Table of Contents

[1. Selecting distinct output rows 1](#_Toc408007020)

[1.1. Sorting and distinct 2](#_Toc408007021)

[2. Limit 3](#_Toc408007022)

[2.1. Limit with a skip 4](#_Toc408007023)

1. Selecting distinct output rows

The keyword DISTINCT can be placed after SELECT to specify that any duplicate copies of an **output row** will not be displayed. The decision is based on the uniqueness of the **rows to be displayed,** not the uniqueness of rows in the table.

Using DISTINCT takes extra processing time and should be avoided unless it is necessary to avoid duplicate output lines. If you are writing a query that uses a single table and display the primary key, do not include DISTINCT.

1. Display one output row per row in the table

SELECT z\_type

FROM zoo\_2015;

+-----------+

| z\_type |

+-----------+

| Giraffe |

| Armadillo |

| Lion |

| Lion |

| Giraffe |

| Zebra |

| Zebra |

| Zebra |

| Horse |

| Giraffe |

| Giraffe |

| Giraffe |

| Giraffe |

| Giraffe |

| Giraffe |

| armadillo |

| armadillo |

| Lion |

| Lion |

| Lion |

| Lion |

+-----------+

21 rows in set (0.00 sec)

1. Display one output row for each different value of z\_type. We have three zebras in the table but with Distinct we get only one row for zebra. Also note that the values 'Armadillo' and 'armadillo' return one row only since this is not case-specific.

SELECT DISTINCT z\_type

FROM zoo\_2015;

+-----------+

| z\_type |

+-----------+

| Giraffe |

| Armadillo |

| Lion |

| Zebra |

| Horse |

+-----------+

5 rows in set (0.02 sec)

1. Display one output row for each different combination of values for z\_type and z\_cost. We have two rows with Giraffe because there are two different price values for giraffes.

SELECT DISTINCT z\_type, z\_cost

FROM zoo\_2015

ORDER BY z\_type, z\_cost;

+-----------+---------+

| z\_type | z\_cost |

+-----------+---------+

| Armadillo | 490.00 |

| armadillo | 490.01 |

| Giraffe | 120.95 |

| Giraffe | 1500.00 |

| Giraffe | 3750.00 |

| Giraffe | 5000.00 |

| Giraffe | 5000.25 |

| Horse | 490.00 |

| Lion | 1850.00 |

| Lion | 5000.00 |

| Zebra | 2500.25 |

+-----------+---------+

11 rows in set (0.00 sec)

Please note that Distinct is not a function and it is inappropriate to use parentheses with Distinct. You will often see queries that use the syntax Select Distinct (z\_type) from zoo\_2015, but those parentheses are simply parentheses you could use around any expression. You can write a query such as this where the parentheses are also legal but meaningless. Select ( z\_type) from zoo\_2015;

Sorting and distinct

Return to the query

SELECT DISTINCT z\_type from zoo\_2015;

Can we sort the output? With some dbms, the way a Distinct operation is implemented the output is commonly sorted. But if you care that the output is sorted then you should use an Order By clause.

1. Distinct Z-Type, order by z\_type

SELECT Distinct z\_type

FROM zoo\_2015

order by z\_type;

+-----------+

| z\_type |

+-----------+

| Armadillo |

| Giraffe |

| Horse |

| Lion |

| Zebra |

+-----------+

5 rows in set (0.00 sec)

What if we want to display the animal type and sort the output by the animal name? Before we try this, think about what this means. We are displaying one row that represents all of the zebras, one row that represents all of the giraffes. If we sort by z\_name, how should the rows be returned? MySQL allows an order by clause to sort by the name even if it is not in the Select**.** The result does not look like it is sorted. **This is a MySQL extension and is not allowed in all dbms**.

SELECT distinct z\_type

FROM zoo\_2015

order by z\_name;

+-----------+

| z\_type |

+-----------+

| Horse |

| Armadillo |

| Zebra |

| Lion |

| Giraffe |

+-----------+

MySQL allows some extensions for the purpose of improving efficiency of retrieval. (Every dbms does this.) You need to decide (1) if this sort makes sense and is useful, and (2) if you want to keep your SQL closer to standard SQL. There is nothing wrong with using the MySQL extensions but it helps to be aware of them.

1. Limit

Limit is a MySQL feature. Other dbms have other techniques to limit the result set by a row count.

A Limit clause can be placed at the end of the statement to specify only a specific number of rows should be displayed. If the result is sorted, the sort is done before the limit is applied. Using a value for limit that is more than the number of available rows is not an error.

