

Практична робота 6

Тема: Просторові запити

Мета роботи: навчитися виконувати запити в PostgreSQL та QGIS

Короткі теоретичні відомості:

Мова запитів SQL або “Structured Query Language (Мова структурованих запитів)”, є засобом для створення запитів та оновлення даних в реляційних базах даних. Приклади функцій:

- SELECT – повертає рядки у відповідь на запит;
- INSERT – додає нові рядки до таблиці;
- UPDATE – змінює наявні рядки в таблиці;
- DELETE – видаляє рядки з таблиці.

Для того щоб задати запит пошуку, потрібно:

1. налаштувати з'єднання з DataBase_Surname;
2. відкрити редактор запитів, рис. 6.1.

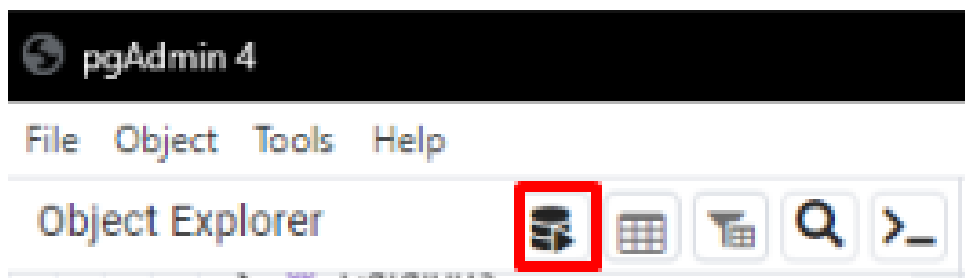


Рис. 6.1. Запуск редактора запитів

3. ввести запит в поле редактора запитів;
4. натиснути кнопку Execute/Refresh або на клавіатурі F5.

! Важливо! Назви полів та таблиць потрібно брати в подвійні лапки.

Вікно редактора має своє меню. Зокрема, можна переглянути історію здійснених запитів та вивести результати за потреби, рис. 18.2.

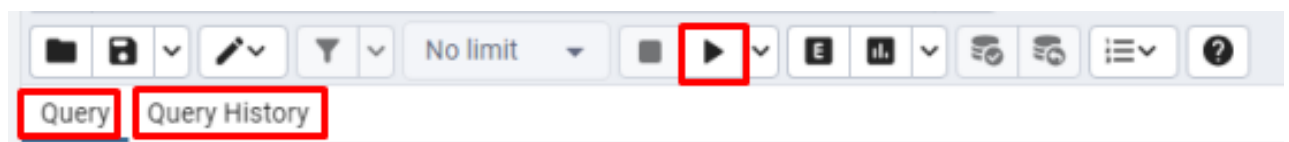


Рис. 6.2. Меню редактора запитів

Вираз **WHERE** використовується для **фільтрації записів**.

Він використовується для видалення лише тих записів (рядків), які відповідають певній умові.

З виразом WHERE можуть використовуватися наступні оператори:

Оператор	Опис
=	Дорівнює
>	Більше, ніж
<	Менше, ніж
>=	Більше, ніж або дорівнює
<=	Менше, ніж або дорівнює
<>	Не дорівнює. Примітка: У деяких версіях SQL цей оператор може бути записаний як !=
BETWEEN	Між певним діапазоном (включно)
LIKE	Пошук по шаблону
IN	Щоб указати кілька можливих значень для стовпця

Умову WHERE можна комбінувати з операторами **AND**, **OR** та **NOT**.

Оператори AND та OR використовуються для фільтрації записів на основі більш ніж однієї умови:

Оператор AND відображає запис, якщо всі умови, розділені AND є TRUE.

Оператор OR відображає запис, якщо будь-яка з умов, розділених OR є TRUE.

Оператор NOT відображає запис, якщо умова (умови) є NOT TRUE.

Ключове слово **ORDER BY** використовується для **сортування набору результатів** у порядку зростання або спадання.

ORDER BY за замовчанням сортує записи в порядку зростання. Щоб відсортувати записи в порядку спадання, використовуйте додаткове ключове слово DESC.

В PostgreSQL з метою полегшення написання запитів назви уже завантажених до бази даних класів (таблиць) можна відкоригувати. Для цього на потрібному класі (таблиці) клікаємо правою клавішею миші – властивості (Properties) – Змінити назву (Name) – Ok.

Шейпфайли містять геометрію, яка після експортування до PostgreSQL міститься в полі під назвою “geom”, рис. 6.3.

