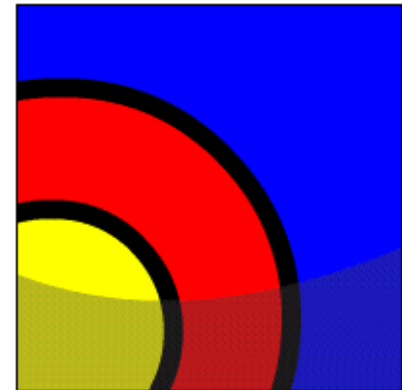


Sorting Rows

What Will I Learn?

In this lesson, you will learn to:

- Construct a query to sort a results set in ascending or descending order
- State the order in which expressions are evaluated and calculated based on the rules of precedence
- Construct a query to order a results set using a column alias
- Construct a query to order a results set for single or multiple columns





Why Learn It?

By nature, most of us need order in our lives. Imagine if each time you had dinner, you had to look in every kitchen drawer or cabinet to find a knife and a fork? Ordering, grouping, and sorting makes finding things easier.

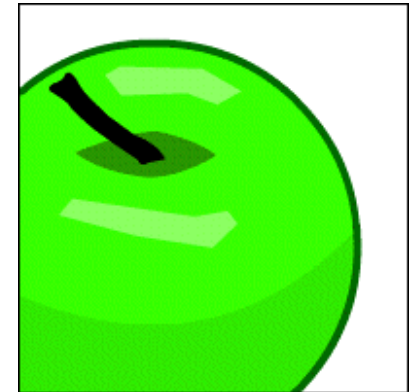
Biologists group animals in phyla, astronomers order brightness of stars by magnitude, and programmers organize Java code in classes. For database design, business functions are ordered by entities and attributes; in database information, SQL uses the ORDER BY clause.

Being able to sort results is a convenient feature in SQL and enables programmers to display information in many different ways.

Tell Me / Show Me

ORDER BY

Information sorted in ascending order is familiar to most of us. It's what makes looking up a number in a phone book, finding a word in the dictionary, or locating a house by its street address relatively easy.



SQL uses the ORDER BY clause following the FROM clause to order data. The ORDER BY clause can specify several ways in which to order rows returned in a query. The DJ on Demand example uses the ORDER BY clause to order the years in ascending (default) order.

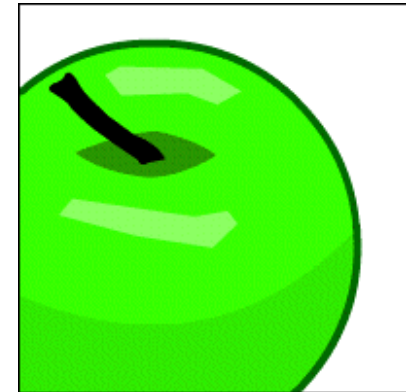
Tell Me / Show Me

ORDER BY

Important note: The ORDER BY clause must be the last clause of the SQL statement.

```
SELECT title, year  
FROM d_cds  
ORDER BY year;
```

| TITLE | YEAR |
|--------------------------------|------|
| The Celebrants Live in Concert | 1997 |
| Graduation Songbook | 1998 |
| Songs from My Childhood | 1999 |
| Party Music for All Occasions | 2000 |
| Carpe Diem | 2000 |
| Here Comes the Bride | 2001 |
| Back to the Shire | 2002 |

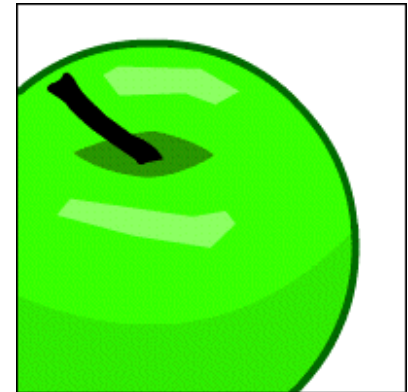


Tell Me / Show Me

ORDER BY

The default sort order is ascending.

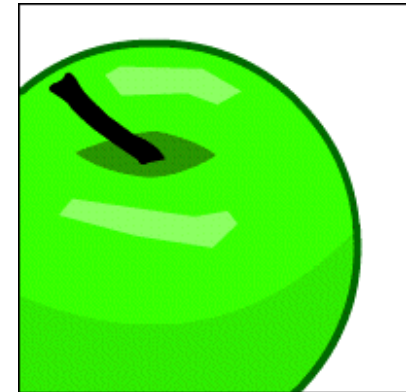
- Numeric values are displayed lowest to highest.
- Date values are displayed with the earliest value first.
- Character values are displayed in alphabetical order.
- Null values are displayed last in ascending order and first in descending order.



Tell Me / Show Me

ORDER BY

If it is necessary to reverse the default order in the ORDER BY clause, ask for descending order. You can do this by specifying the DESC keyword after the column name in the ORDER BY clause.



```
SELECT title, year  
FROM d_cds  
ORDER BY year DESC;
```

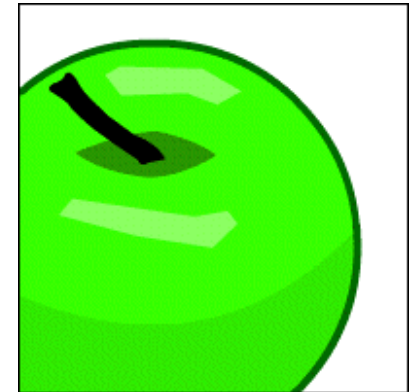
How would you order the following dates in descending order? 22-MAY-85, null, 10-JAN-04, 17-NOV-55, 21-DEC-98

| TITLE | YEAR |
|--------------------------------|------|
| Back to the Shire | 2002 |
| Here Comes the Bride | 2001 |
| Party Music for All Occasions | 2000 |
| Carpe Diem | 2000 |
| Songs from My Childhood | 1999 |
| Graduation Songbook | 1998 |
| The Celebrants Live in Concert | 1997 |

Tell Me / Show Me

ORDER BY

It is also possible to order data by using a column alias. The alias used in the SELECT statement is added to the ORDER BY clause.



