

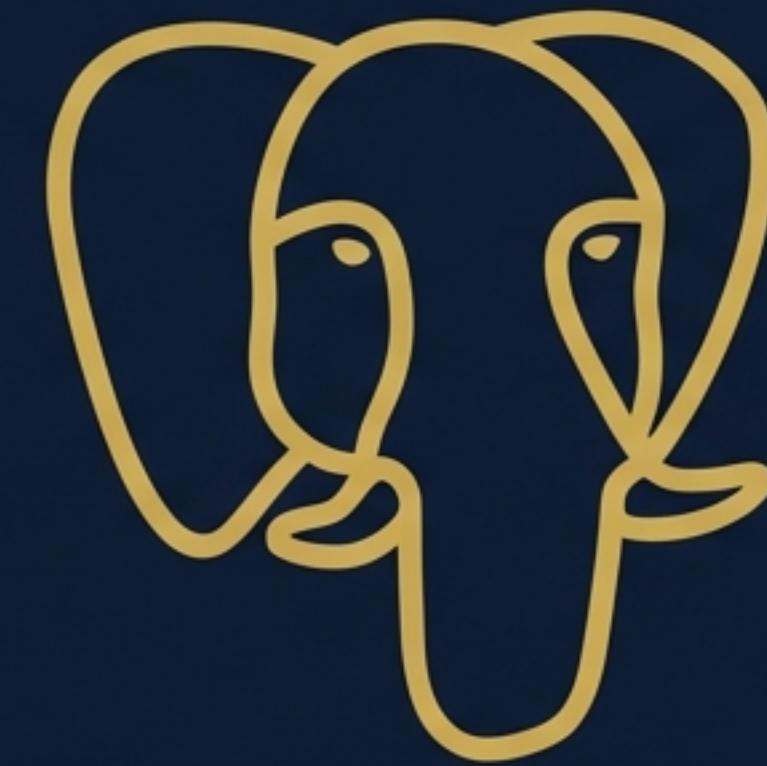
VS CODE ДЛЯ ПРОЄКТУВАННЯ БД POSTGRESQL

ВІД ХАОСУ ДО ІНЖЕНЕРНОГО ПІДХОДУ В АНАЛІТИЦІ



Visual Studio Code
(/vɪʒuəl 'stu:di.oʊ kəʊd/)

Єдине середовище для написання коду
та управління даними.

