



**LibreOffice**  
The Document Foundation

## Base Guide

# *Chapter 1*

## *Introducing Base*

## Copyright

---

This document is Copyright © 2011 by its contributors as listed below. You may distribute it and/or modify it under the terms of the Creative Commons Attribution Share-Alike License (<http://creativecommons.org/licenses/by-sa/3.0/>), version 3.0 or later.

All trademarks within this guide belong to their legitimate owners.

## Contributors

Dan Lewis  
Reizinger Zoltán  
Hazel Russman  
John Viestenz  
Jean Hollis Weber

## Feedback

Please direct any comments or suggestions about this document to:  
[documentation@global.libreoffice.org](mailto:documentation@global.libreoffice.org)

## Publication date and software version

Published 30 November 2011. Based on LibreOffice 3.4.3.

## Note for Mac users

---

Some keystrokes and menu items are different on a Mac from those used in Windows and Linux. The table below gives some common substitutions for the instructions in this chapter. For a more detailed list, see the application Help.

<i>Windows or Linux</i>	<i>Mac equivalent</i>	<i>Effect</i>
<b>Tools &gt; Options</b> menu selection	<b>LibreOffice &gt; Preferences</b>	Access setup options
<i>Right-click</i>	<i>Control+click</i>	Opens a context menu
<i>Ctrl (Control)</i>	<i>⌘ (Command)</i>	Used with other keys
<i>F5</i>	<i>Shift+⌘+F5</i>	Opens the Navigator
<i>F11</i>	<i>⌘+T</i>	Opens the Styles and Formatting window

# Contents

---

Copyright.....	2
Note for Mac users.....	2
Introduction.....	5
What Is Base?.....	5
Embedded databases.....	6
Flat databases.....	6
Relational databases.....	6
Definitions and examples of relational databases.....	6
Example of flat and relational databases.....	7
Primary and foreign keys.....	9
1:1 (one to one) relationship.....	9
1:n (one to many) relationship.....	9
n:n (many to many) relationship.....	9
Spreadsheets and text data sources.....	10
Front end for ODBC and JDBC databases.....	11
How to open a Base file (or start a new one).....	12
Open an existing file.....	12
Create a new Base file.....	12
Open an existing database using the Database Wizard.....	14
Connect to an existing database.....	15
Main database window and its parts.....	15
Database section.....	16
Tasks section.....	16
Tables, Queries, Forms, Reports list.....	16
Document view.....	16
Database objects.....	18
Database purposes.....	18
Tables.....	18
Fields.....	19
Use the wizard to create a table.....	19
Use Design View or SQL to create a table.....	22
Create a View.....	24
Queries: How to use your information.....	24
Create a query using the Query Wizard.....	24
Create a query using Design View.....	34
Create a Query using SQL.....	35
Forms.....	35
Create a simple form using the Form Wizard.....	35
Create a form using Design View.....	39
Reports.....	39
Use the wizard to create a report.....	40
Modify a report.....	45
Use Report Builder to create a report.....	47
Let Wizards do most of the work.....	47
Data input and removal.....	48

<i>Enter data in the table.....</i>	<i>48</i>
<i>Modify data from a table.....</i>	<i>49</i>
<i>Delete data from a table.....</i>	<i>49</i>
<i>Enter or remove data using forms.....</i>	<i>50</i>
<i>Enter or remove data from fields.....</i>	<i>51</i>
<i>Delete records.....</i>	<i>52</i>
<i>Mixing Base with the rest of LibreOffice.....</i>	<i>52</i>
<i>Calc.....</i>	<i>52</i>
<i>Writer.....</i>	<i>53</i>
<i>Impress.....</i>	<i>53</i>
<i>Using Base with other data sources.....</i>	<i>53</i>

## Introduction

---

A database is a collection of information organized so that it can easily be accessed, managed, and updated. (This is how it is defined at [www.whatis.com](http://www.whatis.com).)

Many database programs are available today: Base, MS Access, File Maker Pro, MySQL, Oracle, and PostgreSQL are some examples. Each program creates a database file or files in its native format. With the right program, called a driver, Base can access these databases as well as others.

An example of a simple database is your email client's address book. This might contain only people's names and email addresses. That may be the only information you want to keep about them. Others might use this database to include phone numbers, pictures, and other personal information. In either case, the address book can be accessed to gain the information contained in it. This information can be updated when needed or deleted when it is no longer needed.

Large companies use much more complex databases. These may contain customer data, inventory of the products contained in the company warehouse or warehouses, inventory of the products the company sells, and records of the products sold. Company officials want and need easy access to this data. When information changes within the business, it needs to be changed in the database. The company's survival could depend upon it.

## What Is Base?

---

Base is a part of an office application, which handles databases through ODF (OpenDocument Format) database documents. Base is not a database application.

These database documents form two categories. Some of them contain the complete database, in which case they are known as embedded databases. Other database documents contain the access information instead of the data for the database for a variety of data sources. These documents are known as frontend database documents. The data source becomes the backend.

Base uses the HSQLDB database engine (version 1.8.0.10) to create database documents (ODF version 1.2). This database engine also updates the ODF database documents within an embedded database.

### Note

- 1) The ODF database document files use the `.odb` file extension.
- 2) You can easily check whether your odb file is an embedded database or not. Look at the bottom left part of the database file window. An embedded database document has the phrases "Embedded database" followed by "HSQL database engine". A frontend database document has the type of data source accessed by Base followed by the location of the data source. (For example, a spreadsheet data source has "Spreadsheet" followed by the location of the spreadsheet.)

Base is not just a database engine as are MySQL, PostgreSQL, and Oracle 11g, for example. Base includes additional functionality allowing the user to create full data-driven applications:

- Defining the table structures used.
- Building the UI (user interface with query definitions and forms).
- Data browsing and filtering for ad-hoc analysis/editing functions.

HSQLDB is written in Java. For the beginner, it is best to install JRE (Java Runtime Environment) to avoid possible problems and frustrations. For those who understand what JDBC, ODBC, and SDBC drivers are, JRE is only required when creating an embedded database or using the JDBC driver to access an external database (a database that is not embedded.)

If you know what a flat database is, you can access this type of file without needing JRE. These databases can include email client address books (Outlook, Thunderbird, and LDAP for example).

### Caution



- 1) You may or may not have problems if you do not install JRE (Java Runtime Environment). It **must** be installed if you are going to create embedded databases or use the JDBC driver to access an external database. You **will not** be able to use any of the wizards contained in Base if JRE is **not** installed. The report wizard **requires** JRE regardless of what type of database is being accessed.
- 2) Base can not properly use Microsoft's version of JRE because it does not contain all the properties of JRE used by Base.
- 3) JRE is available at [www.java.com](http://www.java.com). Download JRE and install it if the above statements indicate you may need it. You can always download and install this program later if you find you need it to work with Base.

## Embedded databases

One modification was made to the HSQL database engine when Base was added as a component of LibreOffice. When a new database is created using HSQLDB, several files are created. When Base creates a new database document file, these files are zipped (along with several other files and folders) into a single file. That is why databases created by Base are called embedded databases: the entire database is contained within a single file.

Base can also be used to connect to other databases. In this case, Base creates a file to make the connection. These are not embedded databases since they exist outside the Base file. They are frontend database documents.

## Flat databases

A flat database contains one or more tables, each containing one or more fields. Each table is completely independent of all the other tables in the database. For example, I have an address database with seven tables in it. One has my family's contacts, another has my business contacts, and another has my wife's family contacts. While some contacts are contained in more than one table, different tables may not have the same information about the same person.

dBase is a flat database program. LibreOffice and several database programs can also create flat databases.

## Relational databases

Relational databases contain one or more tables with one or more relationships; each relationship is defined by a pair of fields. One field of each pair belongs to one table and the second field belongs to the same or a different table. Where relationships exist between fields in the same or different tables, a flat database could still be used, but it provides no mechanism for defining the relationship. Instead the same data must be entered in both fields, making data entry errors more likely. A well designed relational database requires the data to be entered only once, reducing possible errors.

### Definitions and examples of relational databases

#### Table

A *table* consists of one or more vertical columns and one or more horizontal rows containing the data of the database. The data of a given table of a database share one or more common characteristics.

**Field**

A *field* is an attribute or characteristic that divides the data of a table into columns, one column for each attribute.

**Record**

A *record* is a row of a table. The word *Tuple* is sometimes used in place of *record* in database literature, but *record* is used more often. The contents of each row share a common characteristic.

**DBMS**

(Database management system) The structure and restrictions placed upon the data of a database that permit the use of the data in a meaningful way. (DBMS can be singular or plural.)

**RDBMS**

(Relational database management system) The structure and restrictions placed upon the data of a relational database that permit the use of the data in a meaningful way and allow relationships to be defined between them. (RDBMS can either singular or plural.)

Database management systems (DBMS) were developed to use the data in a database. The DBMS separates the data into different categories based upon the defined characteristic for each category. We define each category as a field, and the defined characteristic for a field is its attribute. Based upon the characteristics of the fields, they can be grouped into one or more categories. These categories are defined as tables.

The tables are divided into columns and rows, with the column headers given the name of the fields. The data in each column matches that column's attributes (characteristics).

The DBMS also organizes the data in the rows so that the data in a given row has a common characteristic. When entering data into a table, we enter it into the row to which it belongs.

When relational databases were developed, the DBMS for databases was modified to include the restrictions required to manage relational databases. The result of these modifications was the RDBMS model.

**Example of flat and relational databases**

Table 1 is an example of a table in a flat database representing a group of friends. Each column and row has a specific characteristic.

- The first column contains the first names, the second column contains the last names, the third column contains the dates of birth, the fourth column contains the wedding dates, the fifth column contains the first names of each friend's spouse, and the sixth column contains the first name of the friend's child.
- The headers for the columns are the field names. The data in a given column are values of the field named in the header of that column.
- A record is a row in a table containing data about a particular person, place or thing. Each row of this table contains data about a particular person. We have the first and last name, date of birth, wedding date, name of the spouse, and the child in the family of this person.
- This table is part of the DBMS for this database. The database contains data, but the DBMS contains the structure for using the data. This database contains only the data you see entered into the cells of this table. The DBMS places the data into the cells of the table in the location where they belong.

Table 1: Friend table

<b>First Name</b>	<b>Last Name</b>	<b>Date of birth</b>	<b>Wedding Date</b>	<b>Spouse</b>	<b>Child</b>
Jim	Blount	04/01/67	23/07/93	30/12/99	Eric
Janice	White	18/09/88	01/10/07	30/12/99	Kate

Tables 2, 3, and 4 show the same data with some additions when a RDBMS is applied to the data. In place of the Spouse field, we have used First Name and Last Name fields for the spouse. We have also added the child's date of birth.

- Since the Wedding Date, Spouse, and Child fields refer to the family, we have removed them from the original Friend table leaving the first three fields of Table 2. (The RDBMS adds the fourth field.) We create a new table, Family, using the fields we removed. Then we replaced the Spouse field with two fields: First Name and Last Name while adding the Child Date of Birth field.
- Since the Child and Child Date of Birth fields refer specifically to the offspring, we remove them from the original Family table and use them to create the Child table. We use First and Last Name fields in place of the Child field, and we added the Date of Birth field. (The RDBMS adds the fourth field.) The Family table refers only to the spouses, we will change the name from Family table to Spouse table.
- The RDBMS places restrictions on the database.
  - Each table must have a field with unique values. So, we will add a field of whole numbers (zero and positive integers) with the restriction that a whole number can be used only once in this field. We used Friend ID, Spouse ID and Child ID in the Friend, Spouse, and Child tables respectively.
  - Each table that was created from another table has a field that links to a specific field in the other table: the field with unique values. For clarity, the fields to be linked will have the same name. So, the Spouse table will have a field named Friend ID, and the Child table will have a field named Spouse ID.

Table 2: Friend table (amended)

<b>First Name</b>	<b>Last Name</b>	<b>Date of Birth</b>	<b>Friend ID</b>
Jim	Blount	04/01/67	0
Janice	White	18/09/98	1

Table 3: Spouse table

<b>Wedding Date</b>	<b>First Name</b>	<b>Last Name</b>	<b>Spouse ID</b>	<b>Friend ID</b>
23/07/93	Barbara	30/12/99	0	0
01/10/07	William	30/12/99	1	1

Table 4: Child table

<b>First Name</b>	<b>Last Name</b>	<b>Date of Birth</b>	<b>Child ID</b>	<b>Spouse ID</b>
Eric	Blount	25/12/95	0	0
Kate	White	04/07/10	1	1

Below are the definitions of more terms. Use these tables as an example for illustrating these terms.

## **Primary and foreign keys**

Relational databases use a RDBMS (relational database management system) to manage the data. This is derived from the algebra of sets. This system makes more restraints upon the database. Each of the rows (records) of a given table must be unique; no two rows of a table can be identical. A relationship defined between a pair of fields restricts the values of one of the fields based upon the values of the other field.

### **Primary key**

One or more fields of a table that together guarantee that no two records of the table will be identical. No two values of the primary key can be the same.

The primary key is used to identify the rows of a table. This permits access to any row and any column of a given table by specifying the value of the primary key in the row and the field's name. This access to specific fields and records is the basis for how relationships are defined between tables.

The Friend ID field in the Friend table, Spouse ID field in the Spouse table, and Child ID field in the Child table are the primary keys for this database.

### **Foreign key**

A field or fields in one table whose values must either match the primary Key (PK) in another table or be null.

The Friend ID field in the Spouse table and the Spouse field in the Child table are the foreign keys for this database.

A primary key – foreign key pair defines a relationship. A relationship defines what records of the primary key's table are to be linked with what records of the foreign key's table.

The Friend ID field of the Friend table and the Spouse ID field of the Spouse table form one of the two primary key – foreign key pairs. The Spouse ID field of the Spouse table and the Child ID field of the Child table form the second pair.

### **1:1 (one to one) relationship**

In this relationship, the primary keys of two tables are linked. The second primary key acts as the foreign key in the relationship. For each record containing a value in the primary key's column there is a record containing a value in the primary key's column in the other table. Some of these records may have no value in another column with the exception of the primary key column.

Example of a 1:1 relationship: one table contains the vehicle license numbers issued in a location, and the second table contains the registration numbers in that same location. For each vehicle license number there is only one registration number. And for each registration number, there is only one license number.

### **1:n (one to many) relationship**

In this relationship, the primary key of one table is linked to a foreign key of the second table. In this relationship, one record of the first table can be linked to several records of the second table.

Example: One spouse (the parent that is not a friend) may have several children: one parent to 2 or more children.

### **n:n (many to many) relationship**

To have a many to many relationship, three tables are needed. Table 1 contains two foreign keys. The primary key for Table 2 is linked to one of the foreign keys, and the primary key for Table 3 is linked to the other foreign key.

Example: Consider a school. Each student has one teacher, but each teacher has many students. Each student also has one set of parents (no divorce nor remarriage in this school). But one set of parents may have more than one student in the school. We have three relationship here: teachers to students, parents to students, and teachers to parents. The first two relationship are 1:n, but the third relationship is n:n. Many parents see many teachers about how their children are doing at school.

The Student table is the third table. It contains the two foreign keys. One foreign key links to the primary key of the Teachers table, and the other foreign key links to the Parents table.

## Spreadsheets and text data sources

Base creates database document files for embedded databases. It can also connect to other files if they are in the correct format. This is true for spreadsheets as well as some text Documents.

Table 5: Sample spreadsheet entries

	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>
1	FoodID	30/12/99	Vegetables	Meats
2	0	30/12/99	tomato	Beef roast
3	1	30/12/99	zucchini	Pork chop

### Note

When Base accesses a spreadsheet or text document, it creates a database document file containing all the information needed to access the spreadsheet or text document.

