button to access help.
FIELD NAMES

When adding/changing fields, you need to provide some basic information about the field, such as the "name" and "type" of data that will be stored in the field. Field names can be up to 64 characters in length and may contain letters, numbers and spaces. Although you can use up to 64 characters, you should try to keep field names as short as possible since you will be referring to them quite often when creating forms, reports and other objects. However, don't make them too short so they can't be recognized.

FIELD TYPES

Once you have labeled a field with a name, you must identify the "type" of data that will be stored within the field. The following types of fields are available:

<table>
<thead>
<tr>
<th>Field Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text</td>
<td>Stores up to 255 characters (default 50).</td>
</tr>
<tr>
<td>Memo</td>
<td>Stores larger amounts of text - max 65,535.</td>
</tr>
<tr>
<td>Number</td>
<td>Used for numeric data.</td>
</tr>
<tr>
<td>Date/Time</td>
<td>Stores dates (between 100 and 9999) or times.</td>
</tr>
<tr>
<td>Currency</td>
<td>Numbers with dollar signs.</td>
</tr>
<tr>
<td>AutoNumber</td>
<td>Assigns a random or sequential number.</td>
</tr>
<tr>
<td>Yes/No</td>
<td>Contains only one of two values. Used for true/false, on/off and yes/no entries.</td>
</tr>
<tr>
<td>OLE Object</td>
<td>Used for storing objects, such as spreadsheets and graphics.</td>
</tr>
<tr>
<td>Hyperlink</td>
<td>Use this field type if you want to store links to web pages. When entering data for a hyperlink field, simply begin with http:// followed by the web address.</td>
</tr>
<tr>
<td>Lookup</td>
<td>Allows a value to be selected from another table or combo box.</td>
</tr>
</tbody>
</table>

To choose a field type other than text, move to the type column and click on the down arrow to see the list of available types. Click on the desired field type.
FIELD DESCRIPTIONS

Since you will normally keep your field names to a minimum size, you might want to use this section to further clarify the contents of the field. Remember that this description is displayed on the status line whenever the user enters this field. The description can be used to explain the field in more detail or to give special instructions.

INSERTING A FIELD

There may be times when you decide that a new field should be placed between two existing ones. To insert the field, move to the location where the new field should appear.

Click on this button to insert a blank row (field).

You can also click the [RIGHT] mouse button and choose Insert Rows. The current row is pushed down and a blank row appears for you to enter the contents of the new field.
DELETING A FIELD

Click on this button to delete the current row (field).

You can also point to the row containing the field you want to remove and click the [RIGHT] mouse button. Choose **Delete Rows** from the pop-up menu.

If the database contains information, the following confirmation box will be displayed:

Do you want to permanently delete the selected field(s) and all the data in the field(s)?

To permanently delete the field(s), click Yes.

Yes  No

Click on **Yes** to remove the current field and its data.

CHANGING FIELD SIZE

If you find that a field is too small or too long, you can adjust the size to a more appropriate one. The field size is set in the lower half of the design screen where the field **properties** are.

<table>
<thead>
<tr>
<th>General</th>
<th>Lookup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field Size</td>
<td>9</td>
</tr>
<tr>
<td>Format</td>
<td></td>
</tr>
<tr>
<td>Input Mask</td>
<td></td>
</tr>
<tr>
<td>Caption</td>
<td></td>
</tr>
<tr>
<td>Default Value</td>
<td></td>
</tr>
<tr>
<td>Validation Rule</td>
<td></td>
</tr>
<tr>
<td>Validation Text</td>
<td></td>
</tr>
<tr>
<td>Required</td>
<td>No</td>
</tr>
<tr>
<td>Allow Zero Length</td>
<td>No</td>
</tr>
<tr>
<td>Indexed</td>
<td>No</td>
</tr>
</tbody>
</table>
Text fields can be a maximum of 255 characters in length. Access defaults to a size of 50 for text fields. Numeric fields have several different size limitations depending on the size of number you plan to store. You can choose from byte, integer, long integer, single or double. If you need decimals, you should choose single or double.

<table>
<thead>
<tr>
<th>Field Size</th>
<th>Numeric Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Byte</td>
<td>Stores numbers from 0 to 255. No decimals.</td>
</tr>
<tr>
<td>Integer</td>
<td>From -32768 to 32767 without decimals.</td>
</tr>
<tr>
<td>Long Integer</td>
<td>Stores numbers from -2,147,483,648 to 2,147,483,647. No decimals allowed.</td>
</tr>
<tr>
<td>Single</td>
<td>Numbers from -3.402823E38 to 3.402823E38 with up to seven decimal places.</td>
</tr>
<tr>
<td>Double</td>
<td>From -1.79769313486232E308 to 1.79769313486232E308 with 15 decimals.</td>
</tr>
<tr>
<td>Replication ID</td>
<td>A unique ID used for replication purposes.</td>
</tr>
</tbody>
</table>

**TIP:** Select the smallest number possible. Access works faster and more efficiently with smaller values.

**VIEWING THE CHANGES**

When done modifying the table design, click on which switches to the datasheet view. Since you cannot add/view records until the changes are saved, the following box appears:

![You must first save the table.]

Do you want to save the table now?

- [Yes]
- [No]

Click on [Yes] to save your changes.
CREATING A PRIMARY KEY

At some point, you may decide that a field within your table cannot contain duplicates. Because several people may be entering information into your database, it becomes difficult to protect against duplicates. For example, no two employees should have the same social security number. If a mistake occurs and two of the same social security numbers are entered into a payroll file, the company could issue two paychecks each month to one individual.

