**Restricting and Sorting Data**

**Limiting Rows That Are Selected**

* Restrict the rows that are returned by using the WHERE clause:
* The WHERE clause follows the FROM clause.
* The WHERE clause can compare values in columns, literal, arithmetic expressions, or functions. It consists of three elements:
  + Column name
  + Comparison condition
  + Column name, constant, or list of values

Note: You cannot use column alias in the WHERE clause.

**Character Strings and Dates**

* **Character strings and date** values are **enclosed within single quotation marks**.
* **Character** values are **case-sensitive** and **date values are format-sensitive**.
* The default date display format is **DD-MON-RR.**
* Oracle databases store dates in an internal numeric format, representing the century, year, month, day, hours, minutes, and seconds. The default date display is in the DD-MON-RR format.

Note: The symbols **!=** and **^=** can also represent the not equal to condition.

Use the **BETWEEN** operator to display rows based on a range of values.

Use the **IN** operator to test for values in a list:

Note: The set of values can be specified in any **random order**—for example, (201,100,101). The IN operator can be used with **any data type**.

If **characters or dates** are used in a list, they must be enclosed within **single quotation marks ('').**

**Pattern Matching Using the LIKE Operator**

• Use the LIKE operator to perform wildcard searches of valid search string values.

• Search conditions can contain either literal characters or numbers:

– **%** denotes zero or more characters.

– **\_** denotes one character.

Test for nulls with the IS NULL operator.

Three logical operators are available in SQL:

• AND

• OR

• NOT

All the examples so far have specified only one condition in the WHERE clause. You can use several conditions in a single WHERE clause using the AND and OR operators.

1. Precedence of the AND Operator: Example
   1. In this example, there are two conditions:
      1. - The first condition is that the department ID is 80 and the salary is greater than $10,000.
      2. - The second condition is that the department ID is 60.

Therefore, the SELECT statement reads as follows:

“Select the row if an employee’s department ID is 80 and earns more than $10,000, or if the employee’s department ID is 60.”

1. Using Parentheses: Example
   1. In this example, there are two conditions:
      1. - The first condition is that the department ID is 80 or 60.
      2. - The second condition is that the salary is greater than $10,000.

Therefore, the SELECT statement reads as follows:

“Select the row if an employee’s department ID is 80 or 60, and if the employee earns more than $10,000.”

Sort the retrieved rows with the ORDER BY clause:

• ASC: Ascending order, default

• DESC: Descending order

The default sort order is ascending:

• **Numeric** values are displayed with the **lowest values** first (for example, 1 to 999).

• **Date values** are displayed with the **earliest value** first (for example, 01-JAN-92 before 01-JAN-95).

• **Character** values are displayed in the alphabetical order (for example, **“A” first and “Z” last**).

• **Null** values are displayed last for ascending sequences and first for descending sequences.

• You can **also sort** by a **column** that is **not** in the SELECT list