Table 5 contains the entries for the first four columns and three rows of a sheet we will name *food* within a spreadsheet named *inventory*. When Base is used to access the inventory spreadsheet, a table named *food* is created. This table has four fields: FoodID, Fruits, Vegetables, and Meats. These field names are taken from the entries in the first row.

For spreadsheets, Base creates a table from each sheet and uses the sheet name as the name of the table. So if you have three sheets in the spreadsheet but have not named any of the sheets, Base will create these tables: Sheet1, Sheet2, and Sheet3 for you.

### Caution



1. When accessing a spreadsheet, you must first make sure that all the entries in the first row contain the names of the data contained in the columns below them. If the sheet contains only data and no names in the first row, the data in the first row will be used as the names for the fields of the table. You can then no longer access the data in the first row from within Base. So before accessing the data from Base, add an empty row to serve as row one. Add the names you want to use as field names. Then access the spreadsheet.
2. While you can use Base to access this type of file, you cannot change any of the data from within Base. All changes in data must be done from within the spreadsheet or text document.

The database wizard allows you to connect to text files containing data. You are asked in the wizard to specify the separators used in the file. You can later edit the settings for these separators when the Base document file is open. Use **Edit > Database > Properties**. Select the *Advanced Settings* tab, and change the separators which need changing.

The separators must be used consistently in the text file if Base is to access the data in it. But, you have some freedom in how you name the text file. You can use \*.txt. You can also use \*.csv if the file is in the CSV format. Finally, you can use whatever three letter file name extension you want to

use. However, you do have to enter the custom suffix in the database wizard so that Base will recognize it. This is covered in more detail in Chapter 5 of the *Base Guide*.

**Tip**

To access text files, Base requires them to be specially formatted. CSV text files with the ending .csv are one example of this. To be accessed by Base, text files must have specific field, text, decimal, and thousands separators. These separators must be used consistently throughout the file or Base will not be able to accurately identify the data contained in the file.

The figure below contains some data contained in a CSV file and has the format that Base needs to access data in a text file. Each line of the text file contains one row of the data information. Any text data is enclosed within quotation marks, and numbers are not surrounded by any punctuation marks other than those used to separate the data into their respective fields. The first row contains the field names, and all field names are enclosed with quotation marks since they are text.

```
"Date","Time","Phone No.,""Designation","Total Time","Minutes","Sec","Other Party"
"Jun 24","08:59 AM,999-888-7676,""Dialed",1,0,42,"Rachel"
"Jun 26","04:05 PM,999-888-6767,""Dialed",1,0,34,"Dee"
```

Figure 1: Text document with data

**Note**

Time formatted as AM or PM is a number in a CSV file. So is the phone number format above: 999-888-7676.

Base can access the data in this file and produce the table below.

Table 6: Table for data in Figure 1.

<b>Date</b>	<b>Time</b>	<b>Phone No.</b>	<b>Designation</b>	<b>Total Time</b>	<b>Min.</b>	<b>Sec.</b>	<b>Other Party</b>
Jun 24	08:59 AM	999-888-7676	Dialed	1	0	42	Rachel
Jun 26	04:05 PM	999-888-6767	Dialed	1	0	34	Dee

## Front end for ODBC and JDBC databases

Database files are categorized by their type. Flat and relational are just two types of databases. Databases are also categorized by how database engines can access (connect to) specific types of database files. To make the connection, a driver is required by the database engine. JDBC and ODBC are two such drivers. MySQL and PostgreSQL native drivers are two more.

SDBC refers to Base built-in drivers or to drivers added through extensions, whereas JDBC and ODBC are drivers installed on the computer separately. SDBC drivers are available as extensions for LibreOffice or provided as a service by the company that makes the database program that created the database we want to access. These drivers must be properly registered in LibreOffice so that LibreOffice can connect to them.

To properly access a particular database, we need to do so in the same way as the database engine which created the database. These drivers are designed to perform this function.

**Note**

Chapter 8 of the *Base Guide* describes how to connect to these JDBC and ODBC databases. It is more involved than we can discuss in this chapter.

## How to open a Base file (or start a new one)

---

There are several ways to open a Base document file once it has been created. There are at least two ways to create a new database: one uses a wizard contained in LibreOffice, and the other uses a macro. We will discuss only the use of the wizard to create a new file.

### Open an existing file

An easy way to open an existing Base file is to open the LibreOffice program first. Then use **File > Open**, browse to the file, select the file, and click **Open**. The database will open in the database file window.

You can also use *Control+O*, then browse to the file, select it, and click **Open**.

### Create a new Base file

Creating a new Base file is rather simple. The difficulties come in designing the tables, their fields, the field types, the field properties, and the relationships, if any, that will exist between your tables. You should always plan very carefully first as to how to create the structure of the database. Only then are you ready to create the Base file. Finally, you should create the structure of the database as you designed it.

#### Caution



Planning is very important ahead of time or your database will not become what you want it to be. This will require someone to spend extra time changing tables, their relationships, the forms based upon the changed tables, queries, and reports to correct it. It might be easier to create another database from the beginning. (See Chapter 2, Planning/Designing your Database in the *Base Guide*.)

There are three ways to create a new Base file. (LibreOffice must be open at the time.) Each way opens the Database Wizard.

- Choose **File > New > Database** from the menu bar.
- If no document is open, click the **Database** icon in the Start Center.
- If a document is open, click the arrow beside the *New* icon and select **Database** from the dropdown list.

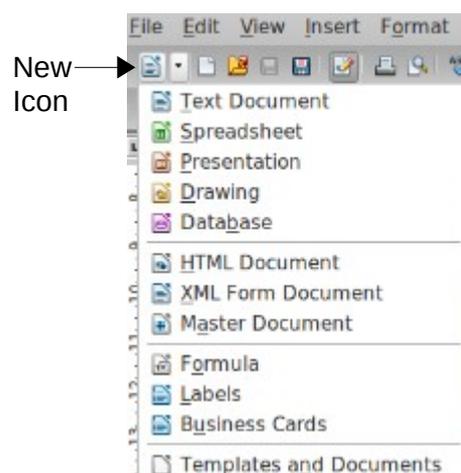


Figure 2: Opening the Database Wizard

In Figure 3, you see the two steps the wizard will take you through.

- 1) Select the database.
- 2) Save and proceed.

### Step 1: Select database

When you open the Database Wizard, you are asked a question with three choices (see Figure 3). The first choice, *Create a new database*, is the Default setting. So if you want to create a new database, you only need to go to Step 2.

- Click **Next**.



Figure 3: Database Wizard (page 1)

### Step 2: Save and proceed

Three things have to be done to accomplish this step: register the database, decide what you want to do next after saving the database, and click the Finish button.

You should register your database. When you do this, you tell Base where you are saving this database. Then the location of this database is added to a file within the LibreOffice folder on your computer. When LibreOffice is open, you can access this list at **Tools > Options > LibreOffice Base > Databases**. (See Figure 5 for an example of this list.)

#### Note

Registering the database is only adding the location of the database to the list kept within the LibreOffice folder. This information is **not** sent to another computer or network.

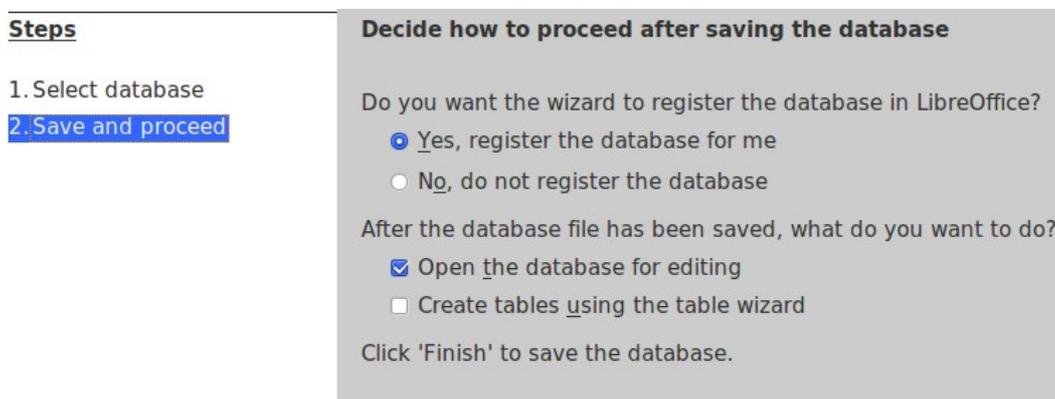


Figure 4: Database Wizard (page 2)

Registered databases	
Registered name <small>Δ</small>	Database file
Addresses	/home/dan/Documents/
Bible Reading	/home/dan/Documents/
Bibliography	/home/dan/.openoffice.
Budget-personal	/home/dan/Documents/
Electricity	/home/dan/Documents/
financial	/home/dan/Documents/
Health	/home/dan/Documents/
schedule	/home/dan/Documents/

Figure 5: Registered databases

This information is for local use only. It allows you direct access to these databases through the Data Source window. (Use *F4* to open this window.) If you register the database, it will be listed in this window. If you do not register the database, it will not be listed.

### Tip

This file does not update itself. If you move the database or change its name, the changes will not appear in the file. You will have to make the changes in this file yourself using the menu path given above. (Help is available where you need to go to change your list information.)

Your next choice depends upon what you want to do with the new database. If you are going to create new tables, select the table wizard. If you want to edit the database, this should be your choice. With a new database, editing the database means creating tables using Design View or using the SQL Statement dialog. (Use **Tools > SQL** to open this dialog.) The Default selection is *Open the database for editing*.

Step by step, here is what to do:

- 1) In the Database Wizard, select *Yes, register the database for me*.
- 2) Use the Default setting: *Open the database for editing*.
- 3) Click **Finish**. The Save As dialog opens.
- 4) In the *File name* box, replace *New Database* with the name you want for your database.
- 5) Browse to where you want to save your database.
- 6) Click the **Save** button.

## Open an existing database using the Database Wizard

### What do you want to do? (page 1 of the database wizard, Figure 3)

This choice refers to opening a database document file (\*.odb file format).

*Open an existing database file* has two methods to select the database you want to open: a drop-down list of recently used databases, or browse for the database using the **Open** button.

- 1) In the Database Wizard, select *Open an existing database file*.
- 2) Select a database file to open.
  - a) *Recently used* drop-down:
    - i. Click the *Recently used* drop-down list to open it. The Default selection is the last database you used.
    - ii. Click the database document file you want to open.
    - iii. Click the **Finish** button.
  - b) OR, click the **Open** button.
    - i. Browse to the desired file.
    - ii. Click the desired file to select it.
    - iii. Click **Open**.

## Connect to an existing database

We have a data source that we want to access. To do this we create a Base file to access the data source. This takes a few more steps than creating a new database.

Base can connect to a variety of database files using the database wizard. The list of these types of database files are below.

- 1) In the Database Wizard, select *Connect to an existing database*.
- 2) Click the drop-down list to open it (default selection: JDBC). The selections you will see may differ from the ones in the example below; some depend on your operating system and are different on Windows, Linux, or Mac OS X.
- 3) Select the type of database (data source) to which you want to connect.
- 4) Click **Next**.

The Database Wizard changes based upon what database you selected. Chapter 8 of the *Base Guide* contains detailed instructions on how to connect to the specific database types listed in this drop-down list.

## Main database window and its parts

Once you have created a new database and saved it, the *main database window* opens. Opening a database file will also open this window. It contains three parts: *Database*, *Tasks*, and a list of objects of the type selected in the *Database* section.

### Note

- 1) Figure 6 and 7 are snapshots of the database created by the instructions contained in this chapter.
- 2) A new empty database opens with the Tables section active; an existing database opens with the Forms section active.

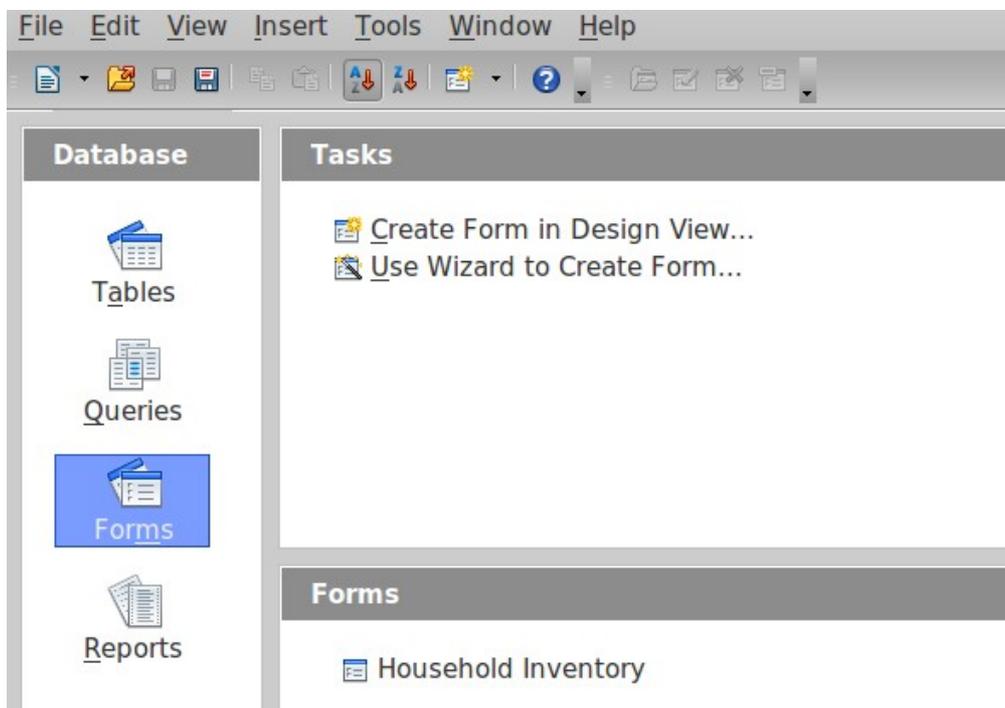


Figure 6: Main database window left side

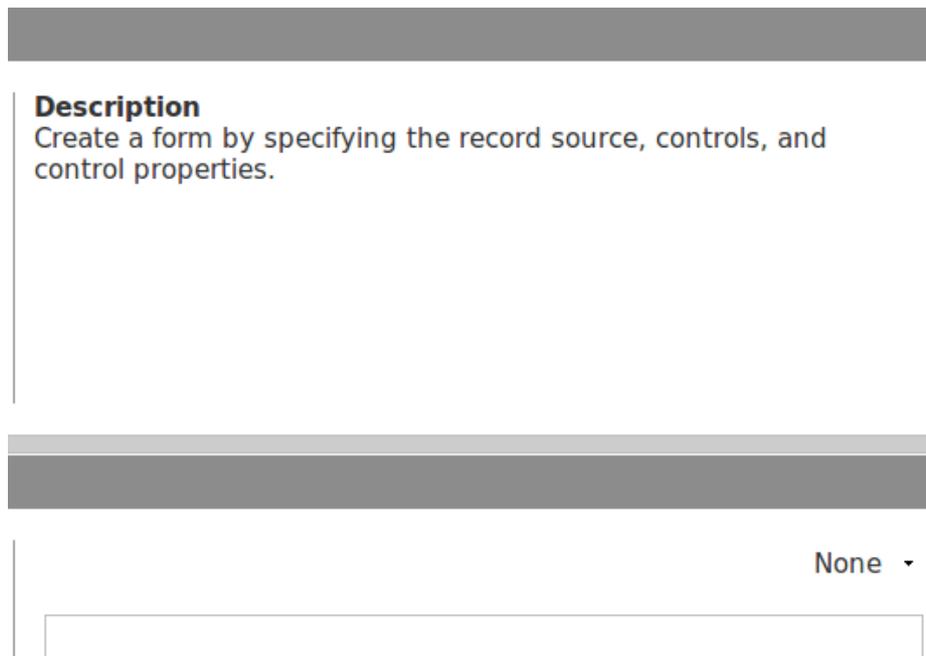


Figure 7: Main document window right side

## Database section

The Database section is the column down the left side containing four icons: *Tables*, *Queries*, *Forms*, and *Reports*. These four icons represent the four types of *Objects* of the database. Select any one of the four objects (icons). What appears in each of the other sections depends upon which icon you select.

