# Dossier

Von Luis Scholten und Jonathan Plum

Zeitaufwand Meilenstein 1:

|  |  |  |  |
| --- | --- | --- | --- |
| Aufgabe | Zeitaufwand geplant | Zeitaufwand tatsächlich | Grund für Abweichung |
| 1 | 5 min | 5 min |  |
| 2 | 3 min | 3 min |  |
| 3 | 10 min | 15 min | Ablenkung |
| 4 | 15 min | 25 min | Recherche auf w3school |
| 5 | 30 min | 30 min |  |
| 6 | 10 min | 20 min | Länger als erwartet |
| 7 | 5 min | 10 min | Kleine Fehler ausgebessert |
| 8 | 15 min | 15 min |  |
| 9 | 40 min | 1 h | Mussten noch viel nachschauen |
| 10 | 1,5 h | 1,5 h |  |

Aufgabe 6: 1)

Ein Bild, das Text enthält.

Automatisch generierte BeschreibungEin Bild, das Text enthält.

Automatisch generierte Beschreibung

* Content-Language: de
  + In welcher Sprache soll die Antwort kommen
* Date
  + Datum der Anfrage
* Connection
  + Soll Verbindung bestehen bleiben oder nicht?
* Server
  + Was für ein Server läuft und auf welchem Betriebssystem
* Cache-Control
  + Wo und ob Cache gespeichert werden soll

2)

Ein Bild, das Text enthält.

Automatisch generierte BeschreibungEin Bild, das Text enthält.

Automatisch generierte Beschreibung

* HTTP/1.1 404 Not Found
  + Da die Seite nicht existiert, kommt der http response code 404
* content-type: text/html
  + Typ der Antwort
* content-length: 92
  + Länge des Headers in byte
* server: PhpStorm 2022.2.2
  + Server + Version
* date: Tue, 04 Oct 2022 13:38:48 GMT
  + Zeitpunkt der Antwort
* x-frame-options: SameOrigin
  + ob man die seite rendern können soll oder nciht
* X-Content-Type-Options: nosniff
  + Sagt, ob die MIME-Types befolgt werden müssen
* x-xss-protection: 1; mode=block
  + stoppt laden einer seite wenn nötig
* access-control-allow-origin: <http://localhost:63342>
  + Die Herkunft/URL des Servers
* vary: origin
  + Zeigt an ob eine Komprimierung der Seite vorliegt
* access-control-allow-credentials: true
  + Ob JavaScript genutzt werden kann/darf

Aufgabe 3.1:

-in Zeile 12 fehlte ein Komma

-in Zeile 21 fehlte eine eckige Klammer

-in Zeile 59 fehlte function

-in Zeile 41 fehlte eine Klammer

-in Zeile 44 str\_pos zu str\_contains geändert

Aufgabe 3.2:

- Dynamischen String generieren (Titel und Überschrift)

-Anzeige der Liste -> GET

-Suche in der Liste -> GET

-Anzeige Bewertungen -> GET

Aufgabe 3.3:

Empty() ->checkt ob Variable leer ist

: float bei funktionen setzt den Rückgabewert auf float

Aufgabe 3.4:

Zeitaufwand Meilenstein 2:

|  |  |  |  |
| --- | --- | --- | --- |
| Aufgabe | Zeitaufwand geplant | Zeitaufwand tatsächlich | Grund für Abweichung |
| 1 | 10 min | 10 min |  |
| 2 | 15 min | 20 min | Dauer des quizzes |
| 3 | 1,5 h | 2 h | Dauer unterschätzt |
| 4 | 5 min | 5 min |  |
| 5 | 1,5 h | 1,5 h |  |
| 6 | 1 h | 1 h |  |
| 7 | 45 min | 1,5 h | Aufwand unterschätzt |
| 8 | 2 h | 2,5 h | Zusätzliches ausprobiert |

Meilenstein 3:

Aufgabe 2.4)

SELECT COUNT(code) FROM allergen -> gibt die Anzahl der Einträge in Column „zurück“.

Zeitaufwand Meilenstein 3:

|  |  |  |  |
| --- | --- | --- | --- |
| Aufgabe | Zeitaufwand geplant | Zeitaufwand tatsächlich | Grund für Abweichung |
| 1 | 30 min | 1,5h | Gab ein paar Probleme bei der Verbindung |
| 2 | 2h | 2h |  |
| 3 | 2h | 2h |  |
| 4 | 1h | 1,5h | verschätzt |
| 5 | 1,5h | 3h | Allergene haben ein bisschen Zeit gekostet |
| 6 | 2h | 2h |  |
| 7 | 30min | 30min |  |
| 8 |  |  |  |
| 9 | 2h | 2h |  |
| 10 | 1h | 1h |  |
| 11 | 2h | 2h |  |

Aufgabe 3:

SELECT \* FROM gericht; Ein Bild, das Text enthält.

