Data Manager

This application allows you to create new Microsoft Jet databases, and examine or map databases in a variety of other formats. Users can also create Attached tables to this data.

Data Manager

This application allows you to create new Microsoft Jet databases, and examine or map databases in a variety of other formats. Users can also create Attached tables to this data. Use Data Manager to create or map databases in the formats shown in the following table.

Database Format	Support
Microsoft Access	Create, Attach, Map
Paradox 3.X and 4.X	Attach, Map
dBASE III and IV	Attach, Map
FoxPro 2.0 and 2.5	Attach, Map
Btrieve	Attach, Map
ODBC	Attach, Map

Visual Basic shares its database engine with Microsoft Access. Databases created with Visual Basic or Data Manager can be manipulated with Microsoft Access, and databases created with Microsoft Access can be manipulated with Visual Basic and Data Manager. This Help file will refer to "Visual Basic" or Jet databases–meaning a database created with Visual Basic, Microsoft Access, or the Data Manager application.

File/New Database

See Also

The databases you create with Data Manager are Microsoft Jet databases and can be edited in both Microsoft Access 2.0 and Visual Basic 4.0.

In the main Data Manager dialog box, select New Database from the File menu. Double-click on a folder that will contain your database, or, to make a new folder, click the Create New Folder button. Type a filename–the filename extension must be .MDB. Do not type or choose the name of an existing .MDB file. Click Save to accept your new database name.

Once your new database file has been created, add new tables, fields, and indexes to contain the data to be stored.

See Also

Creating a New Table

Creating a New Table

See Also

All of the data in your database will be stored in tables. Each table is defined as a set of one or more fields. Each field describes the kind of data to be stored by specifying the data type, size, and other attributes.

Tables group related data together. For example, in the sample BIBLIO.MDB database, most of the information about each book publisher is stored in the Publishers table. Because a publisher can print many books and represent many authors, information about authors and titles are kept in their own tables. To save space, a separate table, Publisher Comments, was created to store extended information about some publishers.

Creating New Tables

1. To create a new table, in the Tables/QueryDefs dialog box, click New. The Add Table dialog box appears.
2. Type a table name, field name, select a data type from the drop down list, and click the top button to add the field name to the list. Click the Up and Down buttons to move field names in the list.
3. To save the table, click OK.

You can manipulate tables using the following commands.

Command	Description
Open	To open an existing table in the specified database and view the data.
New	To create a new table.
Design	To open an existing table and view or modify the table structure.
Delete	To delete an existing table.
Attached Tables	To attach new tables and reattach existing tables.
Relations	Defines how data is related among tables.

The syntax for field names is determined by the database vendor. For Jet databases, the syntax for field names is the same as for any other database object (see the topic Naming Database Objects). The data type determines which kind of information is stored in fields, for example, a number or a string. For most data types, the size is set automatically. For Text fields, type the maximum number of bytes needed in the field. Remember that the total length of your record is limited to 2000 bytes.

Note You can add additional fields to a database table but you cannot rename, modify, or delete fields without deleting the entire table and re-building each field. For this reason, design your tables carefully.

See Also

Opening an Existing Table
Revising Database Design
Deleting a Table

Designing Your Tables

See Also

1. Once you have created all necessary fields, in the Tables/QueryDefs dialog box, select a table from the list and click Design to open the Table Editor dialog box.

   You can change the design of your tables by using the following commands.

Command	Description
Edit	To edit an existing field.
Add	To add a new field.
Remove	To delete an existing field.
Indexes	To edit, add, or remove indexes.
Keys	To select primary and foreign keys.
Close	To close the Design dialog box.

2. In the Table Editor dialog box, click Indexes to open the Indexes dialog box. Choose Add to create any necessary indexes for your table.

   Use indexes to improve performance and to eliminate duplicate records in your table.

   As with fields, the syntax for index names is determined by the database vendor. For Jet databases, the syntax for a index name is the same as for other database objects (see the topic Naming Database Objects).
3. Choose one or more of the field names from the Table Editor list and click Add to add these field names to the Fields in index list.

   The contents of these fields will determine how your table is sorted or returned to your application. You can create an index with as many fields as you like, but the total length of the field names cannot exceed 254 characters. However, not all fields may be used in an index. For example, MEMO or OLE object fields cannot be specified.
4. For Visual Basic to prohibit duplicate records in this table based on this index, select the Required and Unique check boxes. For Visual Basic to consider this to be the only primary index, select the Primary Index check box.
5. Choose either Add (Asc) (Ascending) or Add (Dec) (Descending) to designate how the table is to be sorted based on this index.
6. Click OK when you have created all of the indexes necessary.

Once your database structure is created, you can write a Visual Basic application to fill the database with data or use the Data Manager to open the table and edit it directly (see the topics Opening an Existing Table and Revising Database Design).

See Also

- Opening an Existing Table
- Revising Database Design
- Deleting a Table

Opening an Existing Table

See Also

To open an existing table, from the File menu, choose Open Database, select a file, and click Open. The Tables/QueryDefs dialog box will appear. Select a table name from the list, and click Open.

See Also

Creating a New Table

Deleting a Table

See Also

To delete and existing table, from the File menu, choose Open Database, select a file, and click Open. The Tables/QueryDefs dialog box will appear. Select a table name from the list, and click Delete. Deleting an attached table will not delete the physical table—only the attachment will be deleted.

See Also

Creating a New Table

Revising Database Design

From the Tables/QueryDefs dialog box, select an existing table name from the list and click Design to open the Table Editor dialog box. Here you can edit, add, and remove fields, and designate keys.

