ANGEWANDTE PROGRAMMIERUNG
SQL

Vorlesung 02

Dennis Glüsenkamp

24. März 2023

Hintergrundinformationen zu SQL

SQL-Syntax

Live-Beispiele und Übungen

Hintergrundinformationen zu SQL

Hintergrundinformationen (1/2)

- SQL ist eine relationale Datenbanksprache und als Standardsprache in der Mehrheit der DBMS implementiert [1]
- Vorläufer ist SEQUEL (Structured English Query Language), welche von IBM entwickelt wurde [2]
- Name wird üblicherweise als Akronym für *Structured Query Language* verstanden
- Fokus dieser Vorlesung ist Auswahl, Filterung und Verbindung von Tabellen für Datenauswertung und Analysezwecke

Hintergrundinformationen (2/2)

- SQL besitzt verschiedene Sprachaspekte:
  - *Data Definition Language (DDL):* Beschreibung Datenbankschema und Datenstrukturen (CREATE/ALTER TABLE)
  - *Data Manipulation Language (DML):* Schreiben, lesen, ändern und löschen von Daten (INSERT, UPDATE)
  - *Data Control Language (DCL):* Berechtigungsmanagement (GRANT, REVOKE)
  - *Data Query Language:* Suche nach und Auswahl von Informationen (SELECT)
- Angebot an lokalen oder online verfügbaren Test- und Übungsumgebungen ist vielfältig
- Vorlesung nutzt die Website *SQLiteOnline* für Beispiele

SQL-Syntax

Auswahl von Spalten und Tabellen

- SQL umfasst mehr als die hier aufgeführten Befehle
- Auswahl bezieht sich auf Data Science Nutzungen
- Befehl für die Auswahl von Spalten ist SELECT
- FROM gibt dazu an, von welcher Tabelle gelesen wird
- Durch DISTINCT wird jedes Objekt/jede Objektkombination nur einmal wiedergegeben

Select und From

```
SELECT col1, col2, col3
FROM tableA;
```

Distinct

```
SELECT DISTINCT col4
FROM tableB;
```

Filterung von Daten

- Filterung der Daten erfolgt über WHERE
- Operatoren AND, OR sowie (NOT) IN und BETWEEN können die Filterung ergänzen

Where

```
SELECT col5, col6
FROM tableC
WHERE col5 = 'yes';
```

Or

```
SELECT col7
FROM tableD
WHERE col7 = 'yes' OR col7 = 'no';
```

In

```
SELECT col8
FROM tableE
WHERE col8 IN (10, 20, 30);
```

Sortierung und Aggregation (1/2)

- Sortierung erfolgt über ORDER BY Operator
- Mit GROUP BY werden Gruppen/Aggregationen von Mengen anhand des gewählten Attributs gebildet
- Aggregationsfunktionen sind:
  - sum(...)
  - count(...)
  - min(...)
  - max(...)
  - avg(...)
- Filterung in/Einschränkung einer Gruppierung erfolgt über HAVING
- Einschränkung auf begrenzte Anzahl von Ergebnissen über LIMIT

Sortierung und Aggregation (2/2)

Order by

```
SELECT col10, col20
FROM tableA1
ORDER BY col10 ASC, col20 DESC;
```

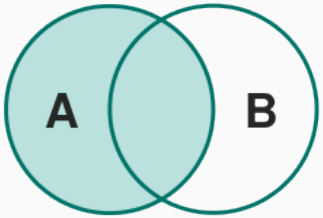
Group by

```
SELECT col30, sum(col40) AS summe,
FROM tableA2
GROUP BY col30;
```

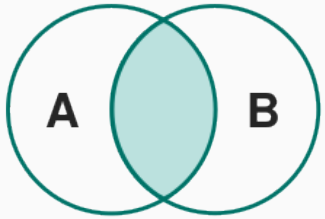
Group by mit Having

```
SELECT col98, sum(col99)
FROM tableA3
GROUP BY col98
HAVING sum(col98) > 100;
```