## Tasks section

The Tasks section is located at the top right. It contains a list of the things you can do with the Database object you have selected. Move the mouse cursor over one of the items in this list, and a description of what this item will do appears.

In Figure 6, two tasks are listed in the Tasks section. Either task will create a new form since the Forms object is selected in the Database section. Select *Tables*, and three tasks appear. Select *Reports*, and two task appear.

## Tables, Queries, Forms, Reports list

Below the Tasks section is the list section; its title is the same as that of the currently selected object. In Figure 6 it contains a list of the forms in the database. If the Tables icon is selected, a list of tables appears. Similarly if the Queries or Reports icon is selected, the list will show these.

## Document view

To the right of the list is a small window giving you more information about the specific document (table, query, form, or report) you have selected. The view is controlled by a drop-down list containing three options: None, Document Information, and Document.

When Tables or Queries is selected in the Database section, only two choices are available: None or Document. When Forms or Reports are selected, all three choices are available. Selecting *None* leaves the Document view window blank. The Document view of a form shows the top left part of it. The Document view of a table shows as much of the entered data for which as there is room.

Figures 8, 9, and 10 are Document Views. The first figure shows the document information for a form. The second figure shows the Document View for a table. The third figure shows the Document View of a form.

Here are samples from one of my database document files. Figure 8 shows the document information for one of my forms. It was last edited April 13, 2011. Figure 9 shows data from one of the tables and figure 10 shows a thumbnail of a form from this same database.

**Modified by:**  
Dan Lewis

**Modified on:**  
04/13/2011, 13:50:18

Figure 8: Document View with Document Information selected of the Transactions form

Auto ID	Odometer	Work done	Next Scheduled	Quantity	Data ID
4	5225			4.512	10
5	45225	Oil change & lu	07/15/10		11
6	5489			12.123	26
7	12233			6.859	30
8	5677			6.318	34
9	5974			9.904	59
10	6246			9.070	62
11	6557			10.365	85
12	12553			11.143	86
13	6848			9.714	89
14	7139			9.715	105
<AutoField:					

Figure 9: Document view of the Auto table

The image shows a document view of a form with the following fields listed on the left side: CategoryID, AppraisedVal, DatePurchase, Description, Insured (with a checkbox), Item, Manufacturer, ModelNumber, PlacePurchase, PurchasePrice, SerialNumber, Notes, and Room. The form content is mostly obscured by a grey overlay.

Figure 10: Document view of a form

## Database objects

---

The flat database described in this chapter with data already entered is available online at:

<http://www.libreoffice.org/get-help/documentation/>. You may download it to see the same things you see in the figures.

A database is a collection of data. For example, a list of the titles of all our music albums or singles could be such data. If our list is short, we can find what we need fairly easily. But if we have a long list, finding what we want can be much more difficult. We need to apply some organization to our list giving it some structure: we need a management system for our list. Then we can much more easily find what we want to know.

This is the purpose of the database objects: to give our collection of data structure and a database management system (DBMS). This system will permit us to add, delete, or modify our collection of data (tables and forms). We can also gather information about parts or all of the data through queries and reports.

We will illustrate by creating a household inventory database, beginning with a list of household items and the information we have about these items. As we discuss each database object, more structure will be added to the database management system.

## Database purposes

We begin with a collection of data. We also need one or more purposes for this data. What are we going to do with it? What information do we want to get from it? With the answers to these questions, we can create data objects with the properties that we want. Without these answers, our database and DBMS may not be any use to us at all.

Think of a household inventory database as a list of household items for a newly constructed house. All of these items are in storage waiting to be moved to the house and put in their places. Our insurance company needs information to make sure everything is covered by our insurance plan. We also need information in case something was stolen. Finally, we need to know where the item was purchased.

Now we have some uses for our database. This will in turn determine what our tables, forms, queries, and reports will be. And we are much more likely to get the correct information that we need from the database.

## Tables

Tables are used to organize our data into smaller collections. What do we know about the data for our inventory database? What else do we know about how the data should be organized?

We have two possibilities. We could divide the list into smaller lists, one for each room. Each one of these lists is a table. This would be fine for the people moving the household items, but the insurance company needs a report about the entire inventory. To achieve this, the entire inventory list would have to be in one table.

In our case, the insurance company wins, and we use only one table. But we will add one more characteristic for each item on the list: the room into which the item will be placed.

### Caution



**First**, determine what you want to accomplish with your database. Then determine what tables you will need and how they are related to each other. What types of fields will you be using and to which table will each of these fields belong? There are many other questions you should ask and answer before creating the first table. **If you do not plan ahead of time, your database will not work the way you expect.** Planning ahead will save time and aggravation later.

## Fields

So, now we have a single table that contains a list of household items. But what are the characteristics of the items that we want to include in this table? We have mentioned two characteristics: the name of the item and what room it is in.

Our insurance company requires the appraised value, date purchased, and purchase price. The information needed to report something stolen are manufacturer, model number, description, whether it is insured or not, serial number, and perhaps some special notes.

So, now we have a list of the characteristics that will apply to each household item. We also need the place where the item was purchased, in case we want to replace an item from the same place, or in case we definitely don't want to buy the replacement in the same store.

LibreOffice Base has a wizard to create tables. This wizard contains many sample tables with their fields. Some of them are for business databases and some for personal databases. Specifically, this wizard contains a household inventory sample table with suggested fields and their field types and properties. We will use the fields that we mentioned above.

We will also add CategoryID as the primary key for the table. It will contain unique whole numbers. We can then specify a given row of the table by requiring the CategoryID field have a certain value. This is done in step 3 of the table wizard.

### Caution



When creating a table, you need to have a primary key for the table. Without this, you will not be able to modify nor delete data from the table. You will be able to open, delete, or edit the table, but this is all you can do without the existence of a primary key.

### Tip

When creating a database, open the table wizard to view the sample lists to see if any of them will meet your goals. Just remember to plan what you want to accomplish first. Then look for a match afterward.

## Use the wizard to create a table

Click **Use Wizard to Create Table** in the Tasks section. The *Table Wizard* opens. The four steps are listed on the left of the wizard with the choices you make during each step on the right.

### Step 1: Select fields

The wizard has two categories: *Business* and *Personal* in the *Select fields for your table* section. Below the Business category is the *Sample tables* drop-down list. Selecting the Business category produces the business drop-down list while selecting the Personal category produces the personal drop-down list. Below the Sample tables list is the *Available fields* list for the Sample table selected.

Select the Personal category, and then select the HouseholdInventory sample table. Figure 11 shows the Personal category selected and the HouseholdInventory sample table chosen. The *Available fields* list for this sample table shows only half of the suggested fields. Scroll down the list if you want to see them all. Of the 16 suggested fields we will select the following: AppraisedValue, DatePurchased, Description, Insured, Item, Manufacturer, ModelNumber, PlacedPurchased, PurchasedPrice, SerialNumber, RoomID, and Notes.

- 1) In the Categories section, click the Personal option button.
- 2) From the *Sample Tables* dropdown list select *HouseholdInventory*.
- 3) In the *Available fields* list, click each of the above listed fields and then click the > button. This moves these fields to the *Selected fields* list.
- 4) Click **Next** to go to Step 2.

**Select fields for your table**

This wizard helps you to create a table for your database. After selecting a table category and a sample table, choose the fields you want to include in your table. You can include fields from more than one sample table.

Category  
 Business  Personal

Sample tables  
 HouseholdInventory

Available fields  
 AppraisedValue  
 CategoryID  
 DatePurchased  
 Description  
 Insured  
 InventoryID  
 Item  
 ItemType

Selected fields

1	Category labeled option buttons	3	Move selected fields from left to right
2	Move selected fields from right to left	4	Change the order of the listed fields

Figure 11: Step 1 of Table Wizard

## Step 2: Set field types and formats

All of the fields have the same field types and formats that we want except for RoomID. We want to change the name, and change the field type (below).

- 1) Click RoomID to enter this name in the Field name box.
- 2) Change RoomID in the Field name box to Room.
- 3) Change the Field type from Integer[INTEGER] to Text[VARCHAR].
- 4) Change the Length from 10 to 20.
- 5) Click **Next** to go to Step 3.

Field information

Field name

Field type

Entry required

Length

Figure 12: View of the changes made

### Tip

For more detailed information on Primary Keys, see Chapter 8, Getting Started with Base, in the *Getting Started Guide*.

### Step 3: Set Primary Key

The primary key is used to specify the row from which we want information. Since the `CategoryID` field contains unique values, it can be used as a primary key, and we will do so.

We could select *Automatically add a primary key*. This will create the field, `ID`, with the field type, `Integer`, and the `AutoValue` field property, `Yes`. But we won't.

We have a third choice: *Define primary key as a combination of several fields*. This choice is available for relational databases although it could be used for a flat one. When you use this choice, you must first determine that the values for the fields you select are unique: no two rows have the same values in the same order for these fields. Selecting a single field as the primary key is simpler than using two or more fields as the primary key.

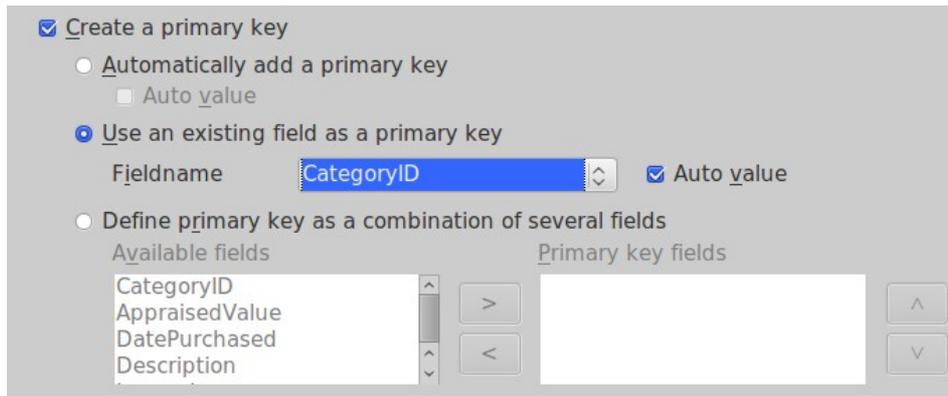


Figure 13: Using existing field as primary key

#### Tip

When the *Auto value* property is selected for a primary key, the values for it are automatically assigned by Base.

Select *Use an existing field as a primary key*.

- 1) Click the *Fieldname* drop-down list, and select `CategoryID` from this list.
- 2) Check the box in front of *Auto value*. Figure 13 shows the correct selections.
- 3) Click **Next** to go to Step 4.

#### Caution



Do not use an existing field or a combination of existing fields as a primary key unless you are sure that it will always be unique. For example, in a table of people, like Table 1 above, it is tempting to use given name + surname as a primary key. But it is quite possible for two people to have the same name.

### Step 4: Create Table

#### Tip

A common naming practice has been to make two or more words into one word for the name of a field or database. `HouseholdInventory` and `CategoryID` are two examples. You may use spaces in names used in a database if you want. Such names may require double quotes when using SQL to create, delete, or modify a table or field. (See Chapter 3 of the *Base Guide* for detailed information.)

Since we are using the fields from a suggested table (`HouseholdInventory`), the table wizard suggests `HouseholdInventory` as the table name.

- 1) Name it *Household Inventory*. (To do this insert a space between the two words: household and inventory.)
- 2) Click **Finish**.

## Use Design View or SQL to create a table

Design View uses the Table Design dialog to create a table. With this dialog, you can name your fields, define the field properties, and further refine the field properties. Creating a table using design view is more complex. But once you have mastered the principles that the table wizard uses, you can apply these principles directly to the Table Design dialog. This dialog is discussed in detail in Chapter 3 of the *Base Guide*.

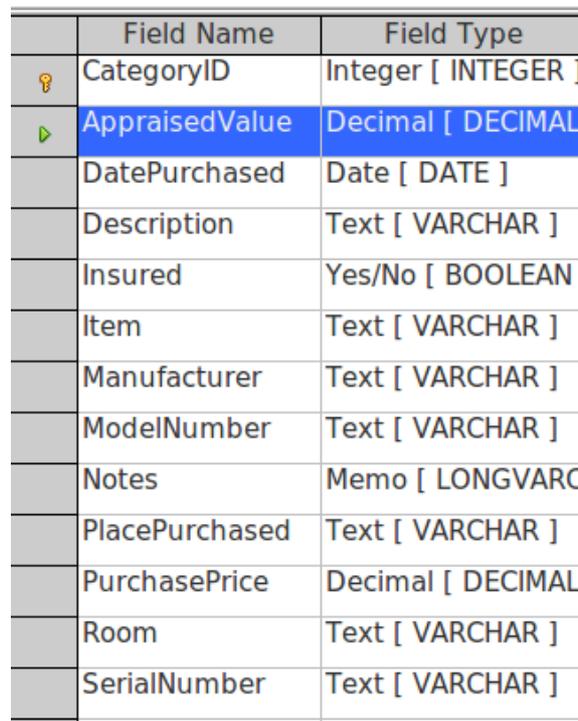
SQL (Structured Query Language) can also be used to create a table. This is the most complex method of creating it. It is also the most exacting. A single typing error while using this dialog can cause problems. This method also requires a knowledge of how to use SQL.

To create a table using SQL, Base uses the SQL Statement dialog. From the main database window, **Tools > SQL...** opens this dialog.

The Table Design and the SQL statement dialogs are discussed in detail in Chapter 3 of the *Base Guide*, Data Input and Removal.

The Design View dialog can also be used to modify a table. Tables are edited using Design View, and we will do so now. We need to do so because the Table Wizard did not do all that it was supposed to do.

- 1) Edit the Household Inventory table using the Table Design dialog.
  - a) In the main database window, click the **Tables** icon in the Database section.
  - b) In the Table list section, right-click **Home Inventory**.
  - c) Select **Edit** in the context menu. The Table Design dialog opens.



	Field Name	Field Type
🔑	CategoryID	Integer [ INTEGER ]
▶	AppraisedValue	Decimal [ DECIMAL ]
	DatePurchased	Date [ DATE ]
	Description	Text [ VARCHAR ]
	Insured	Yes/No [ BOOLEAN ]
	Item	Text [ VARCHAR ]
	Manufacturer	Text [ VARCHAR ]
	ModelNumber	Text [ VARCHAR ]
	Notes	Memo [ LONGVARC ]
	PlacePurchased	Text [ VARCHAR ]
	PurchasePrice	Decimal [ DECIMAL ]
	Room	Text [ VARCHAR ]
	SerialNumber	Text [ VARCHAR ]

Figure 14: Table Design dialog

- 2) Observe the properties of the AppraisedValue field.
  - a) Click the field name, AppraisedValue, to select it. A green arrow appears on the left side of the field selected. (A yellow key is on the left of CategoryID because this is the primary key for the table.)
  - b) Look at the *Field Properties* section at the bottom of the dialog. These are the values we set using the Table Wizard.