Automatisch generierte BeschreibungSELECT erfasst\_am FROM gericht;

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

SELECT erfasst\_am, name FROM gericht ORDER BY name DESC ; Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

SELECT beschreibung, name FROM gericht ORDER BY name ASC LIMIT 5;

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

SELECT beschreibung, name FROM gericht ORDER BY name ASC LIMIT 10 OFFSET 5;

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

SELECT DISTINCT(typ) FROM allergen;

Ein Bild, das Text enthält.

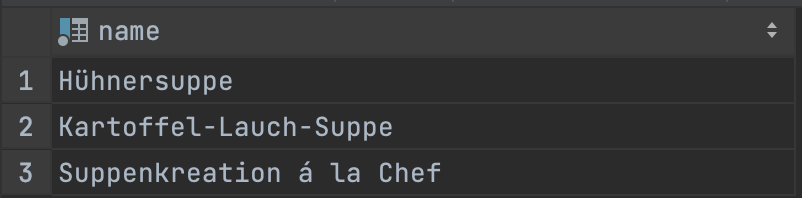
Automatisch generierte Beschreibung

SELECT name FROM gericht WHERE name LIKE 'K%';

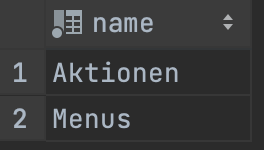
Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

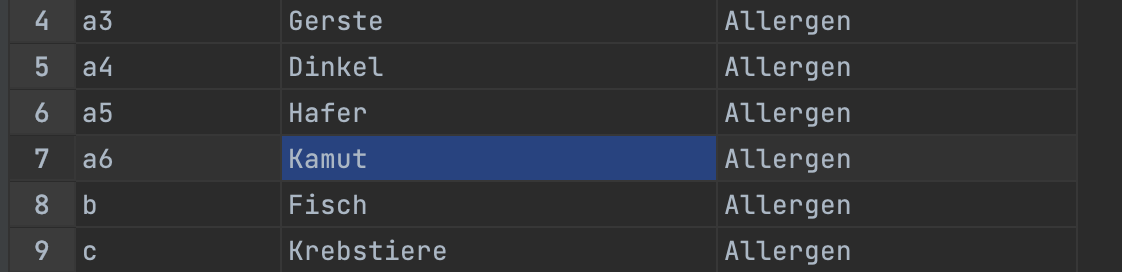
SELECT name FROM gericht WHERE name LIKE '%suppe%';



SELECT name FROM kategorie WHERE eltern\_id is null ;



UPDATE  
 allergen  
SET  
 name = 'Kamut'  
Where  
 code = 'a6';

