

# Mysql Command Cheat Sheet

## MySQL Cheat Sheet

MySQL is a popular open-source relational database management system known for its ease of use and scalability. However, you will need a little help while working on a project. That's why we created this MySQL Cheat Sheet.

Instructions for installing MySQL are available at [MySQLINSTALL](#).

### CONNECTING TO A MYSQL SERVER

Connect to a MySQL server with a username and a password using the `mysql` command-line client.

```
mysql -u <username> -p
```

To connect to a specific database on a MySQL server using a username and a password:

```
mysql -u <username> -p <database>
```

To connect using the `mysql` client tool:

```
mysql -u <username> -p <database>
```

To exit the client:

```
ctrl + c
```

For a full list of commands:

```
help
```

### CREATING AND DISPLAYING DATABASES

To create a database:

```
CREATE DATABASE <name>
```

To list all the databases on the server:

```
SHOW DATABASES
```

To use a specified database:

```
USE <name>
```

To delete a specified database:

```
DROP DATABASE <name>
```

To list all tables in the database:

```
SHOW TABLES
```

To get information about a specified table:

```
DESCRIBE <tablename>
```

It outputs column names, data types, default values, and more about the table.

### CREATING TABLES

To create a table:

```
CREATE TABLE <tablename> (
  <id>
  <name> VARCHAR(255)
)
```

Use `AUTO_INCREMENT` to increment the ID automatically with each new record. An `AUTO_INCREMENT` column must be defined as a primary or unique key.

```
CREATE TABLE <tablename> (
  <id> PRIMARY KEY AUTO_INCREMENT,
  <name> VARCHAR(255)
)
```

To create a table with a foreign key:

```
CREATE TABLE <tablename> (
  <id> PRIMARY KEY AUTO_INCREMENT,
  <name> VARCHAR(255),
  <age> INT,
  FOREIGN KEY (<tablename>_<id>)
  REFERENCES <tablename>(<id>)
)
```

### MODIFYING TABLES

Use the `ALTER TABLE` statement to modify the table structure.

To change a table name:

```
ALTER TABLE <tablename> RENAME <new>
```

To add a column to the table:

```
ALTER TABLE <tablename>
ADD COLUMN <name> VARCHAR(255)
```

To change a column name:

```
ALTER TABLE <tablename>
RENAME COLUMN <id> TO <idnew>
```

To change a column data type:

```
ALTER TABLE <tablename>
MODIFY COLUMN <name> VARCHAR(255)
```

To delete a column:

```
ALTER TABLE <tablename>
DROP COLUMN <name>
```

To delete a table:

```
DROP TABLE <tablename>
```

### QUERYING DATA

To select data from a table, use the `SELECT` command.

An example of a single table query:

```
SELECT <tablename>, <id>, <age>
FROM <tablename>
WHERE <id> = 1
```

Use `ORDER BY` to sort the results.

```
SELECT <tablename>, <id>, <age>
FROM <tablename>
ORDER BY <id>
```

An example of a multiple table query:

```
SELECT <table1>.name, <table2>.name
FROM <table1>
JOIN <table2> ON <table1>.<id> = <table2>.<id>
```

Use `+`, `-`, `*`, `/` to do some basic math.

```
SELECT 10 + 10, 10 - 10, 10 * 10, 10 / 10
```

### AGGREGATION AND GROUPING

- `AVG(<expr>)` - average value of `expr` for the group.
- `COUNT(<expr>)` - count of `expr` values within the group.
- `MAX(<expr>)` - maximum value of `expr` values within the group.
- `MIN(<expr>)` - minimum value of `expr` values within the group.
- `SUM(<expr>)` - sum of `expr` values within the group.

To count the rows in the table:

```
SELECT COUNT(*)
FROM <tablename>
```

To count the non-NULL values in a column:

```
SELECT COUNT(<name>)
FROM <tablename>
```

To select unique values in a column:

```
SELECT DISTINCT <name>
FROM <tablename>
```

### GROUP BY

To report the amounts by species:

```
SELECT species, COUNT(*)
FROM animals
GROUP BY species
```

To get the average, minimum, and maximum ages by habitat:

```
SELECT habitat, <id>, AVG(age),
MIN(age), MAX(age)
FROM animals
GROUP BY habitat
```

### INSERTING DATA

To insert data into a table, use the `INSERT` command:

```
INSERT INTO <tablename> VALUES
(1, 'Puma'),
(2, 'Puma')
```

You may specify the columns in which the data is added. The remaining columns are filled with default values or NULLs.

```
INSERT INTO <tablename> (<name>) VALUES
('Elephant')
```

### UPDATING DATA

To update the data in a table, use the `UPDATE` command:

```
UPDATE <tablename>
SET
  <specimens> = 'Black',
  <name> = 'Black'
WHERE <id> = 1
```

### DELETING DATA

To delete data from a table, use the `DELETE` command:

```
DELETE FROM <tablename>
WHERE <id> = 1
```

This deletes all rows satisfying the `WHERE` condition. To delete all data from a table, use the `TRUNCATE TABLE` statement:

```
TRUNCATE TABLE <tablename>
```

### CASTING

From time to time, you need to change the type of a value. Use the `CAST()` function to do this.