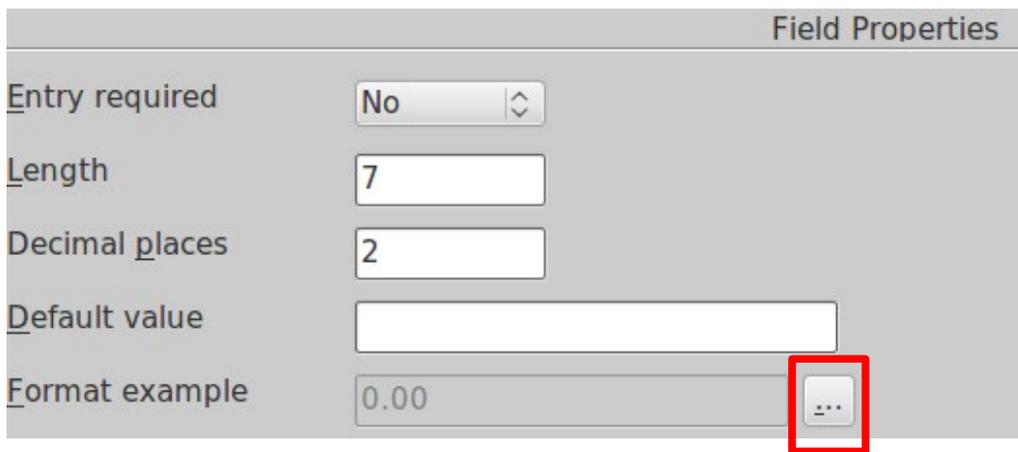


Figure 15: Field Properties section

- c) Change *Format example*.
  - i. Click the button with the ellipse (...). The Field Properties dialog opens in the Format tab.
  - ii. Select Currency in the Category list.
  - iii. From the Format list select the currency format you will use.

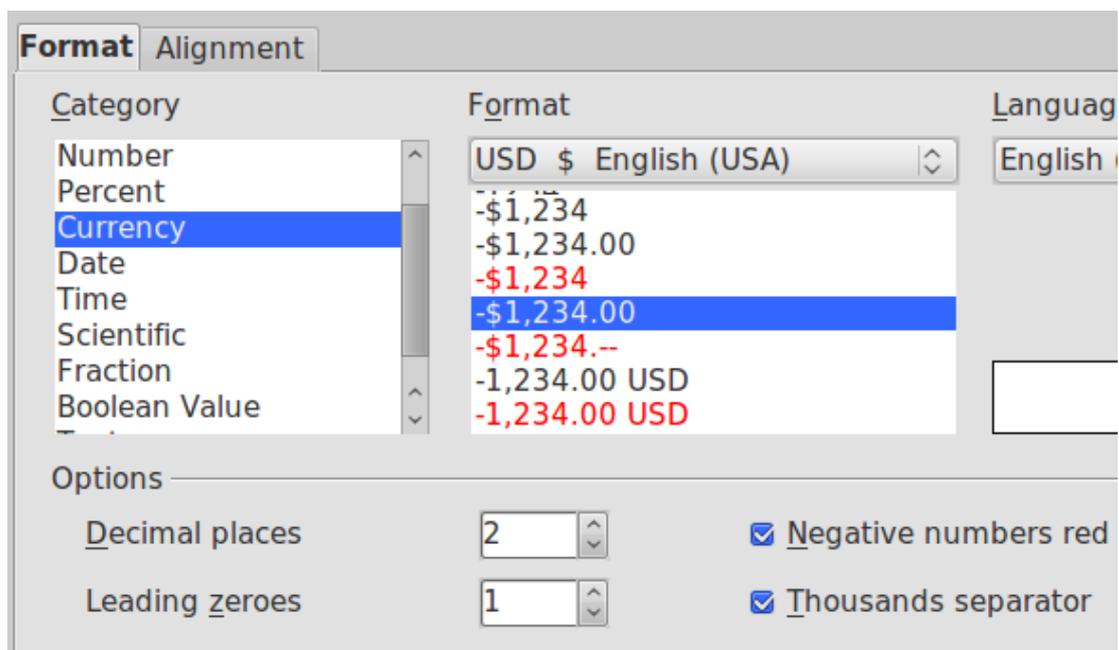


Figure 16: Category and Format selections in the Field Format dialog

- 3) Save the table.
- 4) Close the table.
- 5) Save the database in the main document window.

### Caution



Step 5 is very important that you must do each time you use the Table Design dialog or Table wizard. Step 3 saves the changes made in the Table Design dialog to memory (RAM). Step 5 writes these changes in RAM by writing them to the database file. Before you close the database, you will be warned if you have not saved the database file with all of the changes made to it. If you see this warning, heed it and save the file.

## Create a View

A View is a virtual table or stored query you create using the fields from one or more tables that you have already created. A View allows you to set relationships between tables using selected fields and view the results. View structures are written in the script file in the database folder of the database document file (Queries are written in a different file in another location of the database document file).

When we click **Create View**, the View Design dialog opens. This dialog is very similar to the Query Design dialog. Discussing these dialogs is beyond the scope of this chapter. Both are discussed in detail in Chapter 4 of the *Base Guide*, Data Output.

## Queries: How to use your information

This discussion of the query covers the use of the query wizard although there are other methods for creating queries. Chapter 4 of the *Base Guide* contains a detailed discussion of the more complex methods.

When we are sick, we go to a doctor who gathers data about our condition. We expect the doctor to decide what is wrong with us based upon this data. We would be rather disappointed if the doctor made no diagnosis based upon the data he has.

Your chart in the doctor's office is a database (data source). The knowledge the doctor has from training and education is another database. A diagnosis is a query of his training and education databases for what matches the symptoms you have: the doctor seeks this information. This query will also suggest that some other symptoms may be present. The doctor searches your chart for them: another query.

### Query

A query is a search of the database to locate specific types of information.

Some of this information may be what has been entered. We may want to know the appraised value of an antique lamp. A query can also manipulate the data to obtain desirable information. For example: Based upon the appraised values of the individual items, what is the total appraised value of the entire inventory?

#### Tip

Three tools are available to help you work with your data to get the information you need. The View located in the Table section of the database allows you to view selected data. The Query allows you to both view and manipulate selected data. Finally, the Report allows you to create a text document which uses either a table, a view, or a query from the database for its information.

## Create a query using the Query Wizard

When creating a query, we must consider what we want to know. From this, we decide what field or fields of our database will provide our information. However, unless we are using a relational database, we must make sure that all the fields we want to use are in the same table.

We can create queries using the Query Wizard, the Query Design dialog, or SQL. Later we can also modify a query using the query's design dialog or SQL.

Of these three methods for creating a query, the Query Wizard is best for the beginner. As you make your choices step by step, your query is created without you having to make any decisions other than to choose the things you want the query to do. For simple queries, using the wizard produces good results.

**Tip**

The principles used by the Query Wizard are those that you should use when using the Query Design dialog. If you pay close attention to these principles, you will have few problems with query design.

Since we have only one table, all we need to do is to determine what fields we need for our query. If we had more tables, we would first need to choose the tables we wanted and then what fields we need from these tables.

Queries created by the wizard are fairly simple structurally when using only one table as the source of your fields. While you can use the wizard to create a query containing fields from more than one table, you will have to edit it in Design View or in SQL View to use it.

We will discuss creating a detailed query followed by a summary query.

### Detailed queries

We want a listing of our inventory containing the following information about each item and sorted by their appraised value: the name of each item, its appraised value, the date purchased, and the place purchased. Furthermore, we want our list to contain only those items whose appraised value is greater than \$50. The sorting is to be done from the least valuable to the most valuable.

We have limited the fields for our query to these: Item, AppraisedValue, DatePurchased, and PlacePurchased. We will be using the AppraisedValue field for two additional purposes. We will be sorting our list by this field as well as limiting the list to only those items valued above \$50.

### Step 1 Field selection

- 1) In the database file window, click the Queries icon in the Database section.
- 2) In the Task section, click *Use Wizard to Create Query*. The Wizard opens containing 8 steps (Figure 17).

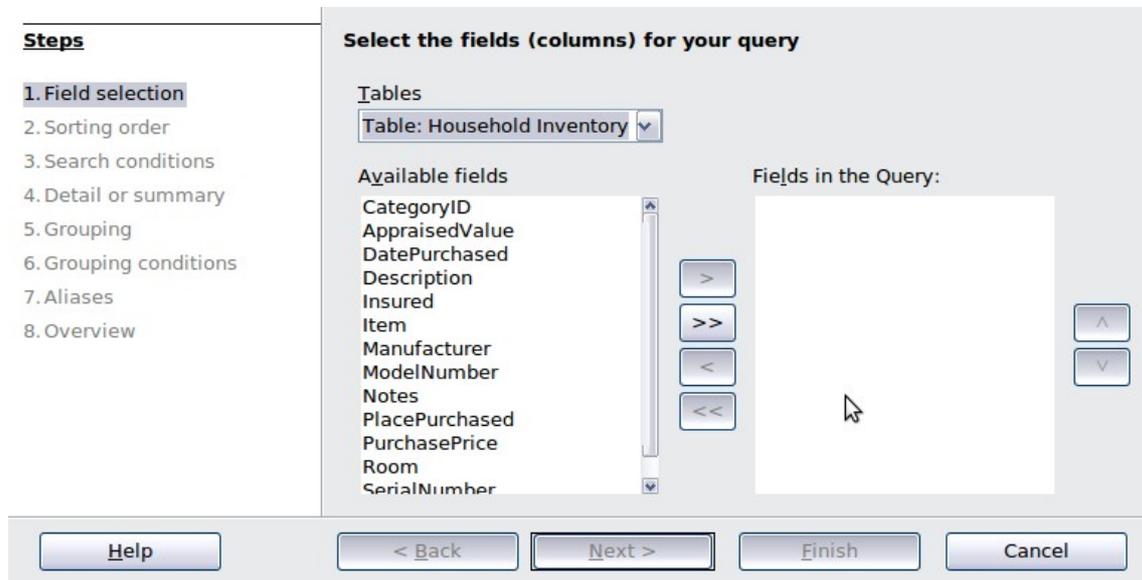


Figure 17: Query Wizard

- 3) Open the drop-down Tables list.
  - Click the table or query you want to use: Household Inventory.

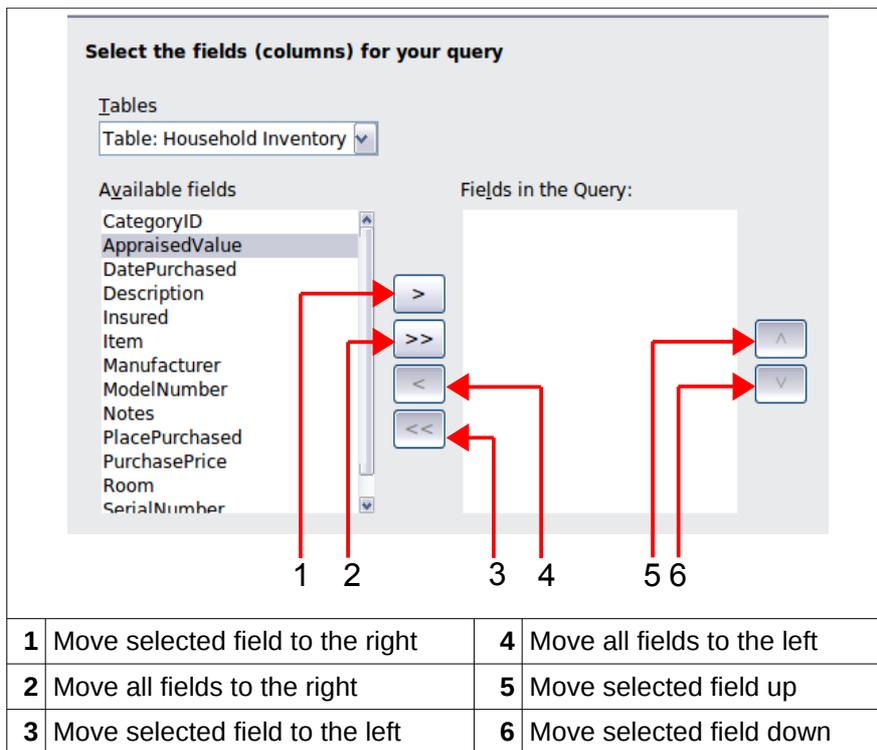


Figure 18: Selection of fields in a query

- 4) Select *Item*, *AppraisedValue*, *DatePurchased*, and *PlacePurchased* fields from the Inventory table.
  - a) Click any of these that are in the Available fields list.
  - b) Click the right arrow to move it to the *Fields in the Query* list.
  - c) Move the rest of the fields mentioned above to the *Fields in the Query* list.
- 5) Change the order of the fields in the *Fields in the Query*.
  - a) Click the field's name that you want to move.
  - b) Use the up arrow to move it higher in the list, or use the down arrow to move it lower.
  - c) Rearrange the order of fields to the order in the *Fields in the Query* list.

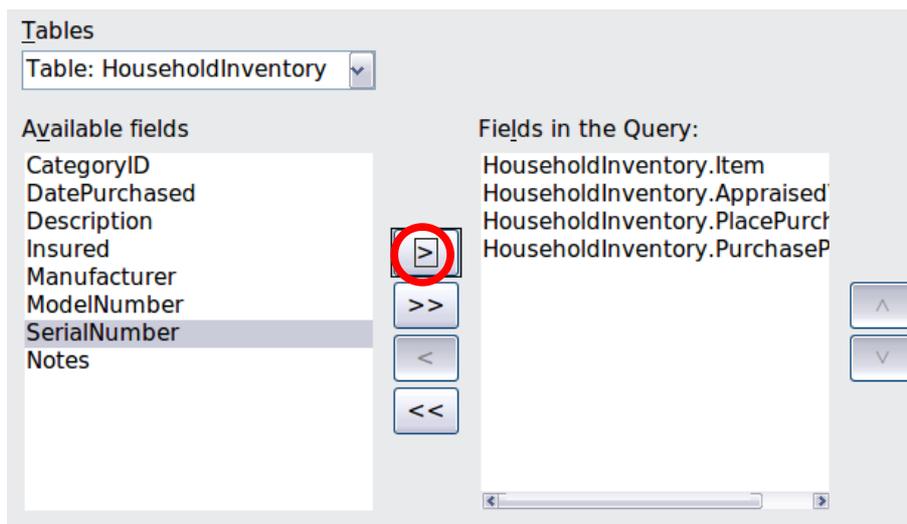


Figure 19: Selected fields for a query

- 6) Click **Next**.

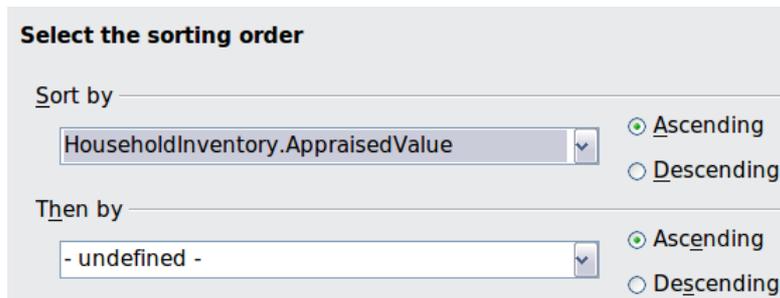
**Tip**

The fields in a query are named using the table name followed by a period followed by the field's name within that table. Hence, *Household Inventory.Item* tells us that this is the *Item* field in the *Household Inventory* table.

**Step 2 Sorting Order**

Sort the data in the query by the *AppraisedValue*.

- 1) Open the *Sort by* dropdown menu by clicking the box labeled *Undefined*.
- 2) Click *AppraisedValue* in this drop-down list to select it (below).