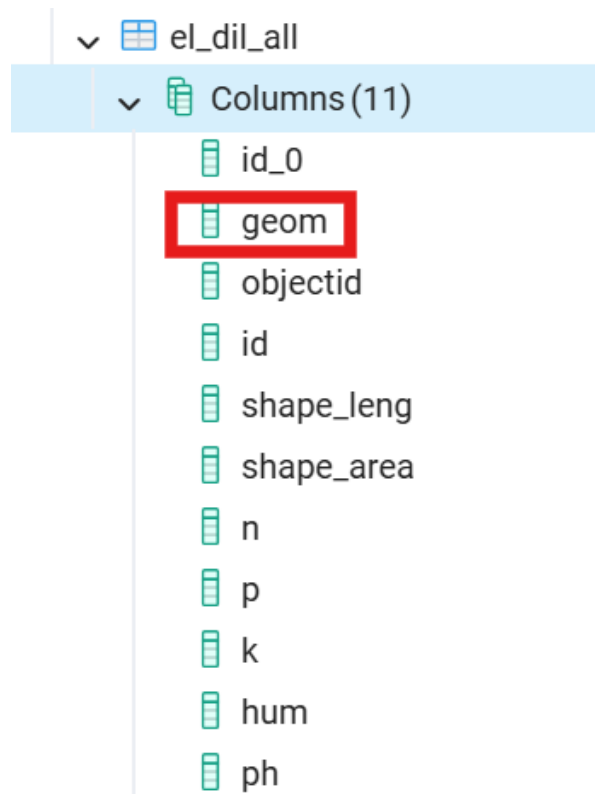


Рис. 6.3. Геометрія елементарних ділянок

Щоб перевести геометрію в площу потрібно застосувати оператор ST_area, а потім отриману площу в метрах квадратних розділити на 10000 щоб перевести її в га. Таким чином, дана функція набуде вигляду: ST_area(geom)/10000.

Як системний інтегратор усіх даних і програмних додатків та візуалізатор супутникових знімків і векторних шарів тестових ділянок у єдиному картографічному інтерфейсі використовується вільна геоінформаційна система **Quantum GIS (QGIS)**. Крім функцій системного інтегратора та візуалізатора, QGIS є потужним аналітичним та функціональним засобом обробки геоінформаційних даних.

QGIS використовується як інтегратор усіх даних і програмних додатків та візуалізатор супутникових знімків і векторних шарів тестових ділянок у картографічному інтерфейсі. QGIS забезпечує відображення інформації з БД із застосуванням технологій об'єктно-орієнтованого програмування та PostgreSQL.

В даній лабораторній роботі розглянуто функціональну взаємодію користувача з середовища QGIS з об'єктно-реляційною БД.

Методичні вказівки:

1. Зробити вибірку всіх елементарних ділянок (список ділянок та вміст гумусу). Функція пошуку SELECT завжди застосовується з виразом FROM (звідки, наприклад, з якої таблиці). Пошук здійснюється за полями класів (таблиць), а умова вибірки – за рядками.

1.1. В pgAdmin відкрити потрібний клас (таблицю) та подивитися назву поля, яке містить дані які потрібно вивести на екран (в даному випадку інформація ПІБ землекористувачів у таблиці Землекористувачі (Owners) містяться у полі класу (таблиці) під назвою “VlasnykNazva”;

1.2. Задати запит та **команду виконати запит**, дивіться рис. 6.4.

Де: "hum" – назва потрібних полів таблиці;

"el_dil_all" – назва таблиці, де містяться дані.

The screenshot shows the pgAdmin 4 interface with the following details:

- Top Bar:** Tabs for 'dependents', 'Processes', 'soils_db/postgres...', and 'soils_db_surname/postgres@PostgreSQL 18*'. The active tab is 'soils_db_surname/postgres@PostgreSQL 18*'. Below the tabs is a toolbar with icons for file operations, filters, and execution.
- Query Editor:** Contains the SQL query:

```
1 SELECT "hum"
2 FROM "el_dil_all"
```
- Data Output:** Shows the result of the query in a table. The table has one column named 'hum' of type 'double precision'. It displays 10 rows of data.
- Table Data:**

	hum double precision
1	5.09
2	5.09
3	5.09
4	5.09
5	5.27
6	5.09
7	3.22
8	4.61
9	4.61
10	3.67

At the bottom of the interface, it shows 'Total rows: 97', 'Query complete 00:00:00.198', and the cursor position 'Ln 2, Col 18'.

