ST.XAVIER’S COLLEGE

**Maitighar, Kathmandu**



**DBMS LAB ASSIGNMENT**

**Submitted By:**

**Narayan Panthi**

013BSCCSIT024

**Submitted To:**

**Er. Sanjay Kumar Yadav** ………..…………….

**Lecturer**

**Department of Computer Science**

**Date of Submission:- *2015 -*** *\_\_****\_-*** *\_\_\_*

# JOIN

A SQL join clause combines records from two or more tables in a relational database. It creates a set that can be saved as a table or used as it is. A JOIN is a means for combining fields from two tables (or more) by using values common to each. Join is a special form of cross product of two tables. It is a binary operation that allows combining certain selections and a Cartesian product into one operation. The join operation forms a Cartesian product of its two arguments, performs a selection forcing equality on those attributes that appear in both relation schemas, and finally removes duplicate attributes.

The most common type of join is: SQL INNER JOIN (simple join). An SQL INNER JOIN return all rows from multiple tables where the join condition is met.

# 

# Types of join:

1. Theta Join
2. Equi Join
3. Semi Join
4. Natural Join
5. Outer Join

# Theta Join:

A theta-join is any Cartesian product that's filtered by a condition which compares values from both Tables. That is, the general theta-join form is:

<Table\_1.Column> relator <Table\_2.Column>

where the relator is almost always "=", as in this example:

Sellers.seller\_name = Sales.seller\_name

This special case of theta-join — where the relation is equality — is called an equijoin.

# Natural Join:

This is the most common and general form of join. If we simply say join, it means the natural join. It is same as equi¬join but the difference is that in natural join, the common attribute appears only once. Now, it does not matter which common attribute should be part of the output relation as the values in both are same.

INNER JOIN: Returns all rows when there is at least one match in BOTH tables

LEFT JOIN: Return all rows from the left table, and the matched rows from the right table

RIGHT JOIN: Return all rows from the right table, and the matched rows from the left table

FULL JOIN: Return all rows when there is a match in ONE of the tables

# RENAME OPERATION:

The results of relational algebra are also relations but without any name. The rename operation allows us to rename the output relation. 'rename' operation is denoted with small Greek letter rho ρ.

Notation − ρ x (E)

Where the result of expression E is saved with name of x.

The REMANE operator returns an existing relation under a new name. This is a Mechanism used for renaming both relations and attributes, this renaming is highly useful when one has to compare a set of tuples of the same relation with other tuples of that relation. To do this in SQL, we specify that we want to change the structure of the table using the **ALTER TABLE** command, followed by a command that tells the relational database that we want to rename the column. The exact syntax for each database is as follows:

In MySQL, the SQL syntax for **ALTER TABLE Rename Column** is,

**ALTER TABLE "table\_name"  
Change "column 1" "column 2" ["Data Type"];**

The ALTER TABLE statement is used to add, delete, or modify columns in an existing table.

### SQL ALTER TABLE Syntax

To add a column in a table, use the following syntax:

ALTER TABLE table\_name  
ADD column\_name datatype

To delete a column in a table, use the following syntax (notice that some database systems don't allow deleting a column):

ALTER TABLE table\_name  
DROP COLUMN column\_name

To change the data type of a column:

ALTER TABLE table\_name  
ALTER COLUMN column\_name datatype

# Assignment Operator

The equal sign (=) is the only Transact-SQL assignment operator. In the following example, the @MyCounter variable is created, and then the assignment operator sets @MyCounter to a value returned by an expression.

DECLARE @MyCounter INT;

SET @MyCounter = 1;

The assignment operator can also be used to establish the relationship between a column heading and the expression that defines the values for the column. The following example displays the column headings First Column Heading and Second Column Heading. The string xyz is displayed in the First Column Heading column heading for all rows. Then, each product ID from the Product table is listed in the Second Column Heading column heading.

5. Division Operation

The division operation will return a Relation R(X) that includes all tuples t[X] in R(Z) that appear in R1 in combination with every tuple from R2(Y), where Z = X ∪ Y. The division operator is symbolized by:

* R1(Z) ∻ R2(Y)

The division operator is the most difficult to implement in SQL as no SQL command is given for division operation. The division operator would be seen as the opposite of the Cartesian product operator; just as in standard math, the relation between division and multiplication. Therefore a series of current SQL commands have to be utilized in implementation of the division operator. An example of the SQL implementation of division operator:

SELECT surname, forenames

FROM employee X

WHERE NOT EXISTS

(SELECT ‘X’

FROM employee y

WHERE NOT EXISTS

(SELECT ‘X’

FROM employee z

WHERE x.empno = z.empno

AND y.surname = z.surname))

ORDER BY empno

# 6. Addition Operations

Arithmetic operators can perform arithmetical operations on numeric operands involved. Arithmetic operators are addition(+), subtraction(-), multiplication(\*) and division(/). The + and - operators can also be used in date arithmetic.

Syntax

SELECT <Expression>[arithmetic operator]<expression>...

FROM [table\_name]

WHERE [expression];

# Set-intersection Operation

# 8. Natural Join Operation

A NATURAL JOIN is a JOIN operation that creates an implicit join clause for you based on the common columns in the two tables being joined. Common columns are columns that have the same name in both tables.

A NATURAL JOIN can be an INNER join, a LEFT OUTER join, or a RIGHT OUTER join. The default is INNER join.

If the SELECT statement in which the NATURAL JOIN operation appears has an asterisk (\*) in the select list, the asterisk will be expanded to the following list of columns (in this order):

• All the common columns

• Every column in the first (left) table that is not a common column

• Every column in the second (right) table that is not a common column

An asterisk qualified by a table name (for example, COUNTRIES.\*) will be expanded to every column of that table that is not a common column.

If a common column is referenced without being qualified by a table name, the column reference points to the column in the first (left) table if the join is an INNER JOIN or a LEFT OUTER JOIN. If it is a RIGHT OUTER JOIN, unqualified references to a common column point to the column in the second (right) table.

Syntax

TableExpression NATURAL [ { LEFT | RIGHT } [ OUTER ] | INNER ] JOIN { TableViewOrFunctionExpression | ( TableExpression ) }

# Reference:

<http://www.tutorialspoint.com/sql/sql-using-joins.htm>

<http://www.w3schools.com/sql/sql_join_right.asp>