**More on SELECT**

It is possible to include the other following items in the SELECT command line:

Arithmetic Expressions

Column Aliases

Concatenated Columns

Literals

Functions

All of these options allow the user to query data and manipulate it for query purposes; for example, performing calculations, joining columns together, or displaying literal text strings.

**Arithmetic Expressions**

Arithmetic Expressions may contain column names, constant numeric values and the following arithmetic operators:

**Operator Description**

* add
* subtract
* multiply
* divide

The priority is \*, / first, then +, - second (left to right if there are several operators with the same priority).

Parentheses may be used to specify the order in which operators are executed, if, for example, addition is required before multiplication.

Shown below, however, is an example of normal use of the \* symbol.

SELECT ENAME, SAL\*12, COMM

FROM EMP;

**SAL \*12**

**- - - - - - -**

**9600**

**19200**

**15000**

**35700**

**15000**

**34200**

**29400**

**36000**

**60000**

**18000**

**13200**

**11400**

**36000**

**15600**

**ENAME**

**- - - - - - -**

**SMITH**

**ALLEN**

**WARD**

**JONES**

**MARTIN**

**BLAKE**

**CLARK**

**SCOTT**

**KING**

**TURNER**

**ADAMS**

**JAMES**

**FORD**

**MILLER**

**COMM**

**- - - - - -**

**300**

**500**

**1400**

**0**

**Column Aliases**

When displaying the result of a query, SQL\*Plus normally uses the selected column's name as the heading. In many cases it may be cryptic or mean little. You can change a column's heading by using an *Alias* .

A column alias gives a column an alternative heading on output. Specify the alias after the column name in the SELECT list. By default alias headings will be forced to upper case and cannot contain blank spaces, unless the alias is enclosed in double quotes (" ").

To display the column heading ANNSAL for annual salary instead of SAL\*12 as in our previous example we can use a column alias:

SELECT ENAME, SAL\*12 ANNSAL, COMM

FROM EMP;

This will produce the following output:

**COMM**

**- - - - - -**

**300**

**500**

**1400**

**0**

**ANNSAL**

**- - - - - - -**

**9600**

**19200**

**15000**

**35700**

**15000**

**34200**

**29400**

**36000**

**60000**

**18000**

**13200**

**11400**

**36000**

**15600**

**ENAME**

**- - - - - - -**

**SMITH**

**ALLEN**

**WARD**

**JONES**

**MARTIN**

**BLAKE**

**CLARK**

**SCOTT**

**KING**

**TURNER**

**ADAMS**

**JAMES**

**FORD**

**MILLER**

**The Concatenation Operator**

The Concatenation Operator (| |) allows columns to be linked to other columns, arithmetic expressions or constant values to create a character expression. Columns on either side of the operator are combined to make one single column.

For example, to combine EMPNO and ENAME and give the expression the alias EMPLOYEE, we would use the command:

SELECT EMPNO | | ENAME EMPLOYEE

FROM EMP;

**EMPLOYEE**

**- - - - - - -**

**SMITH**

**ALLEN**

**WARD**

**JONES**

**MARTIN**

**BLAKE**

**CLARK**

**SCOTT**

**KING**

**TURNER**

**ADAMS**

**JAMES**

**FORD**

**MILLER**

**-- - - -**

**7369**

**7499**

**7521**

**7566**

**7654**

**7698**

**7782**

**7788**

**7839**

**7844**

**7876**

**7900**

**7902**

**7934**

**Literals**

A Literal is any character, expression, or number included on the SELECT list which is not a column name or a column alias.

A literal in the SELECT list is output for each row returned. Literal strings of free format text can be included in the query result, and are treated like a column in the SELECT list.

Date and character literals must be included in single quotes ('); number literals do not need single quotes.

