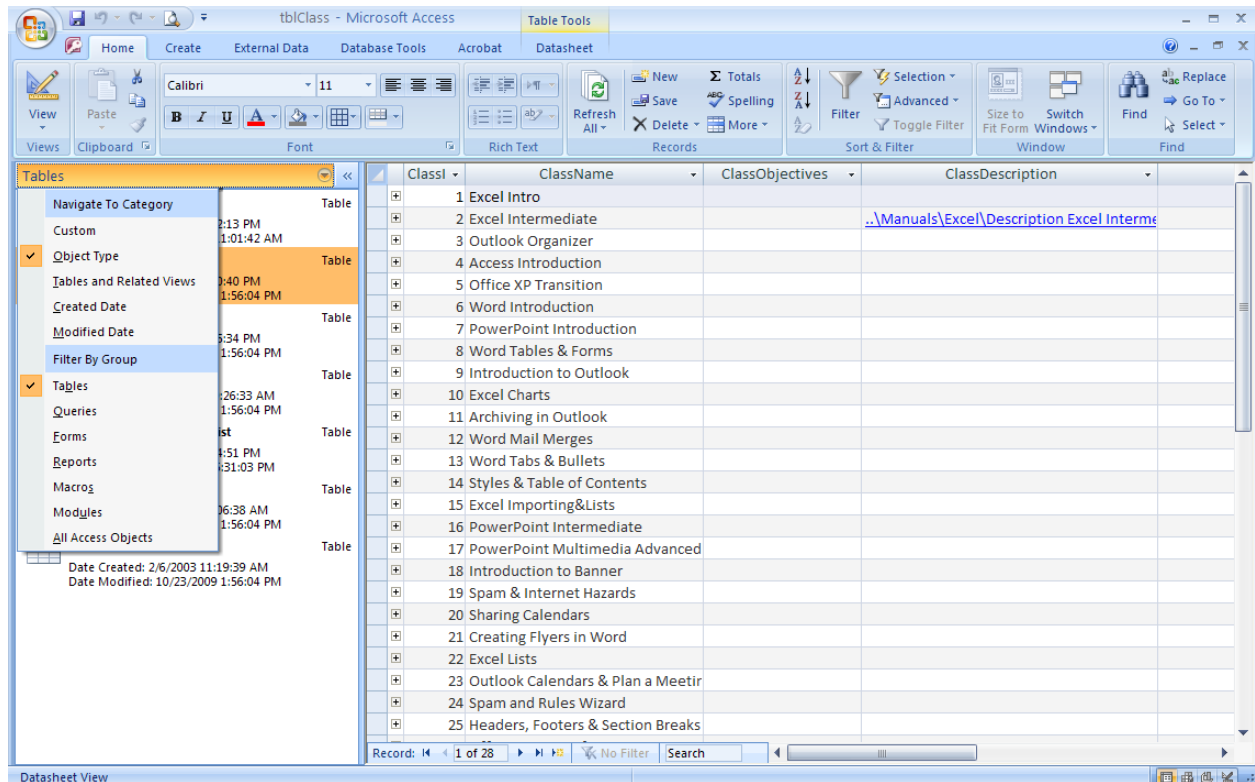


# Introduction To Database Design & Microsoft Access 2007



## Table of Contents

|  |           |
|--|-----------|
| <b>Overview .....</b>                  | <b>1</b>  |
| Objectives.....                        | 1         |
| <b>What is a Database? .....</b>       | <b>2</b>  |
| <b>Flat Database.....</b>              | <b>3</b>  |
| <b>Relational Database .....</b>       | <b>4</b>  |
| <b>Database Design .....</b>           | <b>5</b>  |
| <b>Getting Started .....</b>           | <b>9</b>  |
| <b>Navigating a Database.....</b>      | <b>10</b> |
| <b>Access Database Objects.....</b>    | <b>11</b> |
| <b>Tables.....</b>                     | <b>12</b> |
| Datasheet View.....                    | 12        |
| <b>Forms .....</b>                     | <b>13</b> |
| <b>Queries.....</b>                    | <b>14</b> |
| <b>Reports.....</b>                    | <b>17</b> |
| <b>Working in Datasheet View .....</b> | <b>18</b> |
| Datasheet Toolbar .....                | 18        |
| Add Records.....                       | 19        |
| Save a Record .....                    | 20        |
| Select a Record .....                  | 20        |
| Edit a Record.....                     | 20        |
| Delete a Record .....                  | 20        |
| Check Your Spelling .....              | 21        |
| <b>Filter by Selection .....</b>       | <b>21</b> |
| <b>Filter by Form .....</b>            | <b>21</b> |
| <b>Load from Query .....</b>           | <b>23</b> |
| <b>Table Design View.....</b>          | <b>24</b> |
| Field Properties .....                 | 26        |
| Primary Key .....                      | 30        |
| <b>Create a Database .....</b>         | <b>31</b> |
| Saving a Table.....                    | 34        |
| Creating a Lookup Wizard Field.....    | 37        |
| <b>Multiple-value problems .....</b>   | <b>39</b> |

|   |           |
|---|-----------|
| Intermediary Tables .....                       | 41        |
| Types of Relationships .....                    | 42        |
| <b>Create Relationships.....</b>                | <b>43</b> |
| Referential Integrity .....                     | 44        |
| Modify Table .....                              | 45        |
| Adding a Field .....                            | 45        |
| Deleting a Field.....                           | 45        |
| Moving a Field .....                            | 46        |
| <b>Modifying Database Objects .....</b>         | <b>47</b> |
| Renaming a Table .....                          | 47        |
| Copying a Table .....                           | 48        |
| Deleting a Table.....                           | 50        |
| Printing the Table.....                         | 50        |
| Printing Selected Records.....                  | 50        |
| <b>Queries.....</b>                             | <b>51</b> |
| What is a Query? .....                          | 51        |
| <b>The Simple Query Wizard .....</b>            | <b>52</b> |
| Running Queries .....                           | 54        |
| Switch between Views .....                      | 54        |
| Query Design View (Query-By-Example grid) ..... | 55        |
| <b>Creating a Query in Design View .....</b>    | <b>56</b> |
| Running Queries .....                           | 58        |
| <b>Summary.....</b>                             | <b>59</b> |

## Overview

Many of us have been using Microsoft Office applications such as Word and Excel, but there is another program in the MS Office suite called Access, which many of us have not used. What is Access?

Access is a database management program that allows you to store, organize and retrieve information. Access also allows you to query data and produce some nice looking reports. Many of us may be using Word or Excel to store data. In Word you may be using a table to store data. Excel does have some limited database capability. The difference is that Access is a relational database. Excel does NOT have the functionality or power of a relational database like Access.

[If you are still undecided about what program to choose for the task at hand, watch this video.](#)

## Objectives

In this class, you will learn the how to design and create a database, create tables, become familiar with the different objects in an Access database. When you have completed this class, you will be able to:

- ✧ Be familiar with basic relational database concepts and database design
- ✧ Identify all of the Access database objects
- ✧ Create tables
- ✧ Distinguish between table datasheet and design view
- ✧ Create fields and set their data types and properties
- ✧ Define the primary key
- ✧ Modify table records - edit, add, delete records
- ✧ Navigate in a table
- ✧ Modify table properties

## What is a Database?



A database is a **collection of related data**. The database **stores** information. This information can be manipulated and viewed in many different ways.

A Database Management program such as Access stores information in tables that are related to each other. It is primarily designed for two main purposes:

- \* To store, add, delete, and update the data in the database.
- \* To provide various ways to view the data in the database. For example, you can display data on your screen in a form or print out a report.

With the power of Access you can maintain and manipulate large amounts of data and create reports.

Can you think of some examples where a database could be used? Some examples are:

- \* Store and/or query information on applicants, training or programs.
- \* Query or extract information on donors and donations made to the college and produce reports.
- \* Store a list of students, addresses for a mail merge.
- \* Personnel database with separate but related tables for employees, departments and payroll information.

**Databases allow us to organize, manipulate, analyze information and create reports.**

## Flat Database

Many of you have probably created a **flat database** which consists of only one table. Below is an example of a Word table that stores information on computer books.

|   | ISBN          | Title        | AuID | AuName      | AuPhone      | PubID | PubName     | PubPhone     | Price   |
|---|---------------|--------------|------|-------------|--------------|-------|-------------|--------------|---------|
|   | 0-12-345678-9 | Jane Eyre    | 1    | Bronte      | 123-444-4566 | 3     | Small House | 714-000-0000 | \$49.00 |
|   | 0-99-999999-9 | Emma         | 1    | Austen      | 111-111-1111 | 1     | Big House   | 123-456-7890 | \$20.00 |
|   | 0-55-123456-9 | Main Street  | 10   | Jones       | 123-333-3333 | 3     | Small House | 714-000-0000 | \$22.95 |
|   | 0-321-32132-1 | Balloon      | 11   | Snoopy      | 321-321-2222 | 3     | Small House | 714-000-0000 | \$34.00 |
|   | 0-321-32132-1 | Balloon      | 12   | Grumpy      | 321-321-0000 | 3     | Small House | 714-000-0000 | \$34.00 |
|   | 0-321-32132-1 | Balloon      | 13   | Sleepy      | 321-321-1111 | 3     | Small House | 714-000-0000 | \$34.00 |
|   | 0-11-345678-9 | Moby Dick    | 2    | Melville    | 222-222-2222 | 3     | Small House | 714-000-0000 | \$49.00 |
|   | 0-103-45678-9 | Iliad        | 3    | Homer       | 333-333-3333 | 1     | Big House   | 123-456-7890 | \$25.00 |
|   | 1-22-233700-0 | Visual Basic | 4    | Roman       | 444-444-4444 | 1     | Big House   | 123-456-7890 | \$25.00 |
|   | 1-1111-1111-1 | C++          | 4    | Roman       | 444-444-4444 | 1     | Big House   | 123-456-7890 | \$29.95 |
|   | 0-91-045678-5 | Hamlet       | 5    | Shakespeare | 555-555-5555 | 2     | Alpha Press | 999-999-9999 | \$20.00 |
|   | 0-555-55555-9 | Macbeth      | 5    | Shakespeare | 555-555-5555 | 2     | Alpha Press | 999-999-9999 | \$12.00 |
|   | 0-99-777777-7 | King Lear    | 5    | Shakespeare | 555-555-5555 | 2     | Alpha Press | 999-999-9999 | \$49.00 |
|   | 0-123-45678-0 | Ulysses      | 6    | Joyce       | 666-666-6666 | 2     | Alpha Press | 999-999-9999 | \$34.00 |
|   | 0-91-335678-7 | Faire Queen  | 7    | Spencer     | 777-777-7777 | 1     | Big House   | 123-456-7890 | \$15.00 |
|   | 0-12-333433-3 | On Liberty   | 8    | Mill        | 888-888-8888 | 1     | Big House   | 123-456-7890 | \$25.00 |
|   | 0-55-123456-9 | Main Street  | 9    | Smith       | 123-222-2222 | 3     | Small House | 714-000-0000 | \$22.95 |
| * |               |              |      |             |              |       |             |              |         |

What problems do you see with a single flat database?

- \* Unnecessary duplication of data. Wastes storage space.
- \* It is more difficult to update and maintain.
- \* It can be cumbersome to find or summarize information.
- \* Greater chance for data entry errors.

The main problem with a flat table is redundancy.

- \* The publishers name and phone number are repeated. If their phone number changes, how many updates would you have to make?
- \* The authors' name and phone number is duplicated. If an authors Phone number changes how many updates would you have to make?

To avoid these problems, you need to break up the information into **separate related** tables. This is where the power and versatility of a relational database such as Access should be used. In this example, you would have the following tables: Publisher's, Author's, Books, and an Intermediary table Book/Author. Intermediary tables consist of many to many relationships. For example, an author can write many books and a book can have more than 1 author. So this table could have ISBN and Author ID. An ISBN can be repeated many times depending on how many authors wrote the book. An author can be listed many times for different books.

| Same ISBN     | 2 authors (ID) | 2 ISBN        | 1 author (ID) |
|---------------|----------------|---------------|---------------|
| 0-55-123456-9 | 9              | 1-1111-1111-1 | 4             |
| 0-55-123456-9 | 10             | 1-22-233700-0 | 4             |

## Relational Database

Access is a relational database. In a relational database data is organized in related tables. In related tables, one or more fields are linked to fields in another table. This link ensures that you can enter only those values that have corresponding entries in the other table. For example, you can't enter an order without a customer!

A relational database can have multiple tables that contain data about various entities, such as customers, orders products. An entity is any object that has a distinct set of properties.

A relationship is a connection between two or more tables based on common fields.



*Linking tables together through common fields.*

Tables are the building blocks of a database. From the tables you can generate forms, queries, macros, or reports. Relationships must be established between the tables before we can create queries using multiple tables.

### Definitions of a few key terms

A **table** is a collection of related information that is organized in columns (fields) that describe an entity (record). In the most basic sense, an entity is a person, place, thing, or idea.

A **Record** is a collection of information about a specific entity, such as a student, applicant, program, or project. If we have pieces of information, on one subject, we have a **record**. For example, if we know that Jane Doe is a programmer, with a 2002 Blue Toyota Camry, license plate number 4XN746. We now have a record on Jane Doe. Each of these pieces of information by itself (blue car) is data, but the information altogether makes a record.

A **Field** is a category of information. A formal definition of field: a column of data sharing properties and a data type. We organize this type of information into categories or **fields**. In our Jane Doe example we would have fields such as name, occupation, car color, year, make and model. This would give us a total of six fields.

**Primary key:** The fields or set of fields that **uniquely identify** any given row. It ensures no two rows have the same value. Nulls are not allowed. Primary keys are an important concept because they provide the only way of exactly addressing a specific record in a table. In our Jane Doe example, the license plate number could be designated as the primary key.

## Database Design

**Step 1:** Identify the **purpose** of your database.

What type of information are you trying to **store** and **retrieve** from your database? Each database should be set up for a specific purpose. For example if you're keeping track of donations, you need to plan and design a database that can store related data such as donors and events as well. Consider the following issues:

- ✧ What types of queries do you need to perform on the database?
- ✧ What types of reports will need to be produced?
- ✧ What forms will be needed for data entry purposes?
- ✧ How will the data be sorted, grouped or filtered?

**Tip:** Look for database models that you can follow. There are many resources (online, books, etc) from which you can garner information. Microsoft Access offers templates on your computer (local templates) and featured online templates that you can download and modify it to suit your needs.





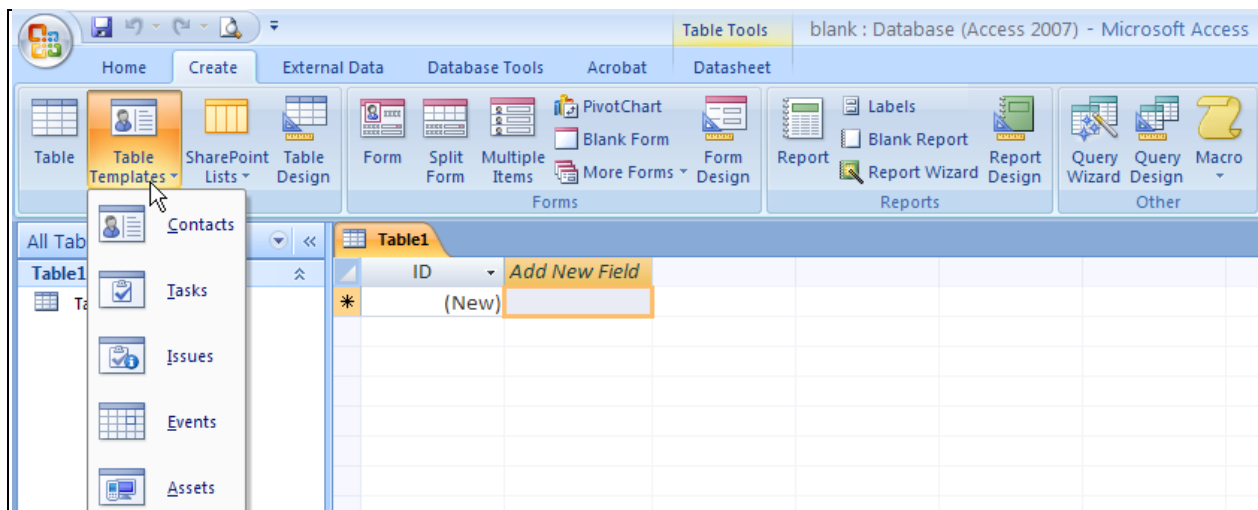
**Step 2:** Determine the **tables** you need.

Each table should focus on one topic and should store only the data related to that topic. For example, to track donations you may have separate tables for Donors, Donations and Events.

Remember our flat database example? You should separate fields into related tables for the following reasons:

- ✱ Eliminate redundant information. For example a customer address change is much easier to implement if the data is stored only in the Customers table. It is only necessary to make the change once, thereby reducing the amount of data entry required and potential for data-entry errors. It also saves storage space.
- ✱ Manage data easier. Large tables can be difficult to work with. It may be cumbersome to find or summarize information.
- ✱ To make future changes to the database design easier. Maintaining separate tables makes it easier to accommodate change. You can avoid having to frequently restructure the fields to accommodate new data.
- ✱ Join or relate the tables to one another to view information from multiple tables in a query.

Even if you choose a blank database, Microsoft Access offers Table Templates that you can use and modify for your own purposes. Just go to the Create tab and select Table Templates. Contacts, Tasks, Issues, Events and Assets table templates are available.



**Step 3: Breakdown the tables into **fields**.**

- \* The fields in a table should relate to the subject of the table. For example, it is intuitive for a user to look in the Customers table for the address of a particular customer.
  - Avoid inconsistent dependencies that can occur when you add fields to a table that are not directly related to the table. For example, it does not make sense to look for the salary of the employee who calls on that customer in the customer table. The employee's salary is related to, or dependent on, the employee and thus should be moved to the Employees table. Inconsistent dependencies can make data difficult to access.
- \* Consider the **type of data** that you plan to put in each field. Some common types are Text, Integer, Currency, and Date/Time.
- \* Store information in its **smallest logical part**. If you combine more than one kind of information in a field, it is difficult to search or sort the data.
  - For example, create separate fields for first name and last name.
  - Don't create one field to store all the address information, break it down into street, city, state, zip.
- \* Do not include calculated data. In most cases you do not want to store the result of a calculation in a table. Access should only perform the calculation when you want to see the result (in a form, query or report).

Note: If you have repeating (the same) information in several tables, it is a clue that you have unnecessary fields in some tables.

**Step 4: Determine the **Relationships**.**

Joining or relating the tables to one another will allow you to view information from multiple tables at the same time. After you define relationships between the tables, you have to flexibility to bring the data from multiple tables together in a query, form or report.

