

Structured Query Language

DDL Data Definition Language	Erstellen von Datenbanken, Tabellen und Indizes
DQL Data Query Language	Abfragen von Daten
DML Data Manipulation Language	Anlegen, Ändern und Löschen von Datensätzen
DCL Data Control Language	Anlegen von Benutzern, Vergabe von Rechten

Structured Query Language

SELECT – Übersicht

SELECT

[Optionen]

* / Liste_der_Ausgabespalten

= was soll ausgegeben werden

FROM Liste_der_Tabellen

= aus welchen Tabellen

[**WHERE** Auswahlbedingung]

= nur spezielle Werte ausgeben...

[**GROUP BY** Spaltenliste]

= Ergebnisse zusammenfassen

[**HAVING** Auswahlbedingung]

= zweite Einschränkung der Werte

[**ORDER BY** Optionen]

= Ausgabe sortieren

[**LIMIT** ...]

= Zahl der Zeilen begrenzen

Structured Query Language

SELECT – Spalten

- Entweder: Liste von Spaltennamen oder Platzhalter * (= alle Spalten):

- Beispiele:

SELECT * FROM buch;

→ Gibt alle Spalten der Tabelle **buch** aus.

SELECT titel FROM buch;

→ Gibt die Spalte **titel** der Tabelle **buch** aus.

SELECT auflage, ISBN FROM buch;

→ Gibt die Spalten **auflage** und **ISBN** aus.

Bemerkung:

- Reihenfolge der Spalten ist beliebig.
- Mehrere Spalten durch Kommata trennen.

Structured Query Language

SELECT – WHERE

- Kennwort **WHERE** leitet Suchbedingung ein, z.B. einen Vergleich.
- Beispiel:
SELECT titel FROM buch WHERE ISBN = '0596002114';
→ Gibt den Titel des Buches mit der ISBN-Nummer 0596002114 aus.

Vergleichsoperatoren

<	kleiner
<=	kleiner oder gleich
=	gleich
<>	nicht gleich
>=	grösser oder gleich
>	grösser

Structured Query Language

SELECT – NULL-Werte

- Syntax: **IS NULL** oder **IS NOT NULL**

Beispiele:

SELECT titel FROM buch WHERE preis IS NOT NULL;

→ Gibt die Titel der Bücher aus, deren Preis bekannt ist.

Auch möglich: logische Operatoren **NOT**, **AND**, **OR**

SELECT titel FROM buch

WHERE (preis IS NOT NULL) AND (seiten IS NOT NULL);

→ Gibt die Titel der Bücher aus, deren Preis und Seitenzahl bekannt ist.

Structured Query Language

SELECT – sortieren

- Syntax: **ORDER BY Spaltenname [Sortier_Richtung]**

Sortier_Richtung: **ASC** = aufsteigend (ascending), von A nach Z
 DESC = absteigend (descending)

Ohne Angabe: Aufsteigend (ASC)

Beispiele:

SELECT jahr, titel FROM buch ORDER BY jahr;

→ Gibt Erscheinungsjahr und Titel aller Bücher aus, die ältesten zuerst.

SELECT jahr, titel FROM buch ORDER BY jahr DESC;

→ Gibt Erscheinungsjahr und Titel aller Bücher aus, die neuesten zuerst.

Structured Query Language

SELECT – Zahl der Zeilen begrenzen

- Syntax: **LIMIT [Erste_Zeile ,] Anzahl_Zeilen**

Erste_Zeile = Nummer des ersten auszugebenden Datensatzes
die Nummerierung beginnt mit 0 !

Beispiele:

SELECT titel FROM buch LIMIT 10;

→ Gibt den Titel der ersten 10 Bücher aus.

SELECT titel FROM buch LIMIT 10, 15;

→ Gibt den Titel von 15 Büchern aus, beginnend beim 11. Buch.

**SELECT titel FROM buch WHERE format = "Taschenbuch"
ORDER BY jahr DESC LIMIT 10;**

→ Die 10 neuesten Taschenbücher...