Рис. 6.4. Результат виконання запиту

Таким самим чином на екран можна вивести кілька полів таблиці, зазначивши у запиті назви полів через кому. Приклад на рис. 6.5.

2. Зробити вибірку всіх номерів елементарних ділянок Вміст гумусу на яких менший 3.5.

2.1. Дані про номери елементарних ділянок містяться в класі (таблиці) "el_dil_all", тому запит буде направлений саме до цієї таблиці. Згідно проведеного аналізу вмісту таблиці Елементарних ділянок видно, що номери номери містяться в полі (стовпчику) під назвою “id”;

2.2. В запиті присутня умова, яка подається через вираз **WHERE**

2.3. Задати запит, дивіться рис. 6.6.

Query	Query History	Scratch Pad X
1	SELECT "hum","id"	
2	FROM "el_dil_all"	
3		
Data Output	Messages	Notifications
Showing rows: 1 to 97 Page No: 1 of 1 SQL		
	hum double precision	id integer
1	5.09	63
2	5.09	65
3	5.09	64
4	5.09	66
5	5.27	62
6	5.09	64
7	3.22	35
8	4.61	34
9	4.61	33
10	2.67	26
Total rows: 97	Query complete 00:00:00.140	CRLF Ln 1, Col 17

Рис. 6.5. Результат виконання запиту

Query	Query History	S
1	SELECT "hum","id"	
2	FROM "el_dil_all"	
3	WHERE "hum"<3.5	
Data Output	Messages	Notifications
Showing rows: 1 to 9 Page No: 1 of 1 SQL		
	hum double precision	id integer
1	3.22	35
2	3.32	10
3	3.32	11
4	3.33	4
5	3.33	6
6	3.33	5
7	3.3	3
8	3.3	2
9	3.3	1

Рис. 6.6. Результат виконання запиту

- Відфільтрувати всі ділянки, площа яких перевищує 15 га, рис. 6.6.

Де: "ST_area(geom)/10000" – переведення геометрії земельної ділянки в площу в га;
 id AS "№ ділянки" – присвоєння існуючому полю імені
 as "Area" – присвоєння новому полю імені;
 "el_dil_all" – клас (таблиця) елементарних ділянок
 WHERE ST_area(geom)/10000 > 15 – умова вибірки тільки ті поля, площа яких перевищує 15 га.

Query		Query History	Scratch Pad
<pre> 1 SELECT 2 id AS "№ ділянки", 3 ST_Area(geom) / 10000 AS "Area" 4 FROM public.el_dil_all 5 WHERE ST_Area(geom) / 10000 > 15; </pre>			
Data Output			
Messages			
Notifications			
Showing rows: 1 to 6 Page No: 1 of 1			
	№ ділянки integer	Area double precision	
1	20	17.25413639459628	
2	7	15.56187793595964	
3	59	16.97425189598021	
4	52	16.177800880885236	
5	1	16.684557736104434	
6	73	15.537417307462704	
Total rows: 6		Query complete 00:00:00.101	CRLF Ln 5, Col 33

Рис. 6.7. Результат виконання запиту

5. Необхідно об'єднати векторний шар елементарних ділянок із таблицею лабораторних агрохімічних аналізів ґрунту та вивести на екран повну атрибутивну інформацію за кожною ділянкою.

Для вирішення цієї задачі було виконано атрибутивне об'єднання двох джерел даних за допомогою SQL-запиту:

Джерела даних: Таблиця просторових даних el_dil_all (координати та межі ділянок) та атрибутивна таблиця agrohim (результати агрохімічного аналізу: рН, вміст гумусу, рухомих форм азоту, фосфору, калію тощо).

Ключ зв'язку: Об'єднання виконано через спільне ідентифікаційне поле: номер елементарної ділянки id з таблиці el_dil_all відповідає полю field_1 у таблиці agrohim (e.id = a.field_1).

