ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені ІВАНА ФРАНКА Факультет прикладної математики та інформатики

Бази даних та інформаційні системи

ЛАБОРАТОРНА РОБОТА №7

Користувацькі функції на мові запитів SQL

Виконав:
Студент Процьків Назарій
Група <u>ПМі-21</u>
Оцінка
Перевірила:
лон. Малень Р.Б.

Тема: Користувацькі функції на мові запитів SQL.

Мета роботи: Ознайомлення з поняттям користувацьких функцій на мові запитів SQL, їх створенням та застосуванням.

Завдання (Варіант 17):

Розробити базу даних для сайту соціальної мережі. Соціальна мережа підтримує реєстрацію користувачів зі збереженням усіх їхніх деталей (ім'я, прізвище, дата народження, місце проживання, телефони, сайти/е-mail/skype іт.д.), місця і періоди перебування, місця і періоди навчання, роботи, служби, приєднані файли (зображення, фільми, аудіо), які можна пов'язувати із місцями з деталей. Крім того, кожен користувач має можливість розміщувати свої повідомлення на власній сторінці, отримувати на повідомлення «лайки» та коментарі, а також додавати інших користувачів в друзі, або в «чорний список». Додатково користувач повинен мати змогу шукати нових друзів за довільними критеріями.

Хід роботи

- 1. Опрацював теоретичний матеріал.
- 2. Відповідно до свого завдання написав 4 користувацькі функції на мові запитів SQL. Виконання функцій і результати проілюструвати скрінами, а також поясненнями роботи функції. Функції мають бути різних типів з базовими типами чи складними типами, які повертають стрічку таблиці чи всю таблицю (TABLE), і т.п.

Функція №1 (Повертає стрічку):

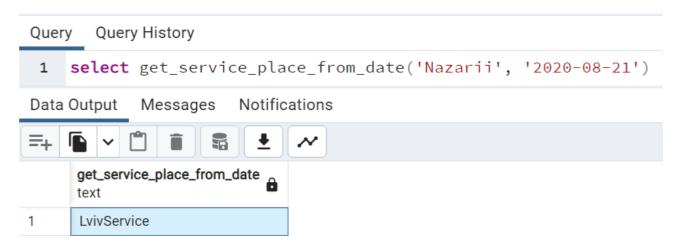
Функція приймає два параметри: ім'я користувача та дату. Повертає стрічку, в якій міститься інформація про те, де користувач з таким іменем служив станом на цю дату.

```
Query Query History
1 create function get_service_place_from_date(firstname_of_user text, date_to_check date)
2
       returns text
       language plpgsql
3
4 as
5 $$
6 declare
7
     to_return text;
8▼ begin
9
     select service into to_return
10
       from serviceplaces where serviceid =
11
          (select "servicePlace"
12
13
          where firstname = firstname_of_user) and (startdate < date_to_check and date_to_check < enddate);</pre>
14
       return to_return;
15 end;
16 $$;
```

Пояснення:

get_service_place_from_date — назва функції. Після неї в дужках написані параметри, які вона буде приймати. Слово returns вказує на те, що буде повертати функція. Слово declare створює змінну які присвоїться результат query. Між begin та end я написав query, яка буде дізнаватись де саме служив користувач з таким іменем на таку дату.

Результат роботи функції:



Користувач з іменем Nazarii станом на 21 серпня 2020 року служив у Львові.

Функція №2 (Повертає таблицю):

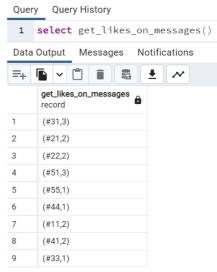
Функція не приймає параметрів. Повертає таблицю, в якій міститься інформація про те, скільки сумарно лайків отримало кожне повідомлення в соціальній мережі.

```
Query
       Query History
    create function get_likes_on_messages()
 1
      returns table (
 2
        msg id messageid check,
 3
        likes_count bigint
 4
 5
 6
      language plpgsql
 7
    as
    $$
 8
9 ▼ begin
    return query
10
        select message_id, count(*) as count_of_likes
11
        from likes
12
        group by message_id;
13
    end;
14
    $$;
15
```

Пояснення:

get_likes_on_messages — назва функції. Після неї в дужках нічого не написано, бо вона не приймає параметрів. Слово returns вказує на те, що буде повертати функція. В даному випадку це ϵ дві колонки. Між begin та end я написав query, яка буде дізнаватись сумарну кількість лайків на кожне повідомлення.

Результат роботи функції:



Повідомлення з іd #31 отримало 3 лайки. Повідомлення з іd #21 отримало 2 лайки. І так далі.

Функція №3 (Нічого не повертає):

Функція приймає два параметри. Оновлює ім'я користувача зі старого на нове.

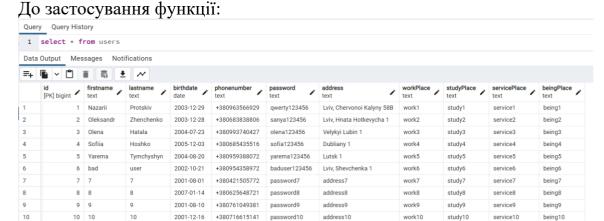
```
Query Query History

1    create function change_firstname(user_id_to_change user_id, new_firstname text)
2    returns void as
3    $$
4    update users
5    set firstname = new_firstname
6    where id = user_id_to_change;
7    $$
8    language sql;
```

Пояснення:

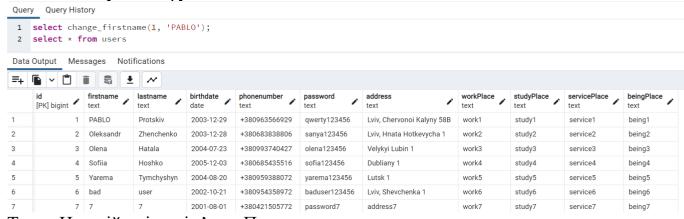
change_firstname – назва функції. Після неї в дужках user_id_to_change – параметр типу домену іd користувача, ім'я якого потрібно замінити. Другий параметр new_firstname – нове ім'я, яке потрібно записати замість старого. returns void вказує на те, що буде функція нічого повертати не буде.

Результат роботи функції:



Уявімо, що Назарій захотів змінити ім'я.

Після застосування функції:



Тепер Назарій змінив ім'я на Павло.

Функція №4 (Нічого не повертає):

Функція приймає два параметри. Видаляє користувача з чорного списку іншого користувача.

```
Query Query History

1    create function remove_from_blacklist(first_user_id user_id, second_user_id user_id)
2    returns void as
3    $$
4    delete from blacklist
5    where userid = first_user_id and enemyid = second_user_id
6    $$
7    language sql;
```

Пояснення:

remove_from_blacklist — назва функції. Після неї в дужках first_user_id — параметр типу домену іd користувача, з чорного списку якого потрібно забрати іншого користувача. Другий параметр second_user_id — параметр типу домену іd користувача, якого потрібно забрати з чорного списку першого користувача. returns void вказує на те, що буде функція нічого повертати не буде.

Результат роботи функції: До застосування функції:



Користувач з id 1 додав до чорного списку користувача з id 6. Користувач з id 2 додав до чорного списку користувача з id 6. І так далі.

Після застосування функції:

Наприклад користувач з id 1 захотів розблокувати користувача з id 6.



Висновок: ознайомився з поняттям користувацьких функцій на мові запитів SQL, їх створенням та застосуванням. Написав власні 4 функції, які можуть використовуватись в реальних базах даних соціальних мереж.