

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет електроніки та комп'ютерних технологій

ЗВІТ
про виконання ЛАБОРАТОРНОЇ РОБОТИ № 8
з дисципліни “Бази даних”

Виконав
студент 2 курсу
групи ФЕП-23
Чепара Станіслав Богданович

Перевірив доцент:
Анохін Володимир Євгенович

Львів 2025

Мета роботи: Навчитися створювати програмний клієнт для роботи з базою даних через прямі SQL-запити, використовуючи стандартні бібліотеки доступу до БД (Python DB API 2.0, драйвер PostgreSQL).

Хід роботи

код [Client.py](#) можна переглянути на github:

<https://github.com/Stanislawwx/Databases/tree/main/lab8>

Встановлення та запуск PostgreSQL на Fedora

```
stanislav@fedora:~$ systemctl status postgresql.service
• postgresql.service - PostgreSQL database server
  Loaded: loaded (/usr/lib/systemd/system/postgresql.service; enabled;
  Drop-In: /usr/lib/systemd/system/service.d
           └─10-timeout-abort.conf
  Active: active (running) since Fri 2025-11-21 15:56:47 EET; 19s ago
  Invocation: 18556dfb990742848f556b6981d01bf7
  Process: 35923 ExecStartPre=/usr/libexec/postgresql-check-db-dir postg
  Main PID: 35926 (postgres)
  Tasks: 7 (limit: 21192)
  Memory: 17.4M (peak: 18.5M)
  CPU: 90ms
  CGroup: /system.slice/postgresql.service
          └─35926 /usr/bin/postgres -D /var/lib/pgsql/data
            └─35927 "postgres: logger "
              └─35928 "postgres: checkpointer "
                └─35929 "postgres: background writer "
                  └─35931 "postgres: walwriter "
                    └─35932 "postgres: autovacuum launcher "
                      └─35933 "postgres: logical replication launcher "
```

лис 21 15:56:47 fedora systemd[1]: Starting postgresql.service - PostgreSQ
лис 21 15:56:47 fedora postgres[35926]: 2025-11-21 15:56:47.474 EET [35926
лис 21 15:56:47 fedora postgres[35926]: 2025-11-21 15:56:47.474 EET [35926
лис 21 15:56:47 fedora systemd[1]: Started postgresql.service - PostgreSQ
lines 1-24/24 (END)

Створення БД та користувача

```

stanislav@fedora:~$ sudo -u postgres psql
psql (16.9)
Type "help" for help.

postgres=# CREATE DATABASE lab8db;
CREATE USER labuser WITH PASSWORD '👀👀';
ALTER DATABASE lab8db OWNER TO labuser;
\q
CREATE DATABASE
CREATE ROLE
ALTER DATABASE
stanislav@fedora:~$ █

```

Створення таблиці clients і наповнення її тестовими даними

```

stanislav@fedora:~$ psql -h 127.0.0.1 -U labuser -d lab8db
Password for user labuser:
psql (16.9)
Type "help" for help.

lab8db=> CREATE TABLE IF NOT EXISTS clients (
    id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL,
    age INTEGER,
    created_at TIMESTAMP DEFAULT NOW()
);

INSERT INTO clients (name, email, age) VALUES
('Ivan Petrenko', 'ivan@example.com', 25),
('Olena Shevchenko', 'olena@example.com', 30),
('Taras Bondar', 'taras@example.com', 22);

SELECT * FROM clients;
CREATE TABLE
INSERT 0 3

```

id	name	email	age	created_at
1	Ivan Petrenko	ivan@example.com	25	2025-11-21 16:15:51.354524
2	Olena Shevchenko	olena@example.com	30	2025-11-21 16:15:51.354524
3	Taras Bondar	taras@example.com	22	2025-11-21 16:15:51.354524

```

(3 rows)
lab8db=> █

```

Створення Python-проекту, встановлення psycpg2 у віртуальному середовищі

```
stanislav@fedora:~/Desktop/Databases$ python3 -m venv venv
stanislav@fedora:~/Desktop/Databases$ source venv/bin/activate
(venv) stanislav@fedora:~/Desktop/Databases$ pip install psycpg2-binary
Collecting psycpg2-binary
  Downloading psycpg2_binary-2.9.11-cp313-cp313-manylinux2014_x86_64.manylinux_2_17_x86_64.whl.metadata (4.9 kB)
Downloading psycpg2_binary-2.9.11-cp313-cp313-manylinux2014_x86_64.manylinux_2_17_x86_64.whl (4.2 MB)
  ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 4.2/4.2 MB 848.3 kB/s eta 0:00:00
Installing collected packages: psycpg2-binary
Successfully installed psycpg2-binary-2.9.11
```

Результати запуску для кожного пункту меню.

```
(venv) stanislav@fedora:~/Desktop/Databases$ python client.py
```

Підключення до БД успішне.