**Select the sorting order**

Sort by   Ascending  Descending

Then by   Ascending  Descending

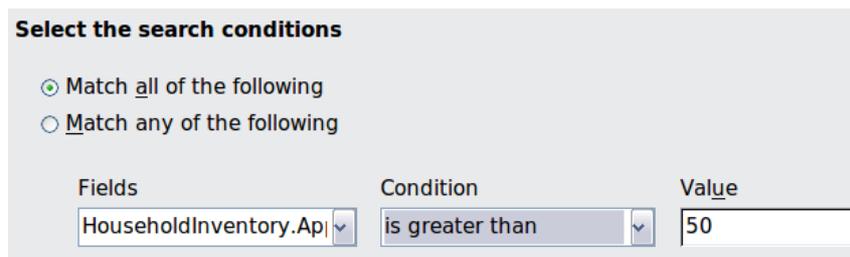
Figure 20: Sorting order selection

- 3) Sorting from least valuable to most valuable is ascending, so the default setting is correct.
- 4) Click **Next**.

**Step 3 Search conditions**

There are many possible search conditions. As an example, we want to see all the data for which *AppraisedValue* is greater than \$50.00.

- 1) Select *Inventory.AppraisedValue* in the first *Fields* drop-down list (below).
- 2) Select *is greater than* from the *Condition* drop-down list.
- 3) Enter 50 as the value.
- 4) Click **Next**.



**Select the search conditions**

Match all of the following  Match any of the following

Fields	Condition	Value
<input type="text" value="HouseholdInventory.Ap"/>	<input type="text" value="is greater than"/>	<input type="text" value="50"/>

Figure 21: Selecting the search conditions

**Step 4 Detail or Summary**

We want a detailed list, so make no changes (Figure 22). Had we wanted the sum, minimum amount, maximum amount, or similar function, we would have selected *Summary query*.

- Click **Next**.

**Select the type of query**

Detailed query (Shows all records of the query.)  
 Summary query (Shows only results of aggregate functions.)

Aggregate functions      Fields

Figure 22: Select type of query

### Step 5 Grouping and Step 6 Grouping Conditions

Grouping is only used with summary queries, so we skip these two steps for now. After Step 8, we will create a summary query that will use grouping.

- Click **Next**.

### Step 7 Aliases

You have the opportunity to give an alias of your choosing for any of the fields in the query. The wizard makes a suggestion for each field. Inventory.Item is given the alias Item, for example. In most cases, it is easier to accept the aliases suggested. We will separate the field names into two words.

- Change AppraisedValue to Appraised Value.
- Change DatePurchased to Date Purchased.
- Change PurchasePrice to Purchase Price.
- Click **Next**.

### Step 8 Overview

**Check the overview and decide how to proceed**

Name of the query:

How do you want to proceed after creating the query?  
 Display Query  
 Modify Query

**Overview**

(Household Inventory.ModelNumber), Notes (Household Inventory.Notes), Place Purchased (Household Inventory.PlacePurchased), Purchase Price (Household Inventory.PurchasePrice), Room (Household Inventory.Room), Serial Number (Household Inventory.SerialNumber)

Sorting order: Appraised Value (ASC)

Search conditions: Appraised Value is greater than 50

No Groups were assigned.

No grouping conditions were assigned.

Figure 23: Overview of a query

Here is where you review the selections you made and name the query. Are the selections listed the same as the ones you thought you made? If not, click the **Back** button until you reach the window you need to correct your error. When you have made the changes, click the **Next** button until you return to Step 8.

- 1) Check the fields selected for the query. (These are correct.)

- 2) Check the sorting order. (These are correct.)
- 3) Check the search conditions. (This is correct.)

If you find incorrect settings, click the **Back** button until you reach the step where you made the mistake and correct it. After that, click the **Next** button until you return to Step 8 Overview.

- 4) Name the query by entering your desired query name in the *Name of the query* box. When giving a query a title, use the name that makes the most sense to you. The Query Wizard suggests Query\_Household Inventory. The use of the word *query* could help some remember that the name tells the type of database object that is named. You may do this, or you may choose not to use the word query as part of your query's name. The choice is yours. For our example, I am going to use the name suggested by the Query Wizard: *Query\_Household Inventory*.

### Note

Base does not permit the use of the same name for a query as for an existing table. Neither can you use the same name for a table as for an existing query. This limitation **only** applies to tables and queries.

- 5) Select what you want to do next.
  - The default setting is to display the query. Accept this setting.
- 6) Click **Finish**.

	Appraised Value	Item	Date Purchased	Purchase Price
▶	75.00	painting	08/01/07	100.00
	75.00	Keyboard	05/16/08	90.00
	125.00	table	10/19/67	825.00
	143.00	TV	03/31/09	249.99
	145.99	Microwave	08/30/11	145.99
	146.00	chair	11/15/00	299.99
	200.00	couch	02/28/00	399.99
	230.00	bed	09/10/77	1199.99
	250.00	Dishwasher	04/20/99	450.95
	351.00	TV	06/30/09	489.99
	400.00	freezer	06/28/03	822.51
	452.00	stove	12/30/99	899.00
	528.00	Refrigerator	12/16/90	1599.99
	550.00	Washer/dryer	03/21/08	884.88
	789.00	couch	01/02/96	1499.99
	1097.00	HP Pavillion	09/04/11	1097.00
	6000.00	Piano	08/12/02	5895.00

Figure 24: Results of running a query (first 17 rows)

Any query is designed to search for specific information while eliminating undesired information from the search. We can apply filters to the query to further reduce the amount of the output that we see. This is done using the tools in the *Table data* toolbar. (See below.)

Some of these tools add additional fields by which the data is sorted. Such sorting is done on the query output. This sorting is in addition to the sorting already done by the query. For example, this query is sorted using the Appraised Value field in ascending order. The only additional sorting that can be done is on the first two rows that have identical values in the Appraised Value column. The rest of the values are in ascending order with no more identical values.

Some of the tools limit the number of rows you will see by limiting the values in given columns.

- 1) The *Sort* tool opens the Sort Order dialog. This dialog permits you to sort the query output by as many as 3 fields. You can also select the direction of the sorting for each of these fields (ascending or descending).

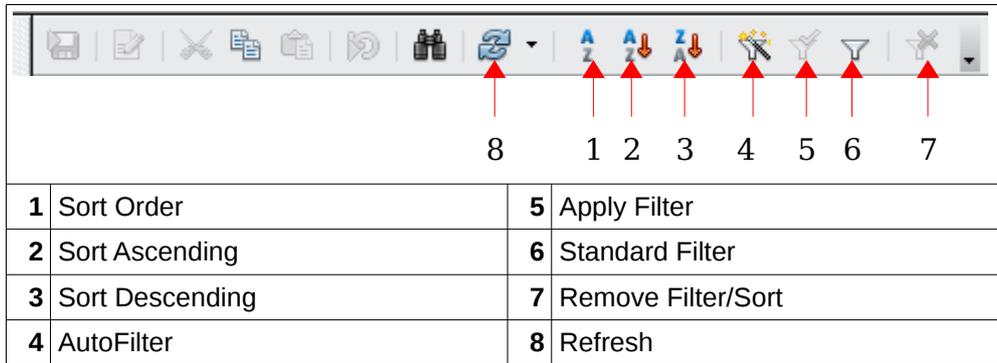


Figure 25: Table data toolbar of the Query Design dialog

- 2) The *Sort Ascending* tool sorts the field you have selected from least to greatest.
- 3) The *Sort Descending* tool sorts the field you have selected from greatest to least.
- 4) The *AutoFilter* tool looks for rows with the same value as contained in the cell you have selected. For example, Figure 24 has \$75.00 selected in the Appraised Value column. If the AutoFilter tool is clicked, only the first two rows of the query output appear. (Both have an appraised value of \$75.00.)
- 5) The *Apply Filter* tool toggles the selected filter on and off. If you just used this tool, you can turn the filter off by clicking the tool again. Then you can apply the filter by clicking the Apply Filter tool again.
- 6) The *Standard Filter* tool uses the Standard Filter dialog. There you can select three fields and enter the conditions for those fields. If more than one field is selected, you must use AND or OR to combine the conditions you have entered.
- 7) The *Remove Filter/Sort* tool removes the filter or sort that has been applied to the query output by the Table data toolbar. It has no effect on the sorting done when the query is run.
- 8) The *Refresh* tool runs the query again. This is usually used after data has been added to the database. If the Refresh tool is clicked after data has been added to one of the fields of the query, the new data will appear in the query output.

Here is an example of using the Standard Filter. We will filter the query to show the rows containing a Purchased Price value greater than \$300.00. The items in the inventory whose purchase price is less than \$300 are no longer listed in the query output.

- 1) Click the Standard Filter tool.
- 2) Select *Purchased Price* from the *Field name* drop-down list.
- 3) Select *>* from the *Condition* drop-down list.
- 4) Enter 300 in the Value box.
- 5) Click **OK**.

	Appraised Value	Item	Date Purchased	Purchase Price
▶	125.00	table	10/19/67	825.00
	200.00	couch	02/28/00	399.99
	230.00	bed	09/10/77	1199.99
	250.00	Dishwasher	04/20/99	450.95
	351.00	TV	06/30/09	489.99
	400.00	freezer	06/28/03	822.51
	452.00	stove	12/30/99	899.00
	528.00	Refrigerator	12/16/90	1599.99
	550.00	Washer/dryer	03/21/08	884.88
	789.00	couch	01/02/96	1499.99
	1097.00	HP Pavillion	09/04/11	1097.00
	6000.00	Piano	08/12/02	5895.00

Figure 26: Applying a filter to Figure 21 above

- Close Query\_Household Inventory without saving it. (We do not want to save the filter we just created.)

### Summary query

Summary queries are very similar to detailed queries, but their goal is to manipulate the data to produce the information you want. The only difference is in steps 4 through 6.

For our example of a summary query using grouping, we will use these fields: Appraised Value, Purchase Price, and Room. The question we seek to answer for our query: What is the total appraised value and total purchased value for each room in the house?

This requires 3 fields: Appraised Value, Purchase Price, and Room. This also requires using the Sum function with 2 fields: Appraised Value and Purchase Price. The sums will be grouped by the field Room.

#### Step 1 Field selection

- 1) In the database file window, click the Queries icon in the Database section.
- 2) In the Task section, click *Use Wizard to Create Query*. The Wizard opens containing eight (8) steps.
- 3) Click *Table: Household Inventory* in the Table list.
- 4) Select *Item, AppraisedValue, Purchase Price, and Room* fields from the Inventory table.
  - a) Click one of these that are in the Available fields list.
  - b) Click the right arrow to move it to the *Fields in the Query* list.
  - c) Change the order of the fields in the *Fields in the Query*
    - i. Click the field's name you want to move.
    - ii. Use the up arrow to move it higher in the list, or down arrow to move it lower.
    - iii. Rearrange the order of fields if necessary. (AppraisedValue, then PurchasePrice, then Room)

#### Step 2 Sorting Order

Sort the data in the query by the *Room* field.

- 1) Open the *Sort by* drop-down menu by clicking the box labeled Undefined (Figure 20).
- 2) Click *Room* in this drop-down list to select it.
- 3) Sorting: use the Default value (ascending).
- 4) Click **Next**.

### Step 3 Search conditions

There are many possible search conditions. As an example, we want to see all the data for which the Appraised Value is greater than \$100.00.

- 1) Select Inventory.AppraisedValue in the first Fields drop-down list (Figure 21).
- 2) Select *is greater than* from the Condition drop-down list.
- 3) Enter 100 as the value.
- 4) Click **Next**.

### Step 4 Detail or Summary

- 1) Select the option button for *Summary query* (Figure 22).
- 2) Click **Next**.

### Step 5 and Step 6: Grouping and Grouping Conditions

While these steps are listed, they are not used. Instead the wizard uses the information contained in steps 2 and 3. When *Summary query* is selected in Step 4, information in step 2 determines how the data is grouped and information in step 3 becomes the grouping conditions.

### Step 7 Aliases

We will separate the field names that can be separated into two words.

- Change AppraisedValue to Appraised Value. (Put a space between words.)
- Change PurchasePrice to Purchase Price.
- Click **Next**.

### Step 8 Overview: Check the information in the Overview box for accuracy.

**Check the overview and decide how to proceed**

Name of the query:

How do you want to proceed after creating the query?

Display Query

Modify Query

**Overview**

Fields in the Query: Appraised Value (Household Inventory.AppraisedValue), Purchase Price (Household Inventory.PurchasePrice), Room (Household Inventory.Room)

Sorting order: Room (ASC)

Search conditions: Appraised Value is greater than 100

Grouped by: Room

No grouping conditions were assigned.

Figure 27: Overview of a summary query

Note that the Sorting order is Room (ASC) and Grouped by is Room. Also, the Search conditions refers to Appraised Value while the query applies the condition (greater than 100) to SUM (Appraised Value). What the query does is make this condition a part of the grouping conditions.

- 1) Check the fields selected for the query. (These are correct.)

- 2) Check the sorting order. (These are correct.)
- 3) Check the search conditions. (This is correct.)
- 4) Check the group by condition. (This is correct.)
  - If you find any incorrect settings, click the Back button until you reach the step where you made mistakes. After correcting it, click the **Next** button until you return to Step 8 Overview.
- 5) Name the query by entering a name in the *Name of the query* box. For our purposes we will use Query\_Household Inventory Summary.
- 6) Select what you want to do next.
  - The default setting is to display the query. Accept this setting.
- 7) Click **Finish**.

The results are shown in the figure below. Each row contains the total Appraised Value and the total Purchase Price for one of the rooms in the house. The only problem is that we don't know what row of values applies to what room. The Query Wizard will not give us this information. However, editing this query using the Query Design dialog, we can add a third column containing the names of the rooms. Then we will know the total appraised value and total purchase price for each room.



	Appraised Value	Purchase Price
▶	328	1580.27
	1117	1132.99
	280	499.99
	364	639.98
	125	825
	1775.99	3918.44
	550	884.88
	1215	2089.98
	6000	5895

Figure 28: Summary query

- 8) Close Query\_Household Inventory Summary.

Filters can be applied to a summary query just as we applied a filter to the detailed query above. but they are only applied to the values shown in the query output. The query output for this query is the total appraised values and total purchase prices. Filtering can be done only on these total values. The rows you see are sorted using the *in ascending order of values* of the Room field.

### Caution



When you have created a filter for a query, you can only remove it by opening the query (double-clicking the query or right-clicking the query and selecting Open) and then clicking the Remove Filter/Sort icon (#7 in Figure 25). Editing the query will not reveal the existence of the filter, nor is this evident when editing the query in the SQL View. If the query results do not look as you expected them, open the query and look for this icon. If the X is red in the icon, the query is being filtered. In that case, click the icon to remove the filter.

## Create a query using Design View

Queries can be created by clicking *Create Query in Design View* in the Task section of the database file window. This opens the Query Design dialog.

The use of this dialog is beyond the scope of this chapter. It is discussed in detail in Chapter 4 of the *Base Guide*, Data Output.

Any query can be edited just as a table can be edited once it is created. One way to edit a query is to open the query using the Query Design dialog. Another way is to use the SQL Statement dialog. Again, both of these ways are discussed in Chapter 4 of the *Base Guide*.

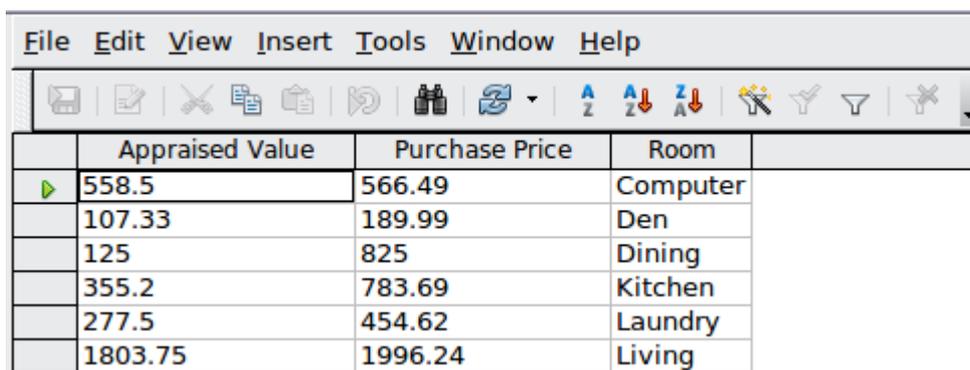
The query wizard does have a limitation. In the Figure 28 , we have two columns of numbers. While the data is sorted by the names of the rooms in the house, we don't see any reference to the rooms. We don't know to what room each row applies. We are going to modify this summary query so that the names appear. To do so, we will edit the query.

