

[Home](#) » [Resources](#) » [Knowledge Base](#) » [Library](#) » [MariaDB Documentation](#) » [SQL Statements & Structure](#) » [SQL Statements](#) » [Administrative SQL Statements](#) » [SHOW](#) » [SHOW PROFILE](#)

# SHOW PROFILE

## Syntax

```
SHOW PROFILE [type [, type] ... ]
            [FOR QUERY n]
            [LIMIT row_count [OFFSET offset]]

type:
    ALL
    | BLOCK IO
    | CONTEXT SWITCHES
    | CPU
    | IPC
    | MEMORY
    | PAGE FAULTS
    | SOURCE
    | SWAPS
```

## Description

The `SHOW PROFILE` and `SHOW PROFILES` statements display profiling information that indicates resource usage for statements executed during the course of the current session. Profiling is controlled by the profiling session variable, which has a default value of `0` (`OFF`). Profiling is enabled by setting profiling to `1` or `ON`:

```
SET profiling = 1;
```

`SHOW PROFILES` displays a list of the most recent statements sent to the master. The size of the list is controlled by the `profiling_history_size` session variable, which has a default value of `15`. The maximum value is `100`. Setting the value to `0` has the practical effect of disabling profiling.

All statements are profiled except `SHOW PROFILES` and `SHOW PROFILE`, so you will find neither of those statements in the profile list. Malformed statements are profiled. For example, `SHOW PROFILING` is an illegal statement, and a syntax error occurs if you try to execute it, but it will show up in the profiling list.

`SHOW PROFILE` displays detailed information about a single statement. Without the `FOR QUERY n` clause, the output pertains to the most recently executed statement. If `FOR QUERY n` is included, `SHOW PROFILE` displays information for statement `n`. The values of `n` correspond to the `Query_ID` values displayed by `SHOW PROFILES`.

The `LIMIT row_count` clause may be given to limit the output to `row_count` rows. If `LIMIT` is given, `OFFSET offset` may be added to begin the output offset rows into the list of rows.

By default, `SHOW PROFILE` displays Status and Duration columns. The Status values are like the State values displayed by `SHOW PROCESSLIST`, although there might be some differences in interpretation for the two statements for some status values (see <http://dev.mysql.com/doc/refman/5.6/en/thread-information.html>).

Optional type values may be specified to display specific additional types of information:

- **ALL** displays all information
- **BLOCK IO** displays counts for block input and output operations
- **CONTEXT SWITCHES** displays counts for voluntary and involuntary context switches
- **CPU** displays user and system CPU usage times
- **IPC** displays counts for messages sent and received
- **MEMORY** is not currently implemented
- **PAGE FAULTS** displays counts for major and minor page faults
- **SOURCE** displays the names of functions from the source code, together with the name and line number of the file in which the function occurs
- **SWAPS** displays swap counts

Profiling is enabled per session. When a session ends, its profiling information is lost.

The `information_schema.PROFILING` table contains similar information.

## Examples

```
SELECT @@profiling;
+-----+
| @@profiling |
+-----+
|          0 |
+-----+

SET profiling = 1;

USE test;

DROP TABLE IF EXISTS t1;

CREATE TABLE T1 (id INT);

SHOW PROFILES;
+-----+-----+-----+
| Query_ID | Duration | Query |
+-----+-----+-----+
| 1 | 0.00009200 | SELECT DATABASE() |
| 2 | 0.00023800 | show databases |
| 3 | 0.00018900 | show tables |
| 4 | 0.00014700 | DROP TABLE IF EXISTS t1 |
| 5 | 0.24476900 | CREATE TABLE T1 (id INT) |
+-----+-----+-----+

SHOW PROFILE;
+-----+-----+
| Status | Duration |
+-----+-----+
| starting | 0.000042 |
| checking permissions | 0.000044 |
| creating table | 0.244645 |
| After create | 0.000013 |
| query end | 0.000003 |
| freeing items | 0.000016 |
| logging slow query | 0.000003 |
| cleaning up | 0.000003 |
+-----+-----+

SHOW PROFILE FOR QUERY 4;
+-----+-----+
| Status | Duration |
+-----+-----+
| starting | 0.000126 |
| query end | 0.000004 |
| freeing items | 0.000012 |
| logging slow query | 0.000003 |
| cleaning up | 0.000002 |
+-----+-----+

SHOW PROFILE CPU FOR QUERY 5;
+-----+-----+-----+-----+
| Status | Duration | CPU_user | CPU_system |
+-----+-----+-----+-----+
| starting | 0.000042 | 0.000000 | 0.000000 |
| checking permissions | 0.000044 | 0.000000 | 0.000000 |
| creating table | 0.244645 | 0.000000 | 0.000000 |
| After create | 0.000013 | 0.000000 | 0.000000 |
| query end | 0.000003 | 0.000000 | 0.000000 |
| freeing items | 0.000016 | 0.000000 | 0.000000 |
| logging slow query | 0.000003 | 0.000000 | 0.000000 |
| cleaning up | 0.000003 | 0.000000 | 0.000000 |
+-----+-----+-----+-----+
```

Comments

No comments

Get Started

Download MariaDB and start working immediately! You can also get subscription details and learn more about the advantages of MariaDB's additional service offers.

[Download Now](#)

Products

- MariaDB Server
- MariaDB MaxScale
- MariaDB ColumnStore
- Why MariaDB
- Get Started
- Pricing
- Product FAQs
- Enterprise Subscriptions Comparison

Services

- Remote DBA
- Technical Support Services
- Professional Services
- Training
- Technical Account Manager
- Migration Practice

Resources

- Customers
- Knowledge Base
- White Papers
- Datasheets & Guides
- Webinars
- Technical Presentations
- Blog
- Books
- Events

About MariaDB

- Investors
- Leadership
- Careers
- Newsroom
- Partners
- MariaDB.org

Contact

Subscribe to our newsletter!

YOUR EMAIL ADDRESS

Add Me

[Legal](#) | [Copyright](#) | [Privacy Policy](#) | [Cookies](#) | [Sitemap](#)

Copyright © 2018 MariaDB. All rights reserved.