Decide how each table will be related to the other tables in the database. If necessary, add fields to create relationships.


To set up a relationship between two tables, add one table's primary key to the other table, so that the field appears in both tables. (Open the secondary table and add the field using the same name and data type.)\* The field in the second table is called the **foreign key**.

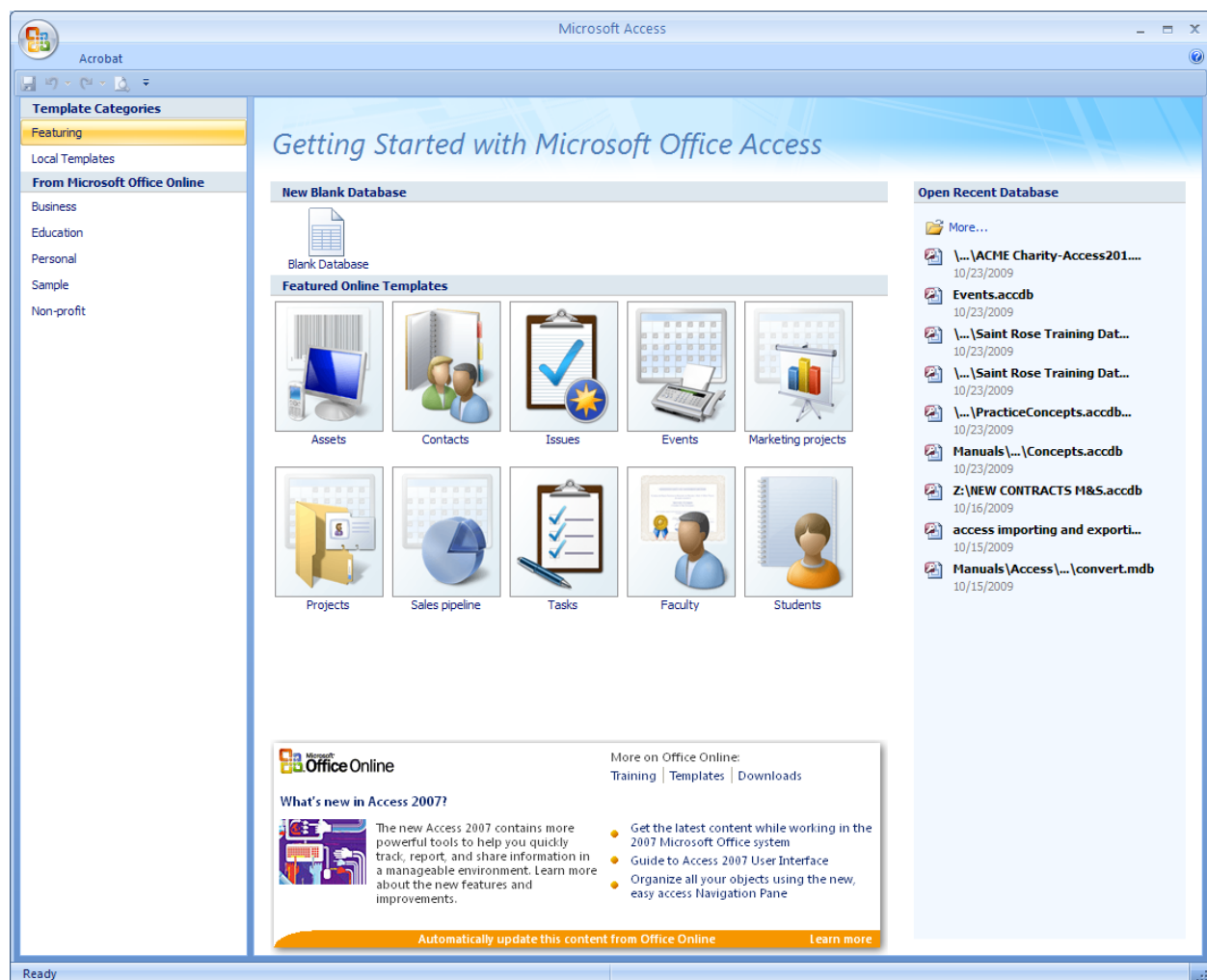
\*Exception If the primary key is an **Auto number**, the foreign key (the corresponding field) must have a data type of a **Long Integer**.

**Step 5: Refine the Design.** Create your tables, specify relationships between the tables, enter a few records and see if the database gives you the answers you want. Create rough drafts of your forms and reports and see if they show the data you expect. Look for unnecessary duplications of data and eliminate them.

- ✧ Did you forget any fields? Go back and add them.
- ✧ The intersection of every column and record should contain one and only one value.
- ✧ Did you choose a good primary key for each table? Can you search for specific records? Make sure that you won't need to enter a value in a primary key field that will result in a duplicate value.
- ✧ Are you repeatedly entering duplicate information in one of your tables? If so you probably need to divide the table into two tables, with a one-to-many relationship.
- ✧ Do you have tables with many fields, a limited number of records and many empty fields in individual records? If so think about redesigning the table so it has fewer fields and more records.

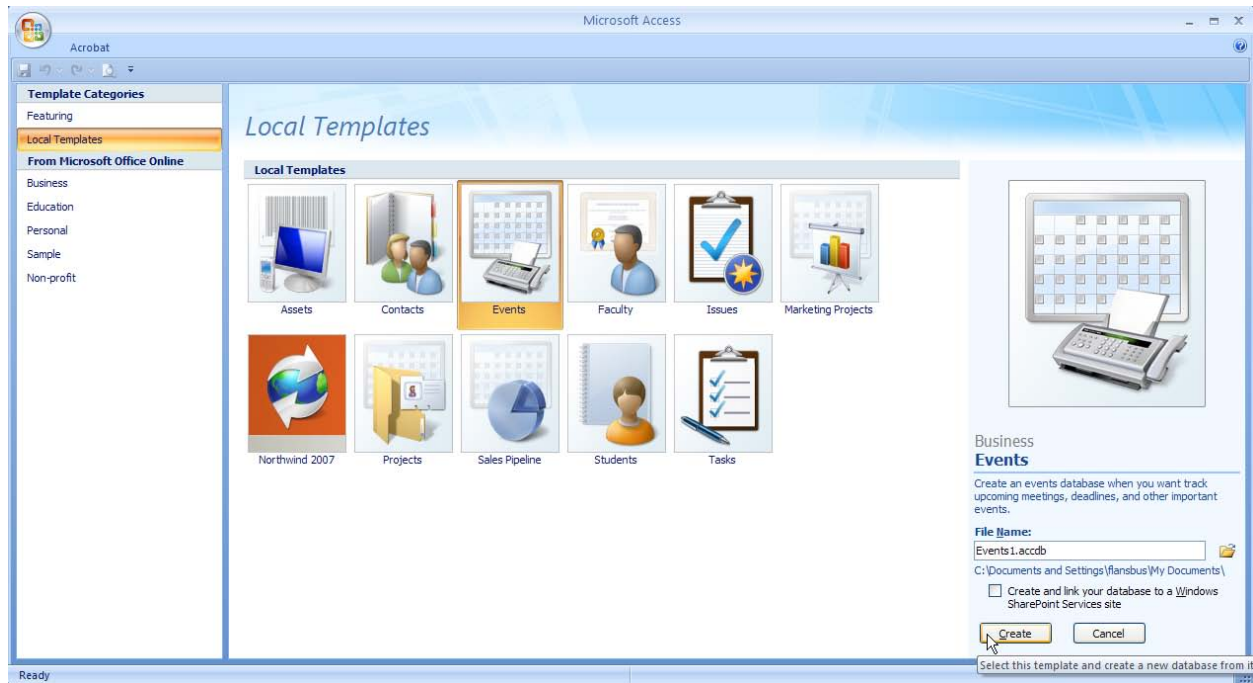
## Getting Started

1. Go to **Start → All Programs → Microsoft Office → Microsoft Access 2007**.
2. You can create a New Blank database, choose from a Template, or open an existing database.
  - \* The **Blank Database** choice allows you to create your own database from scratch.
  - \* **Local Templates** reside on your computer. Microsoft Office Online also offers templates from various categories, that you can download. These templates provide the design and structure of the database, so you just enter the data.
3. **Open Recent Databases** displays a list of currently opened databases from which you can choose, or select **"More"**  to navigate and select a database in the **Open** dialog box.



## Example

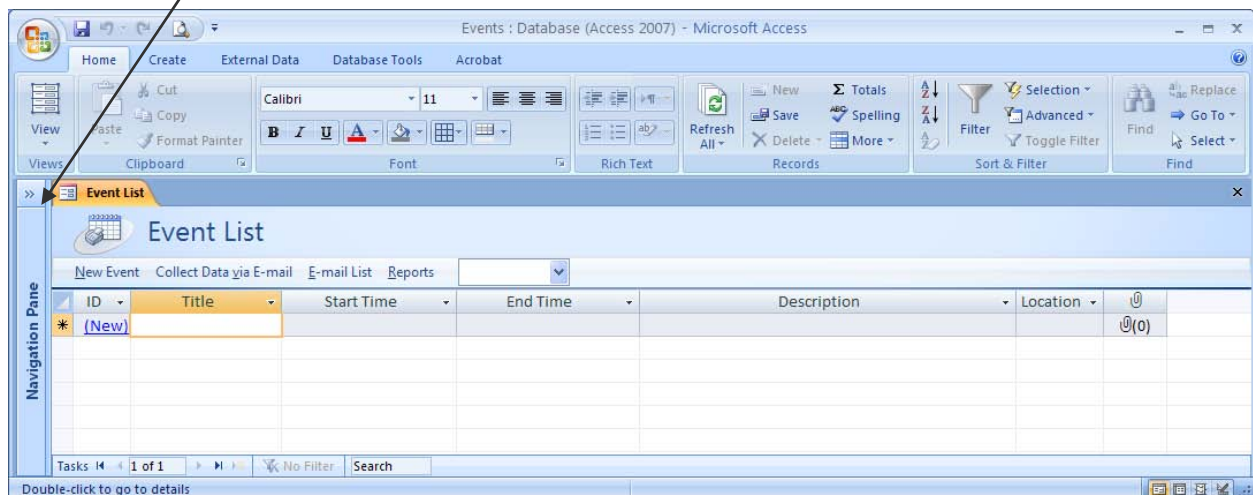
Create a new database from the **Events** template.



## Navigation Pane

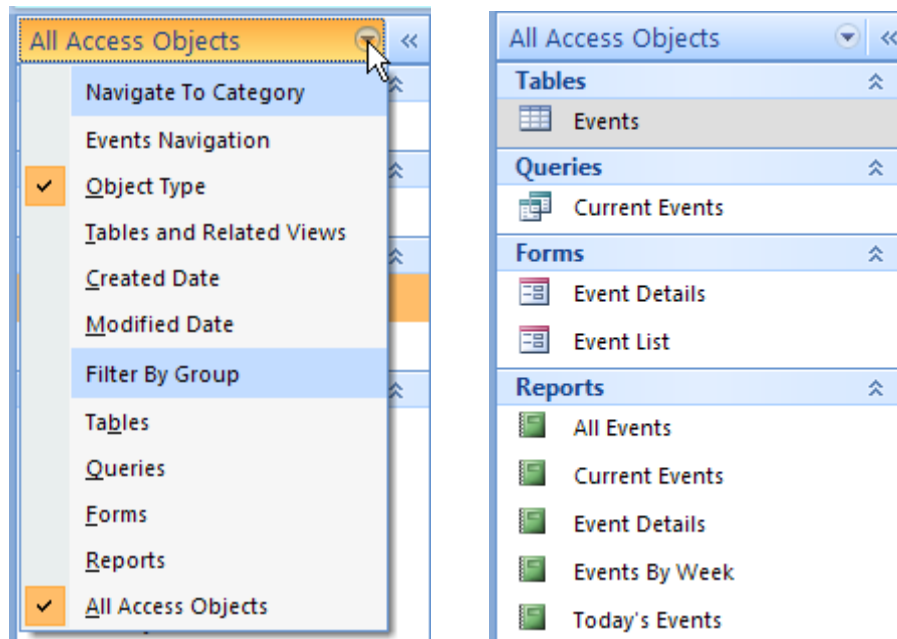
When a database opens, the name and format of the database appears in the title bar, and the navigation Pane opens on the left side of the Access Window.

Click the **Expand Navigation Pane** arrow (chevron).



An empty table with sample fields opens in datasheet view.

The navigation Pane displays the objects in your database. Objects include tables, forms, queries, and reports. You can use the Navigation pane to change which objects display and in what order. For instance, you can display just tables, or all objects as shown below.



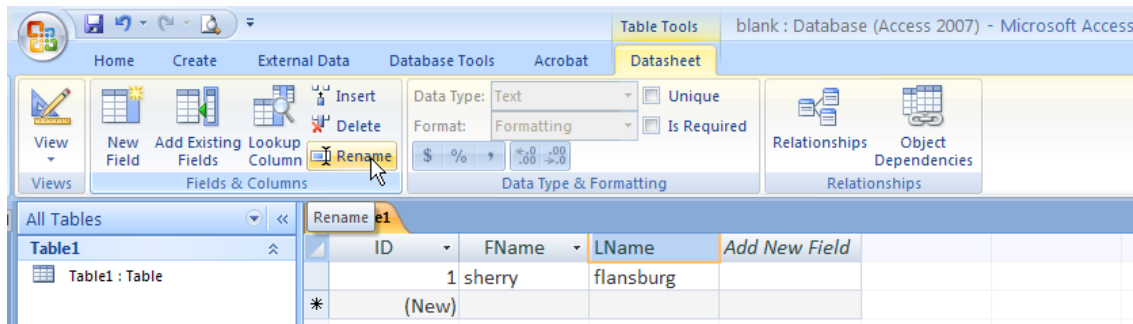
## Access Database Objects

The following table describes the objects you can have in an Access database.

|                |  |
|----------------|--|
| <b>Tables</b>  | A Table stores a collection of related information organized in columns (fields) and rows (records).   |
| <b>Queries</b> | A query is used to extract specific information from your table(s). A Query takes information from the selected tables and displays a subset of data that meets your criteria. |
| <b>Forms</b>   | A form is a graphical representation used to enter or view data into a table.  |
| <b>Reports</b> | Presents data from a query or table and places it in neat, organized and readable form.  |
| <b>Macro</b>   | A stored set of commands that can be used to automate database tasks.  |
| <b>Modules</b> | Automates and customizes database operations using visual basic. Modules are a collection declarations, statements and procedures stored together as a unit.                   |

## Tables

When you create a new blank database or create a database based on a table, Access automatically opens in Datasheet View. When you create a blank database you need to **rename the default field names** as shown below.

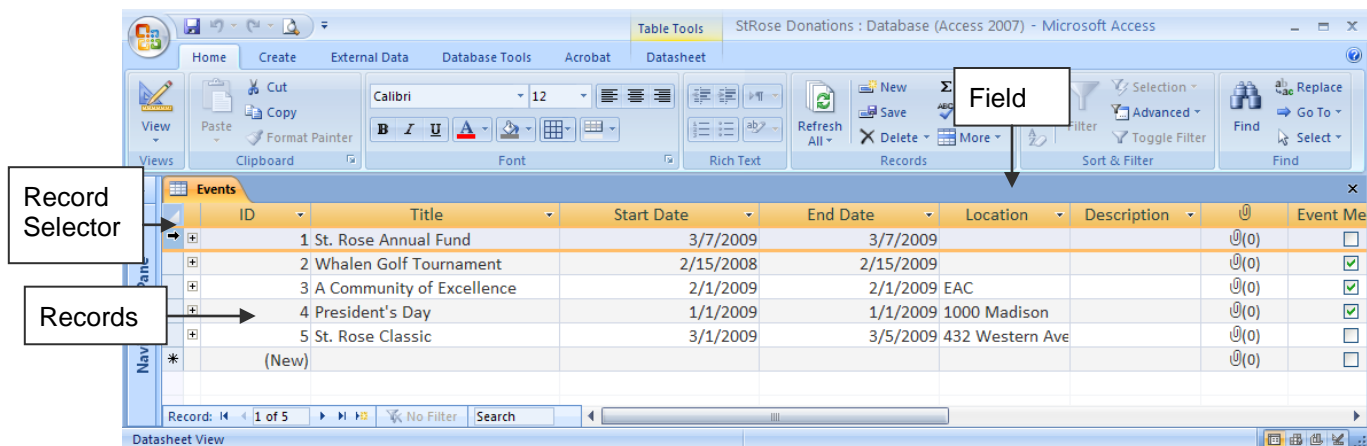


## Datasheet View

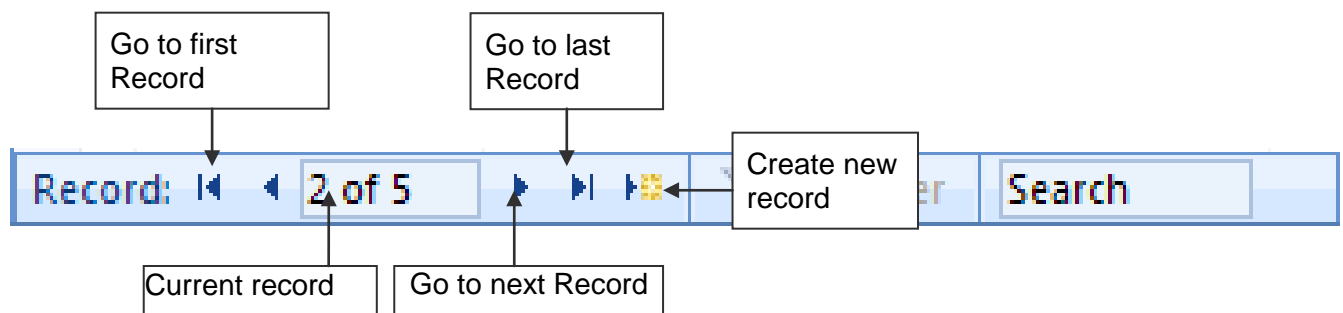
The Datasheet View can be used to view, enter, edit, add, delete, sort and filter data.

The field names appear at the top of each column. Each row represents one record. On the far left of your table is the vertical “record selector.” An arrow located in that area shows you which record is selected.

Because we used a template, the field names already appear in our table. You need to enter the data into the records.



The Record Bar allows you to navigate through the data.



