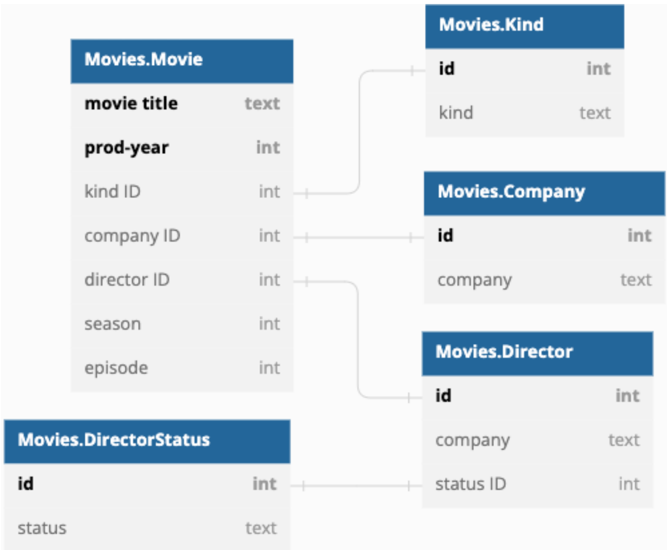
Scalable Data Engineering, Lösungsvorschlag Test Exam WS 2022/23

Henry Haustein

Question 1: Multidimensional Modelling

(a) This is the same as exercise 1, task 4. I decided to don't introduce a `movie ID` but instead make `movie title` and `prod-year` a primary key. Software like KODI (https://kodi.tv/) and Plex (https://www.plex.tv/de/) rely on this when they index your local collection of movies.



(b) It's a Snowflake schema since the schema is fully normalised.

Question 2: Reporting functions

SQL, not tested (I don't have a pgAdmin at hand right now)

```sql
SELECT P_NAME, SUM(L_QUANTITY) AS qty
FROM PART, LINEITEM
WHERE P_PARTKEY = L_PARTKEY
GROUP BY L_PARTKEY
ORDER BY 2 DESC
```

Question 3: Schema Matching

(a) Bigram representations:

- firstname: _f, fi, ir, rs, st, tn, na, am, me, e_
- lastname: _l, la, as, st, tn, na, am, me, e_
- street: _s, st, tr, re, ee, et, t_
- name: _n, na, am, me, e_
- adress: _a, ad, dr, re, es, ss, s_
- forename: _f, fo, or, re, en, na, am, me, e_

(b) Bigramm scores

	firstname	**lastname**	**street**
name	$\frac{2\cdot 4}{10+5} = 0.533$	$\frac{2\cdot 2\cdot 4}{9+5} = 0.571$	$\frac{2\cdot 0}{7+5} = 0$
address	$\frac{2\cdot 0}{10+8} = 0$	$\frac{2\cdot 0}{9+8} = 0$	$\frac{2\cdot 1}{7+8} = 0.133$
forename	$\frac{2\cdot 5}{10+9} = 0.526$	$\frac{2\cdot 4}{9+9} = 0.444$	$\frac{2\cdot 1}{7+9} = 0.125$

Stable Marriage Algorithm:

- firstname proposes to name, agrees: (firstname, name)
- lastname proposes to name, agrees + leaves: (lastname, name)
- firstname proposes to forename, agrees: (lastname, name), (firstname, forename)
- street proposes to adress, agrees: (lastname, name), (firstname, forename), (street, adress)

Question 4: Time Series Data

Trend extraction via moving average with window size 7. Then $trend(1)$, $trend(2)$ and $trend(3)$ all NULL

$$trend(4) = avg(8, 10, 11, 12, 16, 17, 20) = 13.429$$
$$trend(5) = avg(10, 11, 12, 16, 17, 20, 24) = 15.714$$
$$trend(6) = avg(11, 12, 16, 17, 20, 24, 23) = 17.571$$

$trend(7)$, $trend(8)$ and $trend(9)$ are then NULL again.

Question 5: Word Embedding Training

In the input layer we input the vector representation (dimensionality is 10) of every word which is in the sentence (6 words). So we need 60 neutrons.

The hidden layer size is 10 as stated in the task.

In the output layer we want a probability for every words, so we need there 500 neurons.