



Relationship

Mohammad Hassan Adjigol

Feb 2006



Relationship

After you've set up different tables for each subject in your **Microsoft Access database**, you need a way of telling Microsoft Access how to bring that information back together again. The first step in this process is to define **relationships**) **relationship: An association that is established between common fields (columns) in two tables. A relationship can be one-to-one, one-to-many, or many-to-many** (.between your tables. After you've done that, you can create queries, forms, and reports to display information from several tables at once.

Relationship

For example, this form includes information from four tables:

- 1- The Customers table
- 2- The Orders table
- 3- The Products table
- 4- The Order Details table

The customer name in the Bill To box is retrieved from the Customers table, the Order ID and the Order Date from the Orders table, the Product from the Products table, and the Unit Price and Quantity from the Order Details table. These tables are linked to each other in a variety of ways to bring information from each into the form.

Orders

Bill To: Maison Dewey 1

Order ID: 10529 Order Date: 07-Jun-00

Product:	Unit Price:	Quantity:
Gudbrandsdalsost 3	\$24.00	14
Scottish Longbread	\$12.50	20
Pâté chinois	\$36.00	10
*		



Relationship

How relationships work

In the previous example, the fields in four tables must be coordinated so that they show information about the same order. This coordination is accomplished with relationships between tables. A relationship works by matching data in key fields— usually a field with the same name in both tables. In most cases, these matching fields are the **primary key** from one table, which provides a unique identifier for each record, and a **foreign key** in the other table. For example, employees can be associated with orders they're responsible for by creating a relationship between the EmployeeID fields.



Relationship

primary key

One or more fields (columns) whose values uniquely identify each record in a table.

- A primary key cannot allow Null values and must always have a unique index.
- A primary key is used to relate a table to foreign keys in other tables.

foreign key

One or more table fields (columns) that refer to the primary key field or fields in another table.

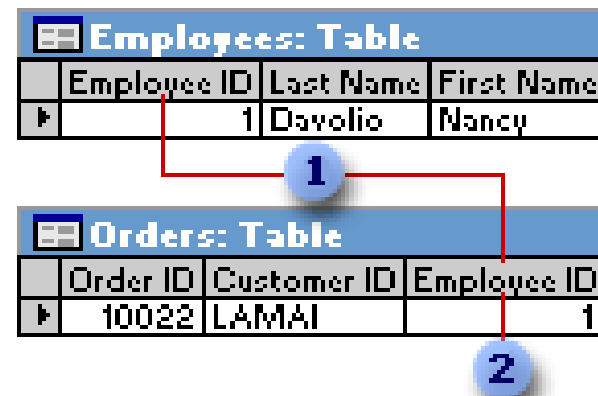
- A foreign key indicates how the tables are related.

Relationship

For example, employees can be associated with orders they're responsible for by creating a relationship between the EmployeeID fields.

1- EmployeeID appears in both tables -as a primary key

2- A foreign key.



Relationship

A one-to-many relationship

A one-to-many relationship is the most common type of relationship. In a one-to-many relationship, a record in Table A can have many matching records in Table B, but a record in Table B has only one matching record in Table A.

Suppliers: Table	
Supplier ID	Company Name
1	Exotic Liquids
2	New Orleans Cajun Delights
3	Grandma Kelly's Homestead
4	Tokyo Traders

Products: Table		
Product ID	Product Name	Supplier ID
1	Chai	1
2	Chang	1
3	Aniseed Syrup	1
4	Chef Anton's Cajun Seasoning	2
5	Chef Anton's Gumbo Mix	2

- 1- One supplier
- 2- can supply more than one product...
- 3- but each product has only one supplier.



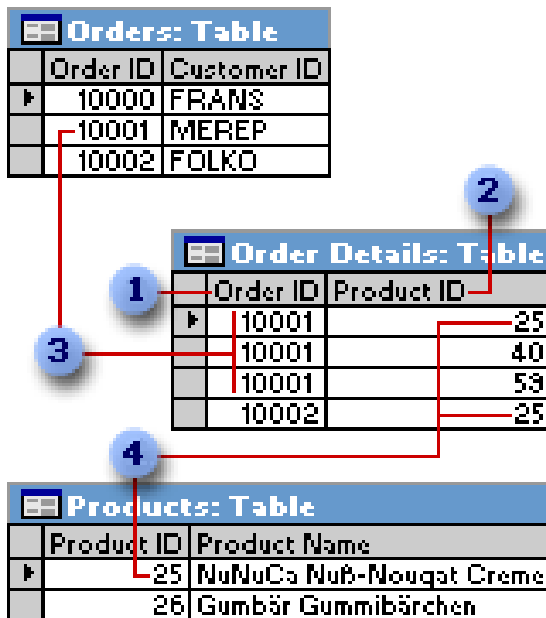
Relationship

A many-to-many relationship

In a many-to-many relationship, a record in Table A can have many matching records in Table B, and a record in Table B can have many matching records in Table A. This type of relationship is only possible by defining a third table (called a **junction table**) whose primary key consists of two fields—the foreign keys from both Tables A and B. A many-to-many relationship is really two one-to-many relationships with a third table.

Relationship

For example, the Orders table and the Products table have a many-to-many relationship that's defined by creating two one-to-many relationships to the Order Details table. One order can have many products, and each product can appear on many orders.