Результат: Завдяки виразу SELECT e.*, a.* сформовано єдиний комбінований набір даних, де для кожного геопросторового полігону ділянки одночасно відображаються всі її фізичні характеристики та повні лабораторні показники якісного стану ґрунту

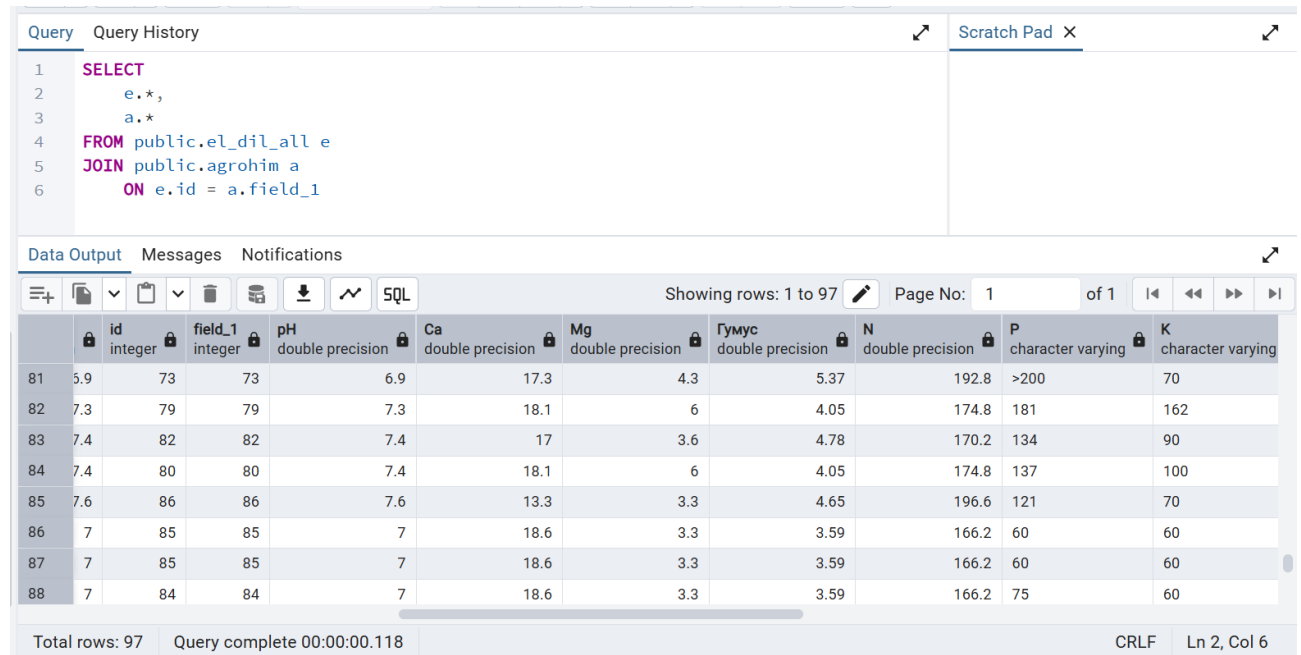
SELECT e.*, a.* - знак зірочки * означає вибір усіх без винятку колонок з обох таблиць: e.* обирає всі колонки з el_dil_all (включаючи геометрію geom,

площу, довжину тощо), а а.* додає всі агрохімічні показники з agrohim (pH, Гумус, N, P, K, Ca, Mg та інші).

FROM public.el_dil_all e – вказує основну таблицю земельних ділянок і задає їй короткий псевдонім e.

JOIN public.agrohim a – приєднує таблицю агрохімічних аналізів із псевдонімом a.

ON e.id = a.field_1 – умова зв'язку: запис із agrohim приєднується до ділянки з el_dil_all, у якої значення в полі id збігається зі значенням у полі field_1.



The screenshot shows a database query interface. The top section displays the SQL query:

```
1 SELECT
2   e.*,
3   a.*
4 FROM public.el_dil_all e
5 JOIN public.agrohim a
6   ON e.id = a.field_1
```

Below the query, the 'Data Output' tab shows the results of the query. The table has 12 columns: id, field_1, pH, Ca, Mg, Гумус, N, P, K. The results are displayed in a table with 97 rows (rows 81 to 88 are visible).

	id integer	field_1 integer	pH double precision	Ca double precision	Mg double precision	Гумус double precision	N double precision	P character varying	K character varying
81	5.9	73	73	6.9	17.3	4.3	5.37	192.8	70
82	7.3	79	79	7.3	18.1	6	4.05	174.8	162
83	7.4	82	82	7.4	17	3.6	4.78	170.2	90
84	7.4	80	80	7.4	18.1	6	4.05	174.8	100
85	7.6	86	86	7.6	13.3	3.3	4.65	196.6	70
86	7	85	85	7	18.6	3.3	3.59	166.2	60
87	7	85	85	7	18.6	3.3	3.59	166.2	60
88	7	84	84	7	18.6	3.3	3.59	166.2	60

At the bottom, the status bar shows: Total rows: 97, Query complete 00:00:00.118, CRLF, Ln 2, Col 6.