For this reason, Access allows you to create **Primary Key Fields**. Primary Key Fields do not allow for duplicate entries. If a key field exists within a table and the user attempts to enter duplicate data, Access will not accept it. Key fields serve another purpose as well. As you enter new records, Access automatically sorts the records based on the key. For example, if you created a key field for social security numbers, Access will automatically sort the entire table based on those numbers. As you add new records, they will automatically be sorted in correct order as they are entered.

Select the table to modify (from within the database window) and click on this tool to enter design mode.
To create a primary key field, select the field to be used and then click on this button once.

A key icon (❽) is used to identify the key field, as shown below:

<table>
<thead>
<tr>
<th>Field</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Cust Id</td>
</tr>
<tr>
<td></td>
<td>Lastname</td>
</tr>
<tr>
<td></td>
<td>Firstname</td>
</tr>
<tr>
<td></td>
<td>Address</td>
</tr>
<tr>
<td></td>
<td>City</td>
</tr>
<tr>
<td></td>
<td>ST</td>
</tr>
<tr>
<td></td>
<td>Zip</td>
</tr>
</tbody>
</table>

The example shown above would result in Access automatically sorting the table based on the customer ID field. From this point on, Access will not allow duplicate entries with the same customer ID code.

If you change your mind and want to remove the key, select the field and click on the key icon (❽) a second time.

**WARNING:** When creating key fields, be careful not to make a single field such as "Last Name" a key as there could be many people with the same last name. If the field contains duplicates, Access displays an error message when you try to save the design.

If you wanted to automatically sort your table based on a field you know contains duplicates (e.g., Last Name), you must make more than one field part of the primary key. For example, you could use the lastname, firstname and middle initial fields together as one primary key.
To make multiple fields part of the primary key, highlight all the required fields and click on this button.

The example shown below illustrates the use of multiple primary key fields:

<table>
<thead>
<tr>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lastname</td>
</tr>
<tr>
<td>Firstname</td>
</tr>
<tr>
<td>MI</td>
</tr>
<tr>
<td>Address</td>
</tr>
<tr>
<td>City</td>
</tr>
<tr>
<td>ST</td>
</tr>
<tr>
<td>Zip</td>
</tr>
</tbody>
</table>

This example would result in Access sorting all records within the table based on the "Lastname" field. If duplicates existed in this field, Access would then sort by the "Firstname", followed by the "MI" field.

The three fields together form a unique key which is now used to identify each record. Access will not allow a new record to be added to the table containing the same Lastname, Firstname and middle initial (MI) as an existing record.

It is unlikely you would have two employees with the exact same first name, middle initial and last name.

**NOTE:** Since Access sorts the records based on the order of the key fields, you may want to move the fields around to get them in the order you want to have the file sorted. For example, making sure that last name is first in the list followed by firstname and then MI.

**CONVERTING FIELD TYPES**

**USAGE:**

There may be times after you create or import a table that you realize the types of fields assigned may not be correct. For example, perhaps you created a table with all alphanumeric field types and then decide you would rather have defined one of the fields (e.g., cost) as containing numeric values so that you could instruct Access to calculate the total of all records contained within this field. Or, perhaps you originally defined a telephone number field as being numeric (because it contains numbers) and then
realize as you begin entering the records that Access will not allow you to add the necessary dashes.

Instead of creating a new table to alleviate these problems, Access allows you to redesign the existing table to convert the field types, as outlined in the steps shown below:

1. Switch to **Design** view.

2. Click in the column labeled **Data Type** beside the name of the field you want to modify. A down arrow is displayed for you to select the new data type.

3. Select the new data type from the pull-down list.

**CONVERTING TO TEXT**

For the most part, converting fields from a numeric, date or other type of field to a text field causes no problems. Problems occur when you discover that certain types of data (phone numbers, zip codes, ssn) are not really "numbers". When defined as a numeric field, Access doesn't let you store the dashes or parentheses along with the number.

You might also use this type if you find that a code, such as an item code, may contain letters and not just numbers.

When you convert numbers or dates, Access uses the "General" format for the conversion and does not store the dollar sign or other special formatting you may have set for the fields.
CONVERTING FROM TEXT

If you are converting from a text field to another type such as numeric, currency, date or yes/no, you will need to be aware that not all data may come across correctly. When you convert an existing text field to numeric and Access finds a record containing a non-numeric character, it will replace the field with a "null" value - basically deleting the field data.

You will see an error message before the data is actually lost, as shown below:

Microsoft Access encountered errors while converting the data.

The contents of fields in 7 record(s) were deleted.
Do you want to proceed anyway?

Yes  No

If you choose Yes, Access will delete the data from the affected fields and you will have to re-enter the correct data. The same holds true when converting to date or dollar types. The information within the field must be appropriate for the field type.

Before converting field types, be sure to examine the table and know what type of information has been entered.

CONVERTING FROM MEMO TO TEXT

The main consideration to take into account when converting from a memo field to a text type, is that the data will be truncated (cut-off) according to the value of the field size. Be sure to increase the size of the new text field to match the longest piece of data you will need to keep. By default, the field size is set to 50.
CHANGING FIELD SIZE

USAGE:

After working with an existing table's design and adding records, you may find that you have assigned a field size that is either too small for the data you want to enter or too large (which can be wasteful).