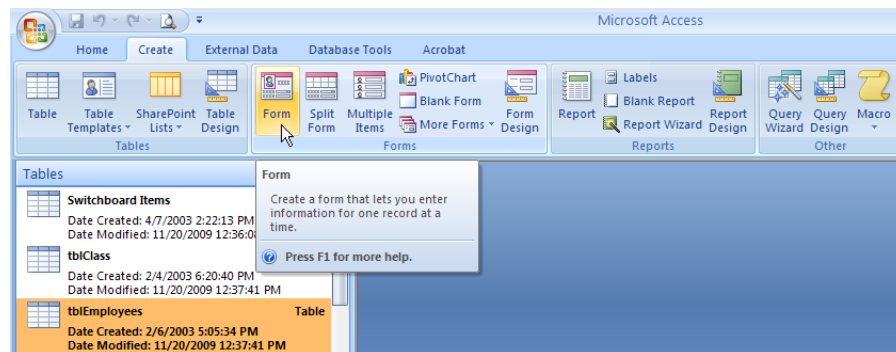
## Forms

A form is an Access database object that allows you to view, edit, and add data to a table. The datasheet view of a table shows you a grid of fields and rows. A form typically shows just one record at a time.

A form is a **graphical representation used to enter data into the tables**. A form is a simple way to enter data all at once rather than in multiple tables. Data input from a form will enter data directly into the respective table/tables. You can build the form from a table or a query.

To create a basic form, do the following:

1. In the navigation pane, select a table on which to base the form.
2. Click on the Create tab
3. In the forms group, click Form. This will create a Columnar form that includes all the fields from the selected table.



An example is shown below. This is a registration form, which can be used to view or add attendees. Once the data is entered in the form, the data is stored in the registration table.

A screenshot of a Microsoft Access form titled 'Class Schedule Registration Form'. The form has a light yellow background. On the left, there is a list of fields: 'RegistrationID' (text box), 'EmployeeID' (dropdown menu showing 'Allen'), 'ScheduleClassID' (dropdown menu showing '1'), 'RegDate' (text box showing '1/1/2002'), 'Attend?' (checkbox checked), 'WaitList?' (checkbox unchecked), and 'Notes' (text area). On the right side of the form, there is a photograph of a stone building with a sign that reads 'THE COLLEGE OF SAINT ROSE'. At the bottom of the form, there is a status bar that says 'Record: 1 of 319' and 'Form View'.



## Queries

- \* A query is a database object that retrieves and displays selective data from one or more tables or from other queries. You can use a query to retrieve data meeting specific conditions. For example, I can display all classes a John Doe attended in June. Query results are similar to the results of a filter, but a query can be saved permanently, whereas a filter provides only a temporary view.
- \* When you run a query it displays the results in datasheet view. The result of the query is dynamic; hence the results are called a dynaset. The dynaset looks and acts like a table; it is a “live” view of one or more tables. **WARNING:** If you make changes to the data in the query, the data in the table will also change!!
- \* A query can be based on data from one or more tables. Queries usually connect two or more tables through a relationship between a common field in both tables, such as a key field (a field unique to each table).
- \* Queries allow you to perform calculations on the data; to create data sources for forms, reports, charts and other queries; to make changes to tables and create new tables.
- \* Select Query - This is the most common type of query; it selects information from one or more tables and returns only the records that meet the criteria.

### **Plan a Query**

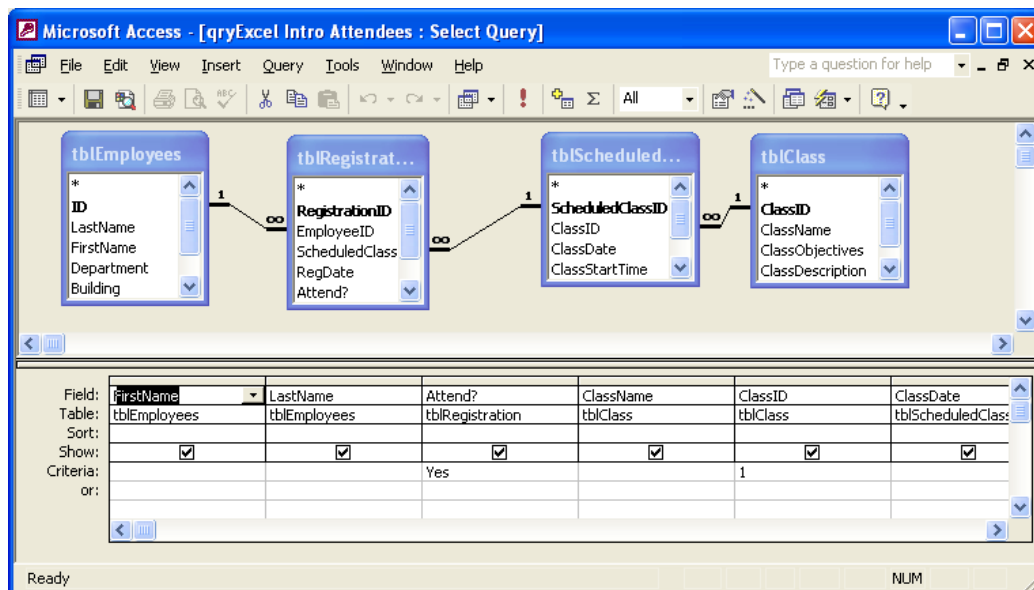
1. Determine the tables from which you will extract the fields
2. Determine the fields that you want to see in the query result
3. Specific the conditions that you want the data to meet.

## Create a query in Design View

1. In the navigation pane, select the Queries object.
2. On the Create tab, click Query Design.
3. In the show table dialog box, select the table you want to add to the query and click **Add**.
4. Click close to return to design view.
5. Select the fields that you want to display in the query result. Apply any sorts or criteria to the fields.
6. Click the Run button or switch to datasheet view to see the results.

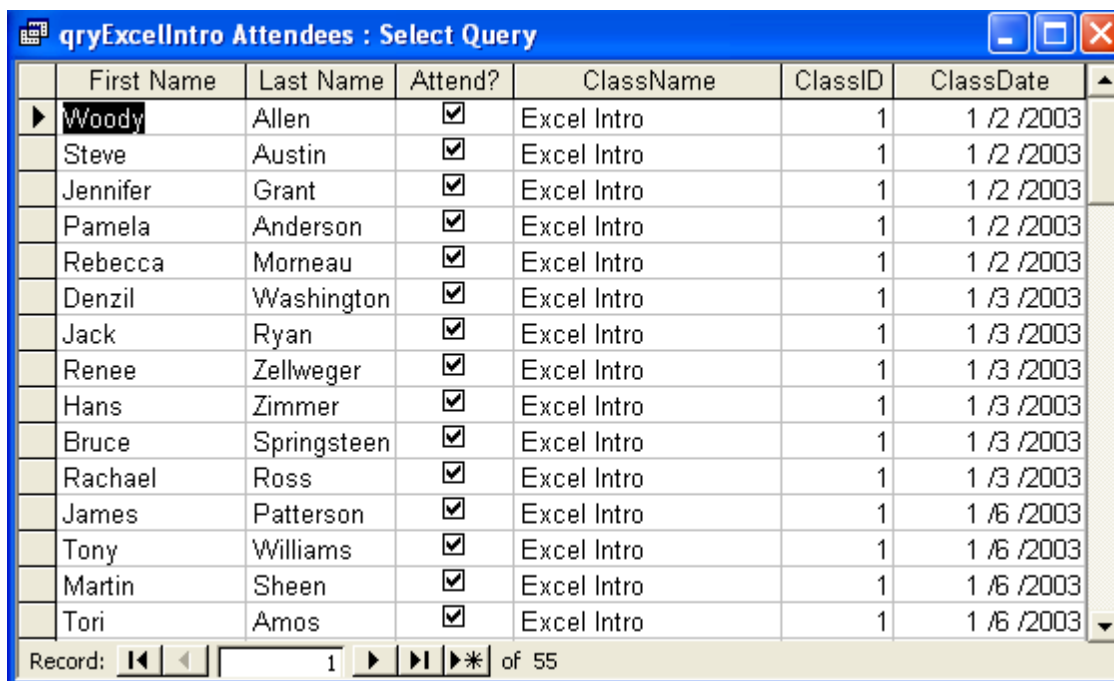
1. Open the query <qryExcel Intro Attendees>.

Below is an example of a query to display all the attendees for Intro to Excel.



## Running Queries

When you run the query it displays the "answer" to your query. It displays the corresponding records from the table that meet the criteria you specify.



|   | First Name | Last Name   | Attend?                             | ClassName   | ClassID | ClassDate  |
|---|------------|-------------|-------------------------------------|-------------|---------|------------|
| ▶ | Woody      | Allen       | <input checked="" type="checkbox"/> | Excel Intro | 1       | 1 /2 /2003 |
|   | Steve      | Austin      | <input checked="" type="checkbox"/> | Excel Intro | 1       | 1 /2 /2003 |
|   | Jennifer   | Grant       | <input checked="" type="checkbox"/> | Excel Intro | 1       | 1 /2 /2003 |
|   | Pamela     | Anderson    | <input checked="" type="checkbox"/> | Excel Intro | 1       | 1 /2 /2003 |
|   | Rebecca    | Morneau     | <input checked="" type="checkbox"/> | Excel Intro | 1       | 1 /2 /2003 |
|   | Denzil     | Washington  | <input checked="" type="checkbox"/> | Excel Intro | 1       | 1 /3 /2003 |
|   | Jack       | Ryan        | <input checked="" type="checkbox"/> | Excel Intro | 1       | 1 /3 /2003 |
|   | Renee      | Zellweger   | <input checked="" type="checkbox"/> | Excel Intro | 1       | 1 /3 /2003 |
|   | Hans       | Zimmer      | <input checked="" type="checkbox"/> | Excel Intro | 1       | 1 /3 /2003 |
|   | Bruce      | Springsteen | <input checked="" type="checkbox"/> | Excel Intro | 1       | 1 /3 /2003 |
|   | Rachael    | Ross        | <input checked="" type="checkbox"/> | Excel Intro | 1       | 1 /3 /2003 |
|   | James      | Patterson   | <input checked="" type="checkbox"/> | Excel Intro | 1       | 1 /6 /2003 |
|   | Tony       | Williams    | <input checked="" type="checkbox"/> | Excel Intro | 1       | 1 /6 /2003 |
|   | Martin     | Sheen       | <input checked="" type="checkbox"/> | Excel Intro | 1       | 1 /6 /2003 |
|   | Tori       | Amos        | <input checked="" type="checkbox"/> | Excel Intro | 1       | 1 /6 /2003 |

Record: 1 of 55

## Reports



A **Report** places the information you have gathered from a query or table and places it in neat, organized and readable form. You would use a report to present your database information.

Below is an example of a report built in access that displays a list of people that attended an Excel Introduction Class.

**Microsoft Access - [RptRosterAttendees]**

File Edit View Tools Window Help Type a question for help

75% Close Setup

**List of Attendees - Excel Introduction**

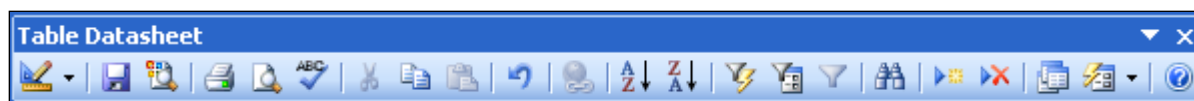
| Class Name  | Last Name | First Name | Class Date |
|-------------|-----------|------------|------------|
| Excel Intro |           |            |            |
| #1          | Allen     | Woody      | 1/2/2003   |
| #2          | Amos      | Tori       | 1/6/2003   |
| #3          | Anderson  | Gillian    | 1/21/2003  |
| #4          | Anderson  | Pamela     | 1/2/2003   |
| #5          | Anderson  | Gillian    | 1/6/2003   |
| #6          | Austin    | Steve      | 1/2/2003   |
| #7          | Austin    | Steve      | 1/7/2003   |
| #8          | Bender    | Matthew    | 1/7/2003   |
| #9          | Bender    | Matthew    | 2/4/2003   |
| #10         | Berry     | Halle      | 2/4/2003   |
| #11         | Blues     | Moody      | 2/4/2003   |
| #12         | Briggs    | Jason      | 2/10/2003  |
| #13         | Broderick | Matthew    | 2/10/2003  |
| #14         | Brokovich | Erik       | 2/10/2003  |
| #15         | Davis     | Kristin    | 2/25/2003  |
| #16         | Davis     | Kristin    | 2/21/2003  |
| #17         | Duchonny  | David      | 3/11/2003  |














Page: 1 NUM










Ready

## Working in Datasheet View


### Datasheet Toolbar

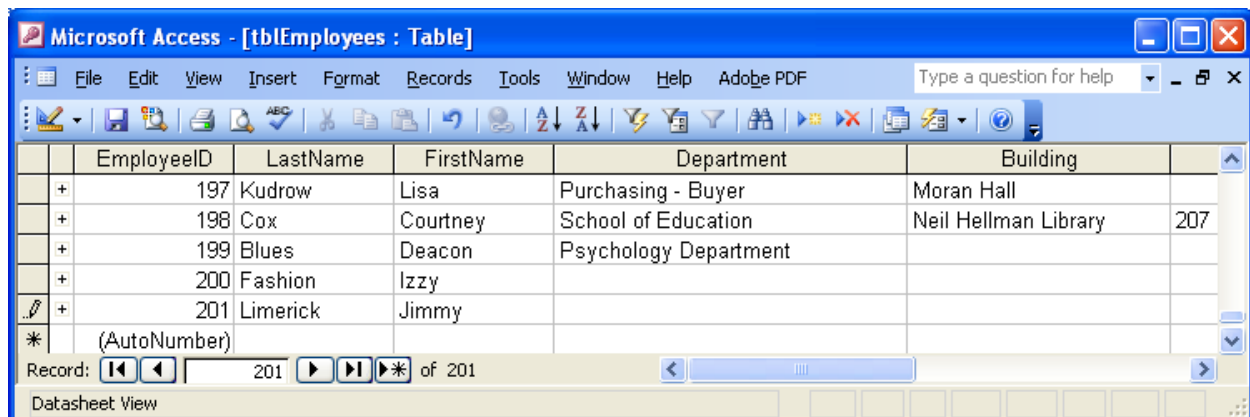


|   |  |
|---|--|
|    | <b>View button</b> - Displays the available views for the current window. Click the button if it displays the desired view, or click the arrow next to the button, and then click the desired view.                  |
|    | <b>Save</b> - Saves the layout of a datasheet; design of a table, query, view, stored procedure, SQL statement, form, report, or data access page; or structure and content of a macro.                              |
|    | <b>Search</b> - Allows you to search for files on a hard disk or a network drive.  |
|    | <b>Print</b> - Prints the selected form, report, datasheet, or data access page immediately without displaying the Print dialog box.   |
|    | <b>Print Preview</b> - Displays the active object, such as a table, form, or report, as it will appear when printed. You can zoom in or out of a page or see several pages at once.                                  |
|  | <b>Check Spelling</b> - Checks the spelling of text entries in table, query, or form Datasheet view or selected text in a text box in Form view.   |
|  | <b>Cut</b> - Removes the selected item (for example, a control or a record) and places it on the Clipboard so you can insert it elsewhere.   |
|  | <b>Copy</b> - Copies the selected item (for example, a record) and places it on the Clipboard so you can insert it elsewhere.  |
|  | <b>Paste</b> - Inserts an item from the Clipboard into an active database object.  |
|  | <b>Undo</b> - click to undo up to 20 of the most recent actions.   |
|  | <b>Hyperlink</b> - Inserts or modifies a hyperlink address or Uniform Resource Locator (URL). In datasheet and form view, the hyperlink address field must be selected.  |
|  | <b>Sort Ascending</b> - Sorts records based on the selected column(s) in ascending order (0 to 9, A to Z). Subsequent sorts replace previous sorts. To undo a sort, choose Remove Filter/Sort on the Records menu.   |
|  | <b>Sort Descending</b> - Sorts records based on the selected column(s) in descending order (9 to 0, Z to A). Subsequent sorts replace previous sorts. To undo a sort, choose Remove Filter/Sort on the Records menu. |

|   |   |
|---|---|
|    | <b>Filter by Selection</b> – Filters records based on selected data. To filter by selection, select a field or part of a field in a datasheet or form, and then click Filter By Selection.  |
|    | <b>Filter By Form</b> - Displays the Filter by Form window, which is a blank version of the active form or datasheet, so you can filter data.   |
|    | <b>Apply Filter/Remove Filter</b> - Applies or removes a filter in the active datasheet or form.  |
|    | <b>Find</b> - Searches for a string, such as an employee's last name in a datasheet or form.  |
|    | <b>New Record</b> - Moves to the end of the form or datasheet so you can enter a new record.  |
|    | <b>Delete record</b> - Deletes the selected record or records from the form or datasheet.   |
|    | <b>Database Window</b> - Displays the Database window, which lists all objects in the current Microsoft Access database. For example, to create, copy, or delete a database object, or to drag an object to a group or to the current window. |
|   | <b>New Object</b> - Automatically creates a form based on the selected table, query, view, or stored procedure.   |
|  | <b>Office Assistant</b> - The Office Assistant provides Help topics and tips to help you accomplish your tasks.   |

## Add Records

1. Click the New Record button  on the toolbar or on the navigation bar.
2. Enter your data.
3. Press your Tab key or the Enter key to move to the next field. You can also use your mouse.

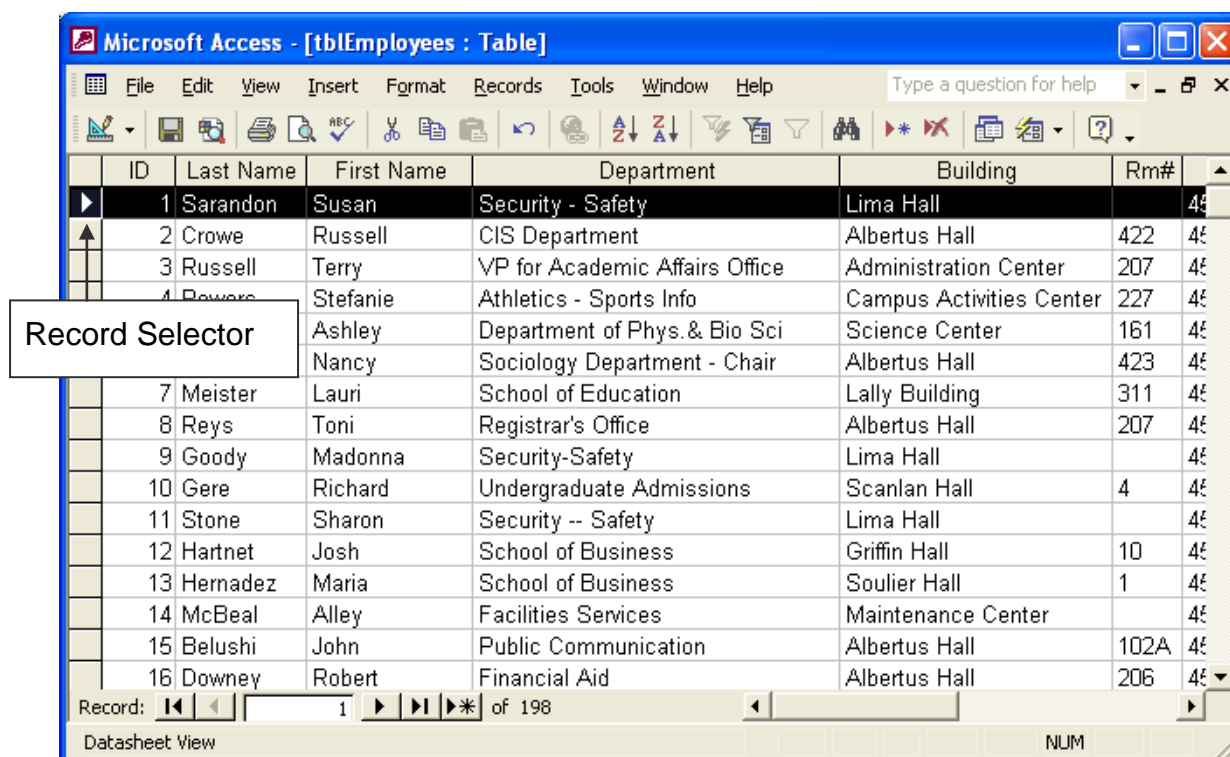


## Save a Record

A record is **automatically saved** by Access when you move to another record. Access automatically saves all records when you close the table.

