2 SQL

Modul: Angewandte Programmierung

Dennis Glüsenkamp ▪ Köln ▪ 26. März 2020

Copyright

Inhalt

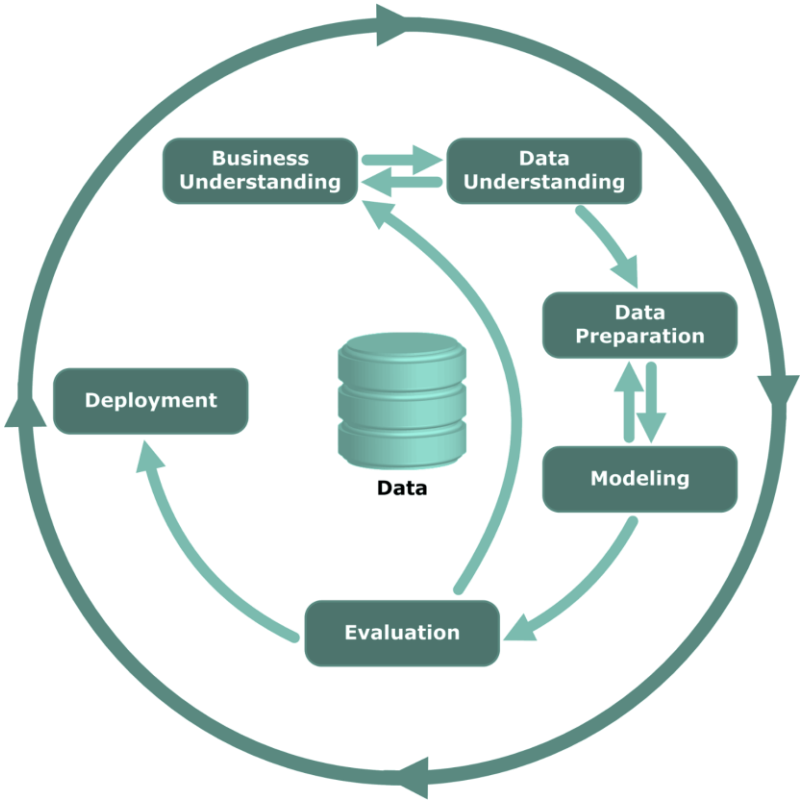
0	Übungsaufgabe der vergangenen Vorlesung
1	Hintergrundinformationen zu SQL
2	SQL-Syntax
3	Beispiele und Übungen

0 Übungsaufgabe der vergangenen Vorlesung

Übungsaufgabe

In Vorbereitung auf die heutige Vorlesung haben Sie folgende Aufgabenstellung bearbeitet:

- Bitte wählen Sie ein mögliches Data Science/Mining Vorhaben (Use Case) aus, dass Ihnen aus Ihrer beruflichen Praxis, den Medien oder aufgrund von persönlichen Interessen bekannt ist.
- Diskutieren Sie anhand dieses Beispiels mögliche Fragen, Arbeitsschritte und Ergebnisse in den sechs Schritten des CRISP-DM Lifecycles.
- Wählen Sie zusätzlich zum Vergleich ein anderes Lifecycle-Modell aus und diskutieren Sie die Gemeinsamkeiten und Unterschiede zu CRISP-DM im Rahmen Ihres Beispiels.



CRISP-DM Prozessmodelldiagramm

Image by Kenneth Jensen, distributed under a CC BY-SA 3.0.

1 Hintergrundinformationen zu SQL

Hintergrundinformationen

- 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
- 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)
- Fokus dieser Vorlesung ist Auswahl, Filterung und Verbindung von Tabellen für Datenauswertung und Analysezwecke
- Angebot an lokalen oder online verfügbaren Test- und Übungsumgebungen ist vielfältig; Vorlesung nutzt die Website https://sqliteonline.com/ für Beispiele

2 SQL-Syntax

Syntax (1/3)

- SQL umfasst mehr als die hier aufgeführten Befehle
- Auswahl bezieht sich auf typische Nutzungen im Bereich Data Science
- 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
- Filterung der Daten erfolgt über **WHERE**
- Operatoren **AND**, **OR** sowie (**NOT**) **IN** und **BETWEEN** können die Filterung ergänzen

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

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

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

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

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

Syntax (2/3)

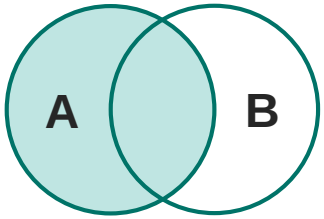
- Sortierung erfolgt über **ORDER BY** Operator
- Mit **GROUP BY** werden Gruppen/Aggregationen von Mengen anhand des gewählten Attributs gebildet
- Aggregationsfunktionen sind:
  - **sum(…)**
  - **count(…)**
  - **min(…)**
  - **max(…)**
  - **avg(…)**
- Filterung in/Einschränkung einer Gruppierung erfolgt über **HAVING**
- Literatur für weitere Befehle und Vertiefung, z.B. [3] und [4] oder als Online-Skript [5] sowie Video-Tutorial [6]

