

5 Einfache SQL-Abfragen

5.1 SQL kennen lernen

SQL (*structured query language*) ist eine Programmiersprache, mit der man auf Datenbanken zugreifen kann. SQL ist eine Vertreterin der so genannten **deklarativen Programmierung**, da man mit ihr nur das Ergebnis beschreibt, nicht aber den Weg dorthin.

5.1.1 Die erste SQL-Abfrage

1. *Rufen* Sie Ihren InstaHub auf und melden Sie sich als Admin dort an!
2. Klicken oben rechts auf das Symbol *Datenbank* und wählen Sie SQL:

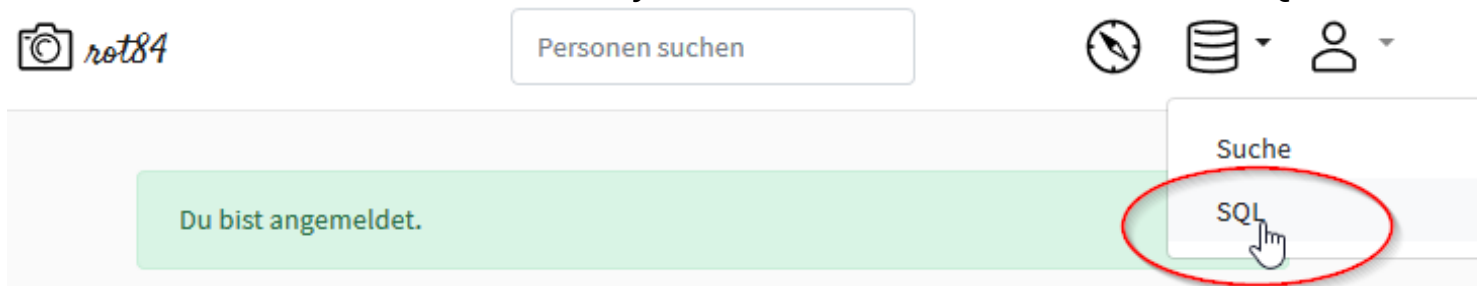


Abbildung 1: Den SQL-Editor öffnen

3. Sie gelangen in den SQL-Editor:

rot84

Personen suchen

SQL

1

Ausführen

Folgende einzelne Tabellen können abgefragt werden:

password_resets: email, token, created_at

users: id, username, email, password, name, bio, gender, birthday, city, country, centimeters, avatar, role, is_active, remember_token, created_at, updated_at

Abbildung 2: Der SQL-Editor

- Setzen Sie den Cursor in das leere, im obigen Screenshot rot markierte, Eingabefeld und tippen Sie die folgende SQL-Abfrage exakt so ein:

```
SELECT  username, name, birthday, city
FROM    users
```

Tipps: Nach dem SELECT bzw. FROM tippen Sie die Tab-Taste, damit der Befehl übersichtlich aussieht. Am Ende der ersten Zeile können Sie ENTER drücken, um in eine neue Zeile zu wechseln.

- Klicken Sie auf Ausführen! Sie sehen nun eine Ergebnistabelle mit gut 200 Zeilen:

The screenshot shows the Instahub SQL interface. At the top, a text area contains the SQL query: `1 SELECT username, name, birthday, city` and `2 FROM users`. Below the text area is a blue button labeled "Ausführen". Underneath the button is a green status bar that says "Anfrage ausgeführt. 204 Ergebnisse gefunden." with a close icon. Below the status bar is a table with the following data:

username	name	birthday	city
niclas258	Niclas Schweizer	1998-01-31 00:00:00	Wremen
rafael54	Rafael Probst	2001-08-06 00:00:00	Leipzig
luis52	Luis Krüger	2001-12-15 00:00:00	Lautertal
mustav480	Gustav Meister	2001-07-12 00:00:00	Halle

Abbildung 3: Das Ergebnis Ihres ersten SELECT-Befehls

5.1.2 Tipps:

- Beachten Sie, dass die Spaltennamen exakt mit den Bezeichnungen nach dem SELECT übereinstimmen!
- SQL unterscheidet nicht zwischen Groß- und Kleinschreibung. Es ist aber üblich, die SQL-Befehle in GROSSBUCHSTABEN zu notieren. Halten Sie sich bitte an diese Gepflogenheit.
- Wenn Sie *alle* Spalten ausgeben wollen, können Sie statt alle Spalten aufzuzählen auch einfach ein `*` notieren, z. B.;

```
SELECT *  
FROM users
```