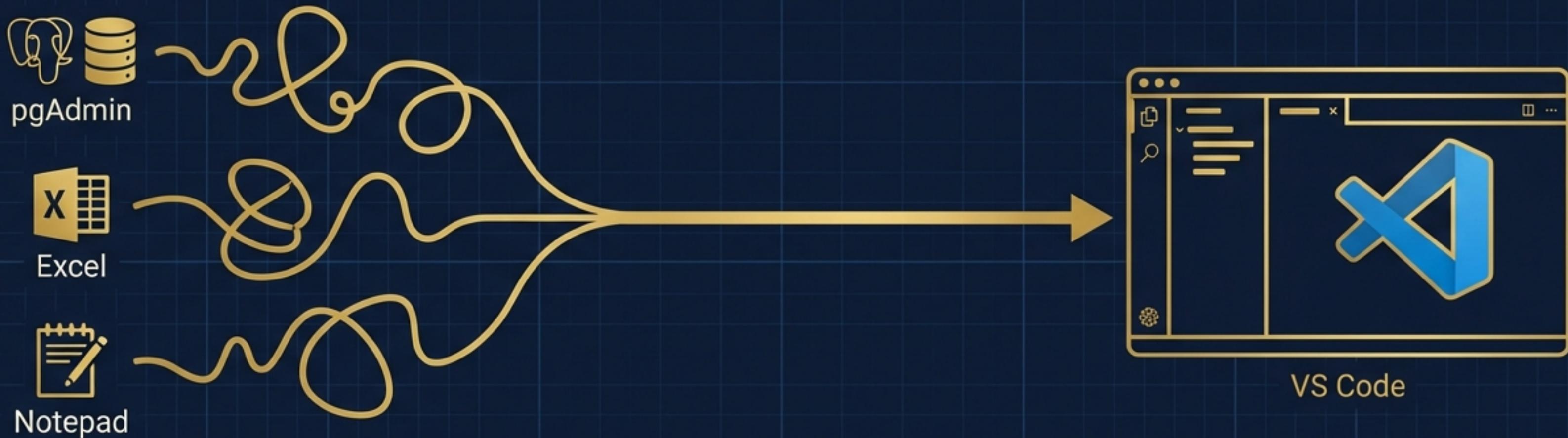


PostgreSQL
(/'poʊstgrɛs,kju:'ɛl/)

Найпотужніша реляційна система баз
даних з відкритим вихідним кодом.

НАВІЩО VS CODE ДЛЯ БД АНАЛІТИКУ?

ЄДИНИЙ РОБОЧИЙ ЦЕНТР (SINGLE PANE OF GLASS)



Централізація

Схеми, SQL-запити, документація та візуалізація — все в одному вікні. Більше немає перемикання між вікнами.

Повторюваність

Підхід «Everything is a Code». Ваша база даних стає проєктом з історією змін ([Git](#)), а не чорною скринькою.

Ефективність

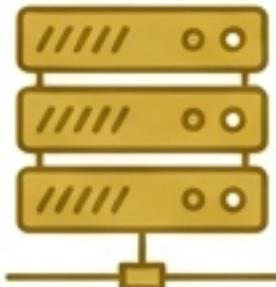
[IntelliSense](#) та автодоповнення прискорює написання запитів та зменшує кількість синтаксичних помилок.

КАРТИНА ПРОЦЕСУ: ВІД ЗАДАЧІ ДО ТАБЛИЦЬ



Інструмент підтримує повний цикл: від концептуального моделювання (ER-діаграми) до фізичного виконання запитів та налагодження.

МІНІМАЛЬНИЙ СТЕК ІНСТРУМЕНТІВ



PostgreSQL Server

/'poustgres,kju: 'el 's3:rvər/

Локально або віддалений сервер
(наприклад, Azure/Neon).



vs Code

Основний редактор коду.



Розширення (Extensions)

- PostgreSQL від Microsoft (або SQLTools)
- PostgreSQL extension

Для підключення та IntelliSense.

