Table of Contents

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

We can use the Limit keyword with a number to limit the number of rows returned in a query. There are a few options you can use with Limit. Limit can take one or two integer numbers; the numbers must be nonnegative constants.

The keyword Limit 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.

1. Simple Limit clause

Demo 01:   Display 5 rows in the result set.

```
Select z_type, z_cost
From a_testbed.zoo
LIMIT 5;
+-----------+---------+
| z_type    | z_cost  |
+-----------+---------+
| Giraffe   | 5000.00 |
| Armadillo |  490.00 |
| Lion      | 5000.00 |
| Lion      | 5000.00 |
| Giraffe   | 5000.25 |
+-----------+---------+
5 rows in set (0.00 sec)
```

Which 5 animals did we get? We did not specify any ordering so we got back the 5 rows that MySQL decided to return. It is possible that if we run the same query with the same table we could get different rows returned.

2. Limit and Order By

Suppose we want the 5 most expensive animals. We can include an Order By clause.

Demo 02:   Display 5 rows from the sorted result set.

```
Select z_type, z_cost
From a_testbed.zoo
Order by z_cost desc
LIMIT 5;
+-----------+---------+
| z_type    | z_cost  |
+-----------+---------+
| Giraffe   | 5000.25 |
| Giraffe   | 5000.00 |
| Lion      | 5000.00 |
| Lion      | 5000.00 |
| giraffe   | 3500.00 |
+-----------+---------+
5 rows in set (0.00 sec)
```

Demo 03:   Suppose I asked for 12 rows.

```
Select z_type, z_cost
From a_testbed.zoo
Order by z_cost desc
LIMIT 12;
+---------+---------+
| z_type  | z_cost  |
+---------+---------+
| Giraffe | 5000.25 |
| Giraffe | 5000.00 |
| Lion    | 5000.00 |
| Lion    | 5000.00 |
| giraffe | 3500.00 |
| Zebra   | 2500.25 |
| Zebra   | 2500.25 |
| Zebra   | 2500.25 |
| penguin |  750.00 |
| penguin |  750.00 |
| penguin |  500.00 |
| anteater |  500.00 |
+---------+---------+
12 rows in set (0.00 sec)
```

I will get 12 rows (assuming I have 12 rows in the table) but I have 6 animals with a z_cost of 500. Limit does not let me says bring back all of the rows tied for that last position and with my Order by clause I did not specify which of those rows to return.

Demo 04:   A slightly more precise Order by clause

```
Select z_type, z_cost
From a_testbed.zoo
Order by z_cost desc, z_type
LIMIT 12;
+----------+---------+
| z_type   | z_cost  |
+----------+---------+
| Giraffe  | 5000.25 |
| Giraffe  | 5000.00 |
| Lion     | 5000.00 |
| Lion     | 5000.00 |
| giraffe  | 3500.00 |
| Zebra    | 2500.25 |
| Zebra    | 2500.25 |
| Zebra    | 2500.25 |
| penguin  |  750.00 |
| penguin  |  750.00 |
| penguin  |  500.00 |
| anteater |  500.00 |
+----------+---------+
```

3. 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.

I am going to use the animals tables in the vets database.

Demo 05:   Limit to 10 rows. Since I am ordering by the an_id- the primary key for the table, I am determining which rows are returned.

```
Select an_id, an_name
From a_vets.vt_animals
Order by an_id
Limit 10;
+-------+-----------+
| an_id | an_name   |
+-------+-----------+
| 10002 | Gutsy     |
| 11015 | Kenny     |
| 11025 | NULL      |
| 11029 | NULL      |
| 12035 | Mr Peanut |
| 12038 | Gutsy     |
| 15001 | Big Mike  |
| 15002 | George    |
| 15165 | Burgess   |
| 15401 | Pinkie    |
+-------+-----------+
10 rows in set (0.00 sec)
```

Demo 06:   Limit with a skip. This also gives us 10 rows, but it skips the first 6 rows before it sends us the 10 rows. That is- it returns rows 7,8,…, 16

```
Select an_id, an_name
From a_vets.vt_animals
Order by an_id
Limit 6, 10;
+-------+----------+
| an_id | an_name  |
+-------+----------+
| 15001 | Big Mike |
| 15002 | George   |
| 15165 | Burgess  |
| 15401 | Pinkie   |
| 16002 | Fritz    |
| 16003 | Ursula   |
| 16004 | Napper   |
| 17025 | 25       |
| 17026 | 3P#_26   |
| 17027 | 3P#_25   |
+-------+----------+
10 rows in set (0.00 sec)
```

There is no special way to say skip the first 10 rows and give me the rest. The technique used is to use a very large number for the second argument.

Demo 07:   Limit with a skip and a large value for the number of rows.

```
Select an_id, an_name
From a_vets.vt_animals
Order by an_id
Limit 10, 999999999999999999;
+-------+-----------------+
| an_id | an_name         |
```

```
+-------+-----------------+
| 16002 | Fritz           |
| 16003 | Ursula          |
| 16004 | Napper          |
| 17025 | 25              |
| 17026 | 3P#_26          |
| 17027 | 3P#_25          |
| 19845 | Pinkie          |
| 21001 | Yoggie          |
| 21002 | Edger           |
| 21003 | Calvin Coolidge |
| 21004 | Gutsy           |
| 21005 | Koshka          |
| 21006 | Koshka          |
| 21007 | NULL            |
| 21314 | Adalwine        |
| 21315 | Baldric         |
| 21316 | Etta            |
| 21317 | Manfried        |
| 21318 | Waldrom         |
+-------+-----------------+
19 rows in set (0.00 sec)
```

Demo 08:   You can think of Limit with one value as using an offset of 0. The following two queries return the same results

```
Select an_id, an_name
From a_vets.vt_animals
Order by an_id
Limit 5;

Select an_id, an_name
From a_vets.vt_animals
Order by an_id
Limit 0, 5;
```