To modify a table's design, make sure you are viewing the database window and highlight the table to be modified.

Once the table has been selected, click on this button to access Design view.

The current table's design will be displayed, as shown below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CustomerID</td>
<td>AutoNumber</td>
<td></td>
</tr>
<tr>
<td>CompanyName</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>ContactFirstName</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>ContactLastName</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>City</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>State</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>PhoneNumber</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>FaxNumber</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>EmailAddress</td>
<td>Text</td>
<td></td>
</tr>
</tbody>
</table>

Click on the field name to modify and then access the property labeled Field Size at the bottom of the window. Enter the new length in the box provided.
You can make a field longer by simply increasing its defined size. However, when you shorten the size of a field, you take the chance that some of the data may be cut off (truncated).

If this is the case, Access displays the following warning box when you exit the design mode:

![Warning Box]

If you choose **Yes**, data extending beyond the size you specified for the field will be cut off at that size.

Make sure you are certain that no important data will be lost by making a field shorter than its original size.

**NOTE:** While text fields are truncated if a field is too long, numeric fields are **deleted** if the number is too big for the size you chose. Make sure you won't be losing important data!
CREATING AN INPUT MASK

USAGE:

You can use the “Input Mask” property to make data entry easier and more precise. You can control how the data will be entered and where within the field it should be placed. For example, if you want a phone number to look like (999)999-9999 but don’t want the user to always have to enter the parentheses and the dash, you can create a “mask” that will automatically display these characters and only allow the user to enter numeric digits in the positions specified.

If not in Design view, click on this button.

The current table design will be displayed, as shown below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CustomerID</td>
<td>AutoNumber</td>
<td></td>
</tr>
<tr>
<td>CompanyName</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>ContactFirstName</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>ContactLastName</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>City</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>State</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>PhoneNumber</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>FaxNumber</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>EmailAddress</td>
<td>Text</td>
<td></td>
</tr>
</tbody>
</table>

General | Lookup
---|---
Field Size | 
New Values | Increment
Formal | 
Caption | 
Indexed | Yes (No Duplicates)

Select the field to modify and then click in the property box labeled **Input Mask**.
If you have a common field, the easiest way of creating an input mask is to choose from the list of predefined formats. Some common formats provided by Access include phone numbers, social security numbers, and zip codes.

Click on this button (located to the far right of the Input Mask property box).

The Input Mask Wizard will be displayed, as shown below:

The first step of the wizard allows you to choose which input mask most closely matches the way you want your data to look. Choices include such formats as social security numbers, telephone numbers, zip codes, and date/time formats.

Once you have chosen a basic mask, click on to move to the next step of the wizard.
The second step of the Input Mask wizard will be displayed:

Do you want to change the input mask?
Input Mask Name: Social Security Number
Input Mask: 000-00-0000

What placeholder character do you want the field to display?
Placeholders are replaced as you enter data into the field.
Placeholder character: _

Try It: 

This second step allows you to customize the selected mask and select a placeholder character to be displayed until data is actually placed in the field.

Once you have chosen a basic mask and added a placeholder for each digit, click on Next to move to the next step of the wizard.
The third step of the Input Mask wizard asks how you want to store the data (with or without symbols). Although you can display data such as social security numbers with dashes and other symbols, you do not need to store these symbols within the actual table since they are repetitive and simply take up extra space.

```
How do you want to store the data?
☑ With the symbols in the mask, like this:
   237-05-7739
☑ Without the symbols in the mask, like this:
   76884176
```

Click on Next to move to the final step of the wizard:

That's all the information the wizard needs to create your input mask.

| Cancel | < Back | Next > | Finish |

When done, click on Finish to complete the process.
If you want to create the mask yourself in the property box provided, you can do so using the following characters:

- **0**: Digit (0 to 9, entry required, plus [+] and minus [û] signs not allowed).
- **9**: Digit or space (entry not required, plus and minus signs not allowed).
- **#**: Digit or space (entry not required; spaces displayed as blanks while in Edit mode, but blanks are removed when data is saved; plus and minus signs allowed).
- **L**: Letter (A to Z, entry required).
- **?**: Letter (A to Z, entry optional).
- **A**: Letter or digit (entry required).
- **a**: Letter or digit (entry optional).
- **&**: Any character or a space (entry required).
- **C**: Any character or a space (entry optional).
- **. , : ; - /**: Decimal placeholder and thousand, date, and time separators. (The actual character used depends on the settings in the Regional Settings Properties dialog box within the Windows Control Panel).
- **<**: Causes all characters to be converted to lowercase.
- **>**: Causes all characters to be converted to uppercase.
- **!**: Causes the input mask to display from right to left, rather than from left to right. Characters typed into the mask always fill in from left to right. You can include the exclamation point anywhere in the input mask.
- **\**: Causes the character that follows to be displayed as the literal character (for example, \A is as just A).
- **password**: Used to prevent displaying typed characters.
There are three parts to the input mask that may be specified. A semicolon (;) separates each of the three sections.

- The first section is used to specify the mask itself. For example, to specify a mask for telephone numbers, you would enter the following:

  Example:  (999)999-9999

  Or if you would like to enter a mask for a social security number, you would enter the following:

  Example:  000-00-0000

  The zeros indicate that the entry is required.

- The second section is used to tell Access whether to store the literal characters with the data. In other words, do you want to store the parentheses and dashes within the table or simply use them for display purposes.