The example given below contains literals selected with concatenation and a column alias:

SELECT EMPNO| | ENAME EMPLOYEE, 'WORKS IN DEPARTMENT', DEPTNO

FROM EMP;

**EMPLOYEE**

**DEPTNO**

**- - - - - - -**

**20**

**30**

**30**

**20**

**30**

**30**

**10**

**20**

**10**

**30**

**20**

**30**

**20**

**10**

**'WORKS IN DEPARTMENT'**

**- - - - - - - - - - - - - - - - - - - - -**

**WORKS IN DEPARTMENT**

**WORKS IN DEPARTMENT**

**WORKS IN DEPARTMENT**

**WORKS IN DEPARTMENT**

**WORKS IN DEPARTMENT**

**WORKS IN DEPARTMENT**

**WORKS IN DEPARTMENT**

**WORKS IN DEPARTMENT**

**WORKS IN DEPARTMENT**

**WORKS IN DEPARTMENT**

**WORKS IN DEPARTMENT**

**WORKS IN DEPARTMENT**

**WORKS IN DEPARTMENT**

**WORKS IN DEPARTMENT**

**- - - - - - -**

**SMITH**

**ALLEN**

**WARD**

**JONES**

**MARTIN**

**BLAKE**

**CLARK**

**SCOTT**

**KING**

**TURNER**

**ADAMS**

**JAMES**

**FORD**

**MILLER**

**-- - - -**

**7369**

**7499**

**7521**

**7566**

**7654**

**7698**

**7782**

**7788**

**7839**

**7844**

**7876**

**7900**

**7902**

**7934**

**Functions**

Functions have a self-descriptive name followed by various parameters (arguments) in parentheses. They transform the column values or literals to which they apply. They are covered in more detail later.

**Handling Null Values**

A *null value* is a value which is unavailable, unassigned, unknown, or inapplicable. A null value is **not** the same as zero. Zero is a number. A null value in SQL is truly non-existent and takes up no disc space at all. Null values are handled correctly by SQL.

If any column values in an expression are null, the result is null. For example, in the following statement only *Salesmen* have a remuneration result:

SELECT ENAME, SAL\*12 + COMM ANNUAL\_SAL

FROM EMP;

**ANNUAL\_SAL**

**- - - - - - - - - - -**

**19500**

**15500**

**16400**

**18000**

**ENAME**

**- - - - - - -**

**SMITH**

**ALLEN**

**WARD**

**JONES**

**MARTIN**

**BLAKE**

**CLARK**

**SCOTT**

**KING**

**TURNER**

**ADAMS**

**JAMES**

**FORD**

**MILLER**

In order to achieve a result for all employees in the EMP table, it is necessary to convert the null value to a number. We use the NVL function to convert a null value to a non-null value.

***Handling Null Values...contd***

Use the NVL function to convert null values from the previous statement to zero :

SELECT ENAME, SAL\*12+NVL(COMM,0) ANNUAL\_SAL

FROM EMP;

**ANNUAL\_SAL**

**- - - - - - - - - - -**

**9600**

**19500**

**15500**

**35700**

**16400**

**34200**

**29400**

**36000**

**60000**

**18000**

**13200**

**11400**

**36000**

**15600**

**ENAME**

**- - - - - - -**

**SMITH**

**ALLEN**

**WARD**

**JONES**

**MARTIN**

**BLAKE**

**CLARK**

**SCOTT**

**KING**

**TURNER**

**ADAMS**

**JAMES**

**FORD**

**MILLER**

NVL expects two arguments :

1. an expression

2. a non-null value

Therefore in the above example, COMM satisfies the first of the above and the value zero satisfies the second. Note that you can use the NVL function to convert a null number, date or even character string to another number date or character string, as long as the data types match :

NVL (DATECOLUMN, '01-AUG-08')

NVL (NUMBERCOLUMN, 9)

NVL (CHARCOLUMN, 'STRING')

**Preventing the Selection of Duplicate Rows**

Unless you indicate otherwise, SQL\*Plus displays the results of a query without eliminating duplicate entries.