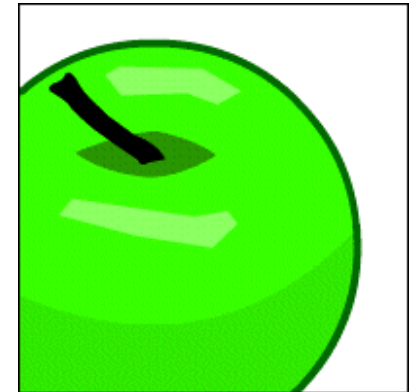
```
SELECT title, year AS "Recording Date"  
FROM d_cds  
ORDER BY "Recording Date";
```

| TITLE | RECORDING DATE |
|--------------------------------|----------------|
| The Celebrants Live in Concert | 1997 |
| Graduation Songbook | 1998 |
| Songs from My Childhood | 1999 |
| Party Music for All Occasions | 2000 |
| Carpe Diem | 2000 |
| Here Comes the Bride | 2001 |

Tell Me / Show Me

ORDER BY

It is also possible to use the ORDER BY clause to order output by a column that is not listed in the SELECT clause.



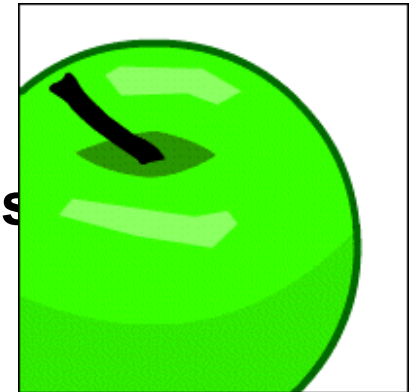
```
SELECT employee_id, first_name  
FROM employees  
WHERE employee_id < 105  
ORDER BY last_name;
```

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

Tell Me / Show Me

ORDER BY

The order of execution of a **SELECT** statement is as follows:



FROM clause -- locates the table that contains the data

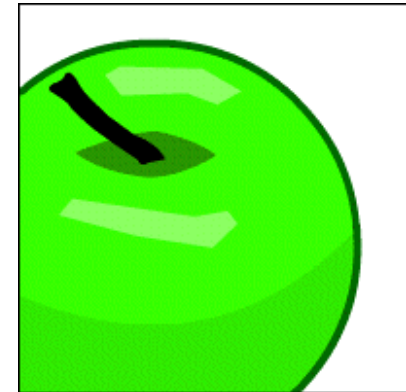
WHERE clause -- restricts the rows to be returned

SELECT clause -- selects from the reduced data set the columns requested

ORDER BY -- orders the results set

Tell Me / Show Me ORDER BY

It is also possible to sort query results by more than one column. In fact, the only limit to how many columns can be added to the ORDER BY clause is the number of columns in the table.



Sorting results by more than one column is like first sorting all the students in the school by class or grade level and then within each grade-level group, sorting students alphabetically by last name.

```
SELECT title, year  
FROM d_cds  
ORDER BY title, year;
```

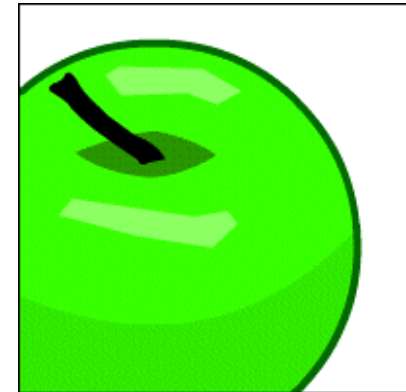
| TITLE | YEAR |
|--------------------------------|------|
| Back to the Shire | 2002 |
| Carpe Diem | 2000 |
| Graduation Songbook | 1998 |
| Here Comes the Bride | 2001 |
| Party Music for All Occasions | 2000 |
| Songs from My Childhood | 1999 |
| The Celebrants Live in Concert | 1997 |

Tell Me / Show Me

ORDER BY

To create an ORDER BY clause to sort by multiple columns, specify the columns to be returned and separate the column names using commas. If after listing the columns you want to reverse the order of a column, add DESC after its name.

```
SELECT title, year  
FROM d_cds  
ORDER BY year, title DESC;
```



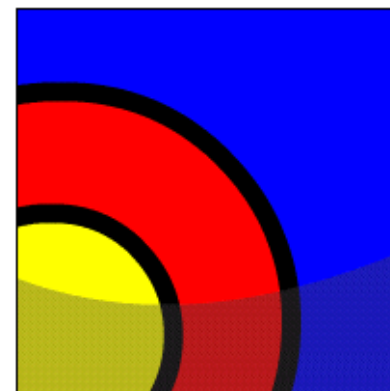
| TITLE | YEAR |
|--------------------------------|------|
| The Celebrants Live in Concert | 1997 |
| Graduation Songbook | 1998 |
| Songs from My Childhood | 1999 |
| Party Music for All Occasions | 2000 |
| Carpe Diem | 2000 |
| Here Comes the Bride | 2001 |
| Back to the Shire | 2002 |



Summary

In this lesson, you have learned about:

- Constructing queries that sorts a results set in ascending or descending order
- Stating the order in which expressions are evaluated and calculated based on the rules of precedence
- Constructing queries to order results sets using column aliases
- Constructing queries to order results sets for single or multiple columns



Summary

Practice Guide

The link for the lesson practice guide can be found in the course outline.

