

# Datenbanken

## Teil 2: Informationen

### Kapitel 10: SQL



## Einführung und Grundlagen zum Thema SQL

In dieser Einleitung werden die folgenden Grundlagen vorgestellt

1. Bedeutung der Abkürzung / Definition
2. Eigenschaften von SQL
3. Aufgaben von SQL
4. Ein erstes Beispiel

# SQL-Grundlagen

*Abkürzung:*

*Bedeutung:*

## Eigenschaften

- 1.
- 2.
- 3.

## Aufgaben

- 1.
- 2.
- 3.
- 4.

# SQL-Grundlagen

**Abkürzung:**

Structured Query Language (strukturierte Abfragesprache)

**Bedeutung:**

## Eigenschaften

- 1.
- 2.
- 3.

## Aufgaben

- 1.
- 2.
- 3.
- 4.

# SQL-Grundlagen

**Abkürzung:**

Structured Query Language (strukturierte Abfragesprache)

**Bedeutung:**

SQL ist eine relationale Datenmanipulations- und Abfragesprache.

## Eigenschaften

- 1.
- 2.
- 3.

## Aufgaben

- 1.
- 2.
- 3.
- 4.

# SQL-Grundlagen

## **Abkürzung:**

Structured Query Language (strukturierte Abfragesprache)

## **Bedeutung:**

SQL ist eine relationale Datenmanipulations- und Abfragesprache.

## Eigenschaften

1. **mengenorientiert** (SQL liefert Menge als Rückgabewert)
- 2.
- 3.

## Aufgaben

- 1.
- 2.
- 3.
- 4.

# SQL-Grundlagen

## **Abkürzung:**

Structured Query Language (strukturierte Abfragesprache)

## **Bedeutung:**

SQL ist eine relationale Datenmanipulations- und Abfragesprache.

## Eigenschaften

1. **mengenorientiert** (SQL liefert Menge als Rückgabewert)
2. **deskriptiv** (SQL ist beschreibend im Gegensatz zu prozedural)
- 3.

## Aufgaben

- 1.
- 2.
- 3.
- 4.

# SQL-Grundlagen

## **Abkürzung:**

Structured Query Language (strukturierte Abfragesprache)

## **Bedeutung:**

SQL ist eine relationale Datenmanipulations- und Abfragesprache.

## Eigenschaften

1. **mengenorientiert** (SQL liefert Menge als Rückgabewert)
2. **deskriptiv** (SQL ist beschreibend im Gegensatz zu prozedural)
3. **RDBMS** (SQL ist ein wesentlicher Bestandteil eines RDBMS)

## Aufgaben

- 1.
- 2.
- 3.
- 4.

# SQL-Grundlagen

## **Abkürzung:**

Structured Query Language (strukturierte Abfragesprache)

## **Bedeutung:**

SQL ist eine relationale Datenmanipulations- und Abfragesprache.

## Eigenschaften

1. **mengenorientiert** (SQL liefert Menge als Rückgabewert)
2. **deskriptiv** (SQL ist beschreibend im Gegensatz zu prozedural)
3. **RDBMS** (SQL ist ein wesentlicher Bestandteil eines RDBMS)

## Aufgaben

1. Datenbanken / Tabellen / Benutzer
- 2.
- 3.
- 4.



# SQL-Grundlagen

## **Abkürzung:**

Structured Query Language (strukturierte Abfragesprache)

## **Bedeutung:**

SQL ist eine relationale Datenmanipulations- und Abfragesprache.

## Eigenschaften

1. **mengenorientiert** (SQL liefert Menge als Rückgabewert)
2. **deskriptiv** (SQL ist beschreibend im Gegensatz zu prozedural)
3. **RDBMS** (SQL ist ein wesentlicher Bestandteil eines RDBMS)

## Aufgaben

1. Datenbanken / Tabellen / Benutzer  
**anlegen, ändern, löschen**
- 2.
- 3.
- 4.

# SQL-Grundlagen

## **Abkürzung:**

Structured Query Language (strukturierte Abfragesprache)

## **Bedeutung:**

SQL ist eine relationale Datenmanipulations- und Abfragesprache.

