

GRUNDLAGEN WIRTSCHAFTSINFOR

DATENBANKEN: SQL

PROF. DR. CHRISTIAN BOCKERMANN

HOCHSCHULE BOCHUM

SOMMERSEMESTER 2023

Inhalt

1 SQL - Structured Query Language

Vom Konzept zur Implementierung

- SQL ist **standardisierte** Sprache für Datenbanken
- kleine Unterschiede / Dialekte zwischen Datenbanken
- SQL erlaubt Abfragen durch Benutzer + Programme



Abfrage (SQL)



Datenbank

Vom Konzept zur Implementierung

- SQL ist **standardisierte** Sprache für Datenbanken
- kleine Unterschiede / Dialekte zwischen Datenbanken
- SQL erlaubt Abfragen durch Benutzer + Programme



A	B	C	D

Abfrage (SQL)

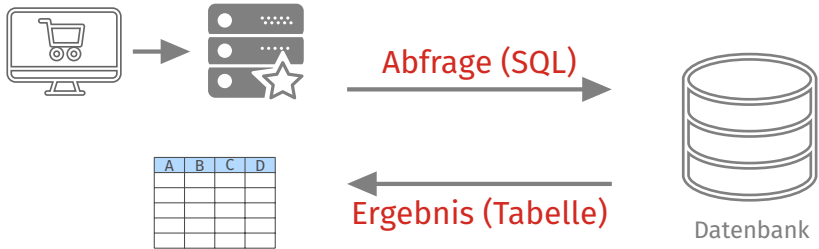
Ergebnis (Tabelle)



Datenbank

Vom Konzept zur Implementierung

- SQL ist **standardisierte** Sprache für Datenbanken
- kleine Unterschiede / Dialekte zwischen Datenbanken
- SQL erlaubt Abfragen durch Benutzer + Programme



Server zum Üben

`https://datascience.hs-bochum.de/sql/`

Die Sprache SQL

Befehle für unterschiedliche Aktionen:

- CREATE – Datenbanken oder Tabellen anlegen
- INSERT – Datensätze in Tabellen eintragen
- UPDATE – Datensätze ändern
- DELETE – Datensätze löschen
- SELECT – Datensätze aus Tabellen selektieren

Anlegen/Definition von Tabellen

Beispiel für das Anlegen der Tabelle **Kunde** als SQL Befehl:

```
CREATE TABLE Kunde (  
    Nr Integer Primary Key,  
    Name Varchar(255),  
    Vorname Varchar(255),  
    PLZ Varchar(5),  
    Ort Varchar(40)  
);
```


SELECT zum Abfragen von Datensätzen

```
SELECT [DISTINCT | ALL] spalten  
FROM tabelle(n)  
WHERE bedingung  
GROUP BY spalten  
HAVING bedingung  
ORDER BY spalten
```

SELECT zum Abfragen von Datensätzen

`SELECT` spalten `FROM` tabellen `WHERE` bedingungen

- spalten steht für Tabellenspalten und/oder Ausdrücke (Formeln/Funktionen)
- * selektiert alle Spalten
- tabellen bezeichnet die Tabelle(n), aus denen abgefragt wird
- bedingungen beschränkt das Ergebnis

Beispiel: Abfrage aller Einträge der Tabelle Kunde

```
SELECT * FROM Kunde
```

Beispiel: Abfrage aller Einträge der Tabelle Kunde

```
SELECT * FROM Kunde
```

SELECT liefert eine Tabelle als Ergebnis:

KundeNr	Name	Vorname	PLZ	Ort
1	Maier	Mia	44801	Bochum
2	Schmidt	Jonas	44801	Bochum
3	Müller	Emma	44802	Bochum
4	Weber	Lukas	45127	Essen

Beispiel: Abfrage mit Bedingung

```
SELECT * FROM Kunde WHERE Ort = "Bochum"
```

Beispiel: Abfrage mit Bedingung

```
SELECT * FROM Kunde WHERE Ort = "Bochum"
```

SELECT liefert eine Tabelle als Ergebnis:

KundeNr	Name	Vorname	PLZ	Ort
1	Maier	Mia	44801	Bochum
2	Schmidt	Jonas	44801	Bochum
3	Müller	Emma	44802	Bochum

Weitere Möglichkeiten von SELECT

- Komplexere Bedingungen über AND und OR
- Selektieren ohne Duplikate mit `distinct`
- Zählen von Zeilen/Werten mit `count(*)` bzw. `count(spalte)`
- Arithmetik und Aggregationen (`min`, `max`, `sum`, `avg`)
- Sortierung mit ORDER BY

Abfrage von mehreren Tabellen

```
SELECT * FROM Kunde, Bestellung
```

KundeNr	Name	Vorname	PLZ	Ort
1	Maier	Mia	44801	Bochum
2	Schmidt	Jonas	44801	Bochum
3	Müller	Emma	44802	Bochum
4	Weber	Lukas	45127	Essen

BestellNr	Datum	KundeNr
1	2016-05-01	1
2	2016-05-04	2
3	2016-05-09	3
4	2016-05-17	1

Abfrage von mehreren Tabellen

```
SELECT * FROM Kunde, Bestellung
```

KundeNr	Name	Vorname	PLZ	Ort
1	Maier	Mia	44801	Bochum
2	Schmidt	Jonas	44801	Bochum
3	Müller	Emma	44802	Bochum
4	Weber	Lukas	45127	Essen

BestellNr	Datum	KundeNr
1	2016-05-01	1
2	2016-05-04	2
3	2016-05-09	3
4	2016-05-17	1

KundeNr	Name	Vorname	PLZ	Ort	BestellNr	Datum	KundeNr
1	Maier	Mia	44801	Bochum	1	2016-05-01	1
1	Maier	Mia	44801	Bochum	2	2016-05-04	2
1	Maier	Mia	44801	Bochum	3	2016-05-09	3
1	Maier	Mia	44801	Bochum	4	2016-05-17	1
2	Schmidt	Jonas	44801	Bochum	1	2016-05-01	1

Abfrage von mehreren Tabellen

```
SELECT * FROM Kunde, Bestellung  
WHERE Kunde.KundeNr = Bestellung.KundeNr
```

KundeNr	Name	Vorname	PLZ	Ort
1	Maier	Mia	44801	Bochum
2	Schmidt	Jonas	44801	Bochum
3	Müller	Emma	44802	Bochum
4	Weber	Lukas	45127	Essen

BestellNr	Datum	KundeNr
1	2016-05-01	1
2	2016-05-04	2
3	2016-05-09	3
4	2016-05-17	1

KundeNr	Name	Vorname	PLZ	Ort	BestellNr	Datum	KundeNr
1	Maier	Mia	44801	Bochum	1	2016-05-01	1
1	Maier	Mia	44801	Bochum	2	2016-05-04	2
1	Maier	Mia	44801	Bochum	3	2016-05-09	3
1	Maier	Mia	44801	Bochum	4	2016-05-17	1
2	Schmidt	Jonas	44801	Bochum	1	2016-05-01	1