РЕКОМЕНДОВАНО



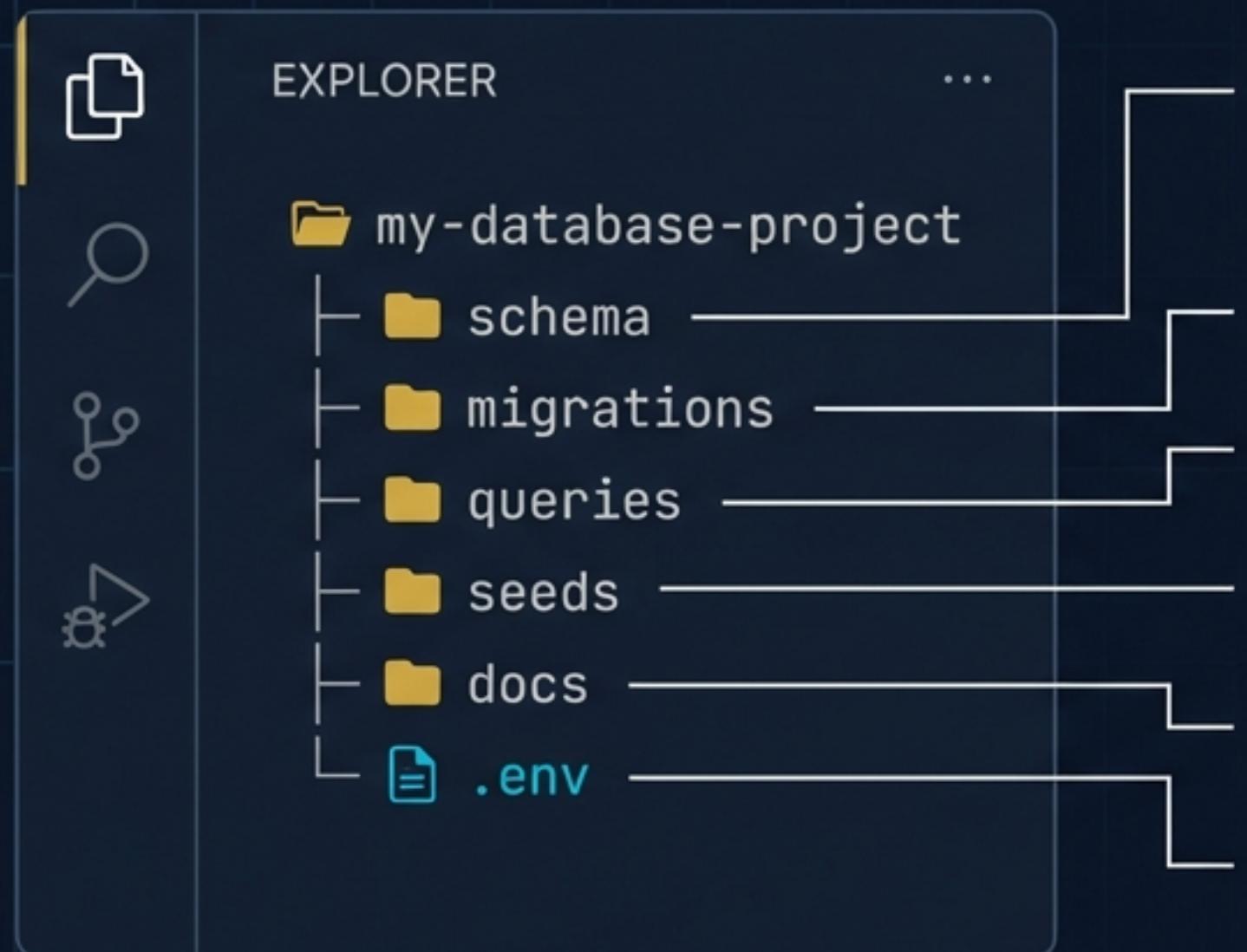
Docker (Опційно)

/'da:kər/

Для миттєвого розгортання локальної версії БД без «сміття» в системі.

СТРУКТУРА ПРОЄКТУ В VS CODE

ОРГАНІЗАЦІЯ «ПАПКИ БД»



schema: DDL-скрипти (створення таблиць, визначення ключів)

migrations: Історія змін версій схеми (up/down scripts)

queries: Каталог типових запитів для аналітики

seeds: Тестові дані (CSV/JSON) для перевірки гіпотез

docs: Опис моделі та словник даних (Data Dictionary)

.env: Змінні оточення (паролі)

Чітка структура дозволяє новому аналітику зрозуміти проект за 5 хвилин, а не 5 днів.

ПІДКЛЮЧЕННЯ ДО POSTGRESQL 3 VS CODE

Connection Profile

Host: localhost

Port: 5432

User: postgres

Password: *****

Database: analytics_db

[Test Connection] [Save Profile]

Використовуйте змінні
оточення (.env)!
Ніколи не зберігайте
паролі у відкритому
вигляді.

ЩО ТАКЕ SQL У ЦЬОМУ КУРСІ?

МОВА ОПИСУ СТРУКТУРИ, А НЕ ПРОСТО ПРОГРАМУВАННЯ

DDL

Data Definition Language

«Я створюю світ»



`CREATE` , `ALTER` , `DROP`

Архітектура, схеми, правила.