## Eigenschaften

1. **mengenorientiert** (SQL liefert Menge als Rückgabewert)
2. **deskriptiv** (SQL ist beschreibend im Gegensatz zu prozedural)
3. **RDBMS** (SQL ist ein wesentlicher Bestandteil eines RDBMS)

## Aufgaben

1. Datenbanken / Tabellen / Benutzer  
**anlegen, ändern, löschen**
2. Benutzerrechte
- 3.
- 4.

# SQL-Grundlagen

## **Abkürzung:**

Structured Query Language (strukturierte Abfragesprache)

## **Bedeutung:**

SQL ist eine relationale Datenmanipulations- und Abfragesprache.

## Eigenschaften

1. **mengenorientiert** (SQL liefert Menge als Rückgabewert)
2. **deskriptiv** (SQL ist beschreibend im Gegensatz zu prozedural)
3. **RDBMS** (SQL ist ein wesentlicher Bestandteil eines RDBMS)

## Aufgaben

1. Datenbanken / Tabellen / Benutzer  
**anlegen, ändern, löschen**
2. Benutzerrechte  
**verwalten**
- 3.
- 4.

# SQL-Grundlagen

## **Abkürzung:**

Structured Query Language (strukturierte Abfragesprache)

## **Bedeutung:**

SQL ist eine relationale Datenmanipulations- und Abfragesprache.

## Eigenschaften

1. **mengenorientiert** (SQL liefert Menge als Rückgabewert)
2. **deskriptiv** (SQL ist beschreibend im Gegensatz zu prozedural)
3. **RDBMS** (SQL ist ein wesentlicher Bestandteil eines RDBMS)

## Aufgaben

1. Datenbanken / Tabellen / Benutzer  
**anlegen, ändern, löschen**
2. Benutzerrechte  
**verwalten**
3. Daten
- 4.

# SQL-Grundlagen

## **Abkürzung:**

Structured Query Language (strukturierte Abfragesprache)

## **Bedeutung:**

SQL ist eine relationale Datenmanipulations- und Abfragesprache.

## Eigenschaften

1. **mengenorientiert** (SQL liefert Menge als Rückgabewert)
2. **deskriptiv** (SQL ist beschreibend im Gegensatz zu prozedural)
3. **RDBMS** (SQL ist ein wesentlicher Bestandteil eines RDBMS)

## Aufgaben

1. Datenbanken / Tabellen / Benutzer  
**anlegen, ändern, löschen**
2. Benutzerrechte  
**verwalten**
3. Daten  
**eingeben, ändern, löschen, gruppieren, sortieren, ausgeben, sichern**
- 4.

# SQL-Grundlagen

## **Abkürzung:**

Structured Query Language (strukturierte Abfragesprache)

## **Bedeutung:**

SQL ist eine relationale Datenmanipulations- und Abfragesprache.

## Eigenschaften

1. **mengenorientiert** (SQL liefert Menge als Rückgabewert)
2. **deskriptiv** (SQL ist beschreibend im Gegensatz zu prozedural)
3. **RDBMS** (SQL ist ein wesentlicher Bestandteil eines RDBMS)

## Aufgaben

1. Datenbanken / Tabellen / Benutzer  
**anlegen, ändern, löschen**
2. Benutzerrechte  
**verwalten**
3. Daten  
**eingeben, ändern, löschen, gruppieren, sortieren, ausgeben, sichern**
4. Unterabfragen / Views

# SQL-Grundlagen

## **Abkürzung:**

Structured Query Language (strukturierte Abfragesprache)

## **Bedeutung:**

SQL ist eine relationale Datenmanipulations- und Abfragesprache.

## Eigenschaften

1. **mengenorientiert** (SQL liefert Menge als Rückgabewert)
2. **deskriptiv** (SQL ist beschreibend im Gegensatz zu prozedural)
3. **RDBMS** (SQL ist ein wesentlicher Bestandteil eines RDBMS)

## Aufgaben

1. Datenbanken / Tabellen / Benutzer  
**anlegen, ändern, löschen**
2. Benutzerrechte  
**verwalten**
3. Daten  
**eingeben, ändern, löschen, gruppieren, sortieren, ausgeben, sichern**
4. Unterabfragen / Views  
**anlegen, ändern, löschen**