There are a few options you can use with Limit. Limit can take one or two integer numbers; the numbers must be nonnegative constants.

1. These are the first 5 rows from the zoo\_2015 table. These are just the first 5 rows that MySQL returned. We have no control here over which rows are returned.

Select z\_type, z\_cost

From zoo\_2015

LIMIT 5;

+-----------+---------+

| z\_type | z\_cost |

+-----------+---------+

| Giraffe | 5000.00 |

| Armadillo | 490.00 |

| Lion | 5000.00 |

| Lion | 5000.00 |

| Giraffe | 5000.25 |

+-----------+---------+

Maybe I want the 2 cheapest animals; I could include an Order by clause

1. Limit with a order by

Select z\_id, z\_type, z\_cost

From zoo\_2015

ORDER BY z\_cost

LIMIT 2;

+------+---------+--------+

| z\_id | z\_type | z\_cost |

+------+---------+--------+

| 261 | Giraffe | 120.95 |

| 47 | Horse | 490.00 |

+------+---------+--------+

2 rows in set (0.01 sec)

But if we look at the data more carefully we have several animals that cost 490.00- maybe we want to see all of the animals tied for the last place. Limit does not return the tied rows.

Try the following and you can see we have three animals at 490.00. Why did the previous query return animal 47? There is no logical reason for that decision- your query said 2 rows and that is what you got.

Select z\_id, z\_type, z\_cost

From zoo\_2015

ORDER BY z\_cost

LIMIT 5;

+------+-----------+--------+

| z\_id | z\_type | z\_cost |

+------+-----------+--------+

| 261 | Giraffe | 120.95 |

| 25 | Armadillo | 490.00 |

| 47 | Horse | 490.00 |

| 370 | armadillo | 490.00 |

| 371 | armadillo | 490.01 |

+------+-----------+--------+

5 rows in set (0.00 sec)

Limit with a skip

If you have two numbers with Limit, the first is the offset- the number of rows to skip before starting to return rows. The offset of the first row is 0- not 1.

1. Limit with a skip. This gives us 10 rows, but it skips the first 5 rows before it sends us the 10 rows. That is- it returns rows 6,7,8,…, 15

Select z\_id, z\_type, z\_cost

From zoo\_2015

ORDER BY z\_cost

LIMIT 5, 10;

+------+---------+---------+

| z\_id | z\_type | z\_cost |

+------+---------+---------+

| 260 | Giraffe | 1500.00 |

| 373 | Lion | 1850.00 |

| 374 | Lion | 1850.00 |

| 372 | Lion | 1850.00 |

| 375 | Lion | 1850.00 |

| 45 | Zebra | 2500.25 |

| 44 | Zebra | 2500.25 |

| 43 | Zebra | 2500.25 |

| 52 | Giraffe | 3750.00 |

| 258 | Giraffe | 5000.00 |

+------+---------+---------+

10 rows in set (0.00 sec)

There is no special way to say skip the first 30 rows and give me the rest. The technique used is to use a very large number for the second argument.

1. Limit with a skip and a large value for the number of rows.

Select z\_id, z\_type, z\_cost

From zoo\_2015

ORDER BY z\_cost

LIMIT 5, 200;

+------+---------+---------+

| z\_id | z\_type | z\_cost |

+------+---------+---------+

| 260 | Giraffe | 1500.00 |

| 375 | Lion | 1850.00 |

| 372 | Lion | 1850.00 |

| 373 | Lion | 1850.00 |

| 374 | Lion | 1850.00 |

| 44 | Zebra | 2500.25 |

| 43 | Zebra | 2500.25 |

| 45 | Zebra | 2500.25 |

| 52 | Giraffe | 3750.00 |

| 259 | Giraffe | 5000.00 |

| 258 | Giraffe | 5000.00 |

| 257 | Giraffe | 5000.00 |

| 57 | Lion | 5000.00 |

| 56 | Lion | 5000.00 |

| 23 | Giraffe | 5000.00 |

| 85 | Giraffe | 5000.25 |

+------+---------+---------+

16 rows in set (0.01 sec)

1. You can think of Limit with 1 value as using an offset of 0. The following two queries return the same results- assuming we do not have additional rows at that low cost point.

Select z\_id, z\_type, z\_cost

From zoo\_2015

ORDER BY z\_cost desc

LIMIT 5;

Select z\_id, z\_type, z\_cost

From zoo\_2015

ORDER BY z\_cost desc

LIMIT 0, 5;