## Select a Record


1. Selecting the record (row) makes it active. An active record will have the arrow in the record selector area. This means that the record is ready for an action to be taken, such as editing or deleting data.
2. To select one record, you can point to it in the record selector area with your mouse or you can choose **Edit, Select Record** from the Menu Bar.
3. To select all records, from the menu bar select **Edit, Select All Records**.



## Edit a Record

You can edit any record except for data in the AutoNumber field. The AutoNumber type produces a unique computer-generated number, which is never repeated in that field. To illustrate this point, if you delete the third record, which has an AutoNumber of 3, the third record will then be AutoNumber 4. Even if you add records, the number 3 will never appear again in that field.

## Delete a Record

1. Select the record you would like to delete.
2. Click the Delete Record  button on the toolbar **or** select Edit, Delete Record from the Menu Bar **or** press the Delete key on the keyboard.

- A dialog box will display a message to confirm the action and warn you that it cannot be undone. **Deletions cannot be undone.** Click **OK** to confirm deletion.

## Check Your Spelling


In Datasheet View you can check the spelling in your records.

- \* Click the Spelling button  on the toolbar

## Filter by Selection

Filter by Selection, filters the records via selected data. To filter by selection, select a field or part of a field in a datasheet or form, and then click **Filter By Selection**.

### Exercise

- Sort the list of employees by last name in ascending order then descending order.
- Filter by selection  all employees in the Department Safety & Security.


|   | EmployeeID   | LastName  | FirstName | Department        | Building           | Rm# | Phone    |
|---|--------------|-----------|-----------|-------------------|--------------------|-----|----------|
| + | 1            | Sarandon  | Susan     | Safety & Security | Lima Hall          |     | 454-5187 |
| + | 12           | Hartnet   | Josh      | Safety & Security | Griffin Hall       | 10  | 454-2065 |
| + | 14           | McBeal    | Alley     | Safety & Security | Maintenance Center |     | 454-2800 |
| + | 101          | Monroe    | Chris     | Safety & Security | Lima Hall          |     | 454-2800 |
| + | 105          | Lopez     | Jenifer   | Safety & Security | Lima Hall          | 105 | 454-5215 |
| + | 150          | Bono      | Cher      | Safety & Security | Moran Hall         | 14  | 458-5345 |
| + | 161          | Degeneres | Ellen     | Safety & Security | Lima Hall          | 411 | 454-5237 |
| + | 188          | Kennedy   | Ted       | Safety & Security | Lima Hall          | 106 | 337-4914 |
| * | (AutoNumber) |           |           |                   |                    |     |          |

- Remove filter 

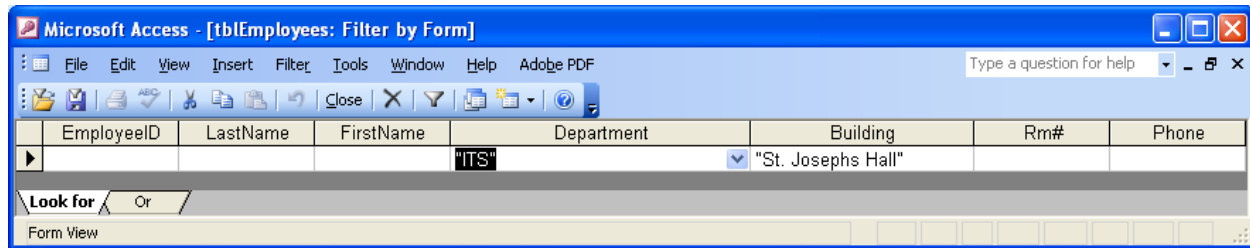
## Filter by Form

**Filter By Form** (Records/Filter menu) displays the Filter by Form window, which is a blank version of the active form or datasheet, so you can filter data.

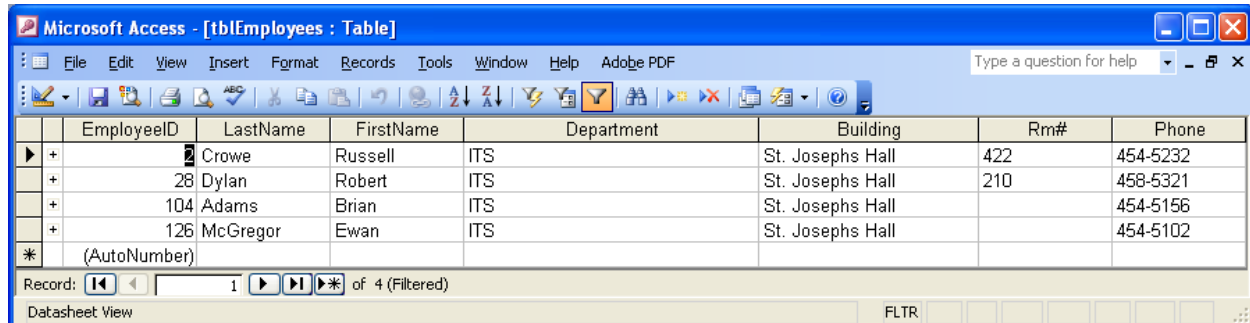
### Exercise

- Click filter by form  button.
- Click the drop-down arrow under the fields you want to filter, and select your criteria. In this example filter Department ITS and the building St. Josephs Hall.





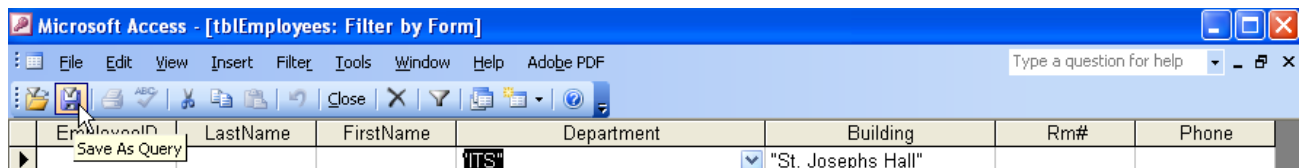
3. Click the Apply Filter  button. The results are shown below.



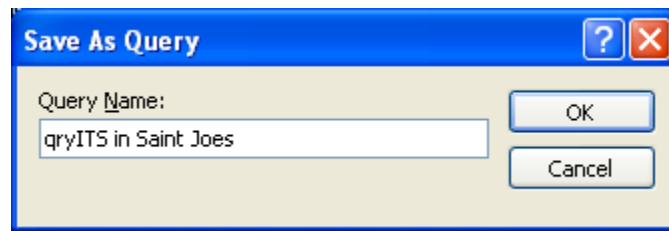
4. Once you set up and apply a filter you can save it as a Query.

5. Click the filter by form  button.

6. In filter by form view click the **Save as Query**  button.




7. In the Save As Query dialog box, type in a name. Click **OK**.

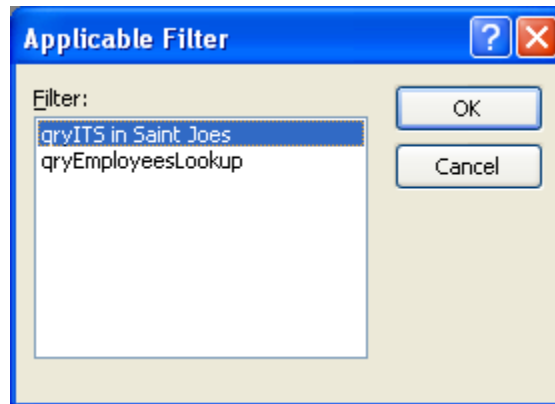



8. Click remove filter to get back into datasheet view.

## Load from Query

Once the query is saved, you can go back at a later date and open the query.

1. Go back into filter by form view. Click Load from query 
2. In the Applicable Filter dialog box select the filter you want to view.



3. Click OK.
4. Click Apply Filter .
5. Your data is displayed.

Note: Since you saved it as a query, you can also open the filter in the query objects tab.

## Table Design View

To open a table in design view, select the table in the database window and click Design. To switch between Design and Datasheet views, click the Table View button in the top left corner of the toolbar. The button will look different depending on which view you are in at the time.

### Table View buttons:



Switch to Datasheet view



Switch to Design view

Design view is recommended when you create a new database. This method allows you to specify the fields in the table and the properties associated with each field.

The top part of this view contains three areas:

**Field Name:** Name the field. Although you can use spaces in the field name, it is not recommended. You can use up to 64 characters, which can include letters, numbers and spaces. They cannot contain periods, exclamation points, a back quote, brackets or ASCII control characters. Names must be unique within a table.

**Data Type:** What type of data will be stored in the field? Will it be text, a date, or a number in which you will perform calculations on? Choose the appropriate data type from the drop down list for the type of data that will be entered into this field. The choice made in this field will affect the second (bottom) half of the form. Different data types allow different options.

**Description:** Use this area to define the field fully. This description will appear in the bottom comment bar of Access when the field is activated.

The screenshot shows the Microsoft Access window titled "Microsoft Access - [tblEmployees : Table]". The top menu bar includes File, Edit, View, Insert, Tools, Window, and Help. Below the menu is a toolbar with various icons. The main area is divided into two sections: a table design grid and a "Field Properties" pane.

| Field Name | Data Type  | Description                    |
|------------|------------|--------------------------------|
| ID         | AutoNumber | computer generated auto number |
| LastName   | Text       |                                |
| FirstName  | Text       |                                |
| Department | Text       |                                |
| Building   | Text       | Location                       |
| Rm#        | Text       |                                |
| Phone      | Text       | work phone number              |

Below the table design grid is the "Field Properties" pane. It has two tabs: "General" and "Lookup". The "General" tab is selected, showing the following properties:

- Field Size: Long Integer
- New Values: Increment
- Format:
- Caption:
- Indexed: Yes (No Duplicates)

On the right side of the "Field Properties" pane, there is a text box with the following text:

The field description is optional. It helps you describe the field and is also displayed in the status bar when you select this field on a form. Press F1 for help on descriptions.

At the bottom of the window, the status bar displays "Design view. F6 = Switch panes. F1 = Help." and "NUM".

After a description is entered, you can press [ENTER] or [Tab] to move to the next field name.

The order of the field names in design view will determine the order they will appear as column headings in your table. The order does not matter to the database, but for your own comfort you may wish to place them in a logical order.

- ✳ To change the order you can click and drag the field to a new location.

### **Data Types**

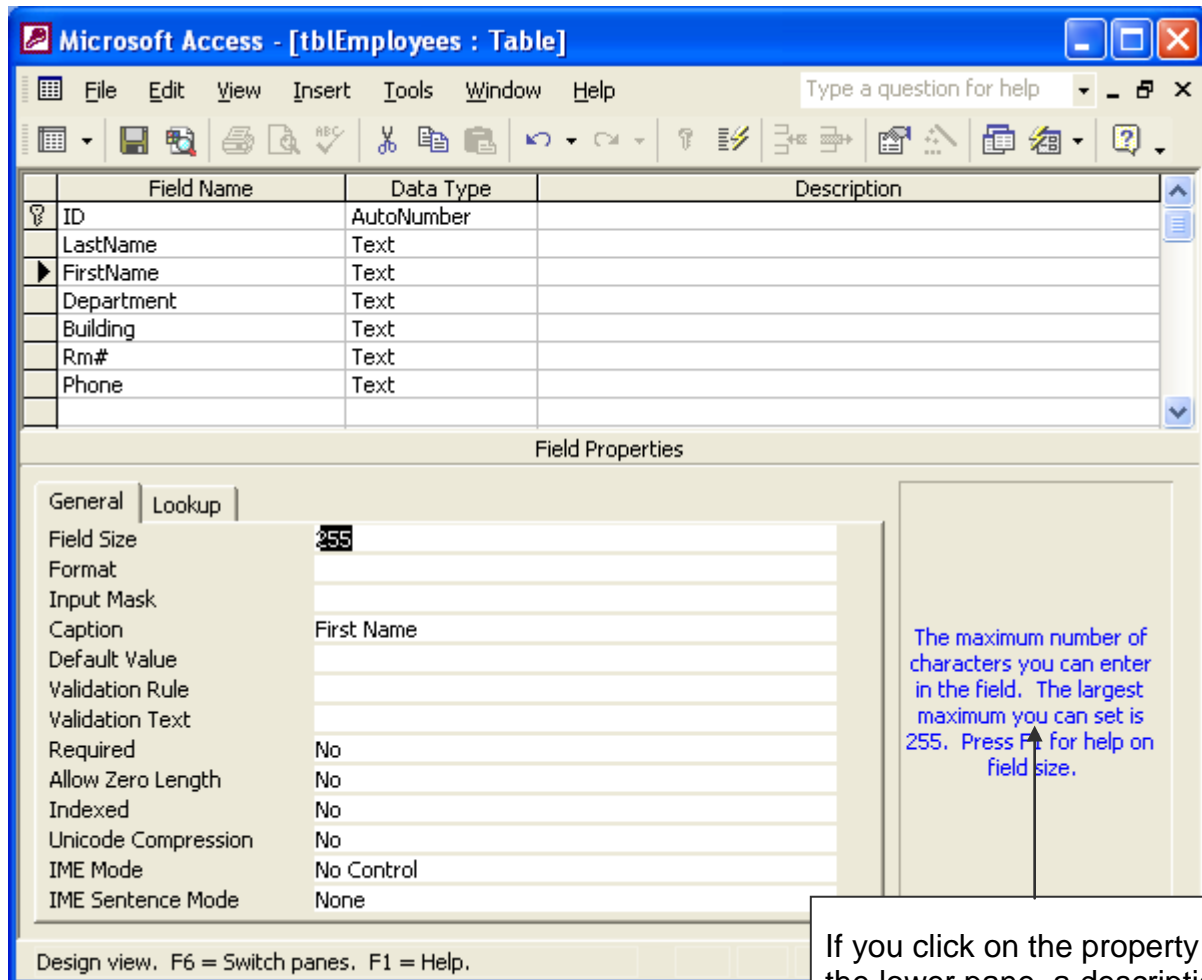
The following are a list of the **data types** available:

- ✳ **Text:** A string of up to 255 characters; however, you can limit the length to a smaller number by specifying the maximum number of characters in the “field size” property on the bottom half of the form. This data type should also be used for numbers that will not be used in calculations.
- ✳ **Memo:** Used to store longer strings of characters. Access places a limit of 64,000 characters for data of this type.
- ✳ **Number:** Used to store numeric data. Access offers a selection of formats for the date and time. This data type is best for numeric data that will be used in calculations.
- ✳ **Date/Time:** Used to store dates and time. Access will let you store a specific format for displaying the date and time. General Date is the default but other formats are available using the drop-down list or creating a custom format.
- ✳ **Currency:** Used for monetary data. This form does not round numbers.
- ✳ **AutoNumber:** Places a computer-generated number in the field. This is a read-only field assigned by Access. This data type is commonly used as a key field (to be discussed) as it is unique to each row in the table.
- ✳ **Yes/No:** Used for Boolean data, when an either/or situation is present.
- ✳ **OLE Object:** Used for “Object Linking and Embedding” from the OLE server provided in the Windows environment. An Excel spreadsheet or Word document can be an OLE object.
- ✳ **Hyperlink:** Insert a hyperlink address.
- ✳ **Lookup Wizard:** Allows you to define a field where the database user can select a value from another table or from a predefined list.

## Field Properties

The *field properties* are located at the bottom of the Design View form. Depending upon your choice of data type, different properties will appear.

You can set field properties to define the characteristics of information in your database.



If you click on the property in the lower pane, a description at the right will appear

### Field Size

The Field Size property determines the maximum number of characters or the range of values that can be stored in a field. The size is important, as the database will set aside that much space regardless of whether the space is used or not. This property can be set for Text, Number and AutoNumber fields.

For **Text Fields**, the field size determines the number of characters you can store in a field. The default size for a text field is 50 characters.

### Field Size Continued

For **Number Fields**, the field size determines the range of values you can store in a field and whether the field can contain fractional values.

Number fields allow the following field sizes:

|                |                             |   |
|----------------|-----------------------------|---|
| Byte           | 0 TO 255                    | No fractions, no decimals, no negative numbers, data is rounded |
| Integer        | -/+32786                    | no fractions, data is rounded                                   |
| Long Integer   | -/+2147483648               | no fractions, data is rounded                                   |
| Single         | -/+ 3.4x10 <sup>38</sup>    | up to seven decimal places                                      |
| Double         | -/+ 1.797x10 <sup>308</sup> | up to fifteen decimal places, the largest field size            |
| Replication ID | Globally unique identifier  | no decimal places available                                     |

### Format

The Format property controls how data is *displayed* in tables and reports. There are predefined settings or you can design your own. This field does not affect how the data is stored. There are the following choices:

|            |  |
|------------|--|
| General    | Displays a number the way it was entered.  |
| Currency   | Uses a comma and the currency symbol. Negative numbers are displayed in parenthesis and the default number of decimal places is two. |
| Fixed      | Displays at least one digit. Rounds to the set number of decimal places.   |
| Standard   | Uses a comma as a thousands separator and defaults to two decimal places.  |
| Percent    | Multiplies the number by 100 and uses a percent ( % ) symbol. Defaults to two decimal places.  |
| Scientific | Uses exponents of 10 for standard scientific notation.   |


### Decimal Places

The Decimal Places property displays for field with numeric data types. This field defines the number of decimal places to be stored in that field. If you choose AUTO, the Format property will control this option.