In MySQL, you can cast to these data types:

```
CHAR, VARCHAR, BINARY, DATE, DATETIME,
DECIMAL, DOUBLE, FLOAT, REAL, SIGNED,
UNSIGNED, TIME, YEAR
```

To get a number as a signed integer:

```
SELECT CAST(1234.567 AS SIGNED)
-- result: 1234
```

To change a column type to double:

```
SELECT CAST(<tablename> AS DOUBLE)
```

Try out the interactive SQL from A to Z in MySQL course at [LearnSQL.com](#), and check out our other SQL courses.

LearnSQL.com is powered by [Technetix](#) for [developers](#), [QA](#), [DBAs](#), [SysAdmins](#), [QA](#).

**MySQL command cheat sheet** is an invaluable resource for developers, database administrators, and anyone who frequently works with MySQL databases. MySQL is one of the most popular open-source relational database management systems, and having a handy reference can significantly improve productivity and efficiency when performing database operations. This article will provide a comprehensive cheat sheet covering the essential MySQL commands, organized into categories for easy reference.

## 1. MySQL Basics

## Understanding how to connect to a MySQL database and perform basic operations is crucial for any user.

### 1.1 Connecting to MySQL

## To connect to a MySQL database, use the following

**command in your terminal or command prompt:**

```

**mysql -u username -p**

```

**After executing this command, you will be prompted to enter your password. Replace `username` with your actual MySQL username.**

## **1.2 Exiting MySQL**

**To exit the MySQL prompt, simply type:**

```

**exit;**

```

**or**

```

**quit;**

```

## **2. Database Operations**

**Creating, selecting, and deleting databases are fundamental tasks for managing your data.**

## **2.1 Creating a Database**

**To create a new database, use:**

```
``sql  
CREATE DATABASE database_name;  
``
```

## **2.2 Listing Databases**

**To view all databases in your MySQL server, use:**

```
``sql  
SHOW DATABASES;  
``
```

## **2.3 Selecting a Database**

**To select a database to work with, execute:**

```
``sql
```

```
USE database_name;
```

```
```
```

## **2.4 Dropping a Database**

**To delete a database and all its contents, use:**

```
```sql
```

```
DROP DATABASE database_name;
```

```
```
```

## **3. Table Operations**

**Working with tables is a significant part of using MySQL.**

### **3.1 Creating a Table**

**To create a new table, use the following syntax:**

```
```sql
```

```
CREATE TABLE table_name (  
column1_name column1_datatype,  
column2_name column2_datatype,
```

```
...  
);  
...
```

## 3.2 Describing a Table

**To see the structure of a table, use:**

```
```sql  
DESCRIBE table_name;  
```
```

**or**

```
```sql  
SHOW COLUMNS FROM table_name;  
```
```

## 3.3 Listing Tables

**To list all tables in the selected database, execute:**

```
```sql  
SHOW TABLES;  
```
```

### 3.4 Dropping a Table

To delete a table, use:

```
```sql
DROP TABLE table_name;
```
```

## 4. Data Manipulation

Manipulating data within your tables is essential for any database operations.

### 4.1 Inserting Data

To insert data into a table, use:

```
```sql
INSERT INTO table_name (column1, column2, ...)
VALUES (value1, value2, ...);
```
```

### 4.2 Selecting Data

**To retrieve data from a table, use:**

```
```sql  
SELECT column1, column2 FROM table_name;  
```
```

**To select all columns, you can use:**

```
```sql  
SELECT FROM table_name;  
```
```

### **4.3 Updating Data**

**To modify existing data in a table, use:**

```
```sql  
UPDATE table_name SET column1 = value1, column2 =  
value2 WHERE condition;  
```
```

### **4.4 Deleting Data**

**To delete records from a table, execute:**

```
```sql
```

```
DELETE FROM table_name WHERE condition;  
```\n
```

## **5. Querying Data**

**Efficient querying is key to retrieving the information needed from your database.**

### **5.1 Filtering Results**

**To filter results based on conditions, use the `WHERE` clause:**

```
```\sql  
SELECT FROM table_name WHERE condition;  
```\n
```

### **5.2 Ordering Results**

**To sort results, use the `ORDER BY` clause:**

```
```\sql  
SELECT FROM table_name ORDER BY column_name  
ASC|DESC;
```

