

SQL - SORTING RESULTS

<http://www.tutorialspoint.com/sql/sql-sorting-results.htm>

Copyright © tutorialspoint.com

The SQL **ORDER BY** clause is used to sort the data in ascending or descending order, based on one or more columns. Some database sorts query results in ascending order by default.

Syntax:

The basic syntax of ORDER BY clause which would be used to sort result in ascending or descending order is as follows:

```
SELECT column-list
FROM table_name
[WHERE condition]
[ORDER BY column1, column2, .. columnN] [ASC | DESC];
```

You can use more than one column in the ORDER BY clause. Make sure whatever column you are using to sort, that column should be in column-list.

Example:

Consider the CUSTOMERS table having the following records:

| ID | NAME | AGE | ADDRESS | SALARY |
|----|----------|-----|-----------|----------|
| 1 | Ramesh | 32 | Ahmedabad | 2000.00 |
| 2 | Khilan | 25 | Delhi | 1500.00 |
| 3 | kaushik | 23 | Kota | 2000.00 |
| 4 | Chaitali | 25 | Mumbai | 6500.00 |
| 5 | Hardik | 27 | Bhopal | 8500.00 |
| 6 | Komal | 22 | MP | 4500.00 |
| 7 | Muffy | 24 | Indore | 10000.00 |

Following is an example, which would sort the result in ascending order by NAME and SALARY:

```
SQL> SELECT * FROM CUSTOMERS
      ORDER BY NAME, SALARY;
```

This would produce the following result:

| ID | NAME | AGE | ADDRESS | SALARY |
|----|----------|-----|-----------|----------|
| 4 | Chaitali | 25 | Mumbai | 6500.00 |
| 5 | Hardik | 27 | Bhopal | 8500.00 |
| 3 | kaushik | 23 | Kota | 2000.00 |
| 2 | Khilan | 25 | Delhi | 1500.00 |
| 6 | Komal | 22 | MP | 4500.00 |
| 7 | Muffy | 24 | Indore | 10000.00 |
| 1 | Ramesh | 32 | Ahmedabad | 2000.00 |

Following is an example, which would sort the result in descending order by NAME:

```
SQL> SELECT * FROM CUSTOMERS
      ORDER BY NAME DESC;
```

This would produce the following result:

| ID | NAME | AGE | ADDRESS | SALARY |
|----|------|-----|---------|--------|
|----|------|-----|---------|--------|

| ID | NAME | AGE | ADDRESS | SALARY |
|----|----------|-----|-----------|----------|
| 1 | Ramesh | 32 | Ahmedabad | 2000.00 |
| 7 | Muffy | 24 | Indore | 10000.00 |
| 6 | Komal | 22 | MP | 4500.00 |
| 2 | Khilan | 25 | Delhi | 1500.00 |
| 3 | kaushik | 23 | Kota | 2000.00 |
| 5 | Hardik | 27 | Bhopal | 8500.00 |
| 4 | Chaitali | 25 | Mumbai | 6500.00 |

To fetch the rows with own preferred order, the SELECT query would as follows:

```
SQL> SELECT * FROM CUSTOMERS
      ORDER BY (CASE ADDRESS
      WHEN 'DELHI' THEN 1
      WHEN 'BHOPAL' THEN 2
      WHEN 'KOTA' THEN 3
      WHEN 'AHMADABAD' THEN 4
      WHEN 'MP' THEN 5
      ELSE 100 END) ASC, ADDRESS DESC;
```

This would produce the following result:

| ID | NAME | AGE | ADDRESS | SALARY |
|----|----------|-----|-----------|----------|
| 2 | Khilan | 25 | Delhi | 1500.00 |
| 5 | Hardik | 27 | Bhopal | 8500.00 |
| 3 | kaushik | 23 | Kota | 2000.00 |
| 6 | Komal | 22 | MP | 4500.00 |
| 4 | Chaitali | 25 | Mumbai | 6500.00 |
| 7 | Muffy | 24 | Indore | 10000.00 |
| 1 | Ramesh | 32 | Ahmedabad | 2000.00 |

This will sort customers by ADDRESS in your own order of preference first and in a natural order for the remaining addresses. Also remaining Addresses will be sorted in the reverse alpha order.