# SQL-Beispiel

## Schüler

| <u>ID_Schüler</u> | Vorname | Name   | Ort     | Straße        |
|-------------------|---------|--------|---------|---------------|
| 1                 | Willi   | Klein  | Olsberg | Vogelweg      |
| 2                 | Elke    | Specht | Menden  | Erlenbruch    |
| 3                 | Uwe     | Groß   | Olsberg | Bahnhofstraße |
| 4                 | Silke   | Müller | Sundern | Hauptstraße   |

## Abfrage





# SQL-Beispiel

## Schüler

| <u>ID_Schüler</u> | Vorname | Name   | Ort     | Straße        |
|-------------------|---------|--------|---------|---------------|
| 1                 | Willi   | Klein  | Olsberg | Vogelweg      |
| 2                 | Elke    | Specht | Menden  | Erlenbruch    |
| 3                 | Uwe     | Groß   | Olsberg | Bahnhofstraße |
| 4                 | Silke   | Müller | Sundern | Hauptstraße   |

## Abfrage

### *Umgangssprachlich*

„Welche Schüler kommen aus Olsberg.“



# SQL-Beispiel

## Schüler

| <u>ID_Schüler</u> | Vorname | Name   | Ort     | Straße        |
|-------------------|---------|--------|---------|---------------|
| 1                 | Willi   | Klein  | Olsberg | Vogelweg      |
| 2                 | Elke    | Specht | Menden  | Erlenbruch    |
| 3                 | Uwe     | Groß   | Olsberg | Bahnhofstraße |
| 4                 | Silke   | Müller | Sundern | Hauptstraße   |

## Abfrage

### ***Umgangssprachlich***

„Welche Schüler kommen aus Olsberg.“

### ***Umformulierung genauer***

„Nenne Vorname und Nachname aller Schüler, die aus Olsberg kommen.“



# SQL-Beispiel

## Schüler

| <u>ID_Schüler</u> | Vorname | Name   | Ort     | Straße        |
|-------------------|---------|--------|---------|---------------|
| 1                 | Willi   | Klein  | Olsberg | Vogelweg      |
| 2                 | Elke    | Specht | Menden  | Erlenbruch    |
| 3                 | Uwe     | Groß   | Olsberg | Bahnhofstraße |
| 4                 | Silke   | Müller | Sundern | Hauptstraße   |

## Abfrage

### *Umgangssprachlich*

„Welche Schüler kommen aus Olsberg.“

### *Umformulierung genauer*

„Nenne Vorname und Nachname aller Schüler, die aus Olsberg kommen.“

### *Umformuliert noch genauer*

Selektiere die Merkmale  
Vor- und Nachname

### *Übersetzung in SQL*

# SQL-Beispiel

## Schüler

| <u>ID_Schüler</u> | Vorname | Name   | Ort     | Straße        |
|-------------------|---------|--------|---------|---------------|
| 1                 | Willi   | Klein  | Olsberg | Vogelweg      |
| 2                 | Elke    | Specht | Menden  | Erlenbruch    |
| 3                 | Uwe     | Groß   | Olsberg | Bahnhofstraße |
| 4                 | Silke   | Müller | Sundern | Hauptstraße   |

## Abfrage

### *Umgangssprachlich*

„Welche Schüler kommen aus Olsberg.“

### *Umformulierung genauer*

„Nenne Vorname und Nachname aller Schüler, die aus Olsberg kommen.“

### *Umformuliert noch genauer*

Selektiere die Merkmale  
Vor- und Nachname

### *Übersetzung in SQL*

```
SELECT Vorname, Name
```

# SQL-Beispiel

## Schüler

| <u>ID_Schüler</u> | Vorname | Name   | Ort     | Straße        |
|-------------------|---------|--------|---------|---------------|
| 1                 | Willi   | Klein  | Olsberg | Vogelweg      |
| 2                 | Elke    | Specht | Menden  | Erlenbruch    |
| 3                 | Uwe     | Groß   | Olsberg | Bahnhofstraße |
| 4                 | Silke   | Müller | Sundern | Hauptstraße   |

