An outer join extends the result of an inner join by including rows from one table (say Table A) that don't have corresponding rows in another table (say Table B). An important thing to note here is that the outer join operation will not include the rows from Table B that don't have corresponding rows in Table A. In other words, an outer join is unidirectional. But there are situations when you may want a bidirectional outer join, i.e., you want to include all the rows from A and B:

* rows from the result of the inner join
* rows from A that don't have corresponding rows in B (left outer join)
* rows from B that don't have corresponding rows in A (right outer join)

Let's look at an example to understand this further. Consider the following two tables:

SQL> desc part

Name Null? Type

----------------------------------------- -------- -----------------

PART\_ID NOT NULL VARCHAR2(4)

SUPPLIER\_ID VARCHAR2(4)

SQL> select \* from part;

PART SUPP

---- ----

P1 S1

P2 S2

P3

P4

SQL> desc supplier

Name Null? Type

----------------------------------------- -------- -----------------

SUPPLIER\_ID NOT NULL VARCHAR2(4)

SUPPLIER\_NAME NOT NULL VARCHAR2(20)

SQL> select \* from supplier;

SUPP SUPPLIER\_NAME

---- --------------------

S1 Supplier#1

S2 Supplier#2

S3 Supplier#3

Notice above that there are two parts (P3 and P4) that don't have a supplier yet. Also, there is a supplier (S3) who doesn't yet supply any part.

**Result of Inner Join**

Generate a report of all the parts and their corresponding suppliers. So we performed a join of these two tables, which gave the following result:

SQL> select p.part\_id, s.supplier\_name

2 from part p, supplier s

3 where p.supplier\_id = s.supplier\_id;

PART SUPPLIER\_NAME

---- --------------------

P1 Supplier#1

P2 Supplier#2

The join shown above is an inner join, which results in just the rows that have corresponding rows in both tables. Therefore, the parts that don't have a supplier, or the suppliers that don't supply any part are excluded from the result set.

**Result of Outer Join**

If we want all parts to be listed in the result set, irrespective of whether they are supplied by any supplier or not, then we need to perform an outer join.

**To perform a left outer join, the WHERE clause is WHERE table1.column1 = table2.column2 (+);**

SQL> select p.part\_id, s.supplier\_name

2 from part p, supplier s

3 where p.supplier\_id = s.supplier\_id (+);

PART SUPPLIER\_NAME

---- --------------------

P1 Supplier#1

P2 Supplier#2

P3

P4

The outer join above lists all of the parts. For the parts that don't have a corresponding supplier, null values are displayed for the SUPPLIER\_NAME column. However, not all the suppliers are displayed. Since supplier S3 doesn't supply any parts, it gets excluded from the result set of the above outer join.

If we want all the suppliers listed in the result set, irrespective of whether they supply any part or not, we need to perform an outer join like the following:

**To perform a right outer join, you switch the position of the outer join operator to the left of the equality operator and the WHERE clause becomes WHERE table1.column1 (+) = table2.column2**

SQL> select p.part\_id, s.supplier\_name

2 from part p, supplier s

3 where p.supplier\_id (+) = s.supplier\_id;

PART SUPPLIER\_NAME

---- --------------------

P1 Supplier#1

P2 Supplier#2

Supplier#3

The outer join above lists all the suppliers. For the suppliers that don't supply any part, null values are displayed for the PART\_ID column. However, not all the parts are displayed. Since parts P3 and P4 are not supplied by any suppliers, they get excluded from the result set of the above outer join.

**Full Outer Join**

If we want all the parts (irrespective of whether they are supplied by any supplier or not), and all the suppliers (irrespective of whether they supply any part or not) listed in the same result set, we have a problem. That's because the traditional outer join (using the '+' operator) is unidirectional, and you can't put (+) on both sides in the join condition. The following will result in an error:

SQL> select p.part\_id, s.supplier\_name

2 from part p, supplier s

3 where p.supplier\_id (+) = s.supplier\_id (+);

where p.supplier\_id (+) = s.supplier\_id (+)

\*

ERROR at line 3:

ORA-01468: a predicate may reference only one outer-joined table

The workaround involves two outer join queries combined by a UNION operator, as in the following example:

SQL> select p.part\_id, s.supplier\_name

2 from part p, supplier s

3 where p.supplier\_id = s.supplier\_id (+)

4 union

5 select p.part\_id, s.supplier\_name

6 from part p, supplier s

7 where p.supplier\_id (+) = s.supplier\_id;

PART SUPPLIER\_NAME

---- --------------------

P1 Supplier#1

P2 Supplier#2

P3

P4

Supplier#3