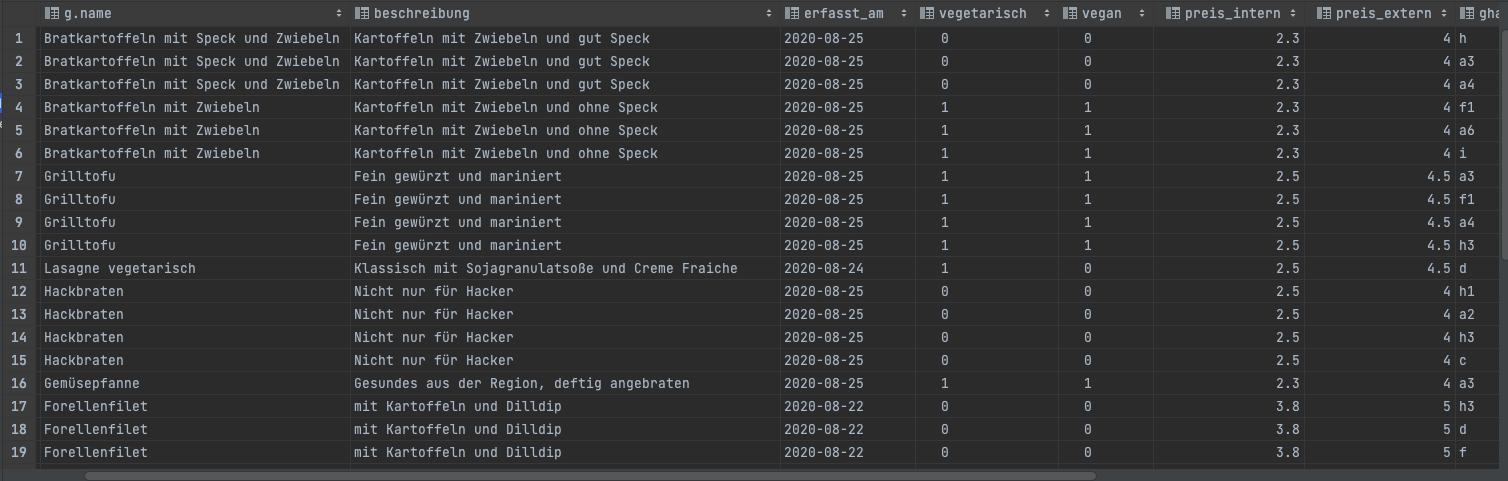


INSERT INTO gericht VALUE (21, 'Pommes mit Currywurst', 'Stinken nach Jonathan', '12.12.12', 0, 0, 10, 11 );



Aufgabe 6

SELECT \* FROM gericht g JOIN gericht\_hat\_allergen gha ON g.id = gha.gericht\_id JOIN allergen a on gha.code = a.code;



SELECT \* FROM gericht g LEFT JOIN gericht\_hat\_allergen gha ON g.id = gha.gericht\_id LEFT JOIN allergen a on gha.code = a.code;



SELECT \* FROM gericht g LEFT JOIN gericht\_hat\_allergen gha ON g.id = gha.gericht\_id RIGHT JOIN allergen a on gha.code = a.code;

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

SELECT *COUNT*(name), name FROM kategorie k RIGHT JOIN gericht\_hat\_kategorie ghk on k.id = ghk.kategorie\_id GROUP BY name ORDER BY *COUNT*(name) ;

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

SELECT *COUNT*(name) AS anzahl, name FROM kategorie k RIGHT JOIN gericht\_hat\_kategorie ghk on k.id = ghk.kategorie\_id GROUP BY name HAVING anzahl > 2 ORDER BY *COUNT*(name) ASC;

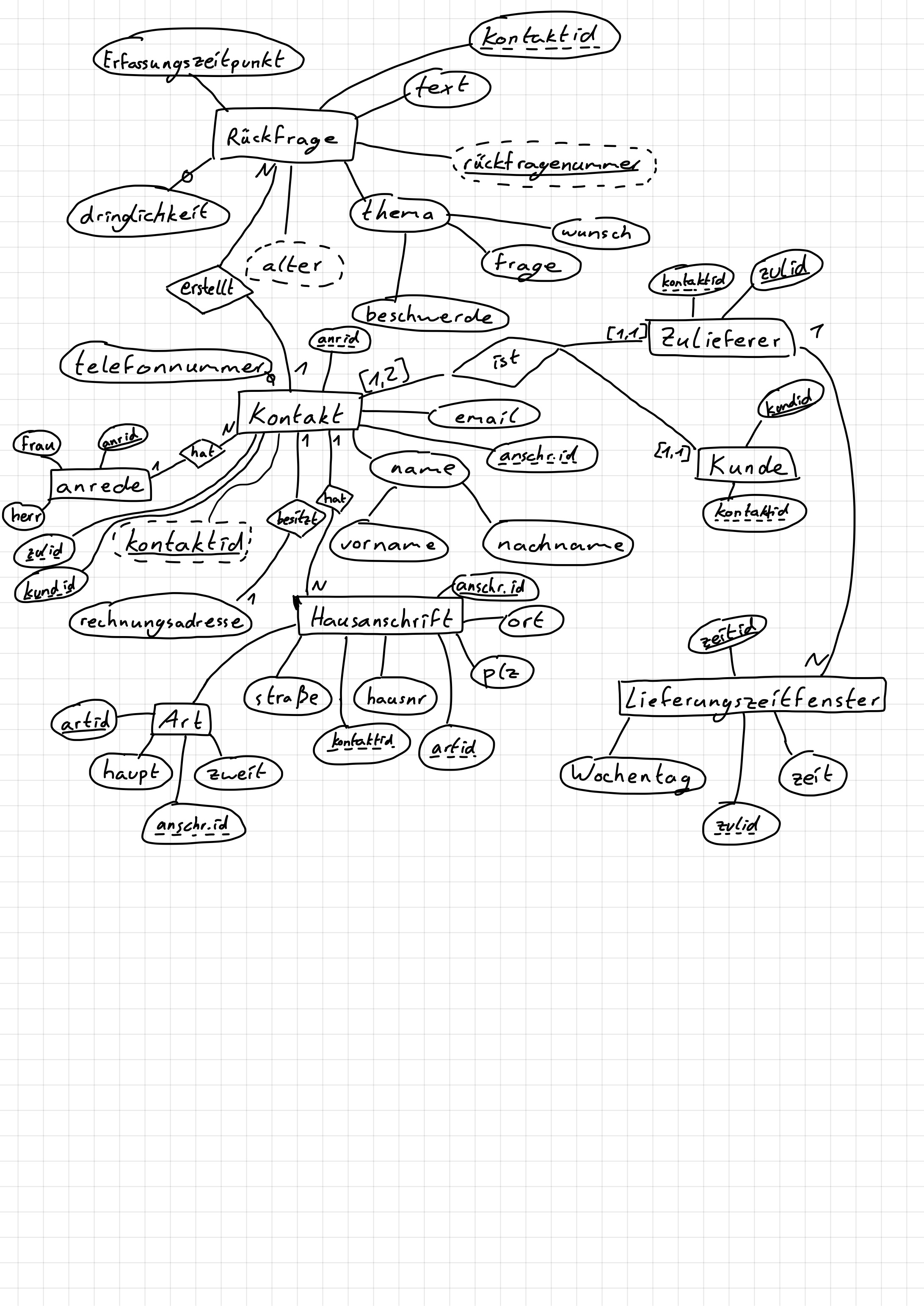
Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