**Caption**

This property defines the **label** that will be placed on the datasheet view of the table, queries, reports or forms when that field is used. If this field is left blank, the field's name will be used.

**Default Value**

A default value automatically appears in the field when a new record is entered. This value may be an expression (a calculation), which can be created activating the Expression Builder. The Expression builder allows you to choose fields and create formulas. To activate the expression builder, just click in the default value property in the lower pane then click the expression builder . For example, if you type **Now()** in this area, the current date and time will appear in that field by default.

**Validation Rule**

Allows you to specify what data can be entered in the field. The Validation Rule is an expression that tests the data that is entered into the field. For example: >1000. The value entered must be greater than 1000.

**Validation Text**

This text appears in a message box when an invalid value is entered into the field.

**Require User Input**

Set this property to say yes if it is required that data is entered into this field. This way the user will not be able to leave this field blank. It is advisable that you do not set this property until you are finished testing your database.

**Indexed**









By creating an index on a field, it speeds up sorting and searching on that field. However, you only want to create Indexes on fields that will be sorted by or searched often because it can slow down data entry.

**Types:** Yes (Duplicates OK) - will allow duplicate values in the field.

Yes (No Duplicates) - cannot have duplicate values in the field.

## Design View Toolbar



|   |   |
|---|---|
|    | View the datasheet  |
|    | Save the table design   |
|    | Search  |
|    | Set Primary Key   |
|    | Define Indexes  |
|    | Insert Row(s) - Creates a blank row above the selected row  |
|    | Delete Row(s) – Deletes selected row(s)   |
|  | Properties - Displays the property sheet for the selected item, such as a table field or form control. Displays the active object's property sheet if nothing is selected   |
|  | Build button - Displays a builder for the selected item or property. Microsoft Access enables this button only if a builder is available for a selected item (such as a field name in table Design view) or a property (such as an input mask property or property set with an expression). |
|  | Displays the main database window   |
|  | Create new database object  |
|  | Office Assistant  |



## Primary Key




Every table should have a primary key. A primary key is a field that uniquely identifies each record in a table. Data that is entered into primary key fields are automatically indexed. Data is retrieved faster from indexed fields, which means that queries and reports run faster and we can find records in our database quickly.

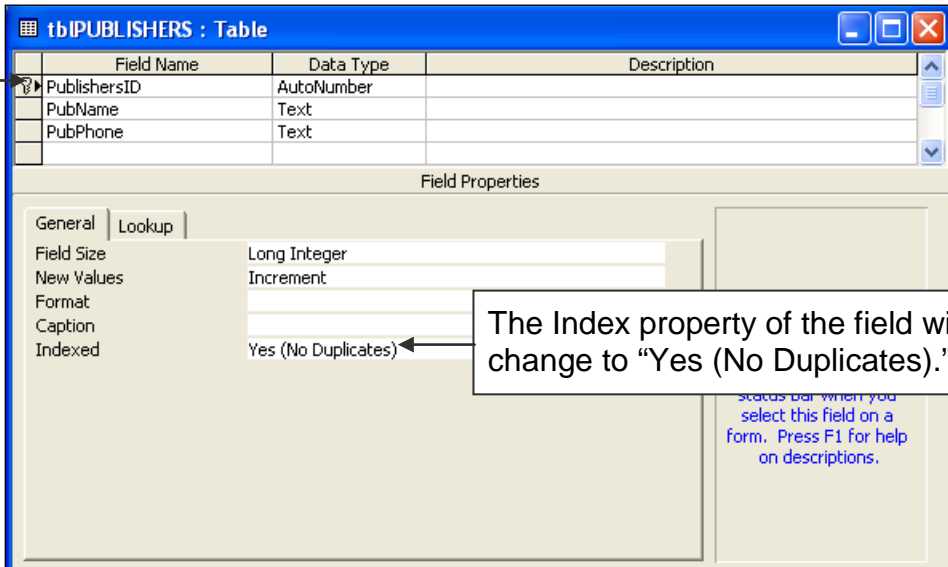
An example of a primary key would be a social security number, phone number or license plate number. An AutoNumber (which is computer generated) is often used as a primary key to uniquely identify a row in that table. The use of the word “ID” for identifier is usually used at the end of the key field name, like PublisherID.

If you do not set a primary key before closing the table, Access will display a message and will automatically select a field for you. (Note: If a field has “ID” in its name, that field will be selected.)

### To set a primary key:

1. Select the field you want to designate as the primary key.
2. Click the primary key button  on the toolbar.

A key symbol will appear by that field.



| Field Name   | Data Type  | Description |
|--------------|------------|-------------|
| PublishersID | AutoNumber |             |
| PubName      | Text       |             |
| PubPhone     | Text       |             |

Field Properties

General | Lookup

Field Size: Long Integer  
New Values: Increment  
Format:  
Caption:  
Indexed: Yes (No Duplicates)

The Index property of the field will change to “Yes (No Duplicates).”

status bar when you select this field on a form. Press F1 for help on descriptions.

## Foreign Key

A *foreign key* is a field that corresponds to a *primary key* in another table. The identification of a foreign key allows you to link two tables together forming a relationship that allows you to display information from both tables using queries, forms or reports. The data type for the primary key should match the data type of the foreign key in the other table. The exception to this is when using an AutoNumber. If the foreign key (the corresponding field) must have a data type of a Long Integer.

## Create a Database

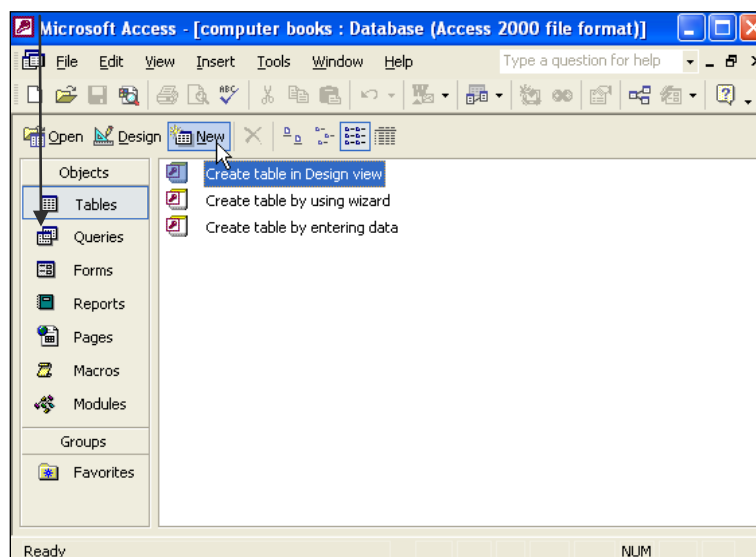
We have examined the two views (design view and datasheet view) that exist in every table. Now it is time to create our own database and tables. **Exercise** – Create a new blank database. See handout or follow your instructor's example.

**Example** - Create a new blank database named **<books>** that stores information on books, authors and publishers. How can we store the flat table information into a relational database? We will start by creating the Author's Table.

|   | ISBN          | Title        | AuID | AuName      | AuPhone      | PubID | PubName     | PubPhone     | Price   |
|---|---------------|--------------|------|-------------|--------------|-------|-------------|--------------|---------|
|   | 0-12-345678-9 | Jane Eyre    | 1    | Bronte      | 123-444-4566 | 3     | Small House | 714-000-0000 | \$49.00 |
|   | 0-99-999999-9 | Emma         | 1    | Austen      | 111-111-1111 | 1     | Big House   | 123-456-7890 | \$20.00 |
|   | 0-55-123456-9 | Main Street  | 10   | Jones       | 123-333-3333 | 3     | Small House | 714-000-0000 | \$22.95 |
|   | 0-321-32132-1 | Balloon      | 11   | Snoopy      | 321-321-2222 | 3     | Small House | 714-000-0000 | \$34.00 |
|   | 0-321-32132-1 | Balloon      | 12   | Grumpy      | 321-321-0000 | 3     | Small House | 714-000-0000 | \$34.00 |
|   | 0-321-32132-1 | Balloon      | 13   | Sleepy      | 321-321-1111 | 3     | Small House | 714-000-0000 | \$34.00 |
|   | 0-11-345678-9 | Moby Dick    | 2    | Melville    | 222-222-2222 | 3     | Small House | 714-000-0000 | \$49.00 |
|   | 0-103-45678-9 | Iliad        | 3    | Homer       | 333-333-3333 | 1     | Big House   | 123-456-7890 | \$25.00 |
|   | 1-22-233700-0 | Visual Basic | 4    | Roman       | 444-444-4444 | 1     | Big House   | 123-456-7890 | \$25.00 |
|   | 1-1111-1111-1 | C++          | 4    | Roman       | 444-444-4444 | 1     | Big House   | 123-456-7890 | \$29.95 |
|   | 0-91-045678-5 | Hamlet       | 5    | Shakespeare | 555-555-5555 | 2     | Alpha Press | 999-999-9999 | \$20.00 |
|   | 0-555-55555-9 | Macbeth      | 5    | Shakespeare | 555-555-5555 | 2     | Alpha Press | 999-999-9999 | \$12.00 |
|   | 0-99-777777-7 | King Lear    | 5    | Shakespeare | 555-555-5555 | 2     | Alpha Press | 999-999-9999 | \$49.00 |
|   | 0-123-45678-0 | Ulysses      | 6    | Joyce       | 666-666-6666 | 2     | Alpha Press | 999-999-9999 | \$34.00 |
|   | 0-91-335678-7 | Faire Queen  | 7    | Spencer     | 777-777-7777 | 1     | Big House   | 123-456-7890 | \$15.00 |
|   | 0-12-333433-3 | On Liberty   | 8    | Mill        | 888-888-8888 | 1     | Big House   | 123-456-7890 | \$25.00 |
|   | 0-55-123456-9 | Main Street  | 9    | Smith       | 123-222-2222 | 3     | Small House | 714-000-0000 | \$22.95 |
| * |               |              |      |             |              |       |             |              |         |

### Create the Authors Table

1. Choose the **Tables** tab in the database window. Click **New**.



2. When you choose **New** you have five choices summarized below.

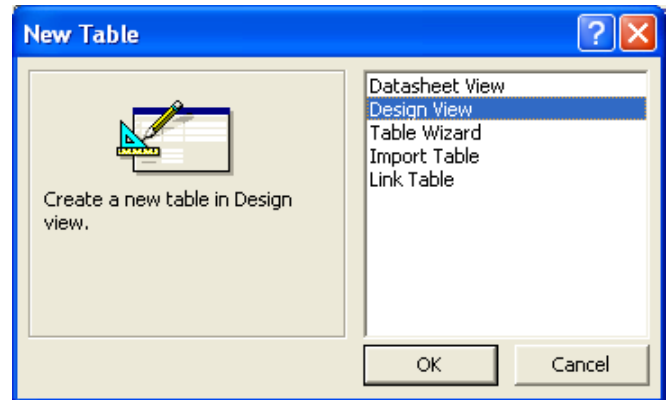
**Datasheet View:** Creates a table based on data you enter into the table.

**Design View:** Creates a table where fields and field properties are identified first (Recommended).

**Table Wizard:** Guides you through built-in templates to create a table.


**Import Table:** Runs a wizard that imports a table to your database from an external source such as Excel, another database, or a delimited text file.

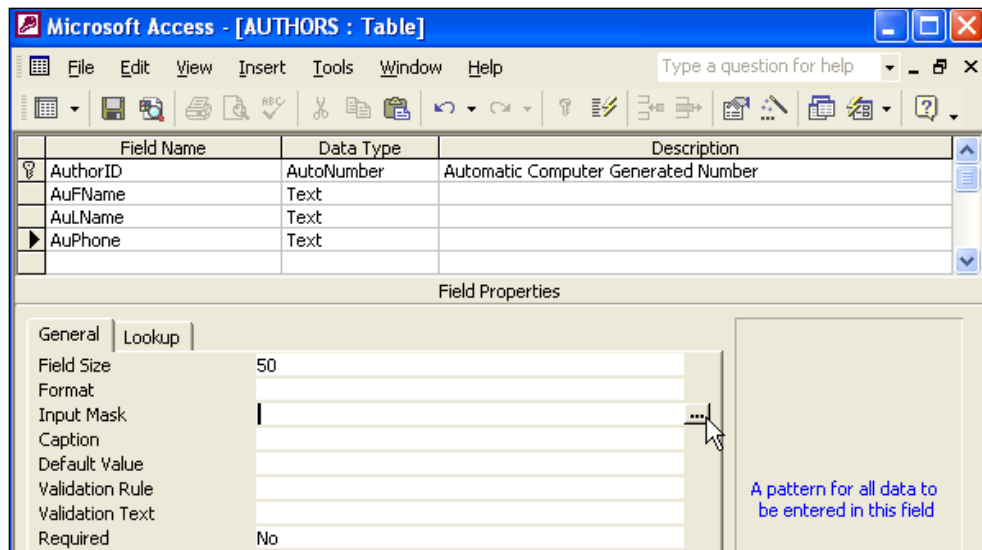
**Link Table:** Runs a wizard that adds a link to a table in another database.



3. Choose **Design View**.
4. Create a table with the field names and properties shown below.
5. Designate the primary key to AuthorID.
  - \* In the top pane, select the AuthorID field. Click the primary key button on the toolbar.

| Field Name | Data Type  |
|------------|------------|
| AuthorID   | AutoNumber |
| AuFName    | Text       |
| AuLName    | Text       |
| AuPhone    | Text       |

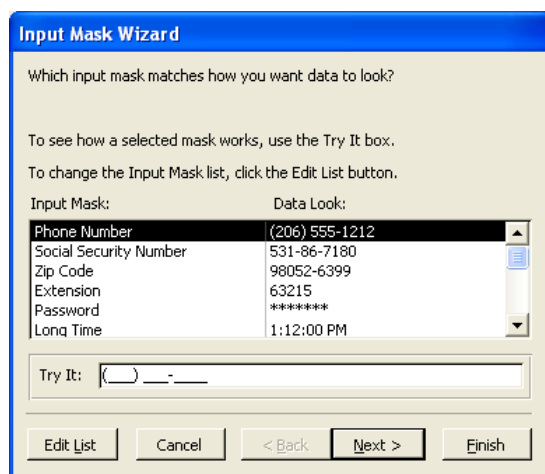
6. Create an Input mask for the author's phone. An input mask provides a pattern to make data entry easier for the user.
7. In the top pane, select the AuPhone field.
8. In the lower pane click the build  button for the Input Mask.



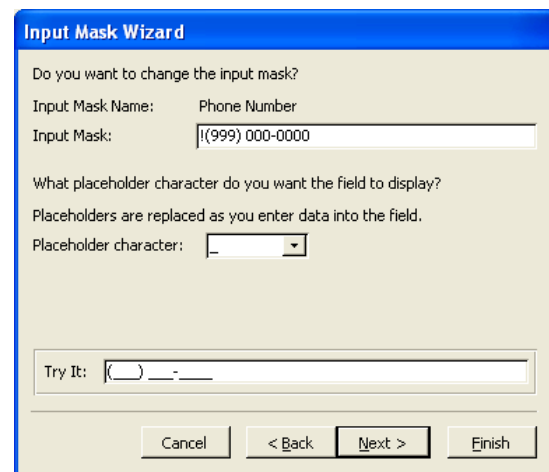
### Input Mask Wizard

**Step 1:** Select a mask that matches how you want your data to appear.

✳ Select Phone Number. Click **Next**.



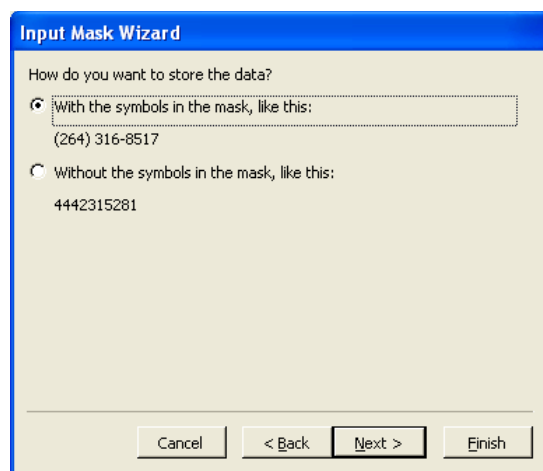
**Step 2** Select the placeholder character you want. Click **Next**.



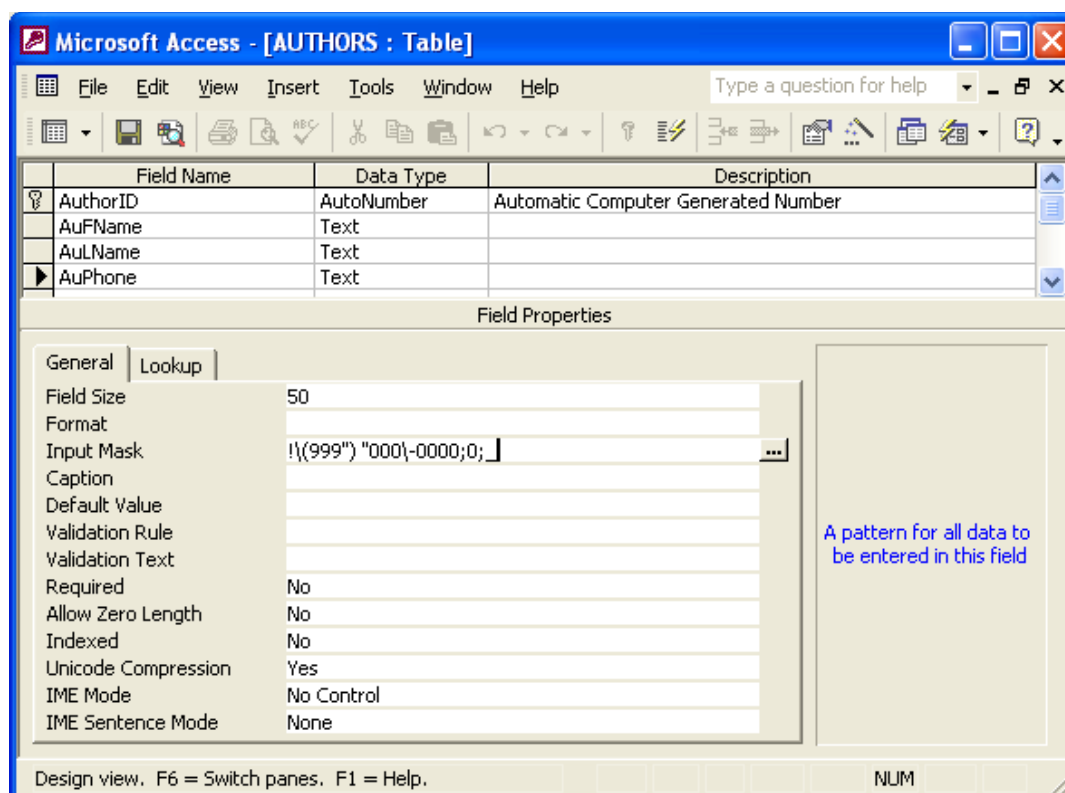
**Step 3** Choose to store the data with the symbols or without the symbols.

