

note

The Data Model is a new feature for Excel 2013

None of the features described in this session existed in Excel 2010 and earlier (though there was a PowerPivot add-in available for Excel 2010 that added similar functionality).

If you use any of the features described in this session they will not work if the workbook is opened in any earlier version of Excel.

note

Virtually all of the world's databases work like this

Dr Edgar F Codd (1923-2003) invented the relational database when working for IBM in the 1970's.

The first relational database products came to market in the late 1970's and were quickly adopted by big business.

So good and great was Codd's design that nothing better has been developed in over thirty years.

The entire world of commerce is now powered by relational databases.

You'll usually find a requirement to analyse data from a relational database in all but the very smallest enterprises.

Lesson 6-1: Understand primary and foreign keys

Later, in: *Lesson 6-2: Create a simple data model*, you'll construct a relationship between two Excel tables. In order to do this you'll need to understand the concept of primary and foreign keys.

What is a primary key?

A relational database consists of several tables, each containing data. A database table is conceptually no different to an Excel table except that it must have a primary key.

Here's an example of an Excel table with a primary key:

CategoryID	CategoryName	Description
1	Beverages	Soft drinks, coffees, teas, beers, and ales
2	Condiments	Sweet and savory sauces, relishes, spreads, and seasonings
3	Confections	Desserts, candies, and sweet breads

The above example comes from a table called *Category*. It is good database design practice to name a primary key as the table name plus the letters *ID*. This designer has followed best practice and called the primary key *CategoryID*.

The only important quality of a primary key is that it must be unique to the table. That's because the primary key is used (by a relationship) to identify a single, specific row in a table. If there were two table rows with the same primary key, a relationship wouldn't be able to correctly identify which single row was being referenced.

In the above example the primary key is a number but primary keys can consist of numbers, letters or both.

What is a foreign key?

Consider the following two Excel tables:

CategoryID	CategoryName	Description
1	Beverages	Soft drinks, coffees, teas, beers, and ales
2	Condiments	Sweet and savory sauces, relishes, spreads, and seasonings
3	Confections	Desserts, candies, and sweet breads

ProductID	ProductName	CategoryID	QuantityPerUnit	UnitPrice
1	Chai	1	10 boxes x 20 bags	18.00
3	Aniseed Syrup	2	12 - 550 ml bottles	10.00
16	Pavlova	3	32 - 500 g boxes	17.45
24	Guaraná Fantástica	1	12 - 355 ml cans	4.50
25	NuNuCa Nuß-Nougat-Creme	3	20 - 450 g glasses	14.00
34	Sasquatch Ale	1	24 - 12 oz bottles	14.00

You can see how it is possible to determine that *Aniseed Syrup* is in the *Condiments* category:

1. In the *Product* table, *Aniseed Syrup* has a *CategoryID* of 2.
2. The item in the *Category* table with a primary key (*CategoryID*) of 2 is *Condiments*.

note

Primary and foreign keys should have the same name

Any database designer worthy of the name will use exactly the same name for the primary key column and the related foreign key column.

I've found that, in the real world of business, I often have to work with a badly designed database where this is not the case. This makes the data a lot more difficult to work with.

anecdote

How meaningful primary keys almost stopped the Welsh from buying cars

Many years ago I implemented a Europe-wide Business Intelligence solution for a very large automotive finance company.

I was not pleased (but unsurprised) to find that the database designer had used meaningful primary and foreign keys.

The designer had used a concatenation of the customer's surname and date of birth as the primary key for the customer table.

The designer believed that the possibility of having two customers with the same surname and date of birth was extremely unlikely to ever happen.

In fact, it transpired that 13.5% of the Welsh population are called Jones, meaning that it was certain to happen in Wales (and actually very likely to happen everywhere)!

The database was, of course, re-designed to use meaningless primary keys.

While *CategoryID* is the primary key column in the *Category* table, it is a *foreign key* column within the *Product* table.

You can think of the *CategoryID* values in the *Product* table as belonging to the *Category* table hence making them foreign keys in the *Product* table.

You can probably now see the wisdom of the naming convention for primary keys. It makes it possible to spot which are the primary and foreign keys in a table at a glance.

- The column named with the table name plus *ID* is the primary key.
- Any other column name that is postfixed with *ID* (has *ID* at the end of the name) is a foreign key.
- Any column not postfixed with *ID* is a regular data field containing information.

Meaningful and Meaningless primary keys

In the above example, the primary key is a *meaningless* number. The number 2 tells you nothing about any attribute of the *Condiments* category. It simply provides a way to find where the correct *Condiments* row is located within the table.

It is also possible to use a *meaningful* primary key (though a professional database designer would never do this). For example, you could argue that because category names are unique, it is fine to use the *CategoryName* column in the above table as the primary key.

If you decided to use the category name as the primary key the tables would look like this:

CategoryID	Description
Condiments	Sweet and savory sauces, relishes, spreads, and seasonings
Beverages	Soft drinks, coffees, teas, beers, and ales
Confections	Desserts, candies, and sweet breads

ProductID	ProductName	CategoryID	QuantityPerUnit	UnitPrice
50	Valkoinen suklaa	Confections	12 - 100 g bars	16.25
1	Chai	Beverages	10 boxes x 20 bags	18.00
77	Original Frankfurter grüne Soße	Condiments	12 boxes	13.00
8	Northwoods Cranberry Sauce	Condiments	12 - 12 oz jars	40.00
68	Scottish Longbreads	Confections	10 boxes x 8 pieces	12.50
70	Outback Lager	Beverages	24 - 355 ml bottles	15.00

From a database design perspective, this is generally a bad idea

You'll often find, however, that you need to create this type of relationship when creating relationships between tables that originate from different databases. In this case, no formal primary key/foreign key relationship will exist.

For example, you may need to join a table from your corporate database containing customer addresses, with a table detailing sales tax rates by state that you've downloaded from the Internet. In this case you would need to create a primary key/foreign key relationship between the state fields in each table.