```

### 5.3 Limiting Results

**To limit the number of results returned, use:**

```
```sql
SELECT FROM table_name LIMIT number;
```
```

### 5.4 Joining Tables

**MySQL supports several types of joins:**

**- Inner Join: Returns records with matching values in both tables.**

```
```sql
SELECT FROM table1 INNER JOIN table2 ON
table1.column_name = table2.column_name;
```
```

**- Left Join: Returns all records from the left table and matched records from the right table.**

```
```sql
```

```
SELECT FROM table1 LEFT JOIN table2 ON  
table1.column_name = table2.column_name;  
``
```

**- Right Join: Returns all records from the right table and matched records from the left table.**

```
``sql  
SELECT FROM table1 RIGHT JOIN table2 ON  
table1.column_name = table2.column_name;  
``
```

**- Full Join: Returns all records when there is a match in either left or right table records.**

```
``sql  
SELECT FROM table1 FULL OUTER JOIN table2 ON  
table1.column_name = table2.column_name;  
``
```

## **6. Indexing**

**Indexes are essential for improving the speed of data retrieval.**

### **6.1 Creating an Index**

**To create an index on a table, use:**

```
```sql  
CREATE INDEX index_name ON table_name  
(column_name);  
```
```

## **6.2 Dropping an Index**

**To remove an index, use:**

```
```sql  
DROP INDEX index_name ON table_name;  
```
```

## **7. User Management**

**Managing users and their permissions is critical for database security.**

### **7.1 Creating a User**

**To create a new user, execute:**

```
```sql  
CREATE USER 'username'@'host' IDENTIFIED BY  
'password';  
```
```

## **7.2 Granting Permissions**

**To grant permissions to a user, use:**

```
```sql  
GRANT ALL PRIVILEGES ON database_name. TO  
'username'@'host';  
```
```

## **7.3 Revoking Permissions**

**To revoke permissions, execute:**

```
```sql  
REVOKE ALL PRIVILEGES ON database_name. FROM  
'username'@'host';  
```
```

## **7.4 Listing Users**

**To view all users, execute:**

```
```sql  
SELECT User, Host FROM mysql.user;  
```
```

## **8. Backup and Restore**

**Backing up and restoring databases is crucial for data integrity and security.**

### **8.1 Backing Up a Database**

**To create a backup of a database, use the  
`mysqldump` command:**

```
```bash  
mysqldump -u username -p database_name >  
backup_file.sql  
```
```

### **8.2 Restoring a Database**

**To restore a database from a backup file, use:**

```
```bash  
mysql -u username -p database_name < backup_file.sql  
```
```

## **9. Advanced Queries**

**For more complex data manipulation, advanced querying techniques can be used.**

### **9.1 Using Subqueries**

**Subqueries allow you to nest one query within another:**

```
```sql  
SELECT column_name FROM table_name WHERE  
column_name IN (SELECT column_name FROM  
other_table);  
```
```

### **9.2 Using Aggregate Functions**

**To perform calculations on a set of values and return a single value, use aggregate functions like ``COUNT``, ``SUM``, ``AVG``, ``MAX``, and ``MIN``:**

```
```sql  
SELECT COUNT() FROM table_name;  
SELECT AVG(column_name) FROM table_name;  
```
```

### **9.3 Grouping Results**

**To group results based on a specific column, use the ``GROUP BY`` clause:**