  Separate this section from the first with a semi-colon (;) and use 0 to store the characters or 1 to only store the data.

  Example:  (999)999-9999;1

- The third section is used to tell Access what character to display in the positions where the user is to type. For example, if you want the display to be blank (other than the literal characters), enter “”. If you want a pound sign to display in each position, enter “#”.

  Separate this section from the second with a semi-colon (;) followed by the character to use enclosed in quotes.

  Example:  (999)999-9999;“#”

  This would display as (###)###-####.
USING THE LOOKUP WIZARD

USAGE:   To help save you typing and possible input errors, you can use a "Lookup" field to choose a field’s value based on a predefined list. For example, if the field you are defining will consist of company departments, the list would contain only those available departments within your company. By adding a lookup field, you can prevent users from entering the wrong data.

If you are not in Design view, click on this button.

Select the field you want to modify and then click on the down arrow beside the current Data Type.

From the list of available data types, select Lookup Wizard...

The Lookup Wizard will be displayed, as shown below:

This wizard creates a lookup column, which displays a list of values you can choose from. How do you want your lookup column to get its values?

- I want the lookup column to look up the values in a table or query.
- I will type in the values that I want.

The first step of the Lookup Wizard asks if you want to create the lookup column from information stored within an existing table or query or if you want to type the list yourself.
Once you have made your selection, click on **Next** to move to the next step.

Assuming you chose to create the lookup list from an existing table or query, the following screen will be displayed:

This step is used to select the table or query to be used for creating the lookup list.

Begin by selecting the radio button (along the bottom of the dialog box) representing the category item you want to use for your lookup list of values. You can choose to get the lookup values from an existing table, query or both.

After selecting the item category, highlight the name of the table or query you want to use.

Once you have chosen the table or query to use for your lookup list, click on **Next** to move to the next step of the wizard.
The next step is used to select the fields that contain the values you want to make available within the lookup list:

To add all of the fields from the selected table, click on \( \rightarrow \). If you only want some of the fields, highlight each field and then click on \( \rightarrow \) to add one field at a time. You can also double-click a field to add it to your list. As you add each field, you will see the list of fields displayed in the box labeled Selected Fields.

If you accidentally added a wrong field, or decide to remove fields from the lookup list, use \( \leftarrow \) to remove all of the fields, or use \( \leftarrow \) to remove one field at a time.

Once you have selected the fields to be included in the lookup list, click on \( \text{Next} \) to move to the next step of the wizard.
The next step is used to define the sort order for your list:

What sort order do you want for your list?
You can sort records by up to four fields, in either ascending or descending order.

1. [Field] - [Ascending]
2. [Field] - [Ascending]
3. [Field] - [Ascending]
4. [Field] - [Ascending]

You can choose to sort the records by up to four fields.

Click on the down arrow ▾ to select the field you want to sort.

You can also choose to sort the columns in either ascending (smallest to largest) or descending (largest to smallest) order.

Once you have selected the sort order, click on Next > to move to the next step of the wizard.
The next step is used to define the width of the lookup columns:

How wide would you like the columns in your lookup column?

To adjust the width of a column, drag its right edge to the width you want, or double-click the right edge of the column heading to get the best fit.

<table>
<thead>
<tr>
<th>Category Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appliance</td>
</tr>
<tr>
<td>Collectible</td>
</tr>
<tr>
<td>Electronic</td>
</tr>
<tr>
<td>Furniture</td>
</tr>
<tr>
<td>Jewelry</td>
</tr>
<tr>
<td>Sports Equipment</td>
</tr>
<tr>
<td>Tool</td>
</tr>
</tbody>
</table>

Place the mouse pointer on the right border of the column you want to adjust. Make sure the mouse pointer is on the column margin line. The pointer changes to a cross-hair indicating you are on the margin.

Click and drag the column margin line either to the right (expanding it) or to the left (shrinking it).

If you double-click on the margin line, Access will adjust the column to fit the largest entry.

Once you have selected the lookup column width, click on Next to move to the next step of the wizard.
This final step is used to enter a label for the lookup column:

In the box provided, enter a descriptive name for the lookup column(s).

Notice there is also a checkbox at the bottom of this dialog box that will enable the Office Assistant to offer help with customizing the lookup column.

When done, click on **Finish** to close the wizard.

You will be prompted to save the table so that the relationships can be created:

Click **Yes** to save the table.

From now on, when you view the table in Datasheet view, instead of entering the data yourself you will be able to select the field contents by clicking on the down arrow ▼ and choosing from the lookup list you created.
ASSIGNING A CAPTION

USAGE:

By default, Access uses the field name as the title for column headings or for field labels on a form. Sometimes the field names do not clearly represent the data they contain. If you would like Access to use a different name to identify the current field, you can assign your own field name (caption).

Although you could also change the field name, this becomes more difficult if other reports/objects have already been created using the current field name. The caption entered while in Design view will be used for all datasheet views and future forms/reports created.

If not already in Design view, click on this button.

The current table design will be displayed, as shown below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CustomerID</td>
<td>AutoNumber</td>
<td></td>
</tr>
<tr>
<td>CompanyName</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>ContactFirstName</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>ContactLastName</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>City</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>State</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>PhoneNumber</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>FaxNumber</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>EmailAddress</td>
<td>Text</td>
<td></td>
</tr>
</tbody>
</table>

Select the field to modify and then click in the property box labeled **Caption**. Enter the new name.
SETTING DEFAULT VALUES

USAGE:

There may be times when a particular field will usually have a set entry. Rather than having to type that entry in each time you add a new record to the table, you can have Access automatically place a default entry in the field and then, should it ever need to be changed, you could manually modify it.