- Wenn Sie statt einer Antwort wie der obigen eine englische Fehlermeldung bekommen, haben Sie bei der Eingabe des Befehls einen Fehler gemacht. Kontrollieren Sie noch einmal ganz genau, ob Sie irgendwo einen kleinen Fehler gemacht haben. Sind die Kommata wirklich da, wo sie sein sollen und sind es wirklich Kommata? Kein kleiner Tippfehler in den Spaltennamen?

5.2 SQL-Abfragen verschönern und sortieren

Wenn Sie die Ergebnistabelle ausdrucken wollen, kann es hilfreich sein, die

5 Einfache SQL-Abfragen

Spaltenbezeichnungen verändern zu können. *Probieren* Sie aus, was passiert, wenn Sie statt `birthday` den Text `birthday AS "Geburts-
tag"` eintragen!

```
SELECT    username, name, birthday AS "Geburtstag",  
          city AS "Stadt"  
FROM      users  
ORDER BY  birthday ASC  
LIMIT     10
```

Mit der Klausel `ORDER BY` können Sie die Ausgabe sortieren lassen, mit `LIMIT` begrenzen Sie die Zahl der ausgegebenen Datensätze.

5.2.1 Kommentare

Beginnt eine Zeile beginnt mit zwei Minuszeichen (`--`), wird diese vom SQL-Interpreter ignoriert:

```
-- verschönerte Anfrage  
SELECT    username, name, birthday AS "Geburtstag",  
          city AS "Stadt"  
FROM      users
```

Man nennt solche ignorierten Zeilen *Kommentar*. Kommentare sind beim Programmieren wichtig, um zu erklären, was man gerade macht. Zum Beispiel kann man der Lehrkraft mitteilen, welche Aufgabe man bearbeitet. Das ist übrigens auch für Sie selbst hilfreich, wenn Sie (z. B. zur Klausurvorbereitung) noch einmal die alten Aufgaben herausholen.

Es gibt da nur ein Problem: InstaHub verarbeitet nur den ersten Befehl, den man in das SQL-Fenster eingibt. Kommentare sind Befehle. Daher wird der Kommentar "ausgeführt" und nicht die Abfrage.

Also: **Keine Kommentare im SQL-Fenster von InstaHub!**

Wenn Sie mehrere Zeilen als Kommentar markieren wollen, verwenden Sie `/*` und `*/`:

```
/* Verschoenerte Abfrage  
   Liefert alle Nutzer*innen nach Geburtstag sortiert. */  
SELECT    username, name, birthday AS "Geburtstag",  
          city AS "Stadt"  
FROM      users  
ORDER BY  birthday ASC  
LIMIT     10
```

5.2.2 ORDER und LIMIT

Statt `ASC` können Sie in der `ORDER`-Klausel auch `DESC` oder gar nichts verwenden. *Beschreiben* Sie, was jeweils passiert!

Beschreiben Sie, welche Bedeutung die LIMIT-Klausel hat!

Aufgabe 5.1: Erste SQL-Abfragen

Erstellen Sie jeweils eine SQL-Abfrage, die die folgenden verbalen Anfragen möglichst schön beantworten:

1. Wer sind die zehn jüngsten InstaHub-User*Innen?
2. Wo wohnen die fünf am längsten angemeldeten InstaHub-User*Innen?
3. Wer ist der erste InstaHub-User*in gewesen?

5.3 Dubletten aussortieren

Uns interessiert brennend, in welchen Städten wir bereits mindestens eine/n Nutzer*In haben. Wir probieren es mit der folgenden Abfrage:

```
SELECT    city
FROM      users
ORDER BY  city ASC
```

Das Ergebnis ist eine lange Liste von Städten, in denen aber einige Städte mehrfach vorkommen.

Probieren Sie es einmal mit

```
SELECT    DISTINCT city
FROM      users
ORDER BY  city ASC
```

Nun werden alle Zeilen aussortiert, die mehr als einmal vorkommen.

