

MySQL Cheat sheet

These are basic MySQL queries that you can use to extract data for a MySQL database. Query terms that are *italicized* represent the names of fields or tables in your database.

SELECT * FROM *mytable*

This uses the wildcard * to return all the records in all the fields in the table named *mytable*.

SELECT * FROM *mytable* ORDER BY *field1*

This uses the wildcard * to return all the records in all the fields in the table named *mytable* but sorted by *field1* in ascending order.

SELECT * FROM *mytable* ORDER BY *field1* DESC

This uses the wildcard * to return all the records in all the fields in the table named *mytable* but sorted by *field1* in descending order.

SELECT *field1*, *field2*, *field3* FROM *mytable*

Returns all records in the fields named *field1*, *field2*, *field3* in the table *mytable*

SELECT *field1*, *field2*, *field3* FROM *mytable* LIMIT 15

Returns records in the fields named *field1*, *field2*, *field3* in the table *mytable* but limits the number of records the number specified after the LIMIT command

SELECT COUNT(*) FROM *mytable*

Returns a single line with a total of number of records in the table.

SELECT SUM(*field1*) from *mytable*

Returns a single line with a sum of the values in *field1*. This only works if *field1* contains a number that can be summed.

SELECT AVG(*field1*) from *mytable*

Returns a single line with a average of the values in *field1*. This only works if *field1* contains a number that can be averaged.

SELECT * FROM *mytable* WHERE *field1* = "Jones"

Returns all records from all fields where *field1* exactly matches "Jones."

SELECT * FROM *mytable* WHERE *field1* = "Jones" and *field2* = "Bob"

Returns all records from all fields where *field1* exactly matches "Jones" and *field2* exactly matches "Bob".

SELECT * FROM *mytable* WHERE *field1* LIKE "%Smith%"

Returns all records from all fields where *field1* contains the characters between the two search wildcards

%. This would return records where *field1* contained Smith, Smithson, or Blacksmith, for example.

SELECT * FROM *mytable* WHERE *field1* LIKE “%Smith%” AND NOT *field2* LIKE “%John%”

Returns all records from all fields where *field1* contains the characters between the two search wildcards % but excludes those in the other phrase This would return records where *field1* contained Smith, Smithson, or Blacksmith, for example but not those where *field2* matched John or Johnny..

SELECT *field1*, COUNT(*) FROM *mytable* GROUP BY *field1*

Returns each unique value in *field1* with the count of the number of records with that value.

SELECT *field1*, SUM(*field2*) FROM *mytable* GROUP BY *field1*

Returns each unique value in *field1* with the sum of the corresponding values in *field2*.

SELECT *field1*, SUM(*field2*) AS *ct* FROM *mytable* GROUP BY *field1* ORDER BY *ct* DESC

Returns each unique value in *field1* with the sum of the corresponding values in *field2* and creates an alias -- a shorter name --- for the summed values. The query also ranks the results in descending order of the sum of *field2*.

**SELECT *field1*, COUNT(*), SUM(*field2*) AS *ct* FROM *mytable* WHERE *field1* LIKE ‘%Smith%’
GROUP BY *field1* ORDER BY *ct* DESC**

Returns each unique value in *field1*, the count of the number of records with that value, and the the sum of the corresponding values in *field2*, but only where *field1* contains the word Swmith. This query also creates an alias for the summed values and ranks the results in descending order.