Creating New Indexes

1. In the Tables/QueryDefs dialog box, select an existing table name and click Design. The Table Editor dialog box appears.
2. Click Indexes, and select Add. Type the Index Name and make your other selections. Choose one or more of the field names from the Fields in table list and click Add.

   The field names are moved to the Fields in Index dialog box. The contents of these fields will determine how your table is sorted or returned to your application. You may create an index with as many fields as you like, but the total length of the field names cannot exceed 254 characters. However, not all fields may be used in an index. For example, MEMO or OLE object fields cannot be specified.Create all necessary indexes for your table.

   As with Fields, the syntax for index names is determined by the database vendor. For Microsoft Jet databases, the syntax for an index name is the same as for other database objects (see the topic Naming Database Objects).
3. Click OK when you have created all of the indexes needed.

Note Use indexes to improve performance and to eliminate duplicate records in your table.

File Menu Commands

The following commands are available from the Data Manager File menu.

Commands
New Database
Open Database
Compact Database
Convert Database
Encrypt Database
Decrypt Database
Repair Database
Exit

Error Adding Index

You may create an index with as many fields as you like, but the total length of the field names cannot exceed 254 characters. Note that not all fields may be used in an index. For example, MEMO or OLE object fields cannot be specified.

Manipulating Table Data

To manipulate data, first open a table. You can perform any of the following tasks:

- To change contents, select an individual field. If you move to another record, the changes are saved to the database.
- To add a new record to the end of your table, click Add and type data into the fields. Click OK to save the changes.
- To delete the current record, click Delete. The record is immediately deleted and cannot be recovered.
- To locate a particular record, click Find. Select the field, operator, and value by which to search. The syntax for this expression matches that for the Find methods as described in Visual Basic online Help. Generally, these follow the syntax for Structured Query Language (SQL) WHERE clauses.
- To retrieve the records again, click Refresh.

Naming Database Objects

There are specific rules for naming Visual Basic database objects. In Visual Basic, database object names can be up to 64 characters long and can include any combination of letters, numbers, spaces, and special characters except a period (.), an exclamation mark (!), and brackets ([ ]). You cannot use leading spaces or control characters (ASCII values 0 through 32).

Relationships Among Tables

Select a table name from the lists in the Tables/QueryDefs dialog box, and click Relations. The Relationships dialog box appears. In the Relationships dialog box, you can define relationships among the tables in your database. Data Manager uses these relationships to find associated data when you run queries or work with forms and reports based on multiple tables.

Primary Table

Lists the tables in the database. Select the table with the primary key field that you want to relate to another table.

Primary Key Fields

Displays the primary key fields for the primary table you select.

Type

Select one of the two options for relationships type: One or Many.

Option	Description
One	Establishes a one-to-one relationship among the tables
Many	Establishes a one-to-many relationship among the tables

Related Table

Lists all tables in the database. Select the table you want to relate to the primary table.

Select Matching Fields

Lists the fields in the related table. Select the fields that match the primary key fields of the primary table.

Enforce Referential Integrity

Select to enforce referential integrity. Clear if you don't want Data Manager to enforce these rules.

Add

Creates the relationship according to your designations.

Delete

Deletes the relationship.

Suggest

Looks for fields in the related table that have the same names as the primary key fields.

Close

Closes the dialog box.

Attaching a Table

1. In the Tables/QueryDef dialog box, click Attached Tables.

   The Attached Tables dialog box appears.

2. Click New, type information in the fields, and select Attach.

   This command creates a link to a table from another database application so that you can view and edit data in the table. You can continue to use attached tables in the original application as well as in Visual Basic. You can also attach tables from other Microsoft Jet databases. Attaching tables from another database allows you to use them without opening the other database.

Note If you attach a table from Paradox, Btrieve, or an SQL database, you may need to supply the password set in the other application. If you attach a table and specify a password, The Data Tool stores this password in your database so that you can open the table later (this is optional for SQL databases). To keep the password secure, you may want to encrypt a database that contains attached tables with passwords.

You can attach tables from the following database applications:

- ■ Microsoft Access (databases other than the open database)
- ■ Paradox (version 3.x and 4.x .DB files)
- ■ FoxPro (version 2.0 and 2.5 .DBF files)
- ■ dBASE III and dBASE IV (.DBF files)
- – Btrieve (with the data definition files FILE.DDF and FIELD.DDF)
- – Microsoft SQL Server, Sybase SQL Server, and ORACLE Server (select <SQL Database>)
- – ODBC

For SQL databases, Data Manager displays a Server Login dialog box. For descriptions of the options in these dialog boxes for each ODBC driver, see ODBC Drivers.

After you attach an external table, set field properties for the table in the Design dialog box.

Although you can't change the structure of an attached table, you can set the field properties.

Defining Many-to-Many Relations

In a many-to-many relationship between two tables, one record in either table can relate to many records in the other table.

To define a many-to-many relationship, create a third table that serves as a junction between the two tables.

1. Create the tables for which you want to define a many-to-many relationship.
2. Create a third (junction) table and include the primary key fields from each of the other two tables in this table. In the junction table, the primary key fields function as foreign keys.
3. In the junction table, set the primary key to include all the primary key fields from the other two tables.
4. Define a one-to-many relationship between each of the two primary tables and the junction table.

Note You can add other fields to the junction table, just as you can add fields to any other table.

SQL Statements

To query a database, first select and open a database. In the SQL Statement dialog box, type a query and click Execute. To save the query, click Save, enter the QueryDef name, and choose OK. To clear the SQL Statement dialog box, click Clear.