Рис. 6.8. Результат виконання запиту

6. Складний просторово-атрибутивний аналіз та формування комплексного звіту за земельними ділянками.

Сформувати підсумкову інформаційну вибірку (звіт) для елементарних земельних ділянок, яка поєднає в собі дані з трьох різних картографічних шарів та однієї практичної атрибутивної таблиці.

Необхідно вивести:

Номер елементарної ділянки (id з el_dil_all).

Номер сівозмінного поля, в межах якого розташована ділянка (поле номер з Field_Surname_2).

Українську назву типу ґрунту (name_ukr) та його бонітет (bonitet) з ґрунтового шару Soil_Surname.

Основні агрохімічні показники — вміст гумусу (Гумус) та кислотність (pH) з практичної таблиці agrohim.

Просторові контури (геометрію geom) для подальшої візуалізації на карті.

Додаткова умова фільтрації: До звіту мають увійти лише ті ділянки, у яких вміст гумусу строго перевищує **3.0%** (а."Гумус" > 3.0).

Аналіз та алгоритм розв'язання:

Для формування звіту виконується комплексний SQL-запит, який комбінує атрибутивні та просторові зв'язки (Рис. 6.9):

✓ Атрибутивне з'єднання (Primary/Foreign Key): Таблиця межових контурів ділянок `el_dil_all` об'єднується з лабораторною таблицею `agrohim` за ключовими полями `e.id = a.field_1`.

✓ Перше просторове з'єднання (Spatial Join): Шар ґрунтів `Soil_Surname` підключається за допомогою функції PostGIS `ST_Intersects(e.geom, s.geom)`, яка географічно зіставляє полігон ділянки з відповідним контуром ґрунтового ареалу.

✓ Друге просторове з'єднання (Spatial Join): Шар полів `Field_Surname_2` підключається так само просторово за допомогою `ST_Intersects(e.geom, f.geom)`, що дозволяє автоматично визначити, до якого саме поля належить кожна ділянка.

✓ Фільтрація та реорганізація колонок: За допомогою виразу `WHERE a."Гумус" > 3.0` відсікаються ділянки з низьким вмістом гумусу, а блок `SELECT` впорядковує вихідні поля й надає їм зрозумілі назви-псевдоніми українською мовою (наприклад, `f."номер" AS "Номер поля"`).

Query	Query History
1	SELECT
2	<code>e.id AS "№ ділянки",</code>
3	<code>f."номер" AS "Номер поля",</code>
4	<code>s.name_ukr AS "Назва ґрунту",</code>
5	<code>s.bonitet AS "Бонітет",</code>
6	<code>a."Гумус",</code>
7	<code>a."pH",</code>
8	<code>e.geom</code>
9	FROM <code>public.el_dil_all e</code>
10	JOIN <code>public.agrohim a</code>
11	<code>ON e.id = a.field_1</code>
12	JOIN <code>public."Soil_Surname" s</code>
13	<code>ON ST_Intersects(e.geom, s.geom)</code>
14	JOIN <code>public."Field_Surname_2" f</code>
15	<code>ON ST_Intersects(e.geom, f.geom)</code>
16	WHERE <code>a."Гумус" > 3.0;</code>

Рис. 6.9. Запит

Data Output

Messages

Notifications

SQL

Showing rows: 1 to 320

Page No: 1 of 1

	№ ділянки integer	Номер поля character varying (80)	Назва ґрунту character varying (7)	Бонітет double precision	Гумус double precision	pH double precision	geom geometry
29	3	1	53 г	83	3.3	6.9	01060000207C7F000C
30	3	1	53 г	83	3.3	6.9	01060000207C7F000C
31	2	1	53 г	83	3.3	6.6	01060000207C7F000C
32	1	1	53 г	83	3.3	6.4	01060000207C7F000C
33	63	1	133 г	95	5.09	7.6	01060000207C7F000C
34	63	1	133 г	95	5.09	7.6	01060000207C7F000C
35	63	4	133 г	95	5.09	7.6	01060000207C7F000C
36	65	1	133 г	95	5.09	7.5	01060000207C7F000C
37	64	1	133 г	95	5.09	7.5	01060000207C7F000C
38	64	1	133 г	95	5.09	7.5	01060000207C7F000C
39	66	1	133 г	95	5.09	7.4	01060000207C7F000C
40	62	1	133 г	95	5.27	7.3	01060000207C7F000C
41	62	1	133 г	95	5.27	7.3	01060000207C7F000C

Total rows: 320

Query complete 00:00:00.163

CRLF

Рис. 6.10. Результат виконання запиту

7. Відкрити QGIS.
8. Встановити з'єднання з потрібною базою даних, ввівши необхідні дані (ім'я користувача Username, пароль password).
9. Відкрити вікно запитів. Для цього клікнути правою клавішею миші на потрібній базі даних в browser (зліва на бічній панелі в QGIS) та обрати **Execut SQL**.