To list all department numbers in the EMP table, enter:

SELECT DEPTNO

FROM EMP;

**DEPTNO**

**- - - - - - -**

**20**

**30**

**30**

**20**

**30**

**30**

**10**

**20**

**10**

**30**

**20**

**30**

**20**

**10**

**The Distinct Clause**

To eliminate duplicate values in the result shown on the previous page, include the DISTINCT clause in your SELECT command:

SELECT DISTINCT DEPTNO

FROM EMP;

**DEPTNO**

**- - - - - - -**

**10**

**20**

**30**

***Preventing the Selection of Duplicate Rows...contd***

Multiple columns may be specified in the DISTINCT clause and the DISTINCT works on all selected columns.

For example, say we want to display all distinct values of DEPTNO and JOB, then we would enter :

SELECT DISTINCT DEPTNO, JOB

FROM EMP;

**DEPTNO JOB**

**- - - - - - - - - - - - - - -**

**10 CLERK**

**10 MANAGER**

**10 PRESIDENT**

**20 ANALYST**

**20 CLERK**

**20 MANAGER**

**30 CLERK**

**30 MANAGER**

**30 SALESMAN**

This displays a list of all the different combinations of jobs and department numbers.

Note that the DISTINCT command word can only be referred to once and must immediately follow the SELECT command word.

**The ORDER BY Clause**

Usually the order of rows returned in a query result is undefined. The ORDER BY clause may be used to sort the rows. If used, ORDER BY must always be the last clause in the SELECT command.

To sort by ENAME, enter:

SELECT ENAME, JOB, SAL\*12, DEPTNO

FROM EMP

ORDER BY ENAME;

**DEPTNO**

**- - - - - - -**

**20**

**30**

**30**

**10**

**20**

**30**

**20**

**10**

**30**

**10**

**20**

**20**

**30**

**30**

**SAL\*12**

**- - - - - - -**

**13200**

**19200**

**34200**

**29400**

**36000**

**11400**

**35700**

**60000**

**15000**

**15600**

**36000**

**9600**

**18000**

**15000**

**JOB**

**- - - - - - - - -**

**CLERK**

**SALESMAN**

**MANAGER**

**MANAGER**

**ANALYST**

**CLERK**

**MANAGER**

**PRESIDENT**

**SALESMAN**

**CLERK**

**ANALYST**

**CLERK**

**SALESMAN**

**SALESMAN**

**ENAME**

**- - - - - - -**

**ADAMS**

**ALLEN**

**BLAKE**

**CLARK**

**FORD**

**JAMES**

**JONES**

**KING**

**MARTIN**

**MILLER**

**SCOTT**

**SMITH**

**TURNER**

**WARD**

**Default Ordering of Data**

The default sort order is ASCENDING :

Numeric values lowest first

Date values earliest first

Character values alphabetically

**Reversing the Default Order**

To reverse this order, the command word DESC is specified after the column name in the ORDER BY clause.

To reverse the order of the HIREDATE column, so that the latest dates are displayed first, enter:

SELECT ENAME, JOB, HIREDATE

FROM EMP

ORDER BY HIREDATE DESC;

**HIREDATE**

**- - - - - - - - -**

**23-JUL-14**

**09-JUL-14**

**11-JUN-14**

**04-JUN-14**

**04-JUN-14**

**14-MAY-14**

**26-MAR-14**

**05-MAR-14**

**05-DEC-13**

**05-DEC-13**

**21-NOV-13**

**31-OCT-13**

**15-AUG-13**

**13-JUN-13**

**JOB**

**- - - - - - - - - -**

**CLERK**

**PRESIDENT**

**MANAGER**

**SALESMAN**

**CLERK**

**MANAGER**

**SALESMAN**

**ANALYST**

**SALESMAN**

**ANALYST**

**CLERK**

**MANAGER**

**SALESMAN**

**CLERK**

**ENAME**

**- - - - - - -**

**JAMES**

**KING**

**BLAKE**

**TURNER**

**ADAMS**

**CLARK**

**WARD**

**SCOTT**

**MARTIN**

**FORD**

**MILLER**

**JONES**

**ALLEN**

**SMITH**

**Ordering by Many Columns**

It is possible to ORDER BY more than one column. The limit is in fact the number of columns in the table. In the ORDER BY clause specify the columns to order by, separated by commas. If any or all are to be reversed specify DESC after any or each column.

