Data Management and Warehousing

SQL Simple Queries

Stéphane Bressan

NUS
National University
of Singapore





We want to develop a sales analysis application for our online gaming store. We would like to store several items of information about our customers: their first name, last name, date of birth, e-mail, date and country of registration on our online sales service and the customer identifier that they have chosen . We also want to manage the list of our products, the games, their name, their version and their price. The price is fixed for each version of each game. Finally, our customers buy and download games. So we must remember which version of which game each client has downloaded. It is not important to keep the download date for this application.

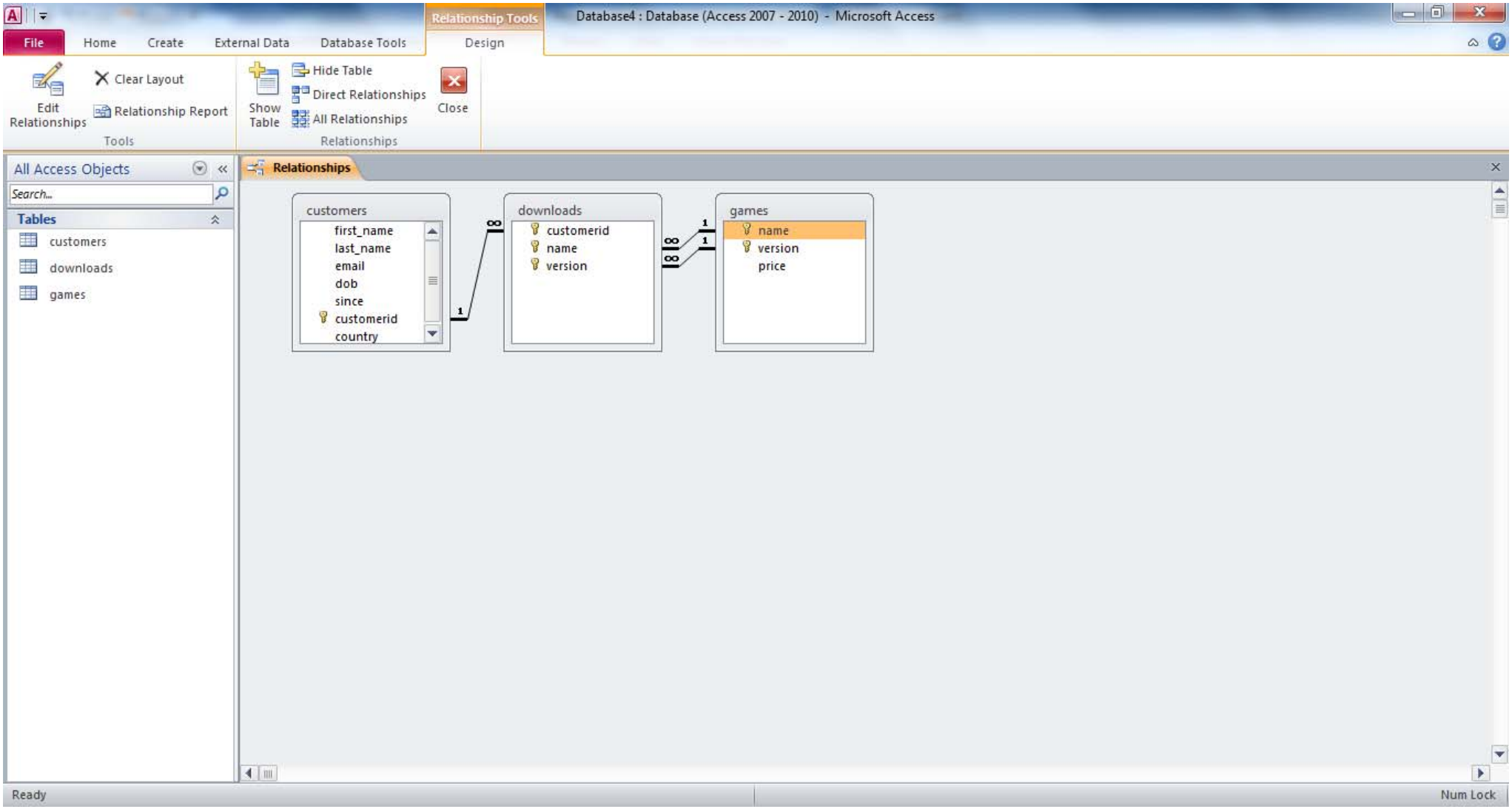
Create a database and run the `customers.sql`, `games.sql` and `download.sql` scripts with pgAdmin (3 or 4)

Here is the complete code of the example with primary and foreign keys.

```
CREATE TABLE customers (
        first_name VARCHAR(64) NOT NULL,
        last_name VARCHAR(64) NOT NULL,
        email VARCHAR(64) UNIQUE NOT NULL,
        dob DATE NOT NULL,
        since DATE NOT NULL,
        customerid VARCHAR(16) PRIMARY KEY,
        country VARCHAR(16) NOT NULL);

CREATE TABLE games (
        name VARCHAR(32),
        version CHAR(3),
        price NUMERIC NOT NULL,
        PRIMARY KEY (name, version));

CREATE TABLE downloads(
        customerid VARCHAR(16) REFERENCES customers(customerid),
        name VARCHAR(32),
        version CHAR(3),
        FOREIGN KEY (name, version) REFERENCES games(name, version),
        PRIMARY KEY(customerid, name, version));
```