Aufgabe 5.2: Dubletten aussortieren

Erstellen Sie jeweils eine SQL-Abfrage, die folgenden verbalen Anfragen möglichst schön beantworten:

1. Welche Körpergrößen haben unser Nutzer*innen?
2. Welche verschiedenen Werte kommen in der Spalte `role` vor?

5.4 Datensätze filtern

Welche InstaHub-Nutzer*Innen leben eigentlich in Dresden?

Wir könnten nun die Liste nach Städten sortieren und dann scrollen, bis wie bei *Dresden* angekommen sind. Das muss einfach gehen. Na klar. Die wohl wichtigste Klausel des SELECT-Befehls fehlt uns noch: `WHERE`

```
SELECT    username, name, city
WHERE     city = "Dresden"
FROM      users
```

Aufgabe 5.3: Deutschlandreise I

Erstellen Sie jeweils eine SQL-Abfrage, die folgenden verbalen Anfragen möglichst schön beantworten:

1. Welche Nutzer*Innen wohnen in Leipzig?
2. Welche Nutzer*Innen haben die Rolle dba?
3. Welche Nutzer*Innen wohnen in Bokholt-Hanredder?
4. Welche Nutzer*Innen wohnen in Hamburg?
5. Ist Justin Schuster bei uns angemeldet?

Aufgabe 5.4 Ups...

Überprüfen Sie Ihre Ergebnisse aus den beiden letzten Abfragen der vorangegangenen Aufgabe noch einmal genau und *begründen* Sie, warum Ihre Ergebnisse offenbar nicht ganz korrekt sind!

5.4.1 Zeichenketten vergleichen

Beim Vergleichen von Zeichenketten ist es oft hilfreich, wenn man nach Teilzeichenketten suchen kann. Hierfür hat SQL den Operator `LIKE`, der anders als `=` nicht die exakte Übereinstimmung erfordert, sondern mit Hilfe von Platzhalten (*Wildcards*) auch Wortteile finden kann. So würde `WHERE name LIKE 'Justin%Schuster%'` auch die folgenden Personen finden, wenn Sie denn als Nutzer*innen angemeldet wären:

- Justine Schuster
- Justin Frederik Augustus Frederikus Freiherr von Schuster-Boomzwinkel

Während mit `%` beliebig viele Zeichen (inklusive keinem) gefunden werden, wird mit dem Zeichen `_` (Unterstrich, Shift+Minuszeichen) nur genau ein Zeichen gefunden.

Aufgabe 5.5 Deutschlandreise II

Erstellen Sie jeweils eine SQL-Abfrage, die folgenden verbalen Anfragen möglichst schön beantworten:

1. Welche Nutzer*Innen wohnen in Hamburg?
2. Welche Nutzer*Innen mit dem Namen Schuster sind bei uns angemeldet?
3. Welche Nutzer*Innen mit dem haben wir, die so klingen wie Meier?
(Hier können Sie einen Beifang irrelevanter Datensätze wohl nicht völlig verhindern!)

5.4.2 Numerische Vergleiche

Mit den Vergleichsoperatoren $>$, $>=$, $<$ und $<=$ können auch numerische Werte verglichen werden.

Das funktioniert auch mit Kalenderdaten. `WHERE created_at < "2017-12-31"` liefert alle Benutzer*Innen, deren Datensätze vor dem 31.12.2017 erstellt wurden.

Aufgabe 5.6 Riesen und Zwerge

Erstellen Sie jeweils eine SQL-Abfrage, die folgenden verbalen Anfragen möglichst schön beantworten:

1. Welche InstaHub-Nutzer*Innen sind größer als 188 cm?
2. Welche InstaHub-Nutzer*Innen sind kleiner als 155 cm?
3. Welche InstaHub-Nutzer*Innen sind heute am Tag der Aufgabenbearbeitung *nicht* volljährig?
4. Welche InstaHub-Nutzer*Innen sind älter als 70 Jahre?
5. Gibt es eine/n Nutzer*In, die am selben Tag Geburtstag hat wie Sie?
(Hinweis: Wenn Sie am 12.10.1998 geboren sein sollten, suchen Sie, ob jemand am 12.10.1998 geboren ist!)