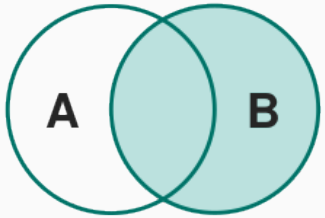
Vereinigungen



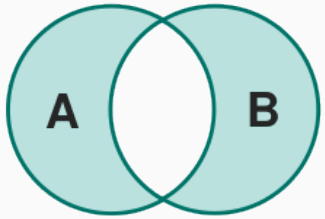

```
SELECT      <cols>
FROM        A
LEFT JOIN   B
ON          A.key = B.key
```



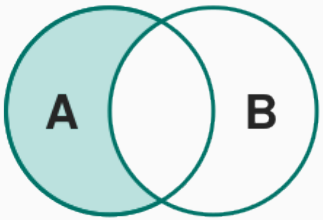

```
SELECT      <cols>
FROM        A
INNER JOIN  B
ON          A.key = B.key
```



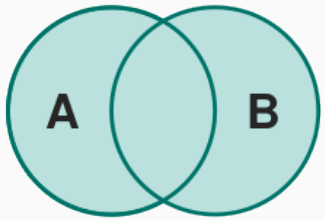

```
SELECT      <cols>
FROM        A
RIGHT JOIN  B
ON          A.key = B.key
```



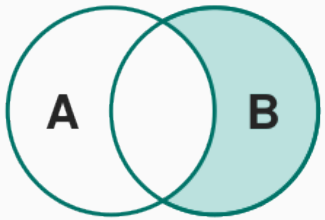

```
SELECT            <cols>
FROM              A
FULL OUTER JOIN   B
ON                A.key = B.key
WHERE             A.key IS NULL
            OR    B.key IS NULL
```




```
SELECT      <cols>
FROM        A
LEFT JOIN   B
ON          A.key = B.key
WHERE       B.key IS NULL
```




```
SELECT            <cols>
FROM              A
FULL OUTER JOIN   B
ON                A.key = B.key
```




```
SELECT      <cols>
FROM        A
RIGHT JOIN  B
ON          A.key = B.key
WHERE       A.key IS NULL
```

Reihenfolge der Befehle

Reihenfolge in der Codierung:

```
SELECT
FROM
WHERE
GROUP BY
HAVING
ORDER BY
LIMIT
```

Reihenfolge in der Ausführung:

```
FROM
WHERE
GROUP BY
HAVING
SELECT
ORDER BY
LIMIT
```

Live-Beispiele und Übungen

Vorbereitung

- Laden Sie die Kaggle-Daten *The History of Baseball* herunter
- Entpacken Sie die ZIP-Datei in ein lokales Verzeichnis
- Öffnen Sie die Website SQLiteOnline
- Laden Sie die soeben entpackte Datei *database.sqlite* über File $\rightarrow$ Open DB in die Online-Umgebung
- Am linken Rand sehen Sie die in der Datenbank enthaltenen Tabellen
- In der Mitte sehen Sie oben das Editor-Fenster und unten die Tabellenansicht

Übungen (1/4)

1. Alle Informationen aus Tabelle `college` abfragen
2. Alle College-Namen und -Städte aus Texas (`state = 'TX'`) aus Tabelle `college` abfragen
3. Alle Stadion-/Parknamen und Städte aus Georgia (`state = 'GA'`) aus Tabelle `park` abfragen
4. Alle Player- und Team-IDs aus Tabelle `manager` für das Jahr 1988 abfragen
5. Alle Bundesstaaten aus Tabelle `college` einmal aufführen
6. Anzahl der Bundesstaaten aus Tabelle `college` bestimmen
7. Alle Städte aus Tabelle `park` einmal aufführen
8. Anzahl der Stadien in Baltimore aus Tabelle `park` bestimmen
9. Alle Informationen zu Spielern aus Tabelle `player_award_vote`, die 1911 bei der Wahl des MVP (Attribut: `award_id`) berücksichtigt wurden abfragen

Übungen (2/4)