Relationship Tools
Database4 : Database (Access 2007 - 2010) - Microsoft Access
File
Home
Create
External Data
Database Tools
Design
Edit Relationships
Clear Layout
Relationship Report
Tools
Show Table
Hide Table
Direct Relationships
All Relationships
Relationships
Close
All Access Objects
Search...
Tables
customers
downloads
games
Relationships
customers
first_name
last_name
email
dob
since
customerid
country
downloads
customerid
name
version
games
name
version
price
1
∞
Ready
Num Lock

A simple SQL query includes a "SELECT" clause, which indicates the fields to be printed, a "FROM" clause, which indicates the table (s) to be queried and possibly a "WHERE" clause, which indicates a possible condition on the records to be printed. We have seen the following query that displays the first and last names of registered customers from Singapore.

```
SELECT first_name, last_name
FROM customers
WHERE country = 'Singapore';
```

Structured Query Language (SQL) is a simplified, specialized query language for updating and querying relational databases.

It is an international industrial standard (unfortunately implemented with slight variations according to the vendors).

The following two queries print all the customer information, that is, all fields in the "customers" table.

```
SELECT first_name, last_name, email, dob, since,
     customerid, country
FROM customers;
```

The asterisk is a shorthand for all the field names.

```
SELECT *
FROM customers;
```

first_name	last_name	email	dob	since	customerid	country
Aaron	Griffin	agriffinfo@zdnet.com	5/31/1986	3/3/2016	Aaron1986	Singapore
Adam	Green	agreenf4@fc2.com	8/22/1983	5/15/2016	Adam1983	Singapore
Adam	Stone	astonea3@businesswire.com	4/12/1990	6/26/2016	Adam1990	Singapore
Adam	Howell	ahowellil@storify.com	9/15/1997	1/14/2016	Adam1997	Indonesia
Adam	Romero	aromerofh@rambler.ru	11/4/1998	12/19/2016	Adam1998	Singapore
Adam	Wijaya	awijaya38@xinhuanet.com	2/21/2000	1/8/2016	Adam2000	Singapore
Alan	Hansen	ahansenp3@webnode.com	11/22/1998	8/1/2016	Al8	Singapore
…						

It is possible to choose, reorder and replicate fields in the "SELECT" clause.

```
SELECT email, email, last_name, first_name
FROM customers;
```

The "SELECT" clause determines the schema of the table containing the result of the query.

Expr1000	email	last_name	first_name
agriffinfo@zdnet.com	agriffinfo@zdnet.com	Griffin	Aaron
agreenf4@fc2.com	agreenf4@fc2.com	Green	Adam
astonea3@businesswire.com	astonea3@businesswire.com	Stone	Adam
ahowellil@storify.com	ahowellil@storify.com	Howell	Adam
aromerofh@rambler.ru	aromerofh@rambler.ru	Romero	Adam
awijaya38@xinhuanet.com	awijaya38@xinhuanet.com	Wijaya	Adam
ahansenp3@webnode.com	ahansenp3@webnode.com	Hansen	Alan
…			

In SQL, the result of a query is a table. You can give a name to the query by using the SQL command "CREATE VIEW".

```
CREATE VIEW customers_basic AS
SELECT last_name, first_name, email
FROM customers;
```

This table can then be reused in other queries. If the "customers" table is updated, the "customers_basic" view also changes.

```
SELECT email
FROM  customers_basic
WHERE last_name='Yoga';
```

It is possible to rename the fields in the "SELECT" clause using the keyword "AS".

```
SELECT last_name AS family_name, first_name
FROM customers;
```

It is possible to make calculations on the fields in the "SELECT" clause.

```
SELECT name || ' ' || version AS game,
      price * 1.18 AS pricetax
FROM games;
```

Caution, the syntax of operations and functions can be specific to the DBMS used. For example, Oracle and MySQL use the "CONCAT ()" function while PostgreSQL and SQLite use "||" instead of "&" used by Microsoft Access 2010 and "+" by SQL Server.

game	pricetax
Aerified 1	14.16
Aerified 1.1	4.7082
Aerified 1.2	2.3482
Aerified 2	5.9
Aerified 2.1	14.16
Aerified 3	4.7082
Alpha 1	14.16
...	

SQL can use the calculated fields, i.e. the fields for which the values are calculated from existing fields, in the "SELECT" and "WHERE" clauses (and also "HAVING" that we will see later) . SQL includes operations and arithmetic functions (for example: addition +, subtraction -, multiplication * and division /, with parentheses if necessary), operations and functions for other domains (types) such as dates and strings characters.

```
SELECT name || ' ' || version
FROM games
WHERE price * 1.18 > 10;
```

Caution, the syntax of operations and functions can be specific to the DBMS used.