This can be useful for adding such things as employee addresses - where the state, and possibly the city, will basically be the same for each employee. Instead of having to type the same state/city for each employee, you can have Access enter it for you. To change the default values of an existing table, make sure you are viewing the database window and highlight the table to be altered.

Once the table is selected, click on this button.

The current table design will be displayed as shown below:

Select the field to modify and then enter the default value in the property box labeled Default Value.
If you want the default value to simply be a number or block of text, enter it in the property box provided. For example, enter 10 in a numeric field to always have that field default to the value 10. Enter CA in a state field to have the state default to CA.

**NOTE:** *Default values are only used when adding new records to a table. Existing records are not affected.*

If the default value should be based on a formula or equation, you will want to access the Expression Builder.

Click on this button (located to the far right of the Default Value property box) to access the Expression Builder.

The resulting dialog box is divided into three sections. The top portion of the box is where you actually build the expression. The middle portion contains the operators used to build the expression and the bottom section contains elements and categories of elements that will be used in the expression.

You can either paste elements and operators from the middle and bottom sections or type them directly in the expression box.
The bottom portion of the dialog box contains three sections:

- The left column contains folders that list available tables, queries, forms, reports and other database objects. It also contains the built-in and user-defined functions, constants, operators, and common expressions.

- The middle column contains elements or categories of elements for the currently selected folder (from the left side).

- The right column lists values for the elements selected in the left and middle sections. For example, if you had selected **Built-In Functions** from the far left and **Date/Time** from the middle section, the right side would display the available date/time functions. You could then paste the function in the expression portion at the top of the dialog box.

**CREATING AN EXPRESSION**

1. From the bottom left section of the dialog box, select the folder containing the expression element to use.

2. From the bottom middle section, select the category of expression element to use.

3. From the bottom right section, select the expression element by double-clicking it or highlighting it and clicking `Paste`.

4. From the middle portion, select the operator by double-clicking on it or highlighting it and clicking `Paste`.

5. Replace the `<<Expr>>` with the actual value to be used in the expression.

If you see a mistake and want to remove an element, operator or value, select it from the expression box and press `=`.

6. When the expression is complete, click on `OK`.
CREATING VALIDITY RULES

USAGE:

To better control the table data, you may at some point want to establish validity rules. These "rules" allow you to define what conditions must be met in order for Access to accept the data. Access automatically checks the data entered in a field to make sure it matches the field type. For example, it won't let you enter text in a numeric field or a date in a numeric field. While this is fine for basic checking (validation) of data, you may need to be more specific as to what data is allowed to be entered in the field.

A salary field is an example. You may know the lowest monthly salary being paid is $2,300 and the highest salary is $6,700. To ensure that no one is paid $23.00 or $67,000, you can add a validity rule that checks the entered value against the range allowed. To add validity rules to a table, make sure you are viewing the database window and highlight the table to be altered.

If not in Design view, click on this button.
Select the **Validation Rule** section (in the bottom half of the screen). Use standard expressions such as =, <, >, <=, >=, <>, <>.

Some sample validation expressions are shown below:

<table>
<thead>
<tr>
<th>Expression</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;0</td>
<td>Must be more than 0.</td>
</tr>
<tr>
<td>&lt;0</td>
<td>Value cannot be zero.</td>
</tr>
<tr>
<td>0 Or &gt;200</td>
<td>Value must be either zero or larger than 200.</td>
</tr>
<tr>
<td>Between 50 And 100</td>
<td>Between 50 and 100.</td>
</tr>
<tr>
<td>Between #1/1/03# And #6/30/03#</td>
<td>Must fall on/after Jan 1 and on/before June 30,'03.</td>
</tr>
<tr>
<td>&gt;=#1/1/03#</td>
<td>Must be on/after 1/1/03.</td>
</tr>
<tr>
<td>&gt;=#1/1/03# And &lt;=#12/31/03#</td>
<td>Date must be in 2003.</td>
</tr>
<tr>
<td>&lt;=Date()</td>
<td>Date can't be after today's.</td>
</tr>
<tr>
<td>Like &quot;N????&quot;</td>
<td>Must be five characters long and begin with an N.</td>
</tr>
<tr>
<td>In (&quot;AZ&quot;,&quot;NM&quot;,&quot;CA&quot;,&quot;TX&quot;)</td>
<td>Must be one of the four inside the parentheses.</td>
</tr>
</tbody>
</table>
If you want to create a more advanced validity rule, you can access the Expression Builder to help guide you through the process.

Click on this button (located to the far right of the Validation Rule property box) to access the Expression Builder.

The resulting dialog box is divided into three sections. The top portion of the box is where you actually build the expression. The middle portion contains the operators used to build the expression and the bottom section contains elements and categories of elements that will be used in the expression.

You can either paste elements and operators from the middle and bottom sections or type them directly in the expression box.
The bottom portion of the dialog box contains three sections:

- The far left contains folders that list available tables, queries, forms, reports and other database objects. It also contains the built-in and user-defined functions, constants, operators, and common expressions.

- The middle contains elements or categories of elements for the currently selected folder (from the left side).

- The far right lists values for the elements selected in the left and middle sections. For example, if you had selected Built-In Functions from the far left and Date/Time from the middle section, the right side would display the available date/time functions. You could then paste the function in the expression portion at the top of the dialog box.

