

In Verbindung mit der **WHERE** Klausel gibt es eine Mustererkennung mit dem Operator **LIKE**. **LIKE** unterscheidet nicht zwischen Groß- und Kleinschreibung, zumindest nicht in MySQL (PostgreSQL im Vergleich ist CaseSensitiv). Zusätzlich können Platzhalter (Wildcards, bzw. Joker) eingesetzt werden. Das Prozentzeichen **%** ersetzt einen Textteil mit beliebigen Zeichen und beliebiger Länge. Der Unterstrich **_** steht exakt für nur ein einziges, beliebiges Zeichen.

SQL



Suche nach einem Vornamen mit einem **_** Platzhalter

Im Beispiel wird der letzte Buchstabe mit einem Platzhalter für ein einziges Zeichen belegt. Als Ergebnis bekommt man sowohl Christine also auch Christina. **LIKE** ignoriert die Großschreibung!

```
USE PersonalDB;
SELECT Nachname, Vorname FROM Stammdaten
WHERE Vorname LIKE 'christin_';
```

SQL



Suche nach allen Nachnamen die mit einem M beginnen

Das Beispiel gibt alle Nachnamen aus, die mit einem M beginnen. **LIKE** ignoriert die Großschreibung!

```
SELECT Nachname, Vorname FROM Stammdaten
WHERE Nachname LIKE 'm%';
```

SQL



Ausgabe aller Mitarbeiter|innen die im Juni Geburtstag feiern!

Mit die Platzhalter ersetzen das Jahr und den Tag. Somit wird nur nach dem Monat 06 gesucht!

```
SELECT Nachname, Vorname, GebDatum FROM Stammdaten
WHERE GebDatum LIKE '%-06-%';
```

SQL



Alle Mitarbeiter|innen aus den Abteilungen Lager und Verkauf anzeigen!

Mit dem Schlüsselwort **IN** kann eine Liste mit Werten definiert werden. **IN** benötigt eine Klammer - die Suchbegriffe werden durch Beistriche getrennt!

```
SELECT Nachname, Vorname, Abteilung FROM Stammdaten
WHERE Abteilung IN ('Lager', 'Verkauf')
ORDER BY Abteilung;
```



IN und **LIKE** lassen sich nicht kombinieren.
Eine Lösung wäre, die **WHERE**-Klausel mit dem **OR** Operator zu verbinden

```
SELECT Nachname, Vorname, Abteilung FROM Stammdaten
WHERE Abteilung LIKE 'La%' OR Abteilung LIKE 'Ve%'
ORDER BY Abteilung;
```