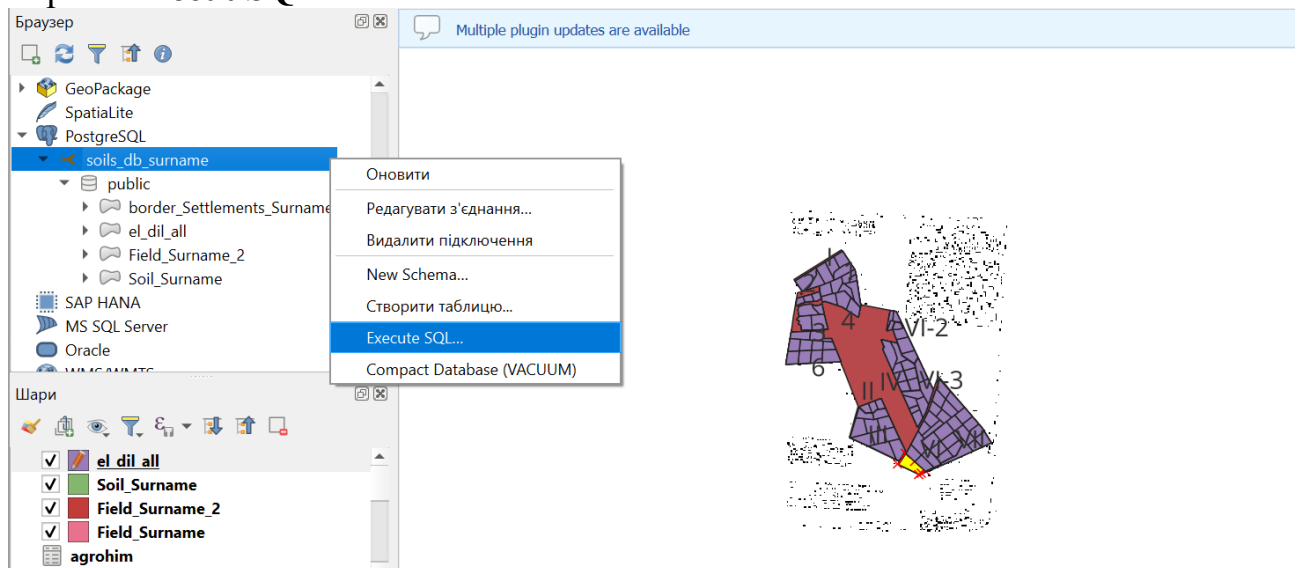


Рис. 6.11. Відкривання вікна запитів в QGIS

10. Виконати запит та вивести на екран всі елементарні ділянки, які мають вміст гумусу менше 3,5%. Для цього потрібно виконати запит, дивіться рис. 6.12.