- 1) Right-click *Query\_Household Inventory Summary*.
- 2) Select **Edit** from the context menu. The Query Design dialog opens *Query\_Household Inventory Summary*.

Field	AppraisedValue	PurchasePrice	Room	AppraisedValue
Alias	Appraised Value	Purchase Price		
Table	Household Inventory	Household Inventory	Household Inventory	Household Inventory
Sort				
Visible	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Function	Sum	Sum	Group	Sum
Criterion				> 100

Figure 29: Table at the bottom of the Query Design dialog

- 3) Go down the column labeled Room until you come to the Visible row.
- 4) Click the checkbox to check it. This will make the values of the Room field visible in the query output.
- 5) Save and close the query.
- 6) Double-click the query to run it. See below for the output.



	Appraised Value	Purchase Price	Room
▶	558.5	566.49	Computer
	107.33	189.99	Den
	125	825	Dining
	355.2	783.69	Kitchen
	277.5	454.62	Laundry
	1803.75	1996.24	Living

Figure 30: Modified query output

Now we can see the sorting contained in the query. Each row now makes more sense.

## Create a Query using SQL

In the main database window, click the Queries icon in the Database section, and look at the choices in the Tasks section. You see three choices there. The third choice is *Create Query in SQL*.

Clicking this choice opens the Query Design dialog, the same dialog you opened to modify the query. The Query Design dialog has two views: Design View and SQL View. Either view can be used to modify or create a query, and a tool in the Query Design toolbar permits us to go from the Design View to SQL View and back to Design View again.

Queries can be created in another way using SQL. Use **Tools > SQL...** while in the database file window. This opens the SQL Statement dialog. You enter your SQL statements into this dialog.

How to use these two dialogs is discussed in Chapter 4 of the *Base Guide*, Data Output.

## Forms

If we had to enter all of our data directly into our table, it would be more time consuming than it needs to be. By creating a form based upon the table we created, we can enter our data in a more meaningful way. When we look at a table, we see many rows of data simultaneously. The form allows us to see only one row (one record) at a time.

### Create a simple form using the Form Wizard

The Form Wizard guides us through some fairly easy steps for creating a form based on the table we have already created. Once the form is created, some modifications are needed that will be much more time consuming than the creation of the form itself. These modifications are necessary to create the appearance we want. (More complex modifications to the form are discussed in Chapter 3 of the *Base Guide*.)

Here are two ways to open the wizard. Use 1a) **OR** 1b). Do not use both.

- 1) Open the wizard to create a form.
  - a) Click the *Table* icon in the Database section of the main database window.
    - i. Right-click the table you want to use. (Right-click the Inventory table.)
    - ii. Select *Form Wizard* from the context menu.
  - b) **OR**, click the *Forms* icon in the Database section of the main window
    - In the *Task* section, click **Use Wizard to create form**

#### Page 1: Select fields (below)

- 1) Select the table, *Inventory*, in the *Tables and queries* drop-down list.
  - a) If you right-clicked the Inventory table and selected *Form Wizard* from the context menu, *Table: Inventory* will already be selected. **OR**
  - b) If you had earlier clicked **Use Wizard to create form**, select *Table: Household Inventory*.

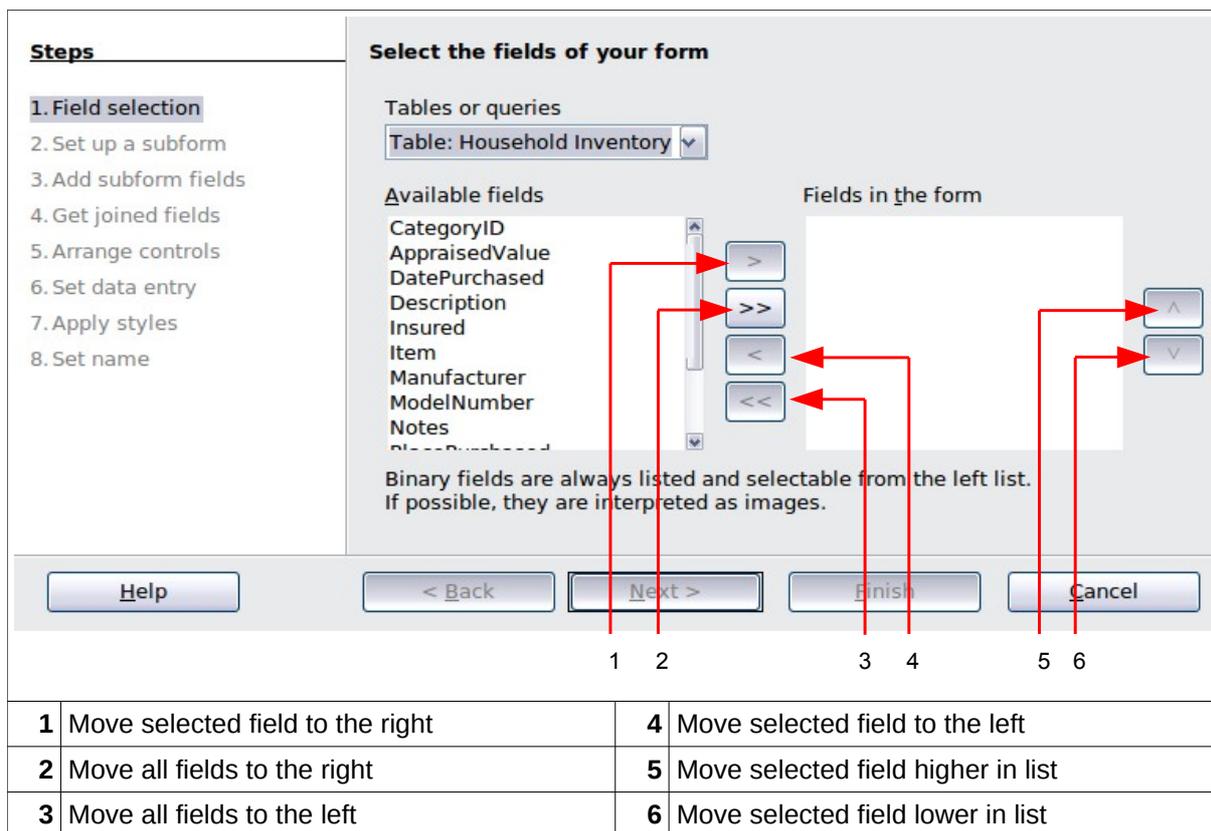


Figure 31: Page 1 of the Form Wizard

- c) Choose fields from the *Inventory* table to be used in the form.
- i. Since we want to use all of the fields, click the right double arrow to move all the fields listed under *Available fields* to the *Fields in the form* list.
  - ii. **OR**, if you only want to move some of the fields:
    - Click the field you want to move.
    - Click the right arrow to move this field.
    - Repeat for the rest of the fields to be moved.
- d) Click **Next**.

### Page 2: Set up a subform

- We will not use another table along with the Household Inventory table. So, we will not be using a subform.
- Click **Next**.

### Page 3, Add Subform fields, and Page 4, Get joined fields

Both of these pages only apply when you are using a second table to create a subform. Since we did not select the second table in page 2, pages 3 and 4 are grayed out.

### Page 5: Arrange the controls on your form (below)

#### Control

A *Control* consists of a label and field. The *label* contains the name of the control, and the *Field* contains the data entered into the form.

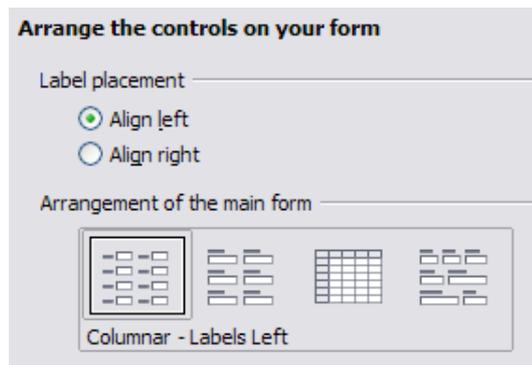


Figure 32: Arranging the controls of the form

You have four choices in arranging your controls as shown in the figure above. From left to right: *Columnar – Label left*, *Columnar – Label on top*, *As Data Sheet*, and *In Blocks – Labels Above*. Each choice gives a slightly different layout.

Both of the columnar layouts create the controls in columns beginning with the first control at the upper left and continuing down the left side. If additional columns are needed, they are filled in from top to bottom and then columns from left to right. The first control is the top control in the left column. The last control is the bottom control in the column on the right.

The *Data Sheet* arrangement looks like a spreadsheet with the labels across the top. This is probably best used for a subform rather than for the main form.

The *In Blocks – Labels Above* arrangement places the controls in rows from left to right. When one row is filled, the next control is placed at the left end of the next row below the row that is filled. So, the first control is at the left end of the top row, and the last control is at the right end of the bottom row.

We will use *Columnar – Labels Left* for the Household Inventory form.

- 1) *Label placement*: I prefer to align the names of the field to the left, so I will do so. If you prefer to align the names of the fields to the right, do so.
  - Accept the Default value of *Align left*, or click the bottom option button to select *Align right*.
- 2) *Arrangement of the main form*: click the graphic on the left. (See Figure 32 for the correct appearance of your window.)
- 3) *Arrangement of the subform*: we are using only one table in this form, so this is grayed out.
  - Click **Next**.

#### Page 6: Select the data entry mode

- The default choice match our needs.
- Click **Next**.

#### Tip

I do advise you look at the choices available, thinking about how each of these choices might meet your future needs.

#### Page 7: Apply the style of your form

- 1) *Apply styles*: page color
  - a) Chose one of the listed colors. The default color is as good as any. (This color can be changed when we modify the form.)

- b) You can create your own color using **Tools > Options > LibreOffice > Colors**. You can use RGB or CMYK values in this dialog. After creating the color you want, you can select your color while modifying the form.
- 2) *Field border*:
  - a) Experiment with the three choices.
  - b) Use the mouse to move the Form window around so that you can see each of the three choices in the form.
  - c) We will use the default value. (This choice can be changed when we modify the form. We will then be able to change the border of the control labels as well.)
- 3) Click **Next**.

## Page 8: Set the name of the form

- 1) The suggested name for the form is the same as the table used to create the form: *Household Inventory*. While we will use the suggested name this time, it is your choice as to what name you use. You can replace the name if you like.
- 2) *How do you want to precede after creating the form?*
  - If you want to add data immediately after creating the form, use *Work with the form*. Select this choice for this example.

---

**Tip**

We almost always have some modifications to make, so *Modify the form* is usually the best choice.

---

- 3) Click **Finish**.
  - a) Since we chose *Work with the form* above, the Household Inventory form opens in the “read only” mode, as in Figure 36. You can start inserting data into your table.
  - b) If you had chosen *Modify the form*, the Household Inventory form would have opened in edit mode. Modifying a form is discussed in Chapter 3 of the *Base Guide*.

---

**Tip**

“Read only” is confusing to many people. In Edit mode, you can modify anything that you earlier created: controls, background, and text you may have added to the page. In this mode, you can not add nor remove any data previously entered into any of the fields unless you click the Design Mode On/Off icon to turn the design mode off. In the “read only” mode, none of the controls nor styles of the form can be changed. In this mode, only data can be added, deleted, or modified in any of the fields. To repeat: “edit mode” is used to change the controls and appearance of the form, and “read only” is used to add or modify the data contained in the controls.

---

Figure 33: Newly created Household Inventory form

### Create a form using Design View

Forms can be created using Design View. But like creating tables or queries in Design View, creating a form in Design View is more complex than using a wizard. See Chapter 3 of the *Base Guide*, Data Input and Removal, for instructions.

### Reports

Reports present information found in the database in a useful way. In this respect they are similar to queries. Reports are generated from the database's tables or queries. They can contain all of the fields of the table or query or just a selected group of fields. Since the data that concerns us is contained in a query, we will use that.

Reports can be static or dynamic. Static reports contain the data from the selected fields at the time the report was created. Dynamic reports can be updated to show the latest data. If you are absolutely sure that data for your report does not change with time, creating a static report is a good choice. A report of your inventory's total appraised value for the year 2008 is an example of a static report. (It is very unlikely that the data for 2008 will change.) But a similar report for your insurance agent for updating your policy needs to be a dynamic report. This report needs to contain any purchases made since the last report was made. When you purchase or replace an item, this report can tell you whether you even need to contact your agent or not.

## Use the wizard to create a report

The report, which we will create, is based upon *Query\_Household Inventory*. We will look for all items listed in this query.

You should now be in the main database window with the list of database objects (Tables, Queries, Forms, and Reports) on the left.

### Step 1 Field Selection (Figure 34)

- 1) In the main database window, click the Report icon in the Database section.
- 2) In the Task section, click **Create Report with Wizard**. The Report Wizard opens.

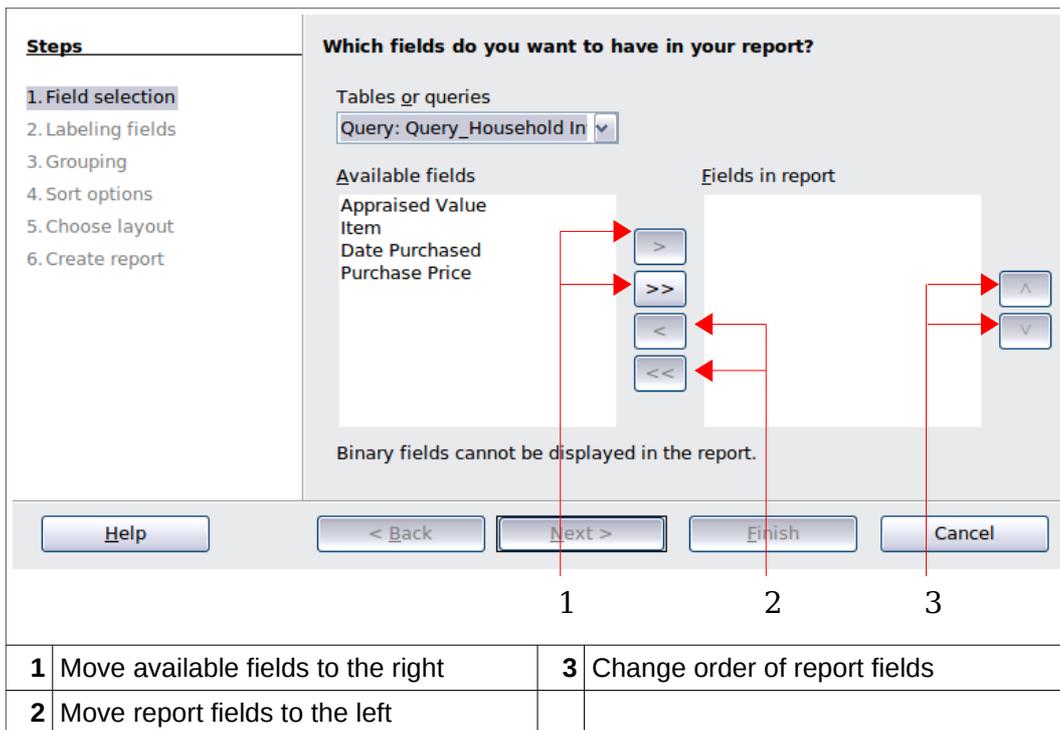


Figure 34: Report Wizard Field Selection

### Tip

- 1) When the wizard opens, a text document opens behind it. This document contains the title, author, and date in the upper left corner. The bottom right corner contains Page 1/1. This shows you what page of the report you are seeing and how many pages are contained in the report. Once the report is created, for example a 3 page report, you will see 1/3, 2/3, 3/3 depending upon what page is shown.
- 2) Since Query comes before Table alphabetically, the queries are listed first in this list. Similarly, the queries are also listed alphabetically in relation to each other. Thus, *Query\_Household Inventory* is the first alphabetically. It is already selected. However, when creating a report based upon a query, most of the time you will have to search through the list to find the one you need.