To order by two columns, for example, and display in reverse order of salary, we may enter:

SELECT DEPTNO, JOB, ENAME

FROM EMP

ORDER BY DEPTNO, SAL DESC;

**DEPTNO JOB ENAME**

**- - - - - - - - - - - - - - - - - - - - - - - - -**

**10 PRESIDENT KING**

**10 MANAGER CLARK**

**10 CLERK MILLER**

**20 ANALYST SCOTT**

**20 ANALYST FORD**

**20 MANAGER JONES**

**20 CLERK ADAMS**

**20 CLERK SMITH**

**30 MANAGER BLAKE**

**30 SALESMAN ALLEN**

**30 SALESMAN TURNER**

**30 SALESMAN WARD**

**30 SALESMAN MARTIN**

**30 CLERK JAMES**

To order by a column *it is not necessary to have SELECTed it* !! Hence, even though the above output is ordered on salary, no overt reference to salary is made in the response.

**SELECT Summary**

We have now met the following clauses in the SELECT command :

SELECT *[DISTINCT] {\*, column [alias], ....}*

FROM *table*

WHERE *condition (s)*

ORDER BY *{column, expr} [ASC/DESC]*  ;

Let us again remind ourselves of the meaning of the clauses we have covered :

**Clause Meaning**

SELECT selects at least one column

Alias may be used for columns on select list only

* denotes all columns

DISTINCT can be used to eliminate duplicates

FROM *Table* denotes the *table* where the columns originate

WHERE restricts query to rows that meet a *condition*. It

may contain column values, expressions and

literals

AND/OR may be used to construct more complex

conditions, AND takes priority over OR

( ) can be used to force priority

ORDER BY always appears last. Specifies sort order. One

or more columns may be specified

ASC ascending order is the default sorting order and

need not be specified

DESC reverses the default sorting order and must be

specified after column name

Clauses may be entered on different lines in the buffer and tabulation used for clarity and ease of editing.

**More on SELECT Worksheet**

The following exercise is intended to introduce, in a workshop setting, all topics covered in the lecture sessions on Introduction to SQL. After each question, a sample of the expected output for a correct solution is given in each case as a guideline.

**Workshop**

1. List all department numbers and names in department name order.

**DEPTNO DNAME**

**- - - - - - - - - - - - - - -**

**10 ACCOUNTING**

**40 OPERATIONS**

**20 RESEARCH**

**30 SALES**

2. Display all the different job types.

**JOB**

**- - - - - - - - -**

**ANALYST**

**CLERK**

**MANAGER**

**PRESIDENT**

**SALESMAN**

3. List the details of the employees in departments 10 and 20 in alphabetical

order of name.

**EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO**

**- - - - - - - - - - - - - - - - - - - - -- - - - - - - - - - - - - - - - - -- - - -**

**7876 ADAMS CLERK 7788 04-JUN-14 1100 20**

**7782 CLARK MANAGER 7839 14-MAY-14 2450 10**

**7902 FORD ANALYST 7566 05-DEC-13 3000 20**

**7566 JONES MANAGER 7839 31-OCT-13 2975 20**

**7839 KING PRESIDENT 09-JUL-14 5000 10**

**7934 MILLER CLERK 7782 21-NOV-13 1300 10**

**7788 SCOTT ANALYST 7566 05-MAR-14 3000 20**

**7369 SMITH CLERK 7902 13-JUN-13 800 20**

4. Display name and total remuneration for all employees.

**REMUNERATION**

**- - - - - - - - - - - - - -**

**9600**

**19500**

**15500**

**35700**

**16400**

**34200**

**29400**

**36000**

**60000**

**18000**

**13200**

**11400**

**36000**

**15600**

**ENAME**

**- - - - - - -**

**SMITH**

**ALLEN**

**WARD**

**JONES**

**MARTIN**

**BLAKE**

**CLARK**

**SCOTT**

**KING**

**TURNER**

**ADAMS**

**JAMES**

**FORD**

**MILLER**

**13 records selected.**

5. Display name, annual salary and commission of all salespeople whose

monthly salary is greater than their commission. The output should be

ordered by salary, highest first. If two or more employees have the same

salary, sort by employee name, within the highest salary order.

**ENAME ANNUAL\_SAL COMM**

**- - - - - - - - - - - - - - - - - - - - - - - - - - -**

**ALLEN 19200 300**

**TURNER 18000 0**

**WARD 15000 500**

6. Select data as displayed.

**SMITH HAS HELD THE POSITION OF CLERK IN DEPT 20 SINCE 13-JUN-13**

**ALLEN HAS HELD THE POSITION OF SALESMAN IN DEPT 30 SINCE 15-AUG-13**

**WARD HAS HELD THE POSITION OF SALESMAN IN DEPT 30 SINCE 26-MAR-14**

**JONES HAS HELD THE POSITION OF MANAGER IN DEPT 20 SINCE 31-OCT-13**

**MARTIN HAS HELD THE POSITION OF SALESMAN IN DEPT 30 SINCE 05-DEC-13**

**BLAKE HAS HELD THE POSITION OF MANAGER IN DEPT 30 SINCE 11-JUN-14**

**CLARK HAS HELD THE POSITION OF MANAGER IN DEPT 10 SINCE 14-MAY-14**

**SCOTT HAS HELD THE POSITION OF ANALYST IN DEPT 20 SINCE 05-MAR-14**

**KING HAS HELD THE POSITION OF PRESIDENT IN DEPT 10 SINCE 09-JUL-14**

**TURNER HAS HELD THE POSITION OF SALESMAN IN DEPT 30 SINCE 04-JUN-14**

**ADAMS HAS HELD THE POSITION OF CLERK IN DEPT 20 SINCE 04-JUN-14**

**JAMES HAS HELD THE POSITION OF CLERK IN DEPT 30 SINCE 23-JUL-14**

**FORD HAS HELD THE POSITION OF ANALYST IN DEPT 20 SINCE 05-DEC-13**

**MILLER HAS HELD THE POSITION OF CLERK IN DEPT 10 SINCE 21-NOV-13**

7. Display all employee information in ascending order of manager number.

**EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO**

**- - - - - - - - - - - - - - - - - - - -- - - - - - - - - - - - - - - - - -- - - -**

**7839 KING PRESIDENT 9999 09-JUL-14 5000 10**

**7788 SCOTT ANALYST 7566 05-MAR-14 3000 20**

**7902 FORD ANALYST 7566 05-DEC-13 3000 20**

**7499 ALLEN SALESMAN 7698 15-AUG-13 1600 300 30**

**7844 TURNER SALESMAN 7698 04-JUN-13 1500 0 30**

**7900 JAMES CLERK 7698 23-JUL-14 950 30**

**7521 WARD SALESMAN 7698 26-MAR-14 1250 500 30**

**7654 MARTIN SALESMAN 7698 05-DEC-13 1250 1400 30**

**7934 MILLER CLERK 7782 21-NOV-13 1300 10**

**7876 ADAMS CLERK 7788 04-JUN-14 1100 20**

**7566 JONES MANAGER 7839 31-OCT-13 2975 20**

**7698 BLAKE MANAGER 7839 11-JUN-14 2850 30**

**7782 CLARK MANAGER 7839 14-MAY-14 2450 10**

**7369 SMITH CLERK 7902 13-JUN-13 800 20**