=== Меню клієнта БД ===

1. Показати список таблиць
2. Показати структуру таблиці
3. Показати всі клієнти (SELECT)
4. Додати клієнта (INSERT)
5. Оновити клієнта (UPDATE)
6. Видалити клієнта (DELETE)
7. Виконати довільний SQL-запит
0. Вихід

Ваш вибір: 1

Таблиці в схемі public:

- clients

=== Меню клієнта БД ===

1. Показати список таблиць
2. Показати структуру таблиці
3. Показати всі клієнти (SELECT)
4. Додати клієнта (INSERT)
5. Оновити клієнта (UPDATE)
6. Видалити клієнта (DELETE)
7. Виконати довільний SQL-запит
0. Вихід

Ваш вибір: 2

Введіть назву таблиці: clients

Структура таблиці clients:

column_name	data_type	nullable	default
id	integer	NO	nextval('clients_id_seq'::regclass)
name	character varying	NO	None
email	character varying	NO	None
age	integer	YES	None
created_at	timestamp without time zone	YES	now()

=== Меню клієнта БД ===

1. Показати список таблиць
2. Показати структуру таблиці
3. Показати всі клієнти (SELECT)
4. Додати клієнта (INSERT)
5. Оновити клієнта (UPDATE)
6. Видалити клієнта (DELETE)
7. Виконати довільний SQL-запит
0. Вихід

Ваш вибір: 3

Клієнти:

id	name	email	age	created_at
1	Ivan Petrenko	ivan@example.com	25	2025-11-21 16:15:51.354524
2	Olena Shevchenko	olena@example.com	30	2025-11-21 16:15:51.354524
3	Taras Bondar	taras@example.com	22	2025-11-21 16:15:51.354524

=== Меню клієнта БД ===

- 2. Показати структуру таблиці
- 3. Показати всі клієнти (SELECT)
- 4. Додати клієнта (INSERT)
- 5. Оновити клієнта (UPDATE)
- 6. Видалити клієнта (DELETE)
- 7. Виконати довільний SQL-запит
- 0. Вихід

Ваш вибір: 4

Введіть ім'я: Ivan Franko

Введіть email: ivan.franko@example.com

Введіть вік (можна порожньо): 35

Клієнта додано з id = 4

=== Меню клієнта БД ===

- 1. Показати список таблиць
- 2. Показати структуру таблиці
- 3. Показати всі клієнти (SELECT)
- 4. Додати клієнта (INSERT)
- 5. Оновити клієнта (UPDATE)
- 6. Видалити клієнта (DELETE)
- 7. Виконати довільний SQL-запит
- 0. Вихід

Ваш вибір: 5

Введіть ID клієнта для оновлення: 4

Новий email (порожньо = не змінювати): franko@example.com

Новий вік (порожньо = не змінювати): 36

Оновлено рядків: 1

=== Меню клієнта БД ===

- 1. Показати список таблиць
- 2. Показати структуру таблиці
- 3. Показати всі клієнти (SELECT)
- 4. Додати клієнта (INSERT)
- 5. Оновити клієнта (UPDATE)
- 6. Видалити клієнта (DELETE)
- 7. Виконати довільний SQL-запит
- 0. Вихід

Ваш вибір: 7

Введіть SQL-запит: SELECT * FROM clients;

Отримано рядків: 4

```
(1, 'Ivan Petrenko', 'ivan@example.com', 25, datetime.datetime(2025, 11, 21, 16, 15, 51, 354524))
(2, 'Olena Shevchenko', 'olena@example.com', 30, datetime.datetime(2025, 11, 21, 16, 15, 51, 354524))
(3, 'Taras Bondar', 'taras@example.com', 22, datetime.datetime(2025, 11, 21, 16, 15, 51, 354524))
(4, 'Ivan Franko', 'franko@example.com', 36, datetime.datetime(2025, 11, 21, 16, 21, 33, 467293))
```

=== Меню клієнта БД ===

- 1. Показати список таблиць
- 2. Показати структуру таблиці
- 3. Показати всі клієнти (SELECT)
- 4. Додати клієнта (INSERT)
- 5. Оновити клієнта (UPDATE)
- 6. Видалити клієнта (DELETE)
- 7. Виконати довільний SQL-запит
- 0. Вихід

Ваш вибір: 6

Введіть ID клієнта для видалення: 4

Видалено рядків: 1

Висновки:Налаштовано локальну СУБД PostgreSQL у середовищі Fedora.

Реалізовано програмний клієнт на Python з використанням бібліотеки psycopg2, яка реалізує стандарт Python DB API.

Клієнт уміє підключатися до БД, показувати список таблиць, структуру таблиці, виконувати CRUD-операції й довільні SQL-запити з обробкою помилок.

Отримано практичні навички роботи з SQL і програмним доступом до БД.