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

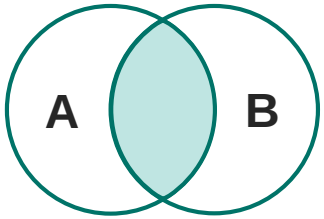
```
SELECT    col30,
          sum(col40) AS summe,
          count(col50) AS anzahl,
          min(col60) AS minimum,
          max(col70) AS maximum,
          avg(col80) AS durchschnitt
FROM      tableA2
GROUP BY  col30;
```

```
SELECT    col98, sum(col99)
FROM      tableA3
GROUP BY  col98
HAVING    col98 = 'nur diese';
```

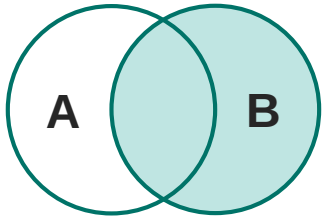
Syntax (3/3)



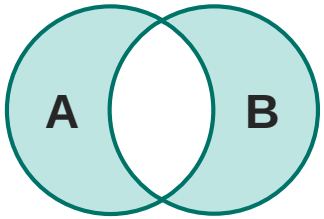

```
SELECT      <cols>
FROM        A
LEFT JOIN   B
ON          A.key = B.key
```



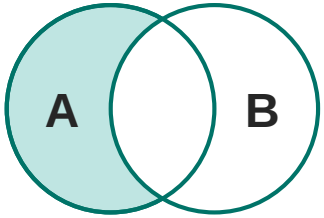

```
SELECT      <cols>
FROM        A
INNER JOIN  B
ON          A.key = B.key
```



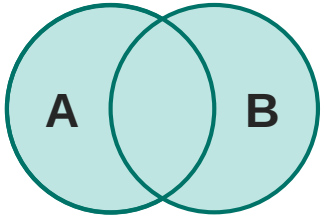

```
SELECT      <cols>
FROM        A
RIGHT JOIN  B
ON          A.key = B.key
```



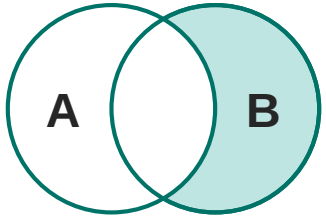

```
SELECT           <cols>
FROM             A
FULL OUTER JOIN  B
ON               A.key = B.key
WHERE            A.key IS NULL
            OR   B.key IS NULL
```




```
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
```

3 Live-Beispiele und Übungen

Livebeispiele und Übungen ▪ Vorbereitung

- Laden Sie die Kaggle-Daten „The History of Baseball“ herunter https://www.kaggle.com/seanlahman/the-history-of-baseball
- Entpacken Sie die ZIP-Datei in ein lokales Verzeichnis
- Öffnen Sie die Website https://sqliteonline.com/
- Laden Sie die soeben entpackte Datei database.sqlite über File >>> 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

Livebeispiele und Übungen

a) Alle Informationen aus Tabelle COLLEGE abfragen

b) Alle College-Namen und -Städte aus Texas (state = 'TX') aus Tabelle COLLEGE abfragen

c) Alle Stadion-/Parknamen und Städte aus Georgia (state = 'GA') aus Tabelle PARK abfragen

d) Alle Player- und Team-IDs aus Tabelle MANAGER für das Jahr 1988 abfragen

e) Alle Bundesstaaten aus Tabelle COLLEGE einmal aufführen

f) Anzahl der Bundesstaaten aus Tabelle COLLEGE bestimmen

g) Alle Städte aus Tabelle PARK einmal aufführen

h) Anzahl der Stadien in Baltimore aus Tabelle PARK bestimmen

i) Alle Informationen zu Spielern aus Tabelle PLAYER_AWARD_VOTE, die 1911 bei der Wahl des MVP (Attribut: award_id) berücksichtigt wurden abfragen

j) Alle Stadien die dem Team team_id = 'NY1' zugeordnet waren oder sind aus Tabelle TEAM einmal aufführen

k) Anzahl der Stadien aus 4.d bestimmen

l) Anzahl der Stadien aus Tabelle PARK pro Bundesstaat bestimmen

m) Anzahl der Stadien aus Tabelle PARK pro Stadt mit Angabe Bundesstaat bestimmen

n) Anzahl der Colleges aus Tabelle COLLEGE pro Bundesstaat bestimmen

o) Anzahl der individuellen Spieler aus Tabelle PLAYER_AWARD_VOTE, die bei der Wahl zum MVP in mindestens einem Jahr nur eine Stimme bekommen haben

p) 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

q) Sortiere das Ergebnis aus 5.d nach der Anzahl, die zuvor mit n benannt wurde

r) Sortiere die Tabelle PARK alphabetisch nach Bundesstaat

s) Anzahl der Städte aus Tabelle PARK pro Bundesstaat bestimmen, mit city_n benennen und von der Anzahl city_n her absteigend sortieren

t) 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

u) Informationen zu Jahr, 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

v) 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.

w) Was muss in der Query geändert werden um die fünf Jahre mit dem geringsten Durchschnitt zu erhalten?

Übungsaufgabe

Bitte lösen Sie die folgenden Aufgaben mit Hilfe einer SQL-Query bis zur übernächsten Vorlesung am 2. April 2020.

Online-SQL-IDE: https://sqliteonline.com/

Kaggle-Datensatz: 18,393 Pitchfork Reviews

https://www.kaggle.com/nolanbconaway/pitchfork-data

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

Anhang

Referenzen

[1] Vossen, G. (2000). Datenmodelle, Datenbanksprachen und Datenbankmanagement-systeme, 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