

# MySQL verwenden

Hier ist eine Schritt-für-Schritt-Anleitung zur Verwendung von MySQL, einem beliebten Open-Source-Relationalen Datenbankverwaltungssystem:

---

## 1. Installation

- **Windows:** Laden Sie den MySQL Installer von [mysql.com/downloads](http://mysql.com/downloads) herunter
- **macOS:**

```
brew install mysql  
brew services start mysql
```

- **Linux** (Debian/Ubuntu):

```
sudo apt update  
sudo apt install mysql-server mysql-client  
sudo systemctl start mysql  
sudo mysql_secure_installation # Sicherheitssetup ausführen
```

---

## 2. Verbindung zu MySQL herstellen

- **Standardmäßiges root-Login:**

```
mysql -u root -p # Geben Sie das Passwort ein, wenn Sie dazu aufgefordert werden
```

- **Verbindung zu einem Remote-Server:**

```
mysql -u username -p -h hostname -P 3306
```

---

## 3. Grundlegende Datenbankoperationen

- **Benutzer erstellen:**

```
CREATE USER 'myuser'@'localhost' IDENTIFIED BY 'mypassword';
```

- **Datenbank erstellen:**

```

CREATE DATABASE mydb;
GRANT ALL PRIVILEGES ON mydb.* TO 'myuser'@'localhost';
FLUSH PRIVILEGES;

```

- **Grundlegende SQL-Befehle:**

```

-- Tabelle erstellen
CREATE TABLE users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(50),
    email VARCHAR(100) UNIQUE
);

-- Daten einfügen
INSERT INTO users (name, email) VALUES ('Bob', 'bob@example.com');

-- Daten abfragen
SELECT * FROM users WHERE name LIKE 'B%';

-- Daten aktualisieren
UPDATE users SET email = 'new@email.com' WHERE id = 1;

-- Daten löschen
DELETE FROM users WHERE id = 2;

```

---

## 4. Erweiterte Funktionen

- **Joins:**

```

SELECT orders.id, users.name
FROM orders
JOIN users ON orders.user_id = users.id;

```

- **Gespeicherte Prozeduren:**

```

DELIMITER //
CREATE PROCEDURE GetUsers()
BEGIN
    SELECT * FROM users;
END //
DELIMITER ;

```

```
CALL GetUsers();
```

- **JSON-Unterstützung** (MySQL 5.7+):

```
CREATE TABLE products (
    id INT PRIMARY KEY,
    details JSON
);
```

---

## 5. Wartung

- **Sicherung:**

```
mysqldump -u root -p mydb > backup.sql
```

- **Wiederherstellung:**

```
mysql -u root -p mydb < backup.sql
```

- **Tabellen optimieren:**

```
ANALYZE TABLE users;
OPTIMIZE TABLE users;
```

---

## 6. Sicherheit

- **Erstellen eines schreibgeschützten Benutzers:**

```
CREATE USER 'reader'@'localhost' IDENTIFIED BY 'readpass';
GRANT SELECT ON mydb.* TO 'reader'@'localhost';
FLUSH PRIVILEGES;
```

---

## 7. Nützliche MySQL-Befehle

- Datenbanken auflisten: SHOW DATABASES;
- Datenbank verwenden: USE mydb;
- Tabellen auflisten: SHOW TABLES;

- Tabelle beschreiben: DESCRIBE users;
  - Benutzer anzeigen: SELECT user FROM mysql.user;
  - MySQL verlassen: EXIT; oder QUIT;
- 

## 8. GUI-Tools

- **MySQL Workbench**: Offizielles GUI-Tool
  - **phpMyAdmin**: Webbasierte Schnittstelle
  - **DBeaver**: Plattformübergreifendes Datenbank-Tool
  - **HeidiSQL**: Leichtgewichtiger Windows-Client
- 

## 9. Lernressourcen

- Offizielle Dokumentation: [dev.mysql.com/doc](http://dev.mysql.com/doc)
  - Interaktives Tutorial: [mysqltutorial.org](http://mysqltutorial.org)
  - Üben: [sqlzoo.net](http://sqlzoo.net) oder [leetcode.com](http://leetcode.com)
- 

**Wichtige Unterschiede zu PostgreSQL:** 1. Verwendet `AUTO_INCREMENT` anstelle von `SERIAL` 2. Tabellennamen standardmäßig groß- und kleinschreibungsunabhängig 3. Verwendet `mysqldump` anstelle von `pg_dump` 4. Standardport: **3306** (im Gegensatz zu PostgreSQLs 5432) 5. Unterschiedliches Authentifizierungssystem und Benutzerverwaltung

**Tipp:** Erstellen Sie immer eine Sicherung der Datenbanken vor größeren Änderungen. Verwenden Sie Transaktionen (`START TRANSACTION/COMMIT`) für kritische Operationen.