```
```sql  
SELECT column_name, COUNT() FROM table_name  
GROUP BY column_name;  
```
```

## **Conclusion**

**The MySQL command cheat sheet outlined above serves as a quick reference guide to some of the most commonly used MySQL commands. By familiarizing yourself with these commands, you can enhance your efficiency when managing databases and performing data operations. Whether you are a beginner or an experienced user, this cheat sheet can help streamline your MySQL experience. Remember to keep it handy for quick access!**

## Frequently Asked Questions

**What is a MySQL command cheat sheet?**

**A MySQL command cheat sheet is a quick reference guide that summarizes the most commonly used MySQL commands and syntax, helping users to efficiently execute database operations.**

**Where can I find a reliable MySQL command cheat sheet?**

**Reliable MySQL command cheat sheets can be found on official MySQL documentation websites, developer blogs, and coding resource platforms like GitHub and Dev.to.**

**What are the basic SQL commands included in a MySQL cheat sheet?**

**Basic SQL commands in a MySQL cheat sheet typically include SELECT, INSERT, UPDATE, DELETE, CREATE TABLE, ALTER TABLE, and DROP TABLE.**

**How can I use the SELECT command effectively in MySQL?**

**The SELECT command can be used effectively by specifying columns, using WHERE clauses for filtering, joining tables, and applying aggregate functions like COUNT, AVG, and SUM.**

**What is the syntax for creating a new database in MySQL?**

**The syntax for creating a new database in MySQL is: CREATE DATABASE database\_name;**

**Can MySQL cheat sheets help with performance optimization?**

**Yes, MySQL cheat sheets often include tips for performance optimization, such as indexing, query optimization, and using EXPLAIN to analyze query performance.**

**What is a common mistake to avoid when using MySQL commands?**

**A common mistake is forgetting to use a WHERE clause with DELETE or UPDATE commands, which can lead to unintentional data loss.**

**How often should I refer to a MySQL command cheat sheet?**

**You should refer to a MySQL command cheat sheet whenever you're learning new commands, troubleshooting issues, or when you need a quick reminder of the syntax.**

**Find other PDF article:**

**<https://soc.up.edu.ph/39-point/Book?docid=EZx09-6659&title=marketing-cheat-sheet.pdf>**

**[Mysql Command Cheat Sheet](#)**

**MySQL**

**The most comprehensive set of advanced features,**

**management tools and technical support to achieve the highest levels of MySQL scalability, security, reliability, and uptime.**

### **MySQL :: Download MySQL Installer**

**MySQL Installer provides an easy to use, wizard-based installation experience for all your MySQL software needs. MySQL 5.7 - 8.0 installers includes the latest of versions of:**

### ***MySQL - Wikipedia***

**MySQL is free and open-source software under the terms of the GNU General Public License, and is also available under a variety of proprietary licenses. MySQL was owned and ...**

### ***MySQL Tutorial - W3Schools***

**Learn MySQL MySQL is a widely used relational database management system (RDBMS). MySQL is free and open-source. MySQL is ideal for both small and large applications. Start ...**

### **MySQL: Understanding What It Is and How It's Used - Oracle**

**Aug 29, 2024 · Unleash the power of this popular open-source relational database. Learn what MySQL is and how to use it to manage your data effectively.**

### **MySQL :: MySQL Downloads**

**MySQL Enterprise Edition for Developers Free! Access**

**the full range of MySQL Enterprise Edition features while learning, developing, and prototyping. Download Now »**

**MySQL :: Download MySQL Community Server**  
**MySQL Community Edition is a freely downloadable version of the world's most popular open source database that is supported by an active community of open source developers and ...**

***What is MySQL? - GeeksforGeeks***

**6 days ago · MySQL is an open-source, relational database management system (RDBMS) that uses Structured Query Language (SQL) to manage and manipulate data. It is one of the most ...**

**MySQL :: MySQL Community Downloads**

**Skip to Main Content The world's most popular open source database Contact MySQL | Login | Register**  
**MySQL.com Downloads**

***MySQL Tutorial: A Comprehensive Guide for Beginners***  
**Sep 28, 2023 · Discover what MySQL is and how to get started in one of the most popular database management systems.**

**MySQL**

**The most comprehensive set of advanced features, management tools and technical support to achieve the highest levels of MySQL scalability, security,**

**reliability, and uptime.**

### **MySQL :: Download MySQL Installer**

**MySQL Installer provides an easy to use, wizard-based installation experience for all your MySQL software needs. MySQL 5.7 - 8.0 installers includes the latest of versions of:**

### **MySQL - Wikipedia**

**MySQL is free and open-source software under the terms of the GNU General Public License, and is also available under a variety of proprietary licenses. MySQL was owned and sponsored by the Swedish company MySQL AB, which was bought by Sun Microsystems (now Oracle Corporation). [8]**

### **MySQL Tutorial - W3Schools**

**Learn MySQL MySQL is a widely used relational database management system (RDBMS). MySQL is free and open-source. MySQL is ideal for both small and large applications. Start learning MySQL now »**

### **MySQL: Understanding What It Is and How It's Used - Oracle**

**Aug 29, 2024 · Unleash the power of this popular open-source relational database. Learn what MySQL is and how to use it to manage your data effectively.**

### **MySQL :: MySQL Downloads**

**MySQL Enterprise Edition for Developers Free! Access**

**the full range of MySQL Enterprise Edition features while learning, developing, and prototyping. Download Now »**

### **MySQL :: Download MySQL Community Server**

**MySQL Community Edition is a freely downloadable version of the world's most popular open source database that is supported by an active community of open source developers and enthusiasts.**

### **What is MySQL? - GeeksforGeeks**

**6 days ago · MySQL is an open-source, relational database management system (RDBMS) that uses Structured Query Language (SQL) to manage and manipulate data. It is one of the most popular database systems used in web applications, ...**

### ***MySQL :: MySQL Community Downloads***

**Skip to Main Content The world's most popular open source database Contact MySQL | Login | Register MySQL.com Downloads**

### ***MySQL Tutorial: A Comprehensive Guide for Beginners***

**Sep 28, 2023 · Discover what MySQL is and how to get started in one of the most popular database management systems.**

**Unlock your MySQL potential with our comprehensive**

**MySQL command cheat sheet! Discover essential commands and tips. Learn more to boost your database skills today!**

**[Back to Home](#)**