Expr1000
Aerified 1
Aerified 2.1
Alpha 1
Alpha 1.2
Alpha 2
…

The keyword "DISTINCT" in the "SELECT" clause (it appears only once after the keyword "SELECT", it eliminates repeated records) eliminates duplicates in the result.

```
SELECT first_name, last_name
FROM customers;
```

```
SELECT DISTINCT first_name, last_name
FROM customers;
```

The result of the first query can contain several occurrences of the same pair of last name and first name, not that of the second.

The query without "`DISTINCT`" displays 1001 records

The query with "`DISTINCT`" displays 983 records

first_name	last_name
Aaron	Griffin
Adam	Green
Adam	Stone
Adam	Howell
Adam	Romero
Adam	Wijaya
…	

The "WHERE" clause is optional. It is used to filter records that satisfy a single or compound condition.

A single condition compares the value of a field with a constant or the values of two fields with each other with the comparison operators =, <, <=, >=, <>, LIKE, BETWEEN, AND and IN.

```
SELECT name, version
FROM games
WHERE price >= 10;
```

```
SELECT name, price
FROM games
WHERE price BETWEEN 4 AND 20;
```

```
SELECT name, version
FROM games
WHERE version IN ('1.0','1.1');
```

name	version
Aerified	1
Aerified	2.1
Alpha	1
Alpha	1.2
. . .	

name	price
Aerified	12
Aerified	5
Aerified	12
Alpha	12
…	

name	version
Aerified	1.0
Aerified	1.1
Alpha	1.0
Alpha	1.1
Alphazap	1.1
…	

"LIKE" allows to make a partial comparison of strings according to a model (or pattern) the model allows to represent characters of replacement (* replaces any character string, replaces a character, ([AZ] replaces a character between A and Z, [0-9] replaces a digit between 0 and 9) and excluding character ranges ([! AZ] replaces A character except those between A and Z, [! 0-9] respectively).

```
SELECT first_name, last_name
FROM customers
WHERE customerid LIKE 'M*88';
```

Some systems use alternative characters (for example, SQLite is useful for replacing a string.) Some systems support regular expression.

first_name	last_name
Margaret	Watkins
Maria	Myers
Marie	Armstrong
Matthew	Vasquez
Mildred	Robinson

The "WHERE" clause is optional. It is used to filter records that satisfy a single or compound condition.

A compound condition combines simple conditions into a Boolean expression with the Boolean operators AND, OR, NOT and parentheses, if necessary.

```
SELECT name
FROM games
WHERE price BETWEEN 4 AND 20
AND version IN ('1.0','1.1');
```

```
SELECT name
FROM games
WHERE price BETWEEN 4 AND 20
AND NOT (version <> '1.0' AND version <> '1.1');
```

name
Aerified
Alpha
Alphazap
Andalax
Andalax
Asoka
…

Logic

P	Q	P AND Q	P OR Q	NOT P
True	True	True	True	False
False	True	False	True	True
True	False	False	True	False
False	False	False	False	True

NULL Values

Every domain (type) has an additional value: the `null` value. In general the semantics of null could be ambiguous. It could be "unknown", "does not exists", "unknown or does not exists". In SQL it is generally "unknown".

`"something = null"` is unknown (even is `"something"` is `"null"`).

`"null IS NULL"` is true.

`"something < null"` is unknown.

`"something > null"` is unknown.

`"10 + null"` is null.

`"0 * null"` is null.

`"COUNT(*)"` counts `NULL` values.

`"COUNT(att)", "AVG(att)", "MAX(att)", "MIN(att)"` eliminate null values.

NULL Values Logic

"`SELECT FROM WHERE condition`" returns results when the condition is true.

P	Q	P AND Q	P OR Q	NOT P
True	True	True	True	False
False	True	False	True	True
Unknown	True	Unknown	True	Unknown
True	False	False	True	False
False	False	False	False	True
Unknown	False	False	Unknown	Unknown
True	Unknown	Unknown	True	False
False	Unknown	False	Unknown	True
Unknown	Unknown	Unknown	Unknown	Unknown

Exercise

```
version IN ('1.0', '1.1')
```

The above condition is the same as the one below.

```
(version = '1.0' OR version = '1.1')
```

Check that it is the above condition is the same as the one below.

```
NOT (version <> '1.0' AND version <> '1.1')
```

NUS National University of Singapore

Exercise

Write a query that displays the names of the games starting with the capital letter "B" or the capital letter "C" and whose version 3.0 costs just over $ 4 after taxes (18%).

And finally, what happens if we query several tables in the same query?

```
SELECT *
FROM customers, downloads, games;
```

The result is the table that contains all the fields and possible combinations of the three tables *.

How useful is it?

* This is called a Cartesian product (named after the French mathematician René Descartes) or a cross product.

NUS National University of Singapore

Credits

Images and clips used in this presentation are licensed from Microsoft Office Online Clipart and Media

For questions about the content of this course and about copyrights, please contact Stéphane Bressan

steph@nus.edu.sg