DATABASE DESIGN & IMPLEMENTATION

ICT Skills

Objectives

- Construct query to sort a result set in ascending or descending order
- Construct a query to order a result set using a column alias
- Construct a query to order a result set for single or multiple columns

ORDER BY Clause

- Information sorted ascending order is familiar to us.
- It’s what makes looking up a number in a phone book, finding a word in a dictionary, or locating a house by its street address relatively easy.
- SQL uses the ORDER BY clause to order data.
- The ORDER BY clause can specify several ways in which to order rows returned in a query.

ORDER BY Clause

- The default sort order is ascending.
- Numeric values are displayed lowest to highest.
- Date values are displayed with the earliest value first
- Character values are displayed in alphabetical order
- Null values are displayed last in ascending order and first in descending order
- NULLS FIRST specifies that NULL values should be returned before non-NULL values.
- NULLS LAST FIRST specifies that NULL values should be returned after non-NULL values.
- You can sort by more than one column (separate with commas).

ORDER BY Clause

- The following employees example uses the ORDER BY clause to order hire_date in ascending (default) order.
- Note: The ORDER BY clause must be the last clause of the SQL statement.

```
SELECT last_name, hire_date
FROM employees
ORDER BY hire_date;
```

LAST_NAME	HIRE_DATE
King	17/Jun/1987
Whalen	17/Sep/1987
Kochhar	21/Sep/1989
Hunold	03/Jan/1990
Ernst	21/May/1991
De Haan	13/Jan/1993
Gietz	07/Jun/1994
Higgins	07/Jun/1994
Rajs	17/Oct/1995
Hartstein	17/Feb/1996

ORDER BY Clause

- You can reverse the default order in the ORDER BY clause to descending order by specifying the DESC keyword after the column name in the ORDER BY clause.

```
SELECT last_name, hire_date
FROM employees
ORDER BY hire_date DESC;
```

LAST_NAME	HIRE_DATE
Zlotkey	29/Jan/2000
Mourgos	16/Nov/1999
Grant	24/May/1999
Lorentz	07/Feb/1999
Vargas	09/Jul/1998
Taylor	24/Mar/1998
Matos	15/Mar/1998
Fay	17/Aug/1997
Davies	29/Jan/1997
Abel	11/May/1996

ORDER BY Clause

- You can order data by using a column alias.
- The alias used in the SELECT statement is referenced in the ORDER BY clause.

```
SELECT last_name, hire_date AS "Date
   Started"
FROM employees
ORDER BY "Date Started";
```

LAST_NAME	Date Started
King	17/Jun/1987
Whalen	17/Sep/1987
Kochhar	21/Sep/1989
Hunold	03/Jan/1990
Ernst	21/May/1991
De Haan	13/Jan/1993
Gietz	07/Jun/1994
Higgins	07/Jun/1994
Rajs	17/Oct/1995
Hartstein	17/Feb/1996

ORDER BY Clause

- It is also possible to use the ORDER BY clause to order output by a column that is not listed in the SELECT clause.
- In the following example, the data is sorted by the last_name column even though this column is not listed in the SELECT statement.

```
SELECT employee_id, first_name
FROM employees
WHERE employee_id < 105
ORDER BY last_name;
```

EMPLOYEE_ID	FIRST_NAME
102	Lex
104	Bruce
103	Alexander
100	Steven
101	Neena

Order of Execution

- The order of execution of a SELECT statement is as follows:
  - *FROM clause: locates the table that contains the data*
  - *WHERE clause: restricts the rows to be returned*
  - *SELECT clause: selects from the reduced data set the columns requested*
  - *ORDER BY clause: orders the result set*