10. Alle Stadien die dem Team `team_id = 'NY1'` zugeordnet waren oder sind aus Tabelle `team` einmal aufführen
11. Anzahl der Stadien aus vorheriger Aufgabe bestimmen
12. Anzahl der Stadien aus Tabelle `park` pro Bundesstaat bestimmen
13. Anzahl der Stadien aus Tabelle `park` pro Stadt mit Angabe Bundesstaat bestimmen
14. Anzahl der Colleges aus Tabelle `college` pro Bundesstaat bestimmen
15. Anzahl der individuellen Spieler aus Tabelle `player_award_vote`, die bei der Wahl zum MVP in mindestens einem Jahr nur eine Stimme bekommen haben

Übungen (3/4)

16. Anzahl der individuellen Teams aus Tabelle `manager` pro Liga, die zumindest in einem Jahr entweder fünf oder weniger Niederlagen erlitten haben oder einen Spielertrainer hatten
17. Sortiere die Tabelle `park` alphabetisch nach Bundesstaat
18. Anzahl der Städte aus Tabelle `park` pro Bundesstaat bestimmen, mit `city_n` benennen und von der Anzahl `city_n` her absteigend sortieren
19. Informationen zu Jahr, Team-ID und Stadion aus Tabelle `team` mit den Informationen zur Stadt und Stadionnamen aus der Tabelle `park` verknüpfen (`LEFT JOIN`), nur für das Jahr 2000 und nach Städten sortiert abfragen

Übungen (4/4)

20. Informationen zu Team-ID und Stadion aus Tabelle `team` für das Jahr 2000 verknüpfen mit der Information des gesamten Jahresgehalts pro Team im Jahr 2000 aus Tabelle `salary`, jedoch nur von Teams mit Gesamtausgaben über 40 Mio. abfragen und in vom Gehalt absteigender Reihenfolge darstellen
21. Bestimmen Sie die fünf Jahre, in denen im Jahresdurchschnitt pro Gewinner-Team in der Postseason das höchste Gesamtgehalt gezahlt wurde und stellen Sie das Ergebnis in absteigender Reihenfolge dar
22. Was muss in der Query geändert werden um die fünf Jahre mit dem geringsten (numerischen) Durchschnitt zu erhalten?

Übungsaufgaben

Bitte lösen Sie die folgenden Aufgaben bis zur Vorlesung am 12. April 2023.

- Laden Sie die Kaggle-Daten *18,393 Pitchfork Reviews* herunter
- Entpacken Sie die ZIP-Datei in ein lokales Verzeichnis
- Öffnen Sie die Website SQLiteOnline
- Laden Sie die soeben entpackte Datei *database.sqlite* über File $\rightarrow$ Open DB in die Online-Umgebung

- Beantworten Sie folgende Fragen mit SQL-Queries:
  1. Wie viele Künstler/Artists (identifiziert durch Schreibweise des Namens) sind in der Datenbank vorhanden?
  2. Wie viele Bewertungen/Reviews sind pro Genre in der Datenbank gespeichert?
  3. In welchen drei Jahren wurden die meisten Reviews verfasst und wie viele waren es?
  4. Welche fünf Labels haben im Jahr 2011 die meisten Reviews erhalten und wie viele waren es pro Label?
  5. Wie lauten die IDs aller Reviews der Bands Metallica, Fugees und Ramones mit Angabe des Jahres in chronologisch ansteigender, tabellarischer Form?

Referenzen

[1] Vossen, G. (2000). Datenmodelle, Datenbanksprachen und Datenbankmanagementsysteme, 4. Auflage. Oldenbourg.

[2] Chamberlin, D. D., & Boyce, R. F. (1974, May). SEQUEL: A structured English query language. In Proceedings of the 1974 ACM SIGFIDET (now SIGMOD) workshop on Data description, access and control (pp. 249-264).

[3] Laube, M. (2019). Einstieg in SQL. Rheinwerk Verlag.

[4] Gennick, J., & Schulten, L. (2006). SQL-kurz & gut. O'Reilly.

[5] Hess, M. (2006). Kleine Einführung in SQL. Universität Zürich, Institut für Computerlinguistik. Retrieved 2020-03-09 from https://files.ifi.uzh.ch/cl/hess/classes/le/sql.0.l.pdf

[6] Pruin, H. (2018). Datenbanken und SQL. YouTube. Retrieved 2020-03-09 from http://y2u.be/fgOiWEGNJ-o