Under *Tables or queries*, you have a drop-down list containing the tables and queries you have created. Select *Query\_Household Inventory*.

- 1) Move the list of *Available fields* to the list of *Fields in report* using the double right arrow button.

- 2) If you want to change the order of the fields in the *Fields in report* list, click the field you want to move up or down in this list. Click one of the arrows (#3 in Figure 34) to move the field up or down to where you want it.
- 3) Click **Next**.

## Step 2 Labeling fields

Here you can give aliases to one or more fields. The report suggests using the name of the fields of the query. If you have added the aliases you wanted when creating the query, you will not have to make any changes in this step. If, however, you want to make some changes, now is the time to make them.

- 1) Review the suggested labels for the fields. Change them if you think your name is better.
- 2) Remember to use the vertical scrollbar if all of the fields are not visible, and then review the rest of the suggested labels.

We added aliases for these fields in the query earlier: AppraisedValue, DatePurchased, and PurchasePrice. So, the report will use the aliases of the query instead: Appraised Value, Date Purchased, and Purchase Price respectively.

- 3) Click **Next**.

## Step 3 Grouping

- 1) Click Date Purchased in the Fields list.
- 2) Click the right arrow button to move it to the Groupings list.
- 3) Make sure your lists match those below. If a field is in the wrong list, select it and use the right or left arrow button to move it the the correct list.

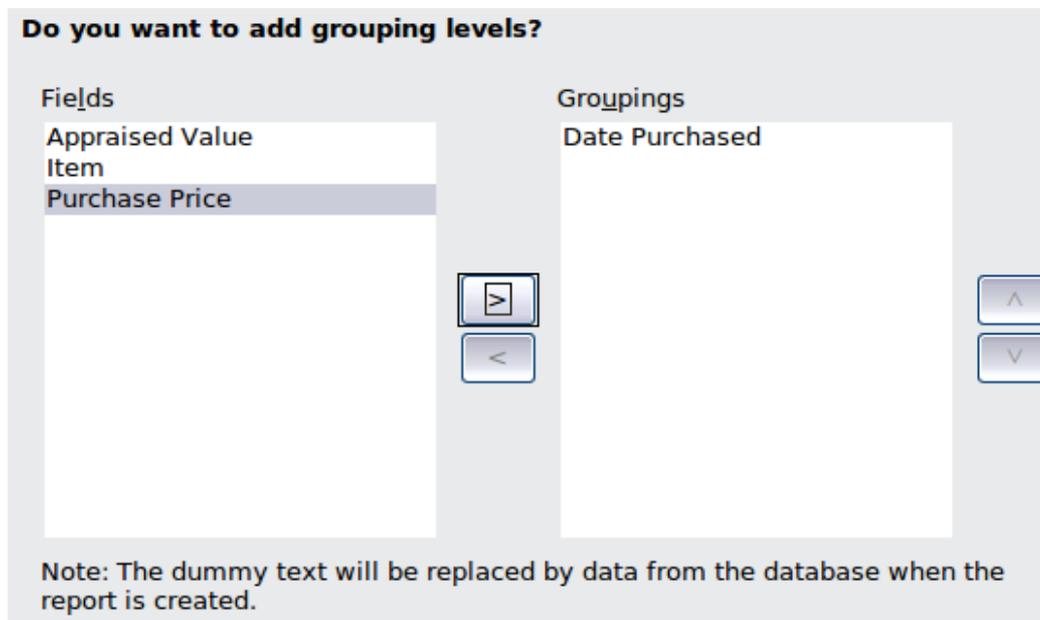


Figure 35: Grouping fields for a report

- 4) Click **Next**.

## Step 4 Sort options (Figure 36)

By selecting how the report should group the data, we have already selected this field, Date Purchased, as the first sort option. We want the sorting order to be ascending. We will not select more.

- 1) If you want to sort by more fields of the report, do so now.
  - a) Click the drop-down list below *Then by* to open the list.
  - b) Select the field you want to use to sort the report by. (This sorting will be in addition to the sorting by the *Date Purchased* field.)
    - Suggestion: select *Purchase Price* as the second field by which to sort the data.
- 2) Click **Next**.

Figure 36: Selecting Sort options

### Step 5 Choose layout (below)

There are only two choices of page orientations: Landscape and Portrait. Which one you use depends upon the number of fields contained in the report. A large number of fields requires the Landscape orientation. A small number of fields will probably be best using Portrait. For now, choose Landscape.

- 1) Choose Landscape as the layout for the report.

Figure 37: Layout choices

There are a large number of choices for layouts of the data: enough to require a vertical scrollbar. There are fewer choices for the types of headers and footers. When you select one of the layouts, the text document in the background changes to show you what your choice will look like. (Figure 38 has the default values. I selected *Bubbles* as the *Layout of headers and footers*.) I am using *Align Left - Border* for the data layout. (Figure 39)

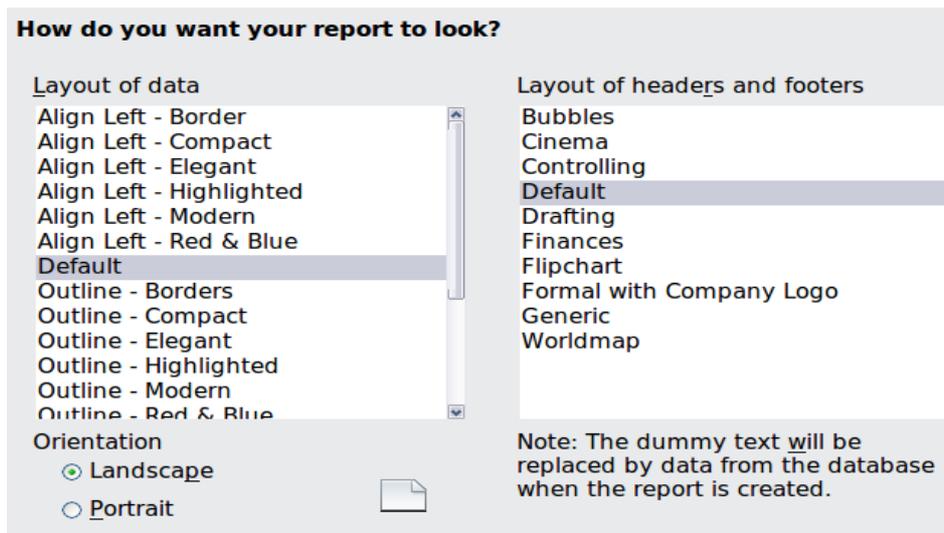


Figure 38: Layout choices

Default		
Author: Dan Lewis		
Date: 9/12/11		
Date Purchased	42510	
Appraised Value	Item	Purchase Price
9876.5	Ut wisi enim ad minim veniam, quis nostrud exerci	9876.5

Figure 39: Align Left-Border Bubbles layouts

**Tip**

As you are creating your report, the text document behind the Report Wizard is changed accordingly. This permits you to see the results of your choices. Then you can change what you don't like to something you like better.

- 2) Select the layout of the data which you want to use. (For this example report use the *Align Left - Border* choice.)
- 3) Select either orientation. I recommend Landscape.
- 4) Select the layout of headers and footers that you want to use. (For this example report use the *Bubbles* choice.)
- 5) Click **Next**.

**Step 6 Create report (below)**

Name the report. The suggestion given you is *Query\_Household Inventory*. This is the same name as the query we used to create this report. Change Query to Report.

**Decide how you want to proceed**

Title of report

What kind of report do you want to create?

Static report

Dynamic report

How do you want to proceed after creating the report?

Modify report layout

Create report now

Figure 40: Final options before creating a report

- 1) A dynamic report is a better choice than a static report especially since you might want to make some changes in the underlying query. Every time you buy something or get rid of something, the inventory changes. The household inventory report needs to change with changes in the query output.
- 2) Click **Finish**.

You have created the report; it opens looking similar to Figure 41. This is what the first page of the report looks like after data has been added to the database.

<b>Author: Dan Lewis</b>		
<b>Date: 5/5/11</b>		
<b>Date Purchased</b> 24764		
<i>Appraised Value</i>	<i>Item</i>	<i>Purchase Price</i>
125	table	825
<b>Date Purchased</b> 28378		
<i>Appraised Value</i>	<i>Item</i>	<i>Purchase Price</i>
230	bed	1199.99
<b>Date Purchased</b> 33223		
<i>Appraised Value</i>	<i>Item</i>	<i>Purchase Price</i>
528	Refrigerator	1599.99
<b>Date Purchased</b> 35066		
<i>Appraised Value</i>	<i>Item</i>	<i>Purchase Price</i>
789	couch	1499.99
<b>Page 1/4</b>		

Figure 41: Final Report

## Modify a report

A report can be edited after it is created to make a variety of modifications. For example, some of the changes that can be made are the report's date format, adding a report title, the Date Purchased date format, converting the numbers in the Appraised Value and Purchase Price into currency, and the column widths.

You use the Report Design dialog to edit the report. When opened it contains 9876.5 in columns that contain numbers, meaningless words for columns containing text, and a five digit Integer for dates in columns. But when the report is opened, the columns contain the data for the report as shown in Figure 41.

### Editing the report

- 1) Right-click Report\_Household Inventory in the Report list.
- 2) Select *Edit* from the context menu. The Report Design dialog opens. (Figure 39 shows the top part of the layout within the dialog.)

### Changing the report's date format

Changing the report's date format is different from changing the date format of a date field in the data section of the report. (The latter change will be discussed next.)

- 1) Double-click the date of the report. This opens the Edit Fields: Document dialog.
- 2) Select the date format from the Format list (right column). Use the scrollbar to find your choice if necessary. (I have selected Friday, December 31, 1999.)
- 3) Click **OK** to change the date format.

#### Tip

The bottom choice of the Format list is *Additional formats...* If you select this, the Number Format dialog also opens (Figure 42). You can use this dialog to create your own date format. Click the Help button to find out how to do this. When you select the format you want to use in the Number Format dialog, click **OK** in the Number Format dialog. Then click **OK** in the Edit Fields: Document dialog.

### Change the format of a date field in the report

- 1) Right-click the date field (Date Purchased, in our case).
- 2) Select *Number Format...* from the context menu. The Number Format dialog opens.

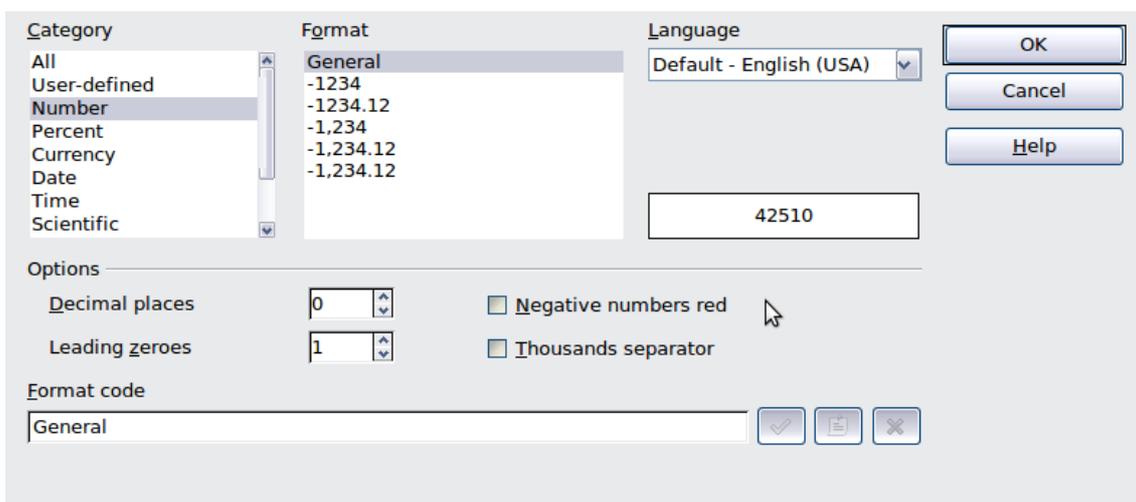


Figure 42: Number Format dialog

- 3) Select the date format you want.
  - a) Select Date in the Category list.
  - b) Select the format you want from the Format list. (I have selected Fri, Dec 31, 99.)  
If you do not like the available choices in the Format list, use the *Format code* box to customize your date format.
  - c) Click **OK** to apply the date format.

### Changing numbers into currency

- 1) Right-click the number you want to change. (You will have to do this for each column having a number.)
- 2) Select *Number Format...* from the context menu. The Number Format dialog opens.
- 3) Select Currency from the Category list.
- 4) Select the number format from the Format list or use the Format code to create your own format.
- 5) Click **OK** to apply the currency format.

#### Tip

- 1) Other numbers can be formatted in a similar way as currency. Open the Number Format dialog, and change the number's format.
- 2) If you know how to use styles (*F11* key opens the Styles and Formatting window), you can modify the text in the report including text in the header of the report.

### Adding a title to the report

- 1) Click to the right of the report author's name.
- 2) Use the *Tab* and *Space* keys to place the cursor where you want the title of the report to begin.
- 3) Type the name of the report.
- 4) Save the Report Design dialog, and then close it.

<b>Author: Dan Lewis</b>		<b>My Household Inventory</b>	
<b>Date: Monday, September 12, 2011</b>			
<b>Date Purchased</b> Fri, May 20, 16			
<b>Appraised Value</b>	<b>Item</b>	<b>Purchase Price</b>	
\$9,876.50	Ut wisi enim ad minim veniam, quis nostrud exerci	\$9,876.50	

Figure 43: Report Design dialog with modifications

Author: Dan Lewis      My Household Inventory		
Date: Monday, November 14, 2011		
<b>Date Purchased</b> Thu, Oct 19, 67		
<i>Appraised Value</i>	<i>Item</i>	<i>Purchase Price</i>
\$125.00	table	\$825.00
<b>Date Purchased</b> Sat, Sep 10, 77		
<i>Appraised Value</i>	<i>Item</i>	<i>Purchase Price</i>
\$230.00	bed	\$1,199.99
<b>Date Purchased</b> Sun, Dec 16, 90		
<i>Appraised Value</i>	<i>Item</i>	<i>Purchase Price</i>
\$528.00	Refrigerator	\$1,599.99
<b>Date Purchased</b> Tue, Jan 2, 96		
<i>Appraised Value</i>	<i>Item</i>	<i>Purchase Price</i>
\$789.00	couch	\$1,499.99
<b>Date Purchased</b> Tue, Apr 20, 99		
Page 1/4		

Figure 44: Final modified report page 1 of 4

### Use Report Builder to create a report

Report Builder is an extension to Base. It is included with many LibreOffice installations or can be downloaded and added to LibreOffice. Using Report Builder, you can create a much more complex report than when using the Wizard.

Lulu ([www.lulu.com](http://www.lulu.com)) sells a book by Dmitri Popov on how to use the Sun Report Builder Guidebook. The LibreOffice Help file contains some basic information for the the Report Builder.

#### Tip

Queries and reports are discussed in much greater detail in Chapter 4 of the *Base Guide*, Data Output.