CREATING AN EXPRESSION

1. From the bottom left section of the dialog box, select the folder containing the expression element to use.

2. From the bottom middle section, select the category of expression element to use.

3. From the bottom right section, select the expression element by double-clicking it or highlighting it and clicking Paste.

4. From the middle portion, select the operator by double-clicking on it or highlighting it and clicking Paste.

5. Replace the <<Expr>> with the actual value to be used in the expression.

If you see a mistake and want to remove an element, operator or value, select it from the expression box and press =.

6. When the expression is complete, click on OK.
ADDITION VALIDITY TEXT

**USAGE:**

When defining a validation rule, you will need to enter text explaining the rule. This text appears in a box when the user "violates" the rule. Use it to explain why the entry is unacceptable.

If not in Design view, click on this button.

The current table design will be displayed, as shown below:

Select the field to be modified and then click in the property box labeled **Validation Text**. Enter the desired text.

**MISCELLANEOUS PROPERTIES**

**USAGE:**

Additional properties (located at the very bottom of the Design window) can be set which allow you to specify such options as whether a field is required, if zero length strings are allowed, and whether the field should be indexed.

**Required**

A field that must be entered. For example, if you had an employee file, you might want to ensure that a social security number is entered for each employee.
Allow zero length

This property is used to specify whether users are allowed to enter a zero length string. A zero length string is created when the user enters two quotation marks together - without any characters in between them. If a user simply leaves a field blank it is referred to as a “null value”. These two types of blank fields differ and can be searched independently. Use zero length strings for fields you do not have information for at the moment but would like to return to and fill in at a later time. You can then use the Format property field to display text depending on whether a field contains a null value (e.g., No Data Available) or a zero length string (e.g., Fill in Later).

Indexed

This is used to pre-sort a field so that Access can find and sort records faster. You should only index fields which you expect to search/sort on a regular basis. For example, a first name field would not be a good candidate for an index since there aren’t too many times you would be searching for someone by only their first name. On the other hand, indexing a last name field would be more useful.
<table>
<thead>
<tr>
<th><strong>Unicode Compression</strong></th>
<th>Access 2000 and later uses the unicode character encoding scheme for storing text, memo, or hyperlink fields. This universal code was developed so that almost all written languages in the world can be represented using a single character set. However, this code uses 2 bytes of storage instead of 1 as was in previous version. Therefore, you can use unicode compression to help reduce the amount of space required to store the data.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>IME Mode</strong></td>
<td>An IME (Input Method Editor) is an application that allows you to enter Asian text by converting your keystrokes into Asian characters. This property is used to set the Kanji Conversion Mode for this field.</td>
</tr>
<tr>
<td><strong>IME Sentence Mode</strong></td>
<td>This property can be used to specify the sentence mode for this field. Click on the down arrow to select the type of sentence mode you want to use.</td>
</tr>
</tbody>
</table>
INDEXING A TABLE

USAGE:

One of the major tasks of working with your database is to get it to run faster and more efficient. A large part of that speed and efficiency will come from the way you design your table structures.

However, you have probably noticed that a lot of your database work involves querying, searching and generating reports. These functions can take a long time if you are working with a large database or are searching several different fields.

One feature that can increase the performance of a table and, yet, is often overlooked in setting up a table, is the **index**.

Indexes are based on fields. Any field that is often used for searching or sorting should be indexed. Indexes work similar to a book index - if you need to find something quickly, Access looks it up in the index and can then go directly to the specified location.

Basically, an index is a pre-sorted list of the field, which means that Access doesn't have to sort each time you use that field.

Although indexes usually do speed up queries and searches, they can slow you down a little while adding or changing records, especially if you have a lot of indexes.

The reason is that Access must update all your indexes each time you add a new record or if you change the data in a field that has been indexed.

Primary keys are automatically indexed.
CREATING AN INDEX

In order to create or modify an index, you must be in the Design view for the table.

NOTE: You cannot index Memo, Hyperlink or OLE Object data type fields.

Once the table is selected, click on this button.

The current table structure will be displayed in the same window that was used to create the table, as shown below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CustomersID</td>
<td>AutoNumber</td>
<td></td>
</tr>
<tr>
<td>CompanyName</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>ContactFirstName</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>ContactLastName</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>City</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>State</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>PhoneNumber</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>FaxNumber</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>EmailAddress</td>
<td>Text</td>
<td></td>
</tr>
</tbody>
</table>

Notice the Indexed property (within the General tab) shown at the bottom of the window. Whenever you select a field in the top portion, this property will reflect the current setting.
To create/change an index for a field, highlight the field in the top portion of the design window and then click on the **Indexed** property in the lower portion of the window.

![Indexed](No)

Click on the down arrow (located to the right of the Indexed property) to view a list of index options.

The following options are available for indexing the field:

**No**
This is the default setting. Use this to remove an index from a field.

**Yes (Duplicates OK)**
Use this option to index the current field and to allow duplicate entries. You will need this for fields such as lastnames, zipcodes, etc. where you will probably have more than one record with the same entry.

**Yes (No Duplicates)**
Use this to index the current field but not allowing duplicate entries. You will need this to make sure that no two records will have the same value for this field. Possible uses are for social security numbers and employee ID numbers.
MODIFYING AN INDEX

Once an index has been created, you may want to modify the index by changing its name or the order in which the index is to be sorted (ascending or descending). To view/modify indexes, you must access the Indexes window.