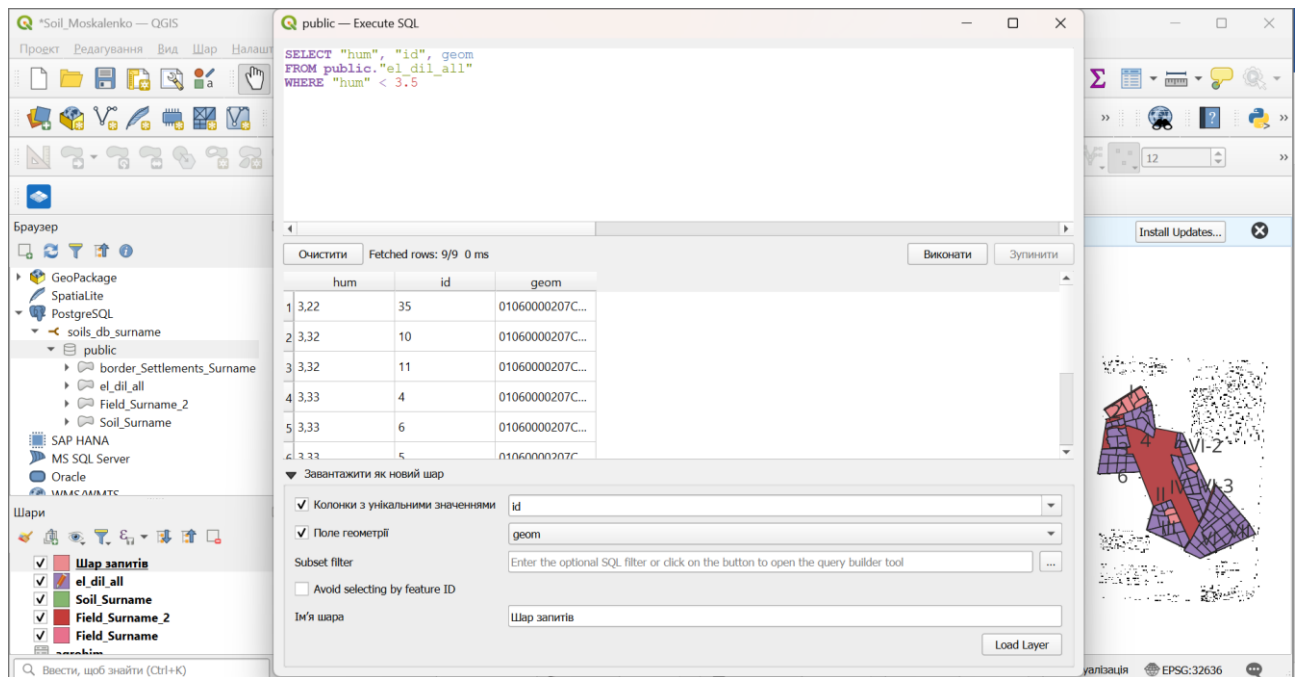


Рис. 6.12. Результат виконання запиту

11. Створення віртуального просторового шару (VIEW) для відбору ділянок за ґрунтовими та агрохімічними параметрами

Розробити та зберегти в базі даних PostgreSQL reusable-представлення (VIEW) з назвою v_soil_53g_hum35, яке поєднує межі елементарних ділянок із картою ґрунтів для вирішення такої аналітичної задачі:

Просторовий та атрибутивний фільтр:

Відібрати елементарні ділянки з таблиці el_dil_all, у яких рівень вмісту гумусу (hum) строго менший за 3.5% ($e.hum < 3.5$).

Враховувати лише ті ділянки, які просторово перетинаються (функція PostgreSQL ST_Intersects) із контурами ґрунтового шару Soil_Surname, що відповідають типу ґрунту "53ґ".

Опрацювання варіативності введення (Синтаксичні адаптації):

В умові пошуку типу ґрунту враховано можливі описки та відмінності у форматах запису в атрибутивних полях (name_ukr, name, namesoil), такі як наявність або відсутність пробілу ('53ґ', '53 ґ') чи помилковий синтаксис ('5 ґ').

Формування підсумкового набору даних:

Включити до створеного представлення унікальний системний ідентифікатор id_0 (із псевдонімом ogc_fid для коректного визначення первинного ключа в QGIS), номер ділянки id, показник гумусу hum, назву ґрунту name_ukr та просторову геометрію geom