## Let Wizards do most of the work

Wizards are designed to do the basic work. Design View dialogs build upon what the wizards have already done. SQL builds upon what the Design View dialogs have done.

The first step is to learn how the wizards work. This will give you a foundation of knowledge when you later decide to use a Design View dialog from the beginning. Then again, learning the principles used by a Design View dialog will give you a good foundation for using SQL.

How we do these things is subjective. One person finds it easier to build a database using one method while another person prefers a different method.

Because our database is fairly simple, it can be created using just the wizards with only minor adjustments. This is why this is the starting point. Once you can create a database using the wizards, you can learn how to replace one wizard at a time using a more complex set of

instructions. Or, you can use the wizards first, modify the database using Design View dialogs second, and do the final modifications using SQL third.

## Data input and removal

You have created the table, query, form, and report. So we have given our database its structure. It is a simple DBMS (database management system).

To use our DBMS, we need to add data to it. Since we may make an error in entering the data, we want to be able to modify the data or even delete it. (Circumstances will determine which of these we will need to do.)

The database we have been creating, Inventory.odt with added data, is available on line at:

<http://wiki.documentfoundation.org/Documentation/Publications>. You may wish to download it and compare what you have created with it.

### Note

This section contains basic instructions for adding and removing data. Chapter 3 of the Base Guide discusses this in more detail.

## Enter data in the table

### Tip

The discussion that follows begins with empty tables. So first you will need to follow the directions for creating tables. Then you can try out what is described.

Open the table and enter the data in the cells. Each row contains thirteen cells. The column headers contain the names of the fields. CategoryID will have <AutoField> in the first row. It will automatically fill in its value in the first row when you move to the second row. Always begin with the second cell of the second row after you have completed entering data in the cells of the first row when the first cell has <AutoField> in it. Enter the data in the table as you would in a spreadsheet, making sure the data in a column belongs to the field named at the top of the column. Use the data in Table 8 to enter into the Household Inventory table. When you have entered the data, save and close the table.

### Tip

To move from one cell to the next one, use the *Enter* or *Tab* keys. Either key will also move from the last cell of a row to the first cell of the next row. To move in the opposite direction, use *Shift+Tab*.

- 1) Double-click the table's name in the Table List section. (For our example, the table's name is Household Inventory.) The table opens with columns and no data. The table below shows the open table. The 13 columns are shown in two parts for convenience.

Table 7: Household Inventory table before data is added.

	CategoryID	AppraisedValue	DatePurchased	Description	Insured	Item	Manufacturer
	<AutoField>				<input type="checkbox"/>		
ModelNumber	Notes	PlacePurchased	PurchasePrice	Room	SerialNumber		

- 2) In the first row click the cell in the AppraisedValue column.
- 3) Enter the data for this cell from table 8.

- 4) Use *Tab* or *Enter* key to move to the next column of the first row.
- 5) Repeat steps 3 and 4 until you reach the Insured column.
- 6) Use *Shift* to make a full check mark. (This usually requires using the *Shift* key twice.)
- 7) Use *Tab* or *Enter* key to move to the next column of the first row.
- 8) Repeat steps 3 and 4 until you reach the last column.
- 9) Use *Tab* or *Enter* key twice to move to the AppraisedValue column of the second row.
- 10) Enter the rest of the data from Table 8.
- 11) Close the table.

Table 8: Data to be entered into the Household Inventory table

AppraisedValue	DatePurchased	Description	Insured	Item	Manufacturer
75.00	05/16/08		<input checked="" type="checkbox"/>	Keyboard	Casio
1097.00	09/04/11	computer w/pri	<input checked="" type="checkbox"/>	HP Pavill	Hewlett Packard
6000.00	08/12/02	upright	<input checked="" type="checkbox"/>	Piano	Steinway
125.00	10/19/67	round	<input checked="" type="checkbox"/>	table	Woodworks, Inc.

ModelNumber	Notes	PlacePurchased	PurchasePrice	Room	SerialNumber
CTK-720		Target	90.00	Den	14268941235
a6503f		Best Buy	1097.00	Computer	KQ946AA#ABA
2002STE4556		Classical Instrument	5895.00	Living	STE112211
WW126	6 chairs	Sonny's Furniture	825.00	Dining	WW2113

## Modify data from a table

This process is very simple. Open the table containing the data. Click the cell containing the data. Change the data just as you would change data in any text box. Save the table. Finally, close the table.

## Delete data from a table

Removing data from a table is done the same way as you modify data in a table. Rather than changing the data, you are erasing data.

Deleting a record from a table is different. You are removing an entire row from the table at one time.

### Caution



Before deleting a record, make sure you want to delete the data in all the fields contained in that record. Once you delete the record, its data is gone forever. If you decide you needed at least some of the data you deleted with the record, you will have to add this data again.

As an example, we will delete one of the records we just created. If you use this table to store your household inventory, you can modify the remaining records so that it applies only to your household items.

Delete a record:

- 1) Open the *Household Inventory* table.
- 2) To delete a record, right-click the gray box to the left of the record you want to delete.

- a) To delete the first record, right-click the box to the left of the value 0 in the CategoryID column. A green arrow appears in the box, and the entire record is highlighted (below).

	CategoryID	AppraisedValue	DatePurchased	Description	Insured	Item	Manufacturer
	1	75.00	05/16/08		<input checked="" type="checkbox"/>	Keyboard	Casio
	2	1097.00	09/04/11	computer w/prin	<input checked="" type="checkbox"/>	HP Pavilli	Hewlett Packard

Figure 45: Selection of a record

- b) Click **Delete Rows** from the context menu.
- c) You will be asked if you want to delete the selected data. If you do, click **Yes**; if you don't, click **No**. Since we want to delete this record, click **Yes**.
- 3) Use Table 8 to reenter the data you just deleted. This will place the data in the bottom row.
- 4) Close the table.

### Note

When you begin reentering that data you had just deleted, the CategoryID value becomes a 4. It was originally a 0 after entering the data the first time. This illustrates that primary key values are never reused.

### Caution



There are times when records may need to be deleted. For example, you might no longer have something that is listed in your inventory or you might have replaced it. Sometimes you might accidentally create a record with no data other than the primary key value. All the other cells are blank in that row. But once you delete a record, it is gone for good. When you open the table, that row is also gone. The primary key column will be missing the number assigned to that record. So, be careful before deciding whether to delete a record or not. Once deleted, the record will have to be reentered if it is really needed.

## Enter or remove data using forms

This is the preferred method for adding, modifying, or removing entries. The data for any of the fields in a record can be entered, removed, or changed at the same time.

Table 9: Data to be added to the Household Inventory form

AppraisedValue	DatePurchased	Description	Insured	Item	Manufacturer
145.99	08/30/11	900W	<input checked="" type="checkbox"/>	Microwav	Sharp
250.00	04/20/99	built-in	<input checked="" type="checkbox"/>	Dishwasl	General Electric
528.00	12/16/90	side by side	<input checked="" type="checkbox"/>	Refrigera	Hot Point
452.00	12/30/99	counter top w/o	<input checked="" type="checkbox"/>	stove	Whirlpool
143.00	03/31/09	19" Digital read	<input checked="" type="checkbox"/>	TV	Motorola
20.00	04/22/04	802.11g compa	<input checked="" type="checkbox"/>	WiFi moc	Linksys

ModelNumber	Notes	PlacePurchased	PurchasePrice	Room	SerialNumber
SHR54678KL		Service Deluxe	145.99	Kitchen	SHR4568de222
GE456ee781	5 cycles	Lowes	450.95	Kitchen	GE4581839315
SBS44588F1990		Brothers Appliances	1599.99	Kitchen	HP4865818634
WP5748we84	oven	Sears	899.00	Kitchen	456789123258
MOT57dde85815		Sanders TV & REpai	249.99	Den	MOT123456789
WRT54GS V7	4-RJ45	Webster's Compute	35.99	Computer	CGNC1H111980

## Enter or remove data from fields

It does not matter whether the form is a simple one or a complex one. Open the form and then make the modifications.

Open the *Household Inventory* form and use Table 9 to enter data into the designated fields. Begin with *AppraisedValue* (*CategoryID* will enter its own value). To see all fields, maximize the size of the form if necessary. When you have filled out all the fields of one record on the form, click the **Next record** button (Figure 46) to go to the next record. All the data entered into the form for this record is saved in the table when you click the **Next Record** button.

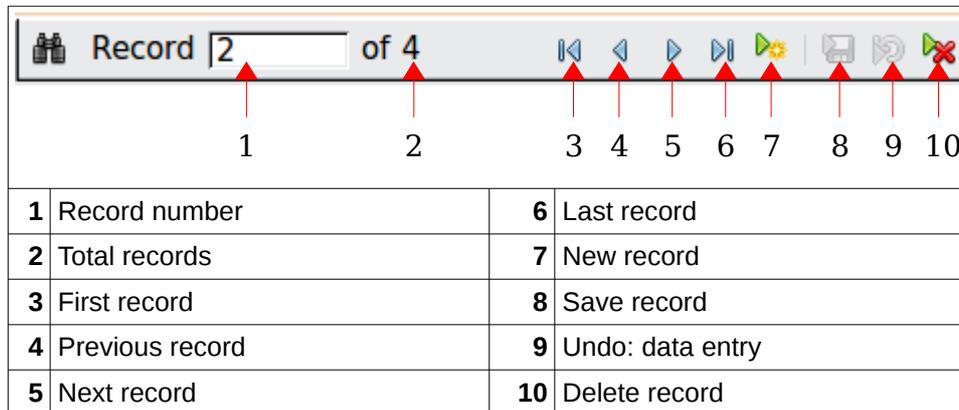


Figure 46: Form Navigation toolbar

### Add data using a form

- 1) Open the form.
  - a) In the main database window, select the Forms icon.
  - b) In the Forms section, double-click *Household Inventory*.
- 2) Enter the data into the fields.
  - a) Click the *AppraisedValue* field.
  - b) Enter the data.
  - c) Press the *Tab* or *Enter* key. This moves you to the next field (*DatePurchased*).
  - d) Enter the data in the *DatePurchased* field and *Tab* into the next field.
  - e) Continue to enter data in each field and then *Tab* to the next field until you have entered the data in the *Notes* field.
  - f) When you have entered the data in the *Notes* field, press the *Tab* key to go to the next record.
  - g) Use the same steps to enter data in all the fields in the new record.

#### Note

When the cursor is in the last field (*Notes*) of a record, use the *Tab* key to go to the first field (*CategoryID*) of the next record. When the cursor is in the first field (*CategoryID*) of a record, use *Shift+Tab* to go to the last field (*Notes*) of the previous record. If the cursor is in the first field of the first record, *Shift+Tab* will do nothing: there is no previous record.

### Modify a field: change or delete its data (Figure 46)

#### Tip

Use the **Previous Record** or **Next Record** button to go to the Record in which you want to make changes.

- 1) Click the field containing data you want to modify.

- 2) Change the value of this field to the correct value.
- 3) Click the **Save Record** button. (Figure 46)
- 4) Close the Household Inventory form.

### Tip

- 1) Delete, add, or erase text the same way you do these things in a word processor.
- 2) Change the selection of a drop-down list by selecting another member of the list.

## Delete records

Deleting a record requires a few steps. We will delete the last record in the Household Inventory form to demonstrate it.

- 1) Open the Household Inventory form. The form opens with the first record showing.
- 2) Click the **Last Record** button (Figure 46).
- 3) Click the **Delete Record** button.
- 4) A warning window appears stating **You intend to delete one record. If you click Yes, you will not be able to undo this operation! Do you want to continue?** Click **Yes**. The record has been deleted.

I recommend that you use Table 50 to reenter the data of the record you just erased. This will give you some idea of what has to be done to correct an error when the wrong record has been erased.

### Caution



**Never, but never, delete any record unless you are absolutely sure you no longer need the data contained in the record. If you accidentally delete a record, you will have to reenter the data later if you discover you really, truly need the data. But unless you have a copy of the data somewhere, you may have deleted some important data that no longer exists anywhere.**

## Mixing Base with the rest of LibreOffice

Base can be used alone or be used with other components of LibreOffice. It all depends upon what we want to do with the information contained in the database that contains our data.

This section contains a few suggestions as to how Base can be used with the other components of LibreOffice. Chapter 5 of the *Base Guide*, Exchanging Data, as well as the individual guides for the other components of LibreOffice contain instructions on how to do this.

## Calc

Base and Calc work together well. Using **F4** or **View > Data Sources** in Calc opens the Data window. Data and column names can be copied from the data source to the spreadsheet opened in Calc. Or, data can be copied from the spreadsheet to the data source. This is discussed in the *Calc Guide* and in Chapter 5 of the *Base Guide*.

Once the data has been copied from the data source to the spreadsheet, you can apply all of the available Calc functions to the data. This means you can create charts from the data as well as all the other things you can do with data in Calc. This is also discussed in the *Calc Guide* and in Chapter 5 of the *Base Guide*.

There are different ways to use the data in a spreadsheet as a data source. One way is to create a Base file to connect to the spreadsheet. A second way is to save the spreadsheet using the dbf file format. This is the file format used by dBase. Base will open this file format. This is discussed in Chapter 5 of the *Base Guide*.

## Writer

One very common database is the address book. Base will connect to several of them. Using Writer with Base, the postal addresses in the address book can be used to print envelopes when needed. Send an email to several people in the address book at one time is also possible. This process is called mail merge. It is explained in the *Mail Merger* chapter of the *Writer Guide*.

We might want to write a document that contains data existing in one of our databases. We can look up the data and type it into our document, but we might make a mistake in typing. By using Base with Writer, we can place a field linked to the desired data in the document. Doing it this way guarantees the data shown in the document will be as accurate as the data in the corresponding database. This is explained in Chapter 14, *Working with Fields*, in the *Writer Guide*.

Tables in text documents can be copied using Writer into a table in a database. A new table can be created in the database including the data in the text document table. Or, a new table can be created in the database without any data. Or, if the data in the table is arranged properly, this data can be added to an existing table of a database. This is part of Chapter 5 of the *Base Guide*.

Similarly, data in a database table can be used to create a table in a text document using Writer. You even have choices as to the table's appearance. This is also part of Chapter 5 of the *Base Guide*.

The forms created by Base are contained within the database file. These forms are used to enter, delete, or view data contained in the database. Writer can also create forms, but they serve a different purpose than the forms created by Base. A Writer form contains the data entered into the controls of it, but a Base form does not. All data is contained in the tables; Base only accesses this data. See Chapter 15, *Using Forms in Writer*, in the *Writer Guide* for more information.

In addition, Writer has the capability to produce XForms. These are web forms created by the World Wide Web Consortium (W3C). These are beyond the scope of the *Base Guide*.

## Impress

Base and Impress are not directly used together. They can be used together in an indirect way. Calc spreadsheets can be inserted into slides. So when you want to use data of a database in a slide, you first copy the data into a spreadsheet and then insert this spreadsheet into the appropriate slide or slides. This is discussed in Chapter 7, *Including Spreadsheets, Charts, and Other Objects*, in the *Impress Guide*.

## Using Base with other data sources

---

Base can be used with a variety of data sources, including JDBC, Oracle JDBC, Adabas D, Groupwise, Evolution LDAP, Evolution Local, KDE Address Book, LDAP Address Book, Mozilla Address Book, Thunderbird Address Book, Spreadsheet, dBASE, Text, MySQL, and ODBC.

Base can connect to some of these data sources directly by making only a few selections in the Base Wizard. These data sources include text files and spreadsheets. Chapter 5 of the *Base Guide* contains instructions for how to work with these data sources.

Base requires a special driver to connect to some data sources. PostgreSQL, MySQL, JDBC, and Oracle JDBC are examples of these data sources. Chapter 8 of the *Base Guide* discusses how to connect to these data sources as well as how to work with them.