

Database Programming

Cross Joins and Natural Joins

Objectives

This lesson covers the following objectives:

- Construct and execute a natural join using ANSI-99 SQL join syntax
- Create a cross join using ANSI-99 SQL join syntax
- Define the relationship between a cross join and a Cartesian product
- Define the relationship between a natural join and an equijoin

Objectives (cont.)

This lesson covers the following objectives:

- Explain the importance of having a standard for SQL as defined by ANSI
- Describe a business need for combining information from multiple data sources

Purpose

Up to now, your experience using SQL has been limited to querying and returning information from one database table at a time.

This would not be a problem if all data in the database were stored in only one table.

EMPLOYEES

EMPLOYEE_ID	LAST_NAME	DEPT_ID
100	King	90
101	Kochhar	90
...		
202	Fay	20
205	Higgins	110
206	Gietz	110

DEPARTMENTS

DEPARTMENT_ID	DEPT_NAME	LOCATION_ID
10	Administration	1700
20	Marketing	1800
50	Shipping	1500
60	IT	1400
80	Sales	2500
90	Executive	1700
110	Accounting	1700
190	Contracting	1700

Obtaining Data from Multiple Tables

EMPLOYEE_ID	DEPT_ID	DEPT_NAME
200	10	Administration
201	20	Marketing
202	20	Marketing
...		
102	90	Executive
205	110	Accounting
206	110	Accounting

Purpose (cont.)

But you know from data modeling that separating data into individual tables and being able to associate the tables with one another is the heart of relational database design. Fortunately, SQL provides join conditions that enable information to be queried from separate tables and combined in one report.

EMPLOYEES

EMPLOYEE_ID	LAST_NAME	DEPT_ID
100	King	90
101	Kochhar	90
...		
202	Fay	20
205	Higgins	110
206	Gietz	110

DEPARTMENTS

DEPARTMENT_ID	DEPT_NAME	LOCATION_ID
10	Administration	1700
20	Marketing	1800
50	Shipping	1500
60	IT	1400
80	Sales	2500
90	Executive	1700
110	Accounting	1700
190	Contracting	1700

Obtaining Data from Multiple Tables

EMPLOYEE_ID	DEPT_ID	DEPT_NAME
200	10	Administration
201	20	Marketing
202	20	Marketing
...		
102	90	Executive
205	110	Accounting
206	110	Accounting

Join Commands

There are two sets of commands or syntax which can be used to make connections between tables in a database:

- Oracle proprietary joins
- ANSI/ISO SQL 99 compliant standard joins

In this course, you will learn to use both sets of join commands.

ANSI

ANSI stands for American National Standards Institute. Founded in 1918, ANSI is a private, non-profit organization that administers and coordinates the U.S. voluntary standardization and conformity assessment system.

The Institute's mission is to enhance both the global competitiveness of U.S. business and the U.S. quality of life by promoting and facilitating voluntary consensus standards and conformity assessment systems, and safeguarding their integrity.

ANSI (cont.)

Reference: <http://www.ansi.org/default.aspx>

SQL

Structured Query Language (SQL) is the information-processing industry-standard language of relational database management systems (RDBMS).

The language was originally designed by IBM in the mid 1970s, came into widespread use in the early 1980s, and became an industry standard in 1986 when it was adopted by ANSI.

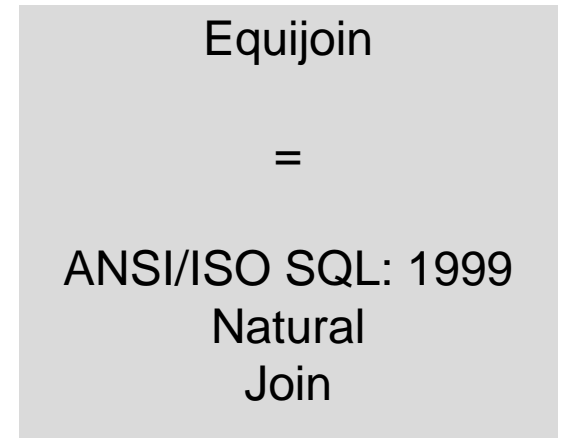
SQL (cont.)

So far there have been three ANSI standardizations of SQL, each one building on the previous one. They are named after the year in which they were first proposed, and are widely known by their short names: ANSI-86, ANSI-92 and ANSI-99.

NATURAL JOIN

An Oracle proprietary equijoin returns all rows whose values match in both tables.

The ANSI/ISO SQL: 1999 join that accomplishes the same result is called a natural join.



NATURAL JOIN (cont.)

A natural join is based on all columns in the two tables that have the same name and selects rows from the two tables that have equal values in all matched columns.

Equijoin

=

ANSI/ISO SQL: 1999
Natural
Join

NATURAL JOIN (cont.)

As shown in the sample code, when using a natural join, it is possible to join the tables without having to explicitly specify the columns in the corresponding table. However, the names and data types in both columns must be the same.

```
SELECT event_id, song_id, cd_number  
FROM d_play_list_items NATURAL JOIN d_track_listings  
WHERE event_id = 105;
```

The WHERE clause was added to apply an additional restriction to one of the tables, to limit the rows of output.

NATURAL JOIN (cont.)

Here is another example:

```
SELECT first_name, last_name, event_date, description  
FROM d_clients NATURAL JOIN d_events;
```

Which column or columns will be used to natural join these two tables ?

Notice that the natural join column does not have to appear in the output.

CROSS JOIN

An Oracle Proprietary Cartesian Product joins each row in one table to every row in the other table.

The ANSI/ISO SQL: 1999 SQL equivalent of the Cartesian product is the cross join.

The results returned from both types of joins are the same. The result set represents all possible row combinations from the two tables. This could potentially be very large!

Cross Join Example

```
SELECT name, event_date, loc_type, rental_fee  
FROM d_events CROSS JOIN d_venues;
```

NAME	EVENT_DATE	LOC_TYPE	RENTAL_FEE
Peters Graduation	14-MAY-2004	Private Home	0
Peters Graduation	14-MAY-2004	Private Home	0
Peters Graduation	14-MAY-2004	Private Home	0
Peters Graduation	14-MAY-2004	School Hall	75/hour
Peters Graduation	14-MAY-2004	National Park	400/flat fee
Peters Graduation	14-MAY-2004	Hotel	300/per person
Vigil Wedding	28-APR-2004	Private Home	0
Vigil Wedding	28-APR-2004	Private Home	0
Vigil Wedding	28-APR-2004	Private Home	0
Vigil Wedding	28-APR-2004	School Hall	75/hour
Vigil Wedding	28-APR-2004	National Park	400/flat fee
Vigil Wedding	28-APR-2004	Hotel	300/per person

Terminology

Key terms used in this lesson included:

- Cross join
- Natural join

Summary

In this lesson, you should have learned how to:

- Construct and execute a natural join using ANSI-99 SQL join syntax
- Create a cross join using ANSI-99 SQL join syntax
- Define the relationship between a cross join and a Cartesian product
- Define the relationship between a natural join and an equijoin

Summary (cont.)

In this lesson, you should have learned how to:

- Explain the importance of having a standard for SQL as defined by ANSI
- Describe a business need for combining information from multiple data sources