Aufgabe 11

* Erleichtert Abfragen mit Bedingungen
* Ist Schneller
* Verknüpfungen zw. Datenbanken leicht
* Von mehreren Nutzern gleichzeitig nutzbar
* Sind skalierbar

Ein Bild, das Text, Quittung enthält.

Automatisch generierte Beschreibung

Zeitaufwand Meilenstein 4:

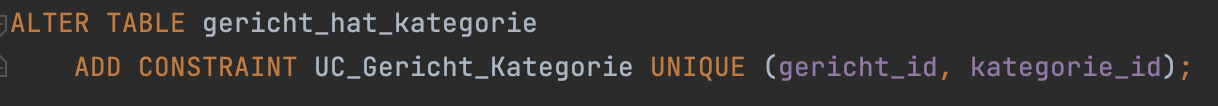
|  |  |  |  |
| --- | --- | --- | --- |
| Aufgabe | Zeitaufwand geplant | Zeitaufwand tatsächlich | Grund für Abweichung |
| 1 | 3h | 5h | Aufgabe falsch gelesen |
| 2 | 1h | 1h |  |
| 3 | - | - | - |
| 4 | 2h | 4h | Besonders Aufgabe 3 hat viel Zeit benötigt |
| 5 | 1h | 1h |  |
| 6 | 1h | 1h |  |
| 7 | 2h | 4h | Mussten uns doch mehr in Blade einfinden |
| 8 | 3h | 4h | Hatten teilweise Probleme bei der Implementierung |

Aufgabe 2:

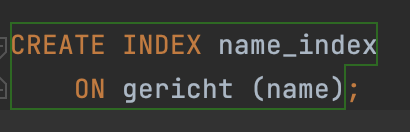
Um für XSS und SQL-Injection vorzubeugen haben wir die commands htmlspecialchars() und mysqli\_real\_escape\_string() bei allen Variablen, die über Post-Requests reinkommen, angewendet.

Aufgabe 4:

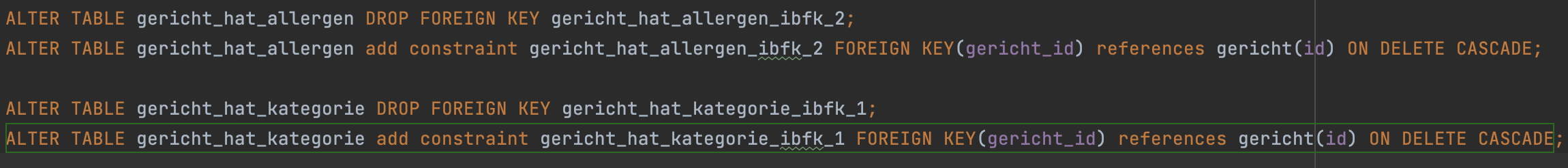
1)



2)



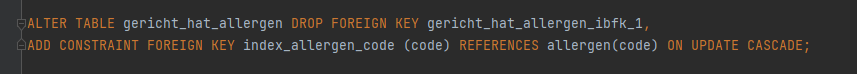
3)



4)



5)



6)

