

## note

### What is a relational database?

Any form of information storage can be said to be a database, but most databases in the world of commerce are *relational* databases.

Relational databases contain multiple tables of data that are joined together with 'relationships'.

By using multiple tables, relational databases avoid repeating the same data more than once, making them more efficient than, for example, an Excel spreadsheet.

You'll learn much more about how relational databases work later in this course, in: *Lesson 20: Understand relationships*.

# Lesson 4: Explore a database

You need to know how a database is structured before you can begin to query it. Fortunately, SSMS makes it easy to examine a database's structure.

- 1 Open SSMS and connect to your SQL Server (if you haven't already done this).
- 2 Examine the tables in the database.

1. Expand the *Databases* folder in the *Object Explorer* pane by clicking the + sign to the left of it.

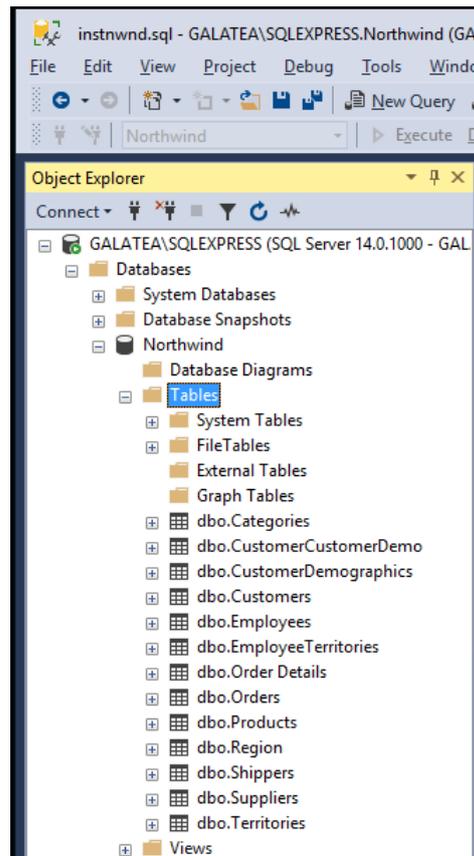
You should see the *Northwind* database, just as you did in the previous lesson.

2. Expand the *Northwind* database.

You should see several subfolders that contain different objects within the *Northwind* database.

3. Expand the *Tables* subfolder.

You can now see all of the tables within the *Northwind* database (the items with the  icon).

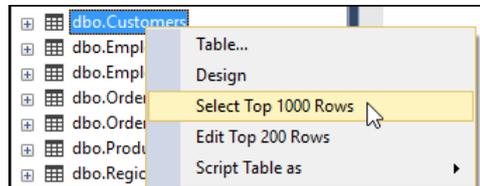


Most databases have several tables that contain the actual data. As you can see, the *Northwind* database has 13 tables.

- 3 Examine the contents of the *Customers* table.

As well as viewing the names of the tables in the database, SSMS also allows you to quickly view a snapshot of their contents. This can be very helpful to establish the types of data that a table contains.

Right click the *Customers* table and click *Select Top 1000 Rows* from the shortcut menu.



SSMS automatically generates and runs a SQL script that displays the top 1000 rows in the *Customers* table. You should see the script in the central pane of the SSMS interface and the query results in the bottom pane.

	CustomerID	CompanyName	ContactName	ContactTitle	Address	City	Region	PostalCode	Country	Phone	Fax
1	ALFKI	Alfreds Futterkiste	Maria Anders	Sales Representative	Obere Str. 57	Berlin	NULL	12209	Germany	030-0074321	030-0076545
2	ANATR	Ana Trujillo Emparedados y helados	Ana Trujillo	Owner	Avda. de la Constitución 2222	México D.F.	NULL	05021	Mexico	(5) 555-4729	(5) 555-3745
3	ANTON	Antonio Moreno Taquería	Antonio Moreno	Owner	Mataderos 2312	México D.F.	NULL	05023	Mexico	(5) 555-3932	NULL
4	AROUT	Around the Horn	Thomas Hardy	Sales Representative	120 Hanover Sq.	London	NULL	WA1 1DP	UK	(171) 555-7788	(171) 555-6750

#### 4 Examine the design of the *Customers* table.

Right click the *Customers* table and click *Design* from the shortcut menu.

Column Name	Data Type	Allow Nulls
CustomerID	nchar(5)	<input type="checkbox"/>
CompanyName	nvarchar(40)	<input type="checkbox"/>
ContactName	nvarchar(30)	<input checked="" type="checkbox"/>
ContactTitle	nvarchar(30)	<input checked="" type="checkbox"/>
Address	nvarchar(60)	<input checked="" type="checkbox"/>
City	nvarchar(15)	<input checked="" type="checkbox"/>
Region	nvarchar(15)	<input checked="" type="checkbox"/>
PostalCode	nvarchar(10)	<input checked="" type="checkbox"/>
Country	nvarchar(15)	<input checked="" type="checkbox"/>
Phone	nvarchar(24)	<input checked="" type="checkbox"/>
Fax	nvarchar(24)	<input checked="" type="checkbox"/>

You can now see the design of the *Customers* table. Database designers and administrators use this interface to design and modify tables, but for reporting purposes you're only interested in the *Data Type* and *Allow Nulls* columns. You'll work with data types and nulls in depth in: *Lesson 16: Work with NULL values and the IN keyword.*