✳ Select **“with the symbols in the mask”**.

**Step 4:** Click **Finish**.



In the Input Mask field properties, Access inserts coding for an input mask that will produce the pattern you selected.



Back in datasheet view when data entry is performed, the input mask will appear when you begin to type a phone number into the AuPhone field, as shown below.

|   | AuthorID     | AuFName | AuLName | AuPhone |
|---|--------------|---------|---------|---------|
| ✎ | 15           | Tim     | Dorado  | ( ) -   |
| * | (AutoNumber) |         |         |         |

## Saving a Table

1. Click on the Save  button or choose **File → Save**.

2. Give the table a meaningful name. All Access tables should begin with the three letter prefix "tbl" and then a descriptive name.

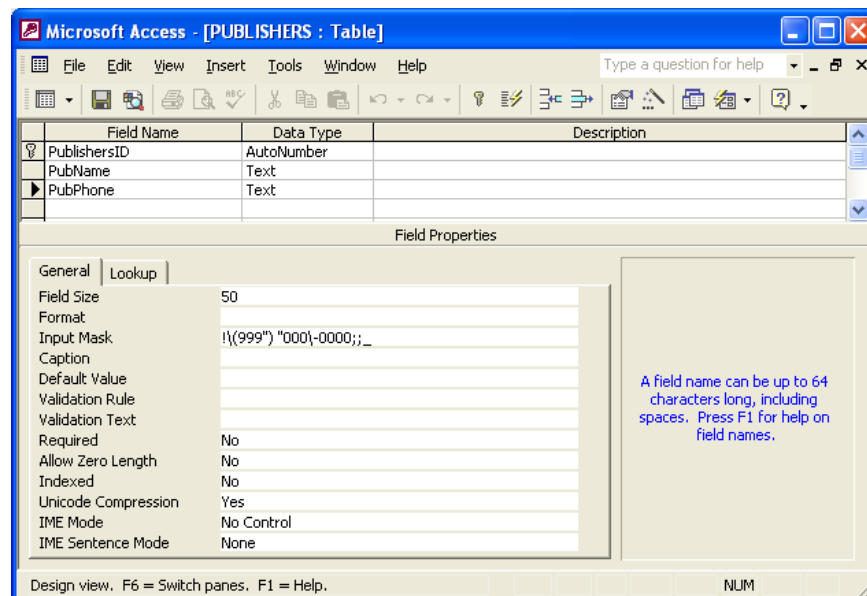


### Create the Publishers Table

1. Create the Publishers table with the field names and properties shown below.
2. The two attributes publisher name and phone number together can uniquely identify the publisher. Nevertheless, for convenience, designate the primary key to PublisherID.

| Field Name  | Data Type  |
|-------------|------------|
| PublisherID | AutoNumber |
| PubName     | Text       |
| PubPhone    | Text       |

3. Create an Input mask for the field named PubPhone.



### Create the Books Table

1. Create the Books table with the field names and properties shown below.
2. Designate the primary key to ISBN.

This is a foreign key. It creates a link back to the primary key in the Publishers table which is an Auto Number so this needs to be Number data type

| Field Name | Data Type |
|------------|-----------|
| ISBN       | Text      |

→

|       |          |
|-------|----------|
| Title | Text     |
| PubID | Number   |
| Price | Currency |

3. For the Price field designate the data type as Currency. Change the properties for that field to currency and 2 decimal points.

The screenshot shows the Microsoft Access interface. At the top, a window titled 'tbIBOOKS : Table' displays a table design view with the following fields:

| Field Name | Data Type | Description |
|------------|-----------|-------------|
| ISBN       | Text      |             |
| Title      | Text      |             |
| PubID      | Number    |             |
| Price      | Currency  |             |

Below the table design view, the 'Field Properties' task pane is open, showing the 'General' tab for the 'Price' field. The properties are as follows:

| Property        | Value    |
|-----------------|----------|
| Format          | Currency |
| Decimal Places  | 2        |
| Input Mask      |          |
| Caption         |          |
| Default Value   |          |
| Validation Rule |          |
| Validation Text |          |
| Required        | No       |
| Indexed         | No       |

On the right side of the task pane, a blue text box contains the following message:

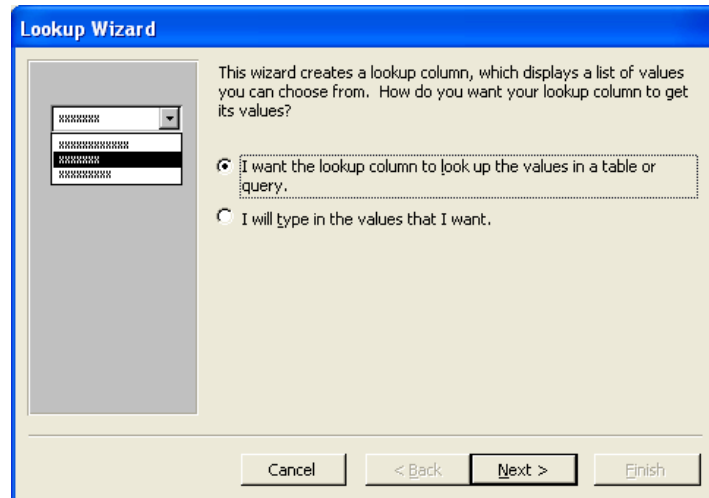
The data type determines the kind of values that users can store in the field. Press F1 for help on data types.

4. For the PubID field, create a lookup to the Publisher's table.
5. Follow the directions on the next page for using the Lookup Wizard to create a lookup field.

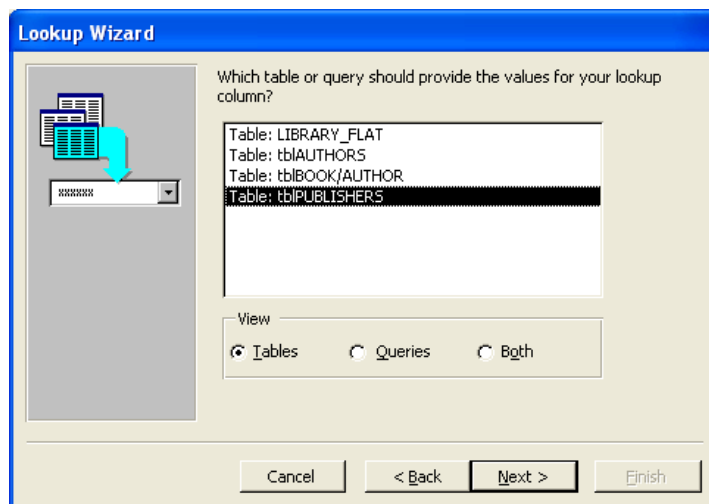
## Creating a Lookup Wizard Field

A Lookup Wizard field let the user select a value from a list of possible values. For the Publisher ID field, the user will be able to select the correct ID from the Publisher's table, rather than having to remember the correct ID or refer back to the table. This makes entering the ID's easier for the user and guarantees that the ID is valid.

1. For the PubID field, choose the data type **Lookup Wizard** from the drop-down list.
2. Choose **I want the lookup column to look up the values in a table or query.** Click **Next**.

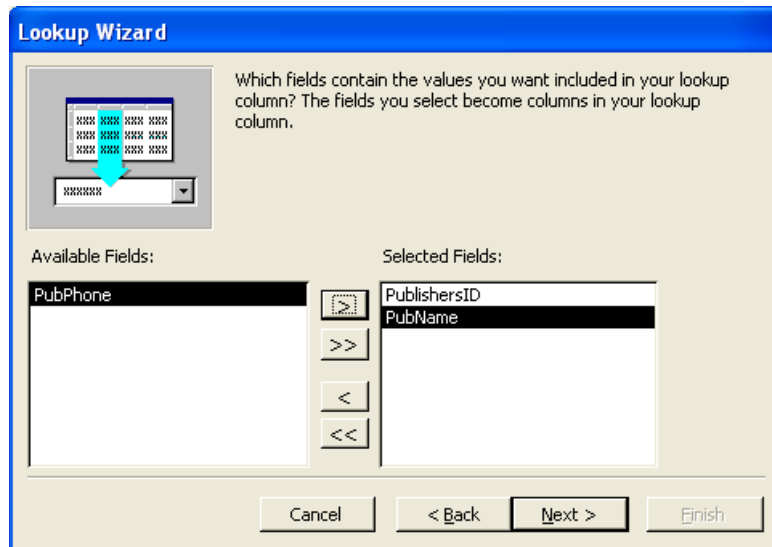


3. Select the table (Publishers table) that provides the values for your Lookup column.

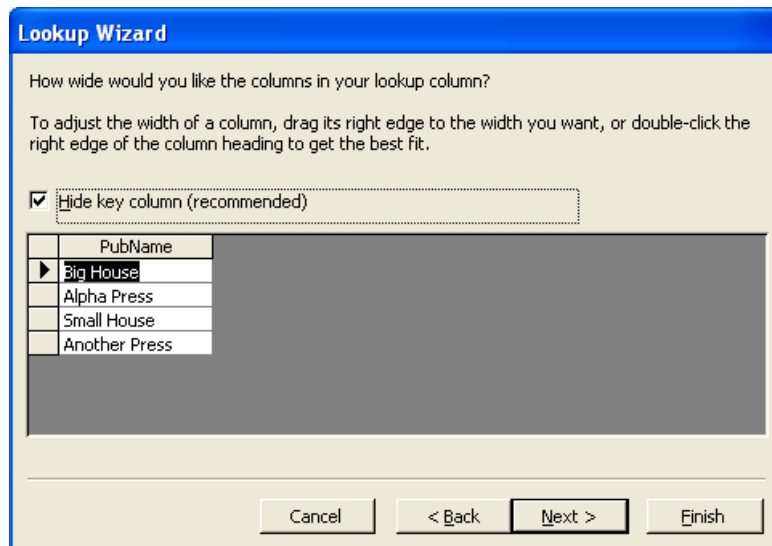




4. Select the fields that contain the values you want included in your lookup column (PublishersID, PubName). For example, you may want the Publishers ID because that is the primary key for this table and the publisher's name. You don't need the phone number.



5. Adjust the width of your lookup column. Choose whether you want to hide the key column.



6. Type a label name (or accept the default name given) for the lookup column and click **Finish**.

## Multiple-value problems

How do we specify the author of a book? What if we have more than one author per book? Authors can write more than one book and a book can have more than one author. What a quandary? How do we represent this in our database?

If we add the AuthorID to the books table as shown in below, each **book that is written by two or more authors** must be represented by two or more rows of repeated data. Look at the example titled Balloon below.

|   | ISBN          | Title        | PubID | Price   | AuthorID |
|---|---------------|--------------|-------|---------|----------|
| + | 0-103-45678-9 | Iliad        | 1     | \$25.00 |          |
| + | 0-11-345678-9 | Moby Dick    | 3     | \$49.00 |          |
| + | 0-12-333433-3 | Oh Liberty   | 1     | \$25.00 |          |
| + | 0-123-45678-0 | Ulysses      | 2     | \$34.00 |          |
| + | 0-12-345678-9 | Jane Eyre    | 3     | \$49.00 |          |
| + | 0-321-32132-1 | Balloon      | 3     | \$34.00 |          |
| + | 0-55-123456-9 | Main Street  | 3     | \$22.95 |          |
| + | 0-555-55555-9 | Macbeth      | 2     | \$12.00 |          |
| + | 0-91-045678-5 | Hamlet       | 2     | \$20.00 |          |
| + | 0-91-335678-7 | Faire Queen  | 1     | \$15.00 |          |
| + | 0-99-777777-7 | King Lear    | 2     | \$49.00 |          |
| + | 0-99-999999-9 | Emma         | 1     | \$20.00 |          |
| + | 1-1111-1111-1 | C++          | 1     | \$29.95 |          |
| + | 1-22-233700-0 | Visual Basic | 1     | \$25.00 |          |
| + | 0-321-32132-1 | Balloon      | 3     | \$34.00 |          |

If we add the ISBN key to the authors table, what do we do when an **author writes more than one book?**

|  |   | AuID | AuName | AuPhone      | ISBN          |
|--|---|------|--------|--------------|---------------|
|  | + | 11   | Snoopy | 321-321-2222 | 0-321-32132-1 |
|  | + | 12   | Grumpy | 321-321-0000 | 0-321-32132-1 |
|  | + | 13   | Sleepy | 321-321-1111 | 0-321-32132-1 |

### **Create the Books /Author Table**

This scenario represents a **many-to-many relationship**. Authors write many books, books are written by many authors.

At first we may think we can just add more than one author's name or author ID into the field as shown below, but this would not be correct.

To include all the author's names or ID in one cell it becomes more difficult to search the database for a single author or sort sort the authors in the table? Remember the rule.

- \* The intersection of every column and record should contain one and only one value.

|   | ISBN          | AuID       |
|---|---------------|------------|
|   | 0-103-45678-9 | 3          |
|   | 0-11-345678-9 | 4          |
|   | 0-12-333433-3 | 8          |
|   | 0-123-45678-0 | 6          |
|   | 0-12-345678-9 | 14         |
| ▶ | 0-321-32132-1 | 11, 12, 13 |
|   | 0-55-123456-9 | 9          |
|   | 0-55-123456-9 | 10         |
|   | 0-555-55555-9 | 5          |
|   | 0-91-045678-5 | 5          |
|   | 0-91-335678-7 | 7          |
|   | 0-99-777777-7 | 5          |
|   | 0-99-999999-9 | 1          |
|   | 1-1111-1111-1 | 4          |
|   | 1-22-233700-0 | 4          |

Second we might think of adding more fields, or a multiple columns to accommodate each author.

- \* The multiple-column approach presents the problem of guessing how many author columns we will ever need, and creates a lot of wasted space (empty fields) for books with only one author.

**incorrect : Table**

|   | ISBN          | Author1 | Author2 | Author3 |
|---|---------------|---------|---------|---------|
| ▶ | 0-103-45678-9 | 3       |         |         |
|   | 0-11-345678-9 | 4       |         |         |
|   | 0-12-333433-3 | 8       |         |         |
|   | 0-123-45678-0 | 6       |         |         |
|   | 0-12-345678-9 | 14      |         |         |
|   | 0-321-32132-1 | 11      | 12      | 13      |
|   | 0-55-123456-9 | 9       |         |         |

## Intermediary Tables

The proper approach to implementing a **many to many** relationship is to create another table called a **junction** or **intermediary table**, in order to break the relationship into two one-to-many relationships.

In this case you would need to create another table to store multiple instances of the (ISBN), once for each author. Below is an example of a Book and Author table where there is more than one author per book or ISBN.

1. Create the Book/Author table with the field names and properties shown below.

| Field Name | Data Type |
|------------|-----------|
| ISBN       | Text      |
| AuID       | Number    |

2. For ISBN number 0-321-321321 there are 3 authors. The table below shows how you would enter all 3 authors for that particular ISBN.

**tbIBOOK/AUTHOR : Table**

|   | ISBN          | AuID |
|---|---------------|------|
|   | 0-103-45678-9 | 3    |
|   | 0-11-345678-9 | 2    |
|   | 0-12-333433-3 | 8    |
|   | 0-123-45678-0 | 6    |
|   | 0-12-345678-9 | 14   |
| ▶ | 0-321-32132-1 | 11   |
|   | 0-321-32132-1 | 12   |
|   | 0-321-32132-1 | 13   |
|   | 0-55-123456-9 | 9    |

Record: 6 of 1

## **Types of Relationships**

### **One-to-One Relationship**

This type of relationship is very rare. If you find you have a one-to-one relationship, you should reevaluate the database design. The tables can most likely be combined into one table.

### **One-to-Many relationship**

A record in Table A has one or more related records in Table B.

- For example a customer places many orders.
- An publisher publishes many books

### **Many-to-Many relationship**


A record in Table A has one or more matching records in Table B, and a record in Table B can be related to more than one record in Table A.

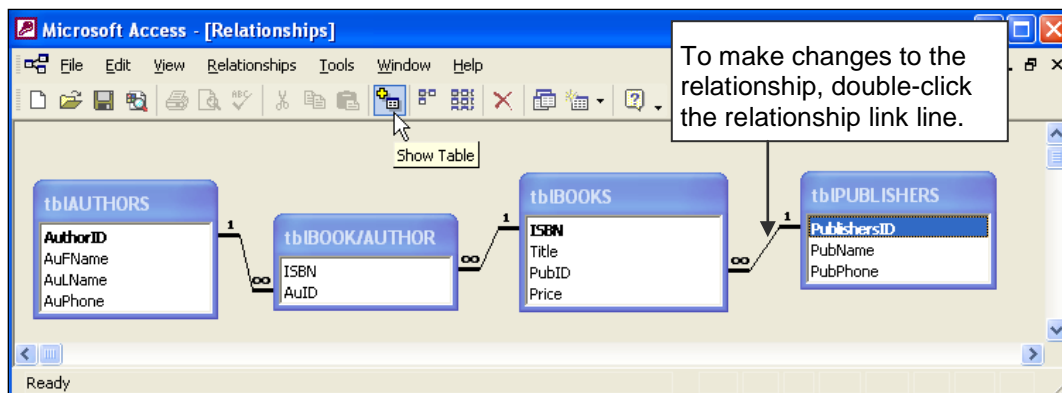
Example: A many-to-many relationship is possible between an author's table and a books table. Each author could have several books and each book could have several authors.


A many-to-many relationship is often a symptom that the two tables are not directly related, but instead are related through some additional table.

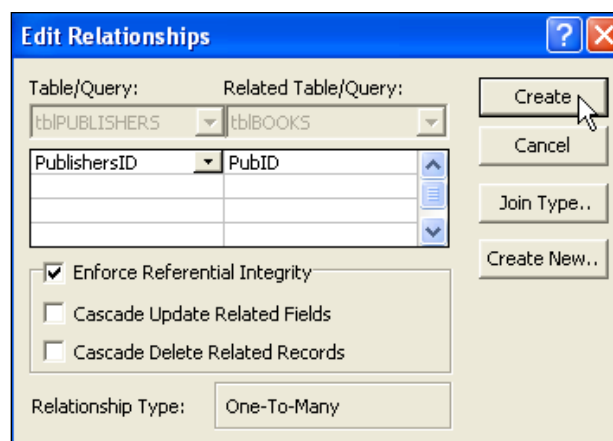
## Create Relationships

Creating relationships allows you to combine data together from multiple tables.

