**The UNION [ALL], INTERSECT, MINUS Operators**

You can combine multiple queries using the set operators UNION, UNION ALL, INTERSECT, and MINUS. All set operators have equal precedence. If a SQL statement contains multiple set operators, Oracle evaluates them from the left to right if no parentheses explicitly specify another order.

**UNION Example**

The following statement combines the results with the UNION operator, which eliminates duplicate selected rows.

select empno,ename,sal from emp   
UNION  
select empno,ename,salary from oldemp

What if you need to select rows from two tables, but tables have different columns?   
In this situation you have to use TO\_CHAR function to fill up missing columns.

For Example

This statement shows that you must match datatype (using the TO\_CHAR function) when columns do not exist in one or the other table:

select empno, ename, sal, to\_char(null) as “Transfer Date” from emp   
 UNION  
select empno,ename,to\_char(null) as “Sal”,tdate from oldemp;

EMPNO     ENAME     SAL       Transfer Date  
-----     -----     ------    -------------  
101       Sami      5000        
102       Smith              11-jul-2000  
201       Tamim              10-AUG-2000  
209       Ravi      2400

**UNION ALL Example**

The UNION operator returns only distinct rows that appear in either result, while the UNION ALL operator returns all rows. The UNION ALL operator does not eliminate duplicate selected rows:

select empno,ename from emp   
union all  
select empno,ename from oldemp;

**INTERSECT Example**

The following statement combines the results with the INTERSECT operator, which returns only those rows returned by both queries:

SELECT empno FROM emp  
INTERSECT  
SELECT empno FROM oldemp;

**MINUS Example**

The following statement combines results with the MINUS operator, which returns only rows returned by the first query but not by the second:

SELECT empno FROM emp  
MINUS  
SELECT empno FROM oldemp;

**SORTING QUERY RESULTS**

To sort query result you can use ORDER BY clause in SELECT statement.

Sorting Examples.

The following query sorts the employees according to ascending order of salaries.

select \* from emp order by sal;

The following query sorts the employees according to descending order of salaries.

select \* from emp order by sal desc;

The following query sorts the employees according to ascending order of names.

select \* from emp order by ename;

The following query first sorts the employees according to ascending order of names.If names are equal then sorts employees on descending order of salaries.

select \* from emp order by ename, sal desc;

You can also specify the positions instead of column names. Like in the following query,which shows employees according to ascending order of their names.

select \* from emp order by 2;

The following query first sorts the employees according to ascending order of salaries.

If salaries are equal then sorts employees on ascending order of names

select \* from emp order by 3, 2;