1. Primary key from the Orders table
2. Primary key from the Products table
3. One order can have many products...
4. and each product can appear on many orders.



Relationship

A one-to-one relationship

In a one-to-one relationship, each record in Table A can have only one matching record in Table B, and each record in Table B can have only one matching record in Table A. This type of relationship is not common, because most information related in this way would be in one table.

You might use a one-to-one relationship to divide a table with many fields, to isolate part of a table for security reasons, or to store information that applies only to a subset of the main table.

Relationship

For example, you might want to create a table to track employees participating in a fundraising soccer game. Each soccer player in the Soccer Players table has one matching record in the Employees table.

Employees: Table			
Employee ID	Last Name	First Name	
1	Davolio	Nancy	
2	Fuller	Andrew	
3	Leverling	Janet	
4	Peacock	Margaret	
5	Buchanan	Steven	

1

Soccer Players: Table			
Employee ID	Player Nickname	Skill Level	
1	Slammin' Man	2	
3	Ace	1	
5	Stevemeister	2	

2

1. Each soccer player has one matching record in the Employees table.
2. This set of values is a subset of the EmployeeID field and the Employees table.



Relationship

The kind of relationship that Microsoft Access creates depends on how the related fields are defined:

✓ A one-to-many relationship is created if only one of the related fields is a **primary key** or has a **unique index** (**unique index**: An index defined by setting a field's **Indexed** property to **Yes (No Duplicates)**). A unique index will not allow duplicate entries in the indexed field. Setting a field as the primary key automatically defines the field as unique.).



Relationship

- ✓ A one-to-one relationship is created if both of the related fields are primary keys or have unique indexes.
- ✓ A many-to-many relationship is really two one-to-many relationships with a third table whose primary key consists of two fields—the foreign keys from the two other tables.



Relationship

Referential integrity

Referential integrity is a system of rules that Microsoft Access uses to ensure that relationships between records in related tables are valid, and that you don't *accidentally delete or change related data*.

You can set referential integrity when all of the following conditions are met:

✓ The matching field from the primary table (primary table: The "one" side of two related tables in a one-to-many relationship. A primary table should have a primary key and each record should be unique.) is a primary key or has a unique index .

M.H.Adjigol



Relationship

✓ The related fields have the same data type. There are two exceptions. An AutoNumber field can be related to a Number field with a FieldSize property setting of Long Integer, and an AutoNumber field with a FieldSize property setting of Replication ID can be related to a Number field with a FieldSize property setting of Replication ID.



Relationship

✓ Both tables belong to the same Microsoft Access database. If the tables are linked tables (**linked table: A table stored in a file outside the open database from which Access can access records. You can add, delete, and edit records in a linked table, but you cannot change its structure.**), they must be tables in Microsoft Access format, and you must open the database in which they are stored to set referential integrity. Referential integrity can't be enforced for linked tables from databases in other formats.



Relationship

The following rules apply when you use referential integrity:

1- You can't enter a value in the foreign key field of the related table that doesn't exist in the primary key of the primary table. However, you can enter a **Null** value in the foreign key, specifying that the records are unrelated. For example, you can't have an order that is assigned to a customer that doesn't exist, but you can have an order that is assigned to no one by entering a **Null** value in the CustomerID field.



Relationship

2- You can't delete a record from a primary table if matching records exist in a related table. For example, you can't delete an employee record from the Employees table if there are orders assigned to the employee in the Orders table.

3- You can't change a primary key value in the primary table, if that record has related records. For example, you can't change an employee's ID in the Employees table if there are orders assigned to that employee in the Orders table.



Relationship

Cascading updates and deletes

For relationships in which referential integrity is enforced, you can specify whether you want Microsoft Access to automatically cascade update (**cascading update: For relationships that enforce referential integrity between tables, the updating of all related records in the related table or tables when a record in the primary table is changed.**) and cascade delete (**cascading delete: For relationships that enforce referential integrity between tables, the deletion of all related records in the related table or tables when a record in the primary table is deleted.**) related records.



Relationship

If you set these options, delete and update operations that would normally be prevented by referential integrity rules are allowed. When you delete records or change primary key values in a primary table, Microsoft Access makes necessary changes to related tables to preserve referential integrity.



Relationship

If you select the **Cascade Update Related Fields** check box when defining a relationship, any time you change the primary key of a record in the primary table, Microsoft Access automatically updates the primary key to the new value in all related records. For example, if you change a customer's ID in the Customers table, the CustomerID field in the Orders table is automatically updated for every one of that customer's orders so that the relationship isn't broken. Microsoft Access cascades updates without displaying any message.



Relationship

Note If the primary key in the primary table is an AutoNumber field, setting the **Cascade Update Related Fields** check box will have no effect, because you can't change the value in an AutoNumber field.



Relationship

If you select the **Cascade Delete Related Records** check box when defining a relationship, any time you delete records in the primary table, Microsoft Access automatically deletes related records in the related table. For example, if you delete a customer record from the Customers table, all the customer's orders are automatically deleted from the Orders table (this includes records in the Order Details table related to the Orders records).



Relationship

When you delete records from a form or datasheet with the **Cascade Delete Related Records** check box selected, Microsoft Access warns you that related records may also be deleted. However, when you delete records using a delete query (delete query: A query (SQL statement) that removes rows matching the criteria that you specify from one or more tables.), Microsoft Access automatically deletes the records in related tables without displaying a warning.