## Abfrage

### *Umgangssprachlich*

„Welche Schüler kommen aus Olsberg.“

### *Umformulierung genauer*

„Nenne Vorname und Nachname aller Schüler, die aus Olsberg kommen.“

### *Umformuliert noch genauer*

Selektiere die Merkmale Vor- und Nachname aus der Tabelle Schüler

### *Übersetzung in SQL*

```
SELECT Vorname, Name
```

# SQL-Beispiel

## Schüler

| <u>ID_Schüler</u> | Vorname | Name   | Ort     | Straße        |
|-------------------|---------|--------|---------|---------------|
| 1                 | Willi   | Klein  | Olsberg | Vogelweg      |
| 2                 | Elke    | Specht | Menden  | Erlenbruch    |
| 3                 | Uwe     | Groß   | Olsberg | Bahnhofstraße |
| 4                 | Silke   | Müller | Sundern | Hauptstraße   |

## Abfrage

### *Umgangssprachlich*

„Welche Schüler kommen aus Olsberg.“

### *Umformulierung genauer*

„Nenne Vorname und Nachname aller Schüler, die aus Olsberg kommen.“

### *Umformuliert noch genauer*

Selektiere die Merkmale Vor- und Nachname aus der Tabelle Schüler

### *Übersetzung in SQL*

```
SELECT Vorname, Name  
FROM Schüler
```

# SQL-Beispiel

## Schüler

| <u>ID_Schüler</u> | Vorname | Name   | Ort     | Straße        |
|-------------------|---------|--------|---------|---------------|
| 1                 | Willi   | Klein  | Olsberg | Vogelweg      |
| 2                 | Elke    | Specht | Menden  | Erlenbruch    |
| 3                 | Uwe     | Groß   | Olsberg | Bahnhofstraße |
| 4                 | Silke   | Müller | Sundern | Hauptstraße   |

## Abfrage

### *Umgangssprachlich*

„Welche Schüler kommen aus Olsberg.“

### *Umformulierung genauer*

„Nenne Vorname und Nachname aller Schüler, die aus Olsberg kommen.“

### *Umformuliert noch genauer*

Selektiere die Merkmale Vor- und Nachname aus der Tabelle Schüler wobei der Wohnort Olsberg ist.

### *Übersetzung in SQL*

```
SELECT Vorname, Name  
FROM Schüler
```

# SQL-Beispiel

## Schüler

| <u>ID_Schüler</u> | Vorname | Name   | Ort     | Straße        |
|-------------------|---------|--------|---------|---------------|
| 1                 | Willi   | Klein  | Olsberg | Vogelweg      |
| 2                 | Elke    | Specht | Menden  | Erlenbruch    |
| 3                 | Uwe     | Groß   | Olsberg | Bahnhofstraße |
| 4                 | Silke   | Müller | Sundern | Hauptstraße   |

## Abfrage

### *Umgangssprachlich*

„Welche Schüler kommen aus Olsberg.“

### *Umformulierung genauer*

„Nenne Vorname und Nachname aller Schüler, die aus Olsberg kommen.“

### *Umformuliert noch genauer*

Selektiere die Merkmale Vor- und Nachname aus der Tabelle Schüler wobei der Wohnort Olsberg ist.

### *Übersetzung in SQL*

```
SELECT Vorname, Name
FROM Schüler
WHERE Ort = 'Olsberg';
```



# SQL-Beispiel

## Schüler

| <u>ID_Schüler</u> | Vorname | Name   | Ort     | Straße        |
|-------------------|---------|--------|---------|---------------|
| 1                 | Willi   | Klein  | Olsberg | Vogelweg      |
| 2                 | Elke    | Specht | Menden  | Erlenbruch    |
| 3                 | Uwe     | Groß   | Olsberg | Bahnhofstraße |
| 4                 | Silke   | Müller | Sundern | Hauptstraße   |

## Abfrage

### *Umgangssprachlich*

„Welche Schüler kommen aus Olsberg.“

### *Umformulierung genauer*

„Nenne Vorname und Nachname aller Schüler, die aus Olsberg kommen.“

### *Umformuliert noch genauer*

Selektiere die Merkmale Vor- und Nachname aus der Tabelle Schüler wobei der Wohnort Olsberg ist.

### *Abfrageergebnis*

| Vorname | Name  |
|---------|-------|
| Willi   | Klein |
| Uwe     | Groß  |

### *Übersetzung in SQL*

```
SELECT Vorname, Name
FROM Schüler
WHERE Ort = 'Olsberg';
```