EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Advanced SQL
Winter Term 2025/2026
Prof. Torsten Grust, Tim Fischer, Björn Bamberg
WSI — Database Systems Research Group

Assignment 5

Hand in this assignment until **Friday, November 28th 2025, 12:00 pm** at the latest.

🤔 Running out of ideas?
Are you hitting a roadblock? Are some of the exercises unclear? Do you just need that one hint to get the ball rolling? Refer to the `#forum` channel on our Discord server—maybe you'll find just the help you need.

⚠️ The lecture evaluation is coming up!
We rely on your feedback to steer Advanced SQL in the right direction. We look forward to both your criticism and your praise! So please take part in the lecture evaluation by Friday, December 5th 2025. Thank you very much!

Task 1: Magic the JSONing

JSON

We provided you with a JSON file `magic-cards.json` which encodes a list of cards for Magic the Gathering (MTG)[1]. Details and rules of the game are not relevant for this task. The format of the JSON file is well documented in `magic-doc.html`.

The given file `magic.sql` provides you with the code to load the data into DuckDB. Write your following SQL queries using the now populated table `cards`.

A Count how many cards exist with type `Enchantment` excluding cards of type `Enchantment - Aura`.

B List the names of the top five cards with the highest `cmc` which satisfy the following predicates: `power` or `toughness` are greater than 14 or `power` is less than `toughness`. Disregard cards with `power` or `toughness` containing the character `*`.

> 💡 Hint
> Remember `TRY_CAST` from the lecture to cast the values from `power` and `toughness` to `float`.

C Count how many cards exist with `mana_cost` of **exactly** `{U}`, `{U}{U}` or `{U}{U}{U}`.

Task 2: Earthquakes

JSON

We provided you with with two files:

- `earthquakes.json` encodes a list of earthquakes detected between April 17 and May 17, 2022. The format of this JSON file is well documented[2][3].
- `earthquakes.sql` copies the JSON data from `earthquakes.json` into a table `earthquakes`:

```sql
CREATE TABLE earthquakes (
  title text,
  quake json
);
```

Run `earthquakes.sql` and then write the following queries which use table `earthquakes`:

A Find the `title` and `depth` of those earthquakes with the highest and lowest `depth`.

B List the `title` and `date` value of all earthquakes with the highest `magnitude` on each day in the dataset.

> 💡 Hint
> Use `to_timestamp(time / 1000)` to convert `time` (in milliseconds since `1970-01-01T00:00:00.000Z`) to a timestamp (which can then be casted into a `date` value).

[1] https://en.wikipedia.org/wiki/Magic:_The_Gathering
[2] https://earthquake.usgs.gov/earthquakes/feed/v1.0/geojson.php
[3] https://earthquake.usgs.gov/data/comcat/data-eventterms.php

C Find the `title` of the earthquake that occured closest to the Sand (latitude: 48.534542, longitude: 9.071296). To calculate the distance between two points p_1, p_2, use the ***haversine*** formula[4], which is provided in the form of a DuckDB macro in `earthquakes.sql`.

```
haversine(lat_p1 float, lon_p1 float, lat_p2 float, lon_p2 float)
```

[4] https://en.wikipedia.org/wiki/Haversine_formula