Click on this button to open the Indexes window.

<table>
<thead>
<tr>
<th>Index Name</th>
<th>Field Name</th>
<th>Sort Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>City</td>
<td>City</td>
<td>Ascending</td>
</tr>
<tr>
<td>CompanyName</td>
<td>CompanyName</td>
<td>Ascending</td>
</tr>
<tr>
<td>PostalCode</td>
<td>PostalCode</td>
<td>Ascending</td>
</tr>
<tr>
<td>PrimaryKey</td>
<td>CustomerID</td>
<td>Ascending</td>
</tr>
<tr>
<td>Region</td>
<td>Region</td>
<td>Ascending</td>
</tr>
</tbody>
</table>

Use the first column to change the name of the index. The second column is used to select the field to be indexed (you can also use this window to create indexes).

The last column is where you define the order in which the field is to be indexed. When you select this column, you can click on the down arrow to choose **Ascending** (A-Z, 0-9) or **Descending** (Z-A, 9-0). In the bottom section there are three Index Properties that can be applied to the current index, as described below:

**Primary**

Specifies whether the current index is the primary key for the table. You can only have one primary key per table.

**Unique**

Specifies whether duplicates are allowed.

**Ignore Nulls**

Indicates whether or not null (empty) field values are to be included in the index.
MULTIPLE-FIELD INDEXES

When working with indexes, you may find that you end up searching on more than one field at a time in order to find a specific record.

For example, if you do a lot of searching based on a last name, you may find that you have several groups of people with the same last name.

To narrow the search down to the person you are looking for, you may end up adding a first name to the query, along with the last name.

Access allows you to combine more than one field into an index to help narrow such searches.

This index can also be used to prevent more than one record from having the same data in all the fields of the index.

Using the previous example, you would create an index that consists of both the last and first name. Since you end up using both fields during a query anyway, you may as well have one index to take care of both fields.

You could then make the index unique, preventing more than one record with the same last name and first name.

If the last name field is defined first in the index, you would be allowed to have multiple records with the same last name.
To create a multiple-field index, you would enter a name for the index and then specify the fields to be indexed on multiple rows.

Access will include all rows as part of the same index until it reads a new index name. Don’t add anything else to the Name column (other than the initial index name). Simply add each field to be included in the combined index on a new row. The order in which you place the fields determines the sorting order.

In the figure shown below, the multi-field index is labeled Name and consists of both the Lastname and Firstname fields:

If you want to make sure that no two records contain the same last and first name, use the index properties to make the index Unique.
CREATING PERMANENT RELATIONSHIPS

USAGE:

Within queries, related tables can come in quite handy when working with large amounts of data that have been divided into multiple tables. However, relating two tables in a query only creates a relationship for that one query. If you need to have the related tables available for other database objects (such as forms and reports), you will not want to create the relationship each time a new object is added. Instead, you will want to create a permanent relationship between the tables.

There are three types of relationships, as discussed below:

One to One

This is the simplest form of a relationship in that each record within one table relates to one record within the other.

For example, for each patient in a hospital there is only one medical record.

One to Many

In this type of relationship, each record within one table is related to one or more records within the other. This is the most commonly used relationship.

For example, for each student in one table, there may be one or more related classes in the other table.

Many to Many

In this type of relationship, one or more records within one table is/are related to one or more records within the other. However, to relate these two tables you must create a third table containing the primary key fields from the other two tables. This third table is referred to as a Junction Table and is used to relate the other two tables together.

For example, to create a detailed order form you might need the order information from one table and the product information from a second table. Each order may contain references to multiple products whereas each product may be part of multiple orders.
To create a relationship, follow the steps outlined below:

1. From within the database window, click on this tool to open the Relationships window.

A large blank window will be displayed in the background with the following dialog box displayed on top:

2. From within this dialog box, highlight each table you want to include in the relationship and click on the [Add] button.

As you add each table, Access will attempt to establish a relationship between the table and the others you have selected. If an identical field name is found in both tables, Access assumes the tables are related and creates a link between the two common fields.

3. When you are done adding each table, click on the [Close] button.

4. If Access has not automatically established a relationship between the tables, you will need to manually connect the tables based on their common fields.
NOTE: The common fields do not have to have the same name, but they do need to be of the same type. For example, you cannot link a text field with a numeric field. Also, the field from the table that you are linking from needs to be a primary key.

As you relate two tables together, the following dialog box will be displayed:

Make sure that the correct fields are linked and then click on the Create button.

Repeat this process for each relationship.

Click on to close the Relationships window.

From now on, whenever you add multiple tables to a query, form, or report, any relationships established between the tables will automatically be recognized.
VIEWING SUBDATASHEETS

When relating multiple tables, you typically have a master/primary record that has several sub/detail records related to it. For example, if you have an employee database with one table containing the departments and another table containing the employees, you might relate the two tables with the department as the primary and the employee table as the details. In this example, for each department record there would be several employees related. If you were going to create a departmental report that shows which employees work in each department, you would display each department followed by its list of employees.

The list of employees is a subdatasheet of the department since the department name determines which employees will be listed.

Subdatasheets can be used to view and edit joined data within related tables, forms, queries, or subforms.

