

## Pierwszy sprawdzian z SQL

**Zadanie 1** Wypisz ile jest wszystkich kobiet, które spełniają jednocześnie następujące warunki:

- urodziły się w styczniu,
- używają przeglądarki Firefox,
- mają mniej niż 30 lat (na dziś).

```
select count(*)
from person
where
    extract(month from birthday) = 1
    and gender = 'female'
    and browserused = 'Firefox'
    and extract(year from age(birthday)) <30;
```

**Zadanie 2** Wypisz imiona i nazwiska osób, które napisały jakiegoś posta oraz jakiś komentarz. Interesują nas wyłącznie osoby z nazwiskami na literę E. Zadbaj aby wyniki się nie powtarzały oraz były posortowane alfabetycznie: w pierwszej kolejności wg nazwisk, w drugiej wg imion.

```
select distinct firstname, lastname
from
    person join
    post_hascreator_person pp on (id=pp.personid) join
    comment_hascreator_person cp on (id=cp.personid)
where
    lastname LIKE 'E%'
order by lastname, firstname;
```

**Zadanie 3** Wypisz imiona i nazwiska osób, które napisały jakiegoś posta oraz komentarz będący odpowiedzią na tego posta. Jak poprzednio interesują nas wyłącznie osoby z nazwiskami na literę E.

```
select firstname, lastname
from
    person join
    post_hascreator_person pp on (id=pp.personid) join
    comment_hascreator_person cp on (id=cp.personid) join
    comment_replyof_post crp on (pp.postid=crp.postid
                                and cp.commentid=crp.commentid)
where
    lastname LIKE 'E%';
```

**Zadanie 4** Wypisz imiona i nazwiska osób, które nie polubiły żadnego komentarza. Zadbaj aby wyniki się nie powtarzały oraz były posortowane alfabetycznie: w pierwszej kolejności wg nazwisk, w drugiej wg imion.

```
select distinct firstname, lastname
from
    person p left join
    person_likes_comment pc on (p.id = pc.personid)
where
    pc.personid is null
order by lastname, firstname;
```

## Drugi sprawdzian z SQL

**Zadanie 1** Dla danej osoby  $p$  oznaczmy przez  $L(p)$  liczbę różnych osób, które polubiły jakikolwiek post stworzony przez  $p$ . Dla każdej osoby  $p$ , dla której  $L(p)$  przekracza 50 wypisz id, imię, nazwisko, płeć oraz liczbę  $L(p)$ . Wyniki posortuj wg  $L(p)$  malejąco. *Wzorcowa odpowiedź zwraca 11 krotek.*

```
select p.id, p.firstname, p.lastname, p.gender, count(distinct plp.personid)
from
  person p join
  post_hascreator_person php on (p.id=php.personid) join
  person_likes_post plp on (php.postid=plp.postid)
group by p.id, p.firstname, p.lastname, p.gender
having count(distinct plp.personid)> 50
order by count(distinct plp.personid) desc;
```

**Zadanie 2** Wypisz id, imię i nazwisko dla wszystkich osób, których komentarze zostały łącznie polubione maksymalnie wiele razy, przy czym liczymy wyłącznie polubienia z dnia 1 grudnia 2011 i późniejszych. *Zapytanie wzorcowe zwraca 2 osoby, których komentarze zostały polubione łącznie po 117 razy.*

```
select p.id, p.firstname, p.lastname, count((plc.personid,plc.commentid))
from
  person p join
  comment_hascreator_person chp on (p.id = chp.personid) join
  person_likes_comment plc on (chp.commentid=plc.commentid)
where
  plc.creationdate >= date '2011-12-01'
group by p.id,p.firstname,p.lastname
having
  count((plc.personid,plc.commentid)) >=
  ALL (select count((plc.personid,plc.commentid))
      from
        person p join
        comment_hascreator_person chp on (p.id = chp.personid) join
        person_likes_comment plc on (chp.commentid=plc.commentid)
      where
        plc.creationdate >= '2011-12-01'
      group by p.id, p.firstname, p.lastname);
```

albo

with tbl as

```

(select p2.id, p2.firstname, p2.lastname, count(*) as ct
 from
   person p join
   person_likes_comment plc on (p.id=plc.personid) join
   comment_hascreeator_person chc on (chc.commentid=plc.commentid) join
   person p2 on (chc.personid=p2.id)
 where
   plc.creationdate >= date '01-12-2011'
 group by p2.id, p2.firstname, p2.lastname)
select id, firstname, lastname, ct
from tbl
where ct=(select max(tbl.ct) from tbl);

```

**Zadanie 3** Tabela `comment_replyof_comment` zawiera wszystkie pary indeksów (`commentid`, `commentid_2`), takie, że komentarz `commentid` jest bezpośrednią odpowiedzią na komentarz `commentid_2`. Powiemy, że komentarz *a pośrednio odpowiada* na komentarz *b* wtedy i tylko wtedy gdy istnieje ciąg komentarzy  $c_0, c_1, c_2, \dots, c_n$  taki, że  $c_0 = a$ ,  $c_n = b$  oraz dla każdej liczby  $0 \leq i < n$ , komentarz  $c_i$  jest bezpośrednią odpowiedzią na komentarz  $c_{i+1}$ .

1. Dla każdego komentarza wypisz ile komentarzy odpowiada na niego pośrednio.
2. Dla każdego komentarza wypisz na ile komentarzy odpowiada on pośrednio.

W obu przypadkach wynik posortuj wg liczb komentarzy malejąco, wypisz pierwsze 20 odpowiedzi.

*Wskazówka: korzystając z konstrukcji WITH RECURSIVE oblicz tranzytywne domknięcie tabeli `comment_replyof_comment`.*

```

WITH recursive tc(cid, cid2) as (
select commentid as cid, commentid_2 as cid2
from comment_replyof_comment
UNION ALL
select crc.commentid as cid, tc.cid2 as cid2
from
  comment_replyof_comment crc join
  tc on (crc.commentid_2=tc.cid)
)
select cid2, count(cid)
from tc
group by cid2
order by count(cid) desc limit 20;

```

```

WITH recursive tc(cid, cid2) as (

```

```
select commentid as cid, commentid_2 as cid2
from comment_replyof_comment
UNION ALL
select crc.commentid as cid, tc.cid2 as cid2
from
    comment_replyof_comment crc join
    tc on (crc.commentid_2=tc.cid)
)
select cid, count(cid2)
from tc
group by cid
order by count(cid2) desc limit 20;
```