Рис. 6.13. Запит

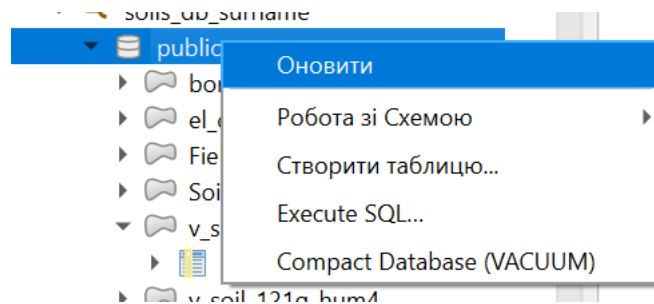


Рис. 6.14. Оновлення бази

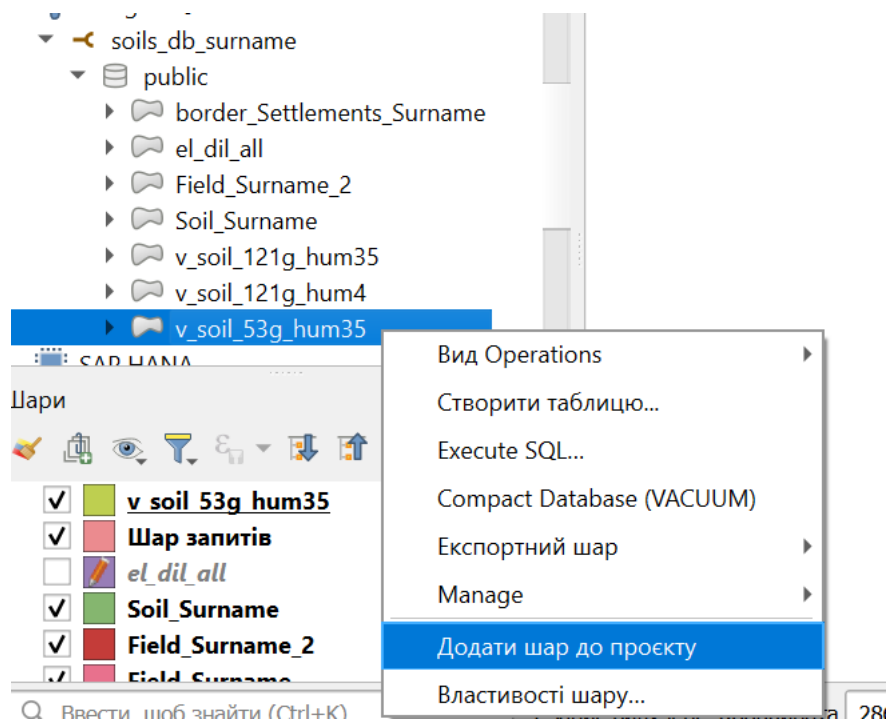


Рис. 6.15. Додавання шару до проекту



Рис. 6.16. Результат виконання запиту

Хід роботи

1. Ознайомтесь з методичними вказівками до практичної роботи.
2. Створіть шаблон звіту у MS Word під назвою Work6_Surname.docx (усі кроки фіксуйте чіткими та читабельними скріншот-підтвердженнями).
3. Виконайте атрибутивні SQL-запити до імпортованих у PostgreSQL таблиць:

- Фільтрація за атрибутом: Вибірка номерів ділянок, у яких вміст гумусу менший за 3.5% (WHERE hum < 3.5).
 - Розрахунок геометрії: Обчислення площі ділянок у гектарах (ST_Area(geom)/10000) та відбір об'єктів площею понад 15 га
 - Об'єднати шар el_dil_all з таблицею агрохімічних аналізів agrohim за спільним ключем e.id = a.field_1 для отримання повних якісних показників ґрунту.
 - Сформувані підсумковий SQL-запит із використанням функції PostGIS ST_Intersects для перетину ділянок із шаром ґрунтів Soil_Surname та полів Field_Surname_2, встановивши додатковий фільтр гумусу понад 3.0
4. Запустіть QGIS та налаштуйте підключення до бази даних PostgreSQL.
 5. Виконайте просторові запити в QGIS:
 - Виконати просторовий запит для відображення на карті ділянок з вмістом гумусу менше 3.5%.
 6. Створити віртуальне представлення (VIEW):
 - Розробити та зберегти в базі даних представлення v_soil_53g_hum35 для вибірки ділянок із вмістом гумусу менше 3.5% та типом ґрунту «53 г».
 - Врахувати варіативність написання назви ґрунту та забезпечити коректне призначення системного ключа id_0 AS ogc_fid для стабільної візуалізації шару в QGIS.
 7. Збережіть проєкт QGIS.
 8. Оформіть звіти (з додаванням усіх скріншотів виконання запитів) та завантажте їх на платформу eLearn.
 9. Підготуйте усні відповіді на контрольні запитання.

Контрольні запитання:

1. Назвіть оператор умови.
2. Який оператор відповідає за вибірку даних?
3. Опишіть запит здійснення фільтру даних з бази даних?
4. Який оператор дає можливість додати дані в таблицю?
5. Який оператор виконує дії з видалення даних з бази даних?
6. За допомогою якого оператора можна замінити дані в базі даних?
7. Опишіть дії як налаштувати з'єднання QGIS з PostgreSQL.
8. Опишіть хід дій як виконати просторовий запит у QGIS.
9. Чим відрізняються за змістом запити в QGIS і PostgreSQL?
10. Чим відрізняється результат виконання запиту в QGIS і PostgreSQL?