The simplest way of viewing a subdatasheet is to follow the steps outlined below:

1. Open the primary table in datasheet view.

   It should look similar to the following diagram in that there are plus signs placed to the left of each record:

<table>
<thead>
<tr>
<th>CostId</th>
<th>FirstName</th>
<th>LastName</th>
<th>Address</th>
<th>City</th>
<th>State</th>
<th>Zipcode</th>
</tr>
</thead>
<tbody>
<tr>
<td>▽</td>
<td>Willy</td>
<td>Cruz</td>
<td>1 School St</td>
<td>Anaheim</td>
<td>CA</td>
<td>92031</td>
</tr>
<tr>
<td>▽</td>
<td>Patrice</td>
<td>Queen</td>
<td>1339 Sunset St</td>
<td>Hollywood</td>
<td>CA</td>
<td>90262</td>
</tr>
<tr>
<td>▽</td>
<td>Nora</td>
<td>Pulsaski</td>
<td>199 Maria Vista</td>
<td>San Jose</td>
<td>CA</td>
<td>92236</td>
</tr>
<tr>
<td>▽</td>
<td>Alan</td>
<td>Fox</td>
<td>999 Piedmont N</td>
<td>San Diego</td>
<td>CA</td>
<td>99402</td>
</tr>
<tr>
<td>▽</td>
<td>Yvonne</td>
<td>Anders</td>
<td>01 E Orange</td>
<td>Anaheim</td>
<td>CA</td>
<td>92002</td>
</tr>
<tr>
<td>▽</td>
<td>Caitlin</td>
<td>May</td>
<td>9033 W Putnam</td>
<td>San Jose</td>
<td>CA</td>
<td>92262</td>
</tr>
</tbody>
</table>
The plus sign indicates that the primary record has subdata that can also be displayed.

Click on the plus sign to open the subdatasheet and display the underlying data, as illustrated in the example shown below:

<table>
<thead>
<tr>
<th>CustId</th>
<th>Firstname</th>
<th>Lastname</th>
<th>Address</th>
<th>City</th>
<th>State</th>
<th>Zipcode</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>Willy</td>
<td>Cruz</td>
<td>1 School St</td>
<td>Anaheim</td>
<td>CA</td>
<td>92020</td>
</tr>
<tr>
<td>101</td>
<td>Patrice</td>
<td>Queen</td>
<td>1339 Sunset #</td>
<td>Hollywood</td>
<td>CA</td>
<td>92362</td>
</tr>
<tr>
<td>102</td>
<td>Nora</td>
<td>Pulaski</td>
<td>199 Maria Vista</td>
<td>San Jose</td>
<td>CA</td>
<td>92362</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DateSold</th>
<th>ItemCode</th>
<th>Quantity</th>
<th>SaleAmount</th>
</tr>
</thead>
<tbody>
<tr>
<td>10/20/92</td>
<td>1</td>
<td>21</td>
<td>$2,394.25</td>
</tr>
<tr>
<td>9/13/92</td>
<td>1</td>
<td>21</td>
<td>$1,096.75</td>
</tr>
<tr>
<td>3/25/93</td>
<td>5</td>
<td>29</td>
<td>$1,300.00</td>
</tr>
<tr>
<td>1/19/93</td>
<td>4</td>
<td>5</td>
<td>$243.01</td>
</tr>
</tbody>
</table>

To close the subdatasheet, simply click on the minus sign "-".
ENFORCING REFERENTIAL INTEGRITY

When you link two tables, for every record within one table there is at least one related record within the other table. For example, if you have one table with a list of customer names and have linked that table to another table containing an order history, you should expect that for each order listed, there will be a matching customer.

However, if you were to delete a customer record, the matching orders would no longer have a customer to refer to. In essence, you would still have orders but not be able to tell which customer placed the orders. These incomplete orders are referred to as “orphan data”. Unless deleted, these orders continue to be stored in the Orders table taking up space yet never being referenced.

Referential integrity ensures that if you delete a primary record (customer) that its detail records (orders) are automatically deleted. This keeps the integrity of the linked tables intact.

Referential integrity also works when changing data in a field on which tables have been linked. For example, changing a customer’s ID in the Customer table without changing the customer’s ID within the Orders table would cause the same problem as mentioned above. Referential integrity would once again fix the problem by changing all of the orders automatically to match the new customer ID.

To enable referential integrity, follow these steps:

1. From within the database window, click on the Relationships tool.

2. Double-click on the relationship line that connects the primary table to the related table.
The following dialog box will be displayed:

![Diagram of Access 2003 Table Design dialog box]

3. Check the box labeled **Enforce Referential Integrity**.

If you want Access to automatically update the primary key in related records whenever you change the primary key in the primary table, check the box labeled **Cascade Update Related Fields**.

If you want Access to automatically remove related records whenever you delete a primary record, check the box labeled **Cascade Delete Related Records**.

4. When done, click on **OK**.

You can test the referential integrity by changing a primary field or deleting a primary field.

Once the primary field has been updated or deleted, the related records should automatically be updated or deleted.

You will, however, be warned about the change or deletion before the action is taken.
PRINTING DATABASE RELATIONSHIPS

If you want to have a hard copy reference of the database relationships, follow these steps:

1. From within the database window, click on the Relationships tool.

2. Next, access the following menu:

```
MENU:
File
New...
Open...
Get External Data
Close
Save
Save As...
Export...
File Search...
Page Setup...
Print Preview
Print Relationships...
Send To
Database Properties
1-Class Data|Sample.mdb
Exit
```

3. To actually print the relationship information within the preview window, click on this tool.