Blatt 7

Jonathan Barthelmes Igor Dimitrov Jacob Rose

note: html Version der Abgabe fuer leichtere Kopierung der Codeblocks.

Aufgabe 1

1.

```
select tu.real_name , tu.twitter_name
from twitter_user tu
where tu.typ = 'politician'
and exists (
    select *
    from twitter_user tu2
    where tu2.twitter_name <> tu.twitter_name
    and tu2.real_name = tu.real_name
)
```

Table 1: A1.1

real_name	twitter_name
Martin Hagen	_MartinHagen
Martin Hagen	MartinHagenHB

2.

```
select
    tu.real_name real_name,
    tu.twitter_name twitter_name,
    tu.follower_count follower_count,
    tu.tweet_count tweet_count,
```

```
    array_length(c.tweets, 1) conversation_length
from tweet t, conversation c, twitter_user tu
where t.id = c.id
and tu.id = t.author_id
and array_length(c.tweets, 1) >= all (
    select array_length(c2.tweets, 1)
    from conversation c2
)
```

Table 2: A1.2

real_name	twitter_name	follower_count	tweet_count	conversation_length
Tom Schreiber	TomSchreiberMdA	5328	48186	86
Christian Lindner	c_lindner	653690	18882	86

3.

```
select ne.txt, ne.id, count(*)
from
    tweet t,
    hashtag_posting hp,
    hashtag h,
    named_entity ne,
    named_entity_posting nep
where t.id = hp.tweet_id
and hp.hashtag_id = h.id
and h.txt ilike 'energie'
and nep.tweet_id = t.id
and nep.named_entity_id = ne.id
group by ne.txt, ne.id
having count(*) >= 4
order by count(*) desc
```

Table 3: A1.3

txt	id	count
Deutschland	31	18
Bayern	240	7
Thüringen	526	6
Anschluss	1741	5

txt	id	count
Berlin	2	4
Bernhard Stengele	11253	4
CDU	65	4
Europa	217	4
Bund	655	4

4.

```
select
    ne.id entity_id,
    ne.txt entity_txt,
    date(t.created_at) datum,
    count(*) anzahl
from
    tweet t ,
    named_entity_posting nep ,
    named_entity ne
where t.id = nep.tweet_id
and ne.id =nep.named_entity_id
group by ne.id, ne.txt, date(t.created_at)
order by count(*) desc
limit 5
```

Table 4: A1.4

entity_id	entity_txt	datum	anzahl
6	Ukraine	2023-02-24	761
2	Berlin	2023-02-12	427
28	Bundestag	2023-03-17	286
2	Berlin	2023-02-10	283
1425	CSU	2023-03-17	259

Aufgabe 2

1.

- **umg**: Was sind die echten Namen von allen Twitter Benutzern, die Lobbyisten sind, die einen Tweet mit ueber 2000 Likes veroeffentlicht haben, der die EU oder die USA erwaehnt?

- **tup**:

$$\{\langle \texttt{tu.real\_name} \rangle \mid \texttt{tu} \in \texttt{twitter\_user} \wedge \texttt{tu.typ='lobby'} \wedge \exists \texttt{t} \exists \texttt{ne} \exists \texttt{nep}(\\ \texttt{t} \in \texttt{tweet} \wedge \\ \texttt{ne} \in \texttt{named\_entity} \wedge \\ \texttt{nep} \in \texttt{named\_entity\_posting} \wedge \\ \texttt{t.id = nep.tweet\_id} \wedge \\ \texttt{ne.id = nep.named\_entity\_id} \wedge \\ \texttt{t.like\_count} > 2000 \wedge \\ \texttt{t.author\_id = tu.id} \wedge \\ (\texttt{ne.txt = 'EU'} \vee \texttt{ne.txt = 'USA'}))\}$$

2.

- **umg**: Was sind die IDs aller Authoren, die zwar einen Tweet mit dem Hashtag “openai” verfasst haben aber keinen mit dem Hashtag “chatgpt”.
- **tup**:

$$\{\langle \texttt{t.author\_id} \rangle \mid \texttt{t} \in \texttt{tweet} \wedge \exists \texttt{h} \exists \texttt{hp}(\\ \texttt{h} \in \texttt{hashtag} \wedge \\ \texttt{hp} \in \texttt{hashtag\_posting} \wedge \\ \texttt{h.id = hp.hashtag\_id} \wedge \\ \texttt{hp.tweet\_id = t.id} \wedge \\ \texttt{h.txt = 'openai'}) \wedge \\ \neg \exists \texttt{h} \exists \texttt{hp}(\\ \texttt{h} \in \texttt{hashtag} \wedge \\ \texttt{hp} \in \texttt{hashtag\_posting} \wedge \\ \texttt{h.id = hp.hashtag\_id} \wedge \\ \texttt{hp.tweet\_id = t.id} \wedge \\ \texttt{h.txt = 'chatgpt'})\}$$