1. To set up the relationship, open the relationship window by going to **Tools** → **Relationships** or click on the relationships  button on the toolbar.
2. Below is an example of the relationships window where relationships have already been created.



3. Click the Show Table  button to display the tables or on the menu bar select **Relationships** → **Show table**.
4. Double-click the tables or queries you want to add to the **Relationships** layout. Close the show table dialog box.
5. Create the relationship by selecting the primary key field in one table and dragging it to the matching foreign key field in another table. The primary key will be bolded in the table.
6. The **Edit Relationships** box will appear. Click the **Create** button to create the relationship.

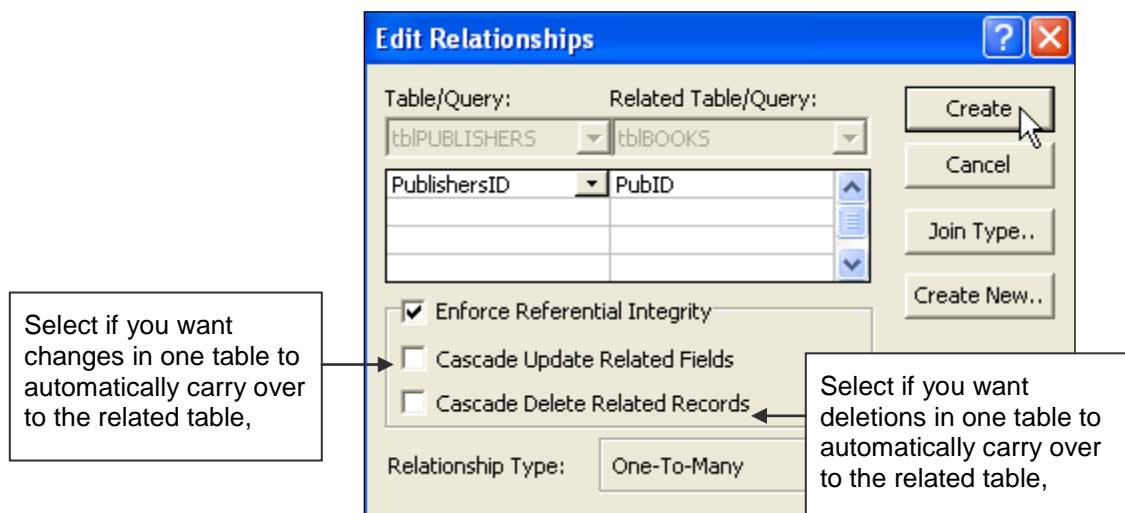


## Referential Integrity

In the edit relationships dialog box, you can choose **Enforce Referential Integrity**. Referential Integrity ensures that the records in related tables are consistent with one another. When you choose Enforce Referential Integrity, certain rules are implemented.

- When you add a record to a related table, a matching record must already exist in the primary table, thereby preventing the possibility of orphaned records.
- If you attempt to change the value of the primary key in the primary table, Access prevents this change if matching records existing in a related table.
  - However, if you choose the **cascade updates option**, access permits the change in value to the primary key and change the appropriate foreign key values in the related table, thereby eliminating the possibility of inconsistent data.
- When you delete a record in the primary table, Access prevents the deletion if matching records exist in a related table.
  - However, if you choose the **cascade deletes option**, Access deletes the record in the primary table and also deletes all records in related tables that have matching foreign key values.

If you want to **enforce referential integrity** between the two tables, check the box. If you choose to enforce referential integrity, you will need to tell Access how to handle Edits and Deletions in the primary table.



**Note:** In order to define referential integrity between tables, the matching field from the primary table must be a primary key or have a unique index, the related fields must have the same data type and both tables must be stored in the same Access database.

## Modify Table



### Changing a Field Name

To change a field name, go to design view and type in a new name. But, before you do this check to see if that field is referenced in any calculations, expressions or validation rules. If so, you will have to change the field name in every reference.


As an alternative, you can leave the field name as is but enter the name you wish to use in the **Caption** property. Captions display in the status bar when the field is activated on forms and reports.

### Changing a Field's Properties

You can change the properties of a field at the bottom of the Design View form. If the modification affects the data already entered into the database, Access will display a message to that effect.


## Adding a Field

To add a new field to the table, select the row where you want the new field to appear, and do one of the following:

- \* Select **Insert, Field** from the menu bar.
- \* Press the **Insert** key on the keyboard.
- \* Click the **Insert Field** button on the toolbar. 

## Deleting a Field

To delete, select the row and do one of the following:

- \* Select Edit, Delete or Edit, Delete Row from the menu bar. Choose Yes on the dialog box to permanently delete the field.
- \* Press the Delete key on the keyboard.
- \* Click the Delete **Field** button on the toolbar. 



## Moving a Field

You can move fields in both Datasheet and Design views.

### To move a field in Design View:

1. Click in the selector bar at the field you wish to move.
2. Point at the selected field. A white arrow appears.
3. Hold the mouse button down and drag the field to its new location.
4. Release the mouse.

Note: You can undo this move immediately after it is performed by going to the **Edit** menu and selecting **Undo**.

### To move a field in Datasheet View:

1. Point to the field name of the column you wish to move. A black down arrow appears.
2. Click once to select the column and release the mouse.
3. Hold the mouse button down and drag the column to its new location.
4. Release the mouse.

This change only appears in Datasheet View. If you choose to save the changes upon closing the table, the change will appear in Design View as well.

## Modifying Database Objects

### Renaming a Table

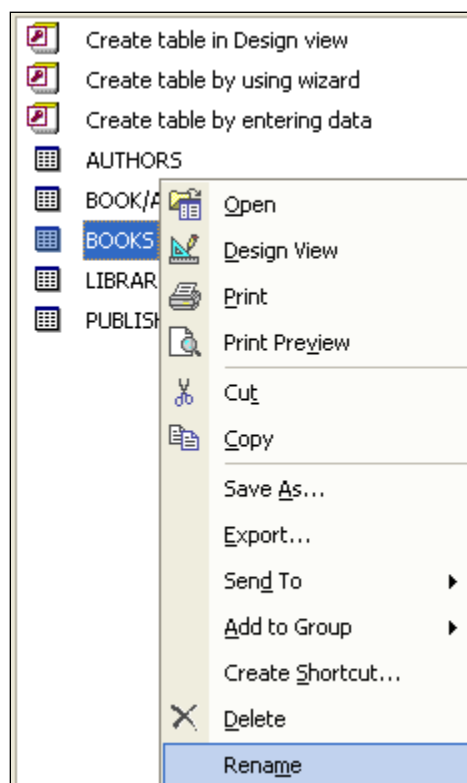
In the database window, do one of the following:

- \* Double-click on the table's name to activate the "edit" mode. Type the new name and press Enter.
- \* Select the table and choose **Edit, Rename** from the Menu Bar
- \* Right-click the table's name and select **Rename** from the context menu.

**CAUTION:** Do not rename a table if it is already being used in any queries, forms and/or reports in the database.

### Exercise

1. Right click on "Books" and choose **Rename**.



2. Rename it **<tblBooks>**.
3. Click off the name, and you have renamed your table!

## Copying a Table

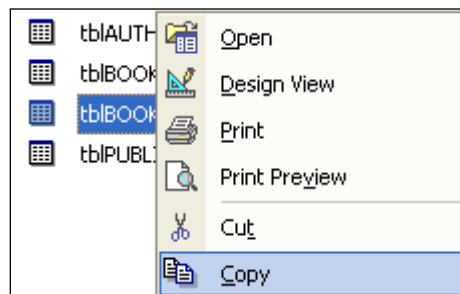
To copy a table, do one of the following from the database window:

- \* Right-click the table's name and choose **Copy** from the context menu; then, right-click a blank area in the window and choose **Paste**. A box appears asking whether you want to copy the structure, the data or both. Make your selection and click OK.
- \* Hold the left mouse button down and drag the table's name to a blank space below the list of tables; release the mouse. This method copies the structure and data of the table.

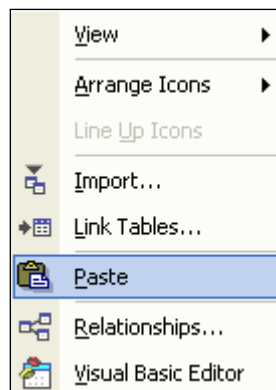
### Exercise

Copy the table **<tblBooks>** and name it **<tblBooksStructure>**.

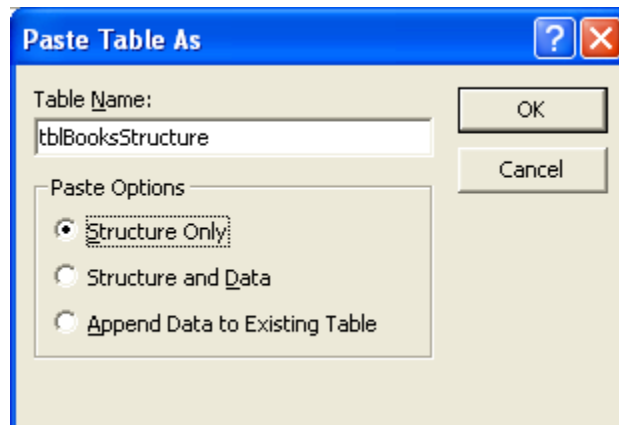
1. Right click on **<tblBooks>**.



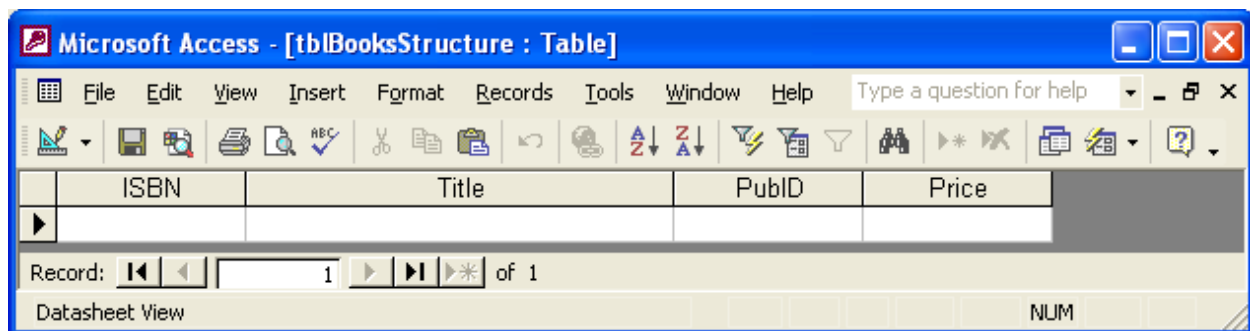
2. Choose **Copy**.
3. Right click in a blank area of the database window.
4. Choose **Paste**.



In the **Paste Table As** dialog box, type in the new table name tblBooksStructure.



5. Under the Paste Options, choose **Structure Only**.
6. Now you have a copy of the tblBooks table structure.
7. When you open the table there will be no records since you copied the structure only. This can be a useful time-saver because you don't have to recreate the table and fields.

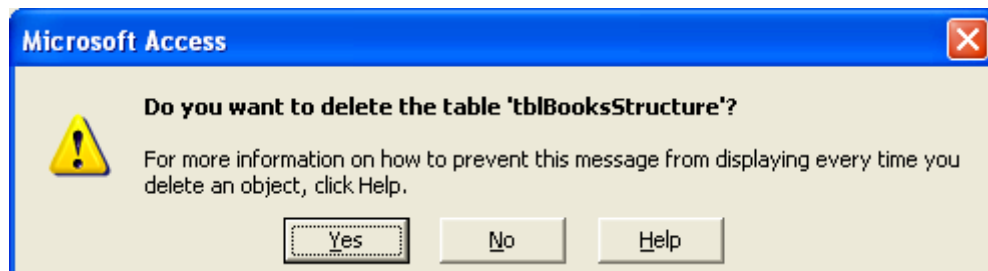


## Deleting a Table

In the database window, select the table and do one of the following:

- \* Choose **Edit, Delete** from the menu bar.
- \* Press the **Delete** key on the keyboard.
- \* Right-click the table and choose **Delete** from the context menu.

Access will warn you prompt you with a warning message before the table is actually deleted. Click **Yes**, to delete the table. Once you confirm the deletion, it cannot be undone.



## Printing the Table

There are two ways to print *all* the data in your table. One is to choose the Print button on the toolbar in Datasheet View. The other is to choose the table in the Database Window and click the Print button from the toolbar.

To change the page setup, choose **File, Page Setup** in Datasheet View and make your changes.

## Printing Selected Records

To print *selected* records, rather than the whole table:

1. Select the group of (adjacent) records you want to print.
2. Select **File, Print** from the Menu Bar.
3. Choose **Selected Records** in the dialog box.
4. Choose **OK**.

## Queries

### What is a Query?

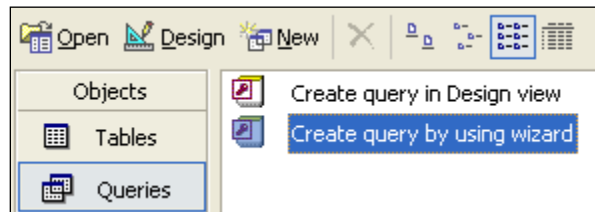
A query is a question that you ask about data. Queries help you to obtain specific information about the data. A Query takes information from the selected tables and displays a subset of data that meets your criteria. Learning how to create queries is the key to unlocking the data in your database!!

- \* The result of the query is temporary or dynamic; hence the results are called a dynaset. The dynaset looks and acts like a table; it is a “live” view of one or more tables. WARNING: If you make changes to the data in the query, the data in the table will also change!!
- \* A query can be based on data from one or more tables. Queries usually connect two or more tables through a relationship between a common field that exists in both tables, such as a key field (a field unique to each table).
- \* Queries allow you to perform calculations on the data; to create data sources for forms, reports, charts and other queries; to make changes to tables and create new tables.
- \* Select Query - This is the most common type of query; it selects information from one or more tables and returns only the records that meet the criteria.

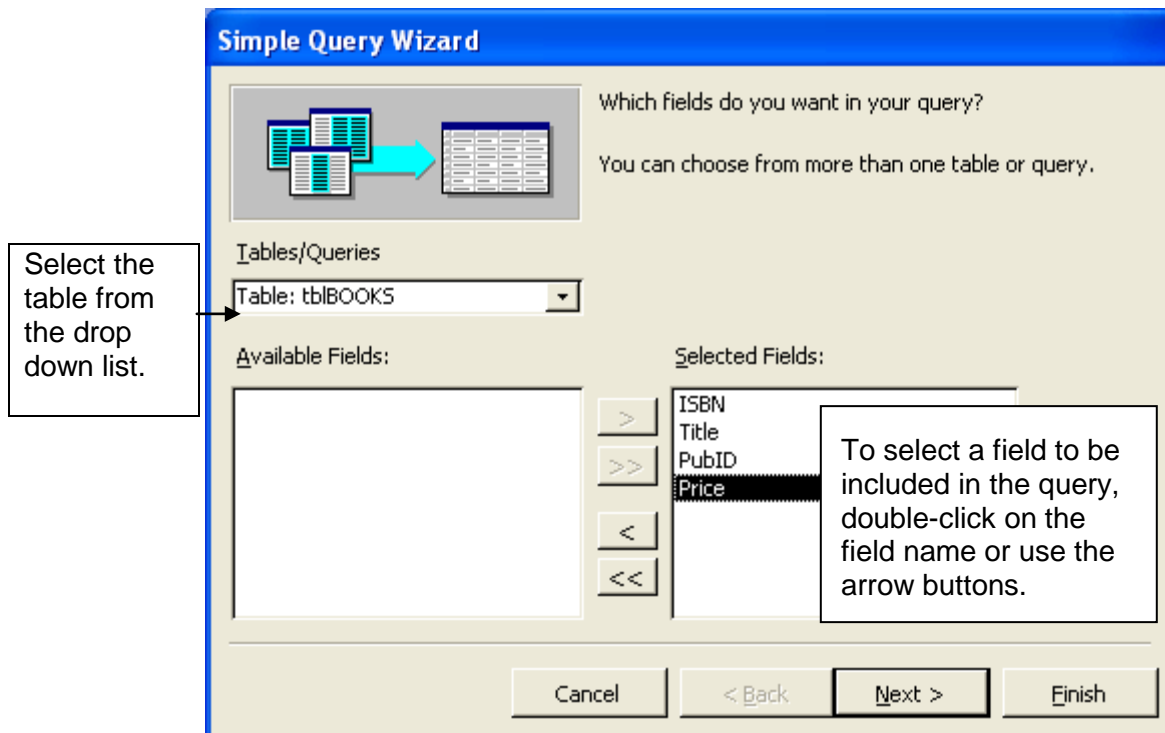
## The Simple Query Wizard

This wizard will help you build the background of the query. It assists with the selection of the tables and fields to be included. You can also create simple calculations. If you want to create a more complicated query, you will need to work in the query design window directly.

1. In the Database Window, click on the **Query** tab and click **Create query by using wizard**.

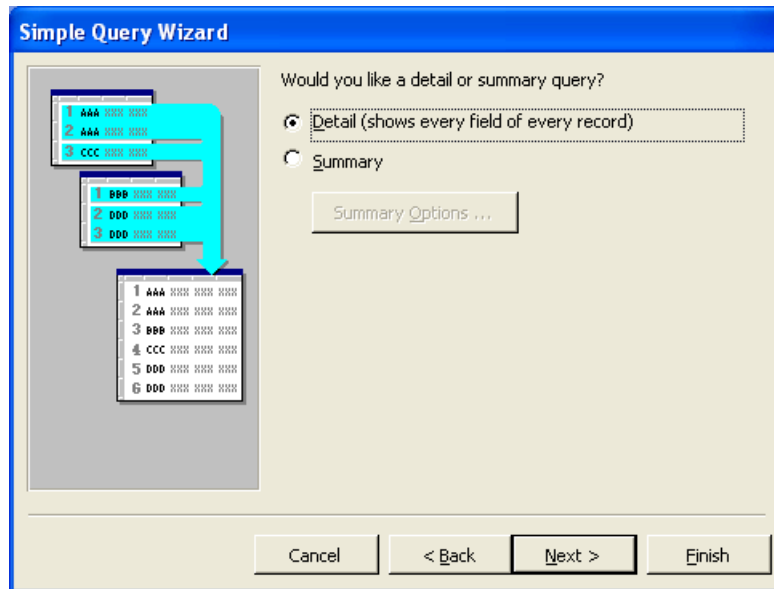


2. Choose the table and fields to include in your query.
  - \* In this example choose the tblBooks table and add all of the fields, as shown below.

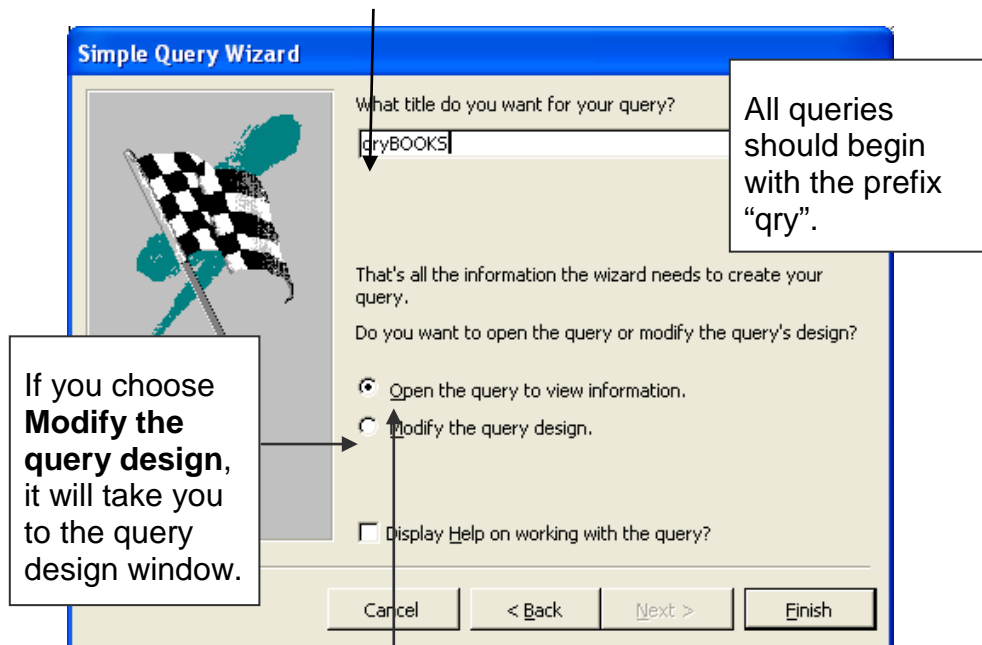


3. After selecting the fields for your query, click **Next**.

4. If you have any numerical data in your table you will be given the option to display your data in detail or summary.
5. Choose **Detail**. Click **Next**.



6. Give your query a meaningful name using naming conventions.



7. Choose **Open the query to view information**. (This view will show you the results of the query in a dynaset.)
8. Click on **Finish** to view the results.



## Running Queries

When you run the query it displays the "answer" to your query. It shows the corresponding records from the table that meets the criteria you specified. The results are shown below.

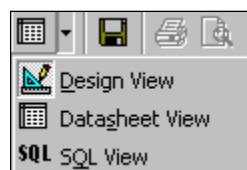
| ISBN          | Title                            | PubID | Price   |
|---------------|----------------------------------|-------|---------|
| 0-103-45678-9 | Running Word 2002                | 1     | \$25.00 |
| 0-11-345678-9 | Running Access 2002              | 3     | \$49.00 |
| 0-12-333433-3 | Running Excel 2002               | 1     | \$25.00 |
| 0-123-45678-0 | Running PowerPoint 2002          | 2     | \$34.00 |
| 0-12-345678-9 | Mastering WordPerfect            | 3     | \$49.00 |
| 0-321-32132-1 | Using Office XP                  | 3     | \$34.00 |
| 0-55-123456-9 | Using Windows Professional       | 3     | \$23.00 |
| 0-555-55555-9 | Troubleshooting Access Databases | 2     | \$12.00 |
| 0-91-045678-5 | Unix for Dummies                 | 2     | \$20.00 |
| 0-91-335678-7 | Excel 97 Simplified              | 1     | \$15.00 |
| 0-99-777777-7 | Internet for Dummies             | 2     | \$49.00 |
| 0-99-999999-9 | DOS for Dummies                  | 1     | \$20.00 |
| 1-1111-1111-1 | C++                              | 1     | \$30.00 |
| 1-22-233700-0 | Visual Basic                     | 1     | \$25.00 |

## Switch between Views

There are three different views for a query: **design view**, **datasheet view**, and **SQL view**.

- Design view is where you create the query.
- Datasheet view is where you can view the results of the query. We saw the datasheet view when we created the query with the Simple Query Wizard.
- SQL (Structured Query Language) view shows the user the actual programming language used to create the query.
  - Go to the **View Menu** and select Design, Datasheet or SQL view.

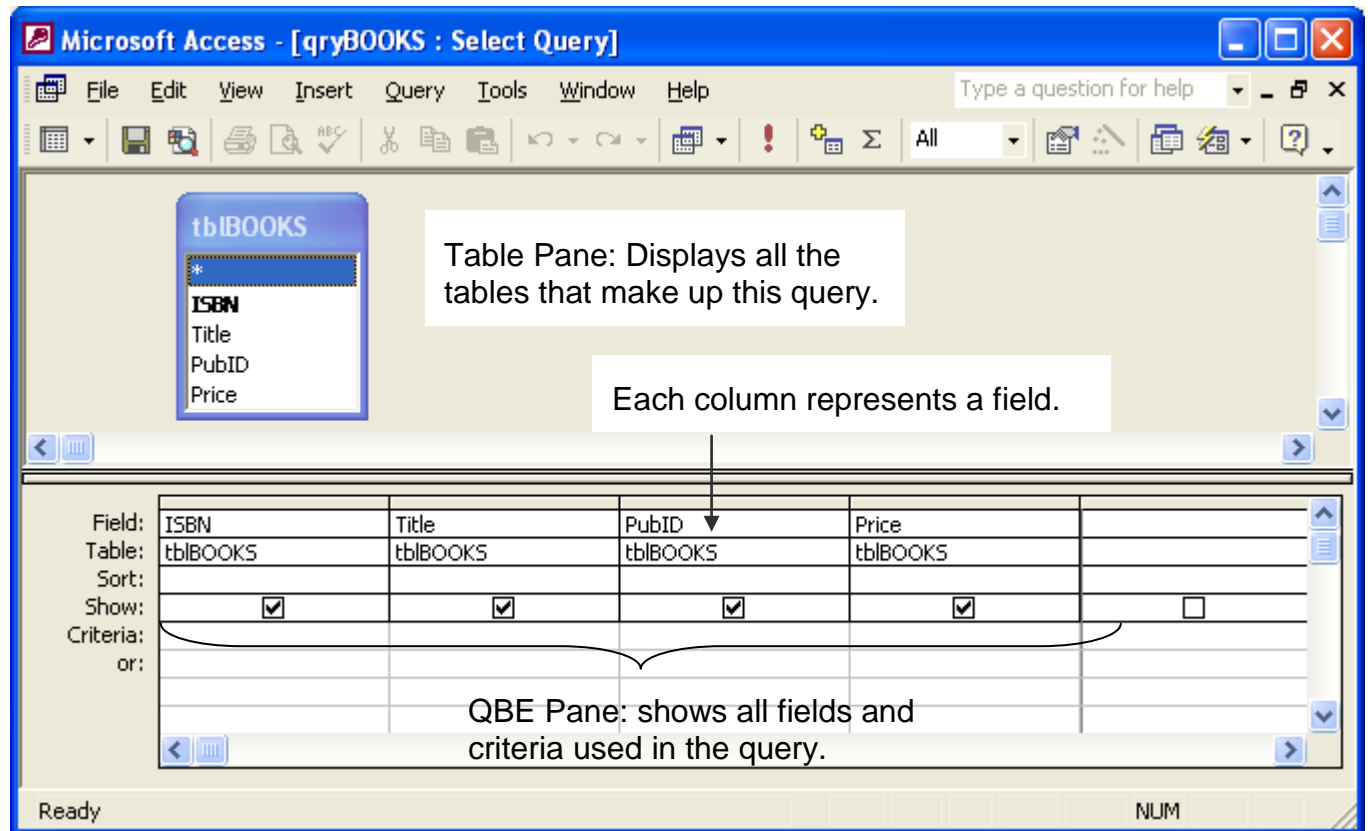
Or



- Click on the arrow for View button and select the view.

## Query Design View (Query-By-Example grid)

Below is the Applicant personal query in design view.

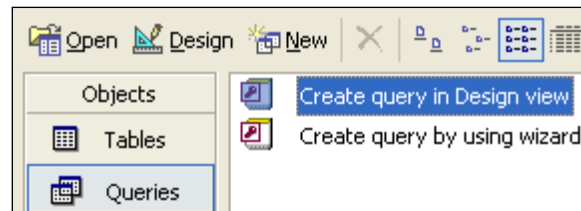


|                 |   |
|-----------------|---|
| <b>Field</b>    | Name of a field or expression.  |
| <b>Table</b>    | Name of the table that the field is from (this is automatically filled in when the field is chosen).                |
| <b>Sort</b>     | Is the data going to be sorted in ascending or descending order.  |
| <b>Show</b>     | If you want the data to appear in the query.  |
| <b>Criteria</b> | Use to filter data.   |
| <b>Total</b>    | Select an option from the list of predefined calculations (To view this option, go to <b>View</b> → <b>Totals</b> ) |

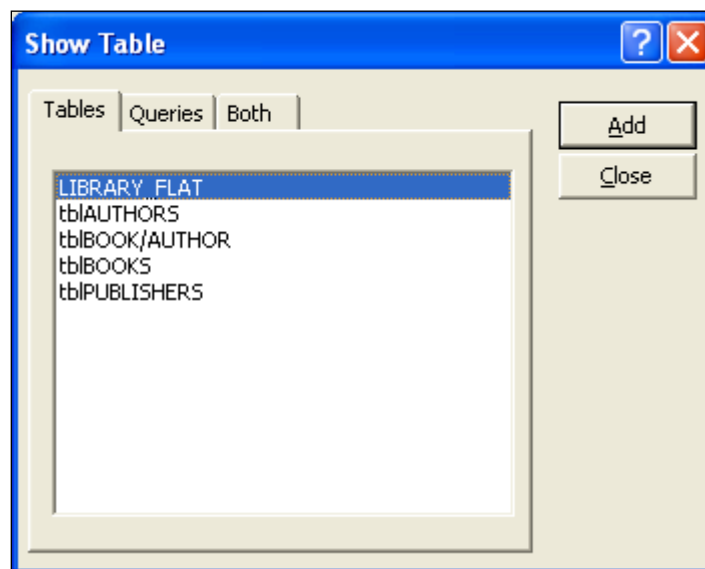
## Creating a Query in Design View

The simple query wizard took care of the simple steps in the design of the query for you. If you want to get into the thick of designing a query, you need to go into the query design view.

1. In the Database Window, click on the **Query** tab and click **Create query in Design View**.



2. The **Show Table** dialog box appears.



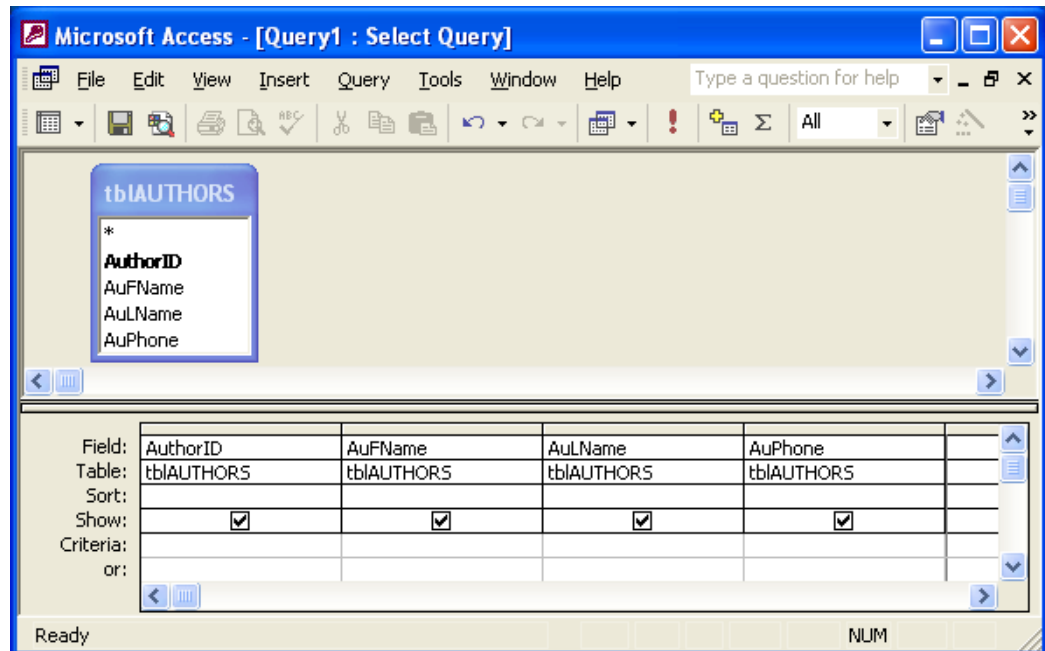
3. Select the tables or queries you want to use for the query and click **Add**. In our example select the **<tblAuthors>** table and click **Add**.
4. Close the dialog box.

## Adding a Field

There are three methods for adding fields to the QBE pane:

- \* Double click on the field name in the table pane.
- \* Click in the **Field** box in the QBE grid and choose the field name from the drop down list.
- \* Click and drag the field from the table pane into the QBE grid.

5. Add all of the fields or the Authors table to the QBE grid as shown below.



## Sort



- Click in the **Sort** field and choose either *Ascending* or *Descending*.
- If you want to do multiple sorts, Access will sort the field on the left first and so on.

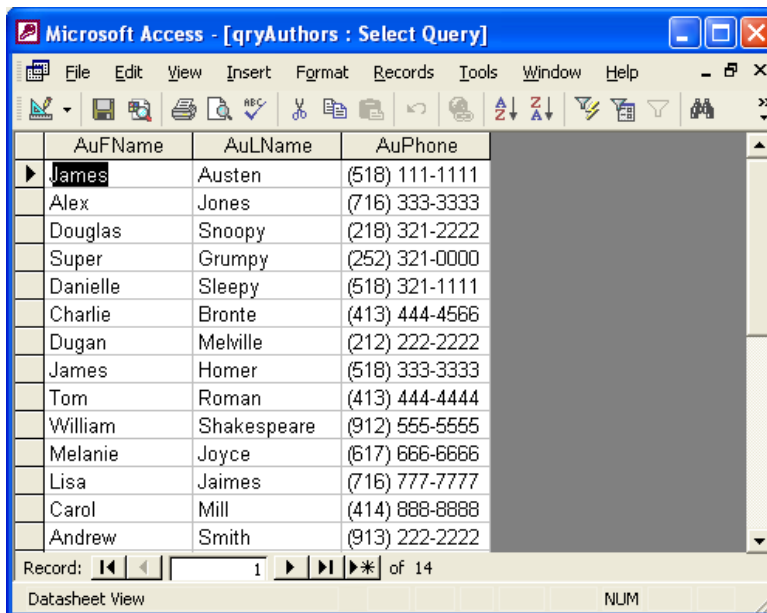
## Show

- You may want to include a particular field in your query for sorting or criteria purposes, but you may not want to show the data. In the Design view, remove the check mark in the Show row.

|           |                          |                                     |
|-----------|--------------------------|-------------------------------------|
| Field:    | AuthorID                 | AuFName                             |
| Table:    | tblAUTHORS               | tblAUTHORS                          |
| Sort:     |                          |                                     |
| Show:     | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| Criteria: |                          |                                     |
|           | <b>Will not Show</b>     | <b>Show</b>                         |

## Running Queries

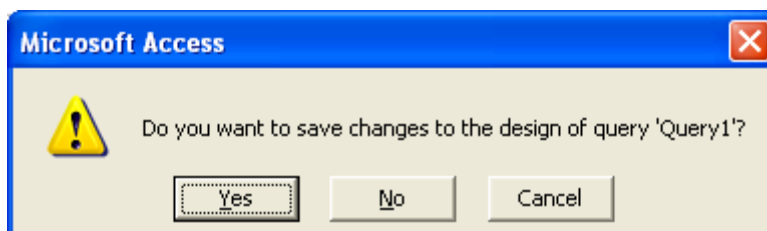
1. Click the **Run**  button or the datasheet view  button, to view the results. A snapshot is shown below.



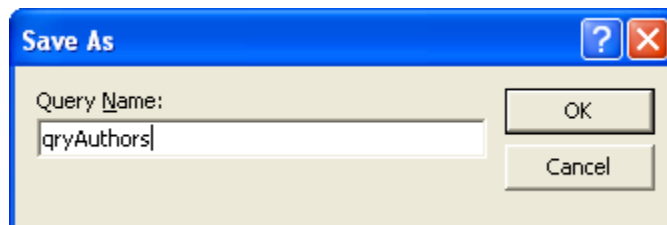
| AuFName  | AuLName     | AuPhone        |
|----------|-------------|----------------|
| James    | Austen      | (518) 111-1111 |
| Alex     | Jones       | (716) 333-3333 |
| Douglas  | Snoopy      | (218) 321-2222 |
| Super    | Grumpy      | (252) 321-0000 |
| Danielle | Sleepy      | (518) 321-1111 |
| Charlie  | Bronte      | (413) 444-4566 |
| Dugan    | Melville    | (212) 222-2222 |
| James    | Homer       | (518) 333-3333 |
| Tom      | Roman       | (413) 444-4444 |
| William  | Shakespeare | (912) 555-5555 |
| Melanie  | Joyce       | (617) 666-6666 |
| Lisa     | Jaimes      | (716) 777-7777 |
| Carol    | Mill        | (414) 888-8888 |
| Andrew   | Smith       | (913) 222-2222 |

## Save the Query

Once you created the query you can save it for future use. You will be prompted to save upon exiting.



Type in a meaningful name for the query and click **OK**. Your query has been saved!



## Summary

In this class we covered the concepts of a relational database and summarized the six steps to designing a sound relational database. Database design is the most difficult and important part in creating a database. Several relational database concepts were covered including: data types, primary and foreign keys, and relationships between tables. You should have a basic look and feel of Access as we viewed the Database Window and the different database objects that make up the Database Window. We developed a table and examined datasheet and design view, fields, data types and primary keys. Now that you have an idea of what Microsoft Access looks like, in the next class you will learn how to create a query to obtain or extract the information you need from your database.

**Source:** Access Database Design & Programming, 3<sup>rd</sup> Edition January 2002, by Steve Roman, O'Reilly publishing, ISBN 1-56592-626-9.

New Perspectives on Microsoft Access 2002, Aadamski, Joseph and Finnegan, Kathleen.