DML

Data Manipulation Language

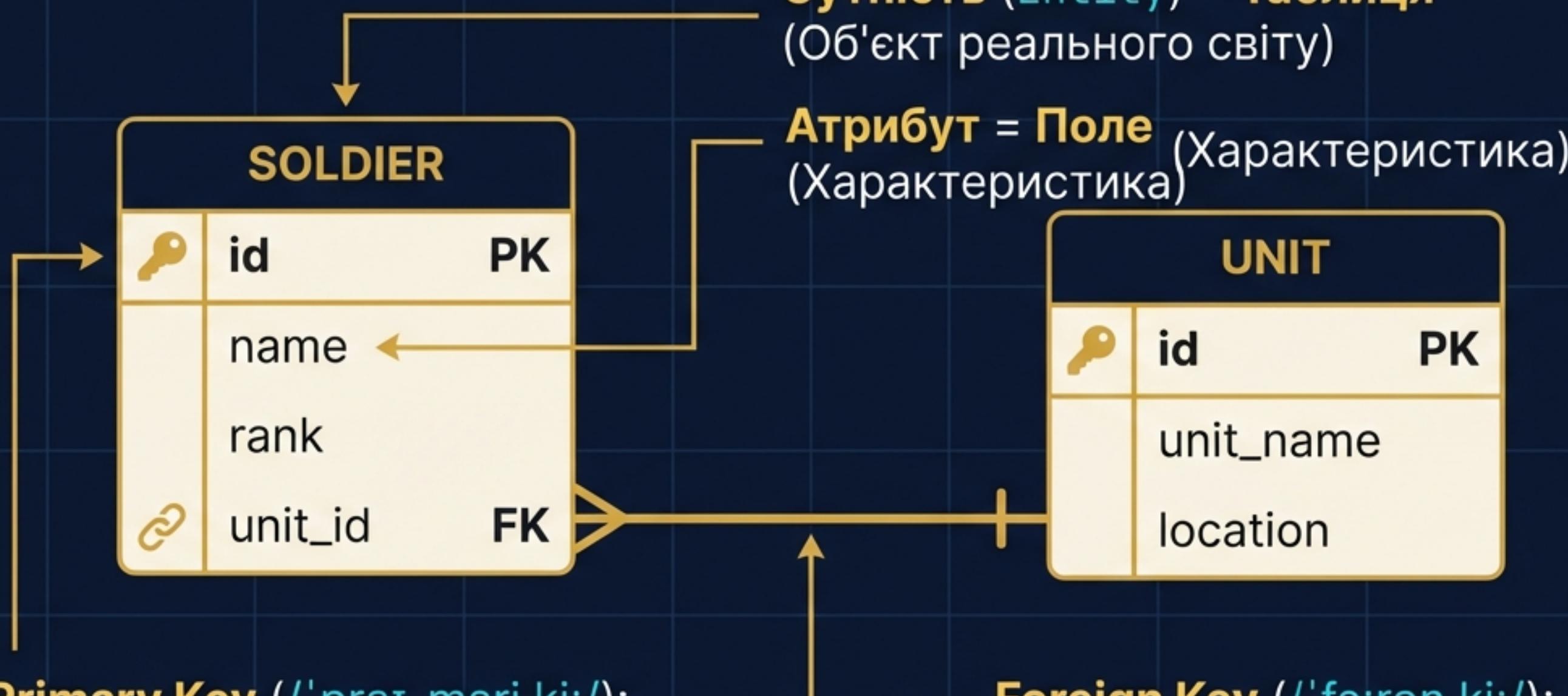
«Я живу в цьому світі»



`SELECT` , `INSERT` , `UPDATE`

Аналітика, звіти, інсайти.

ПРОЕКТУВАННЯ СХЕМИ ЧЕРЕЗ DDL



Сутність (Entity) = Таблиця

(Об'єкт реального світу)

Атрибут = Поле

(Характеристика)

ТИПИ ДАНИХ POSTGRESQL

Чому «Все текстом» — це погана стратегія

Тип	Використання	Перевага
	<code>integer / bigint</code>	Лічильники, ID, кількість. Швидка арифметика.
	<code>text / varchar</code>	Імена, описи, коди. Гнучкість.
	<code>boolean</code>	Пропорці (True/False). Логіка (Активний/Неактивний).
	<code>date / timestamp</code>	Події, час. Обчислення інтервалів (вік, стаж) засобами БД.

Правильний тип даних = Вбудована перевірка
якості (Data Quality) + Швидкість.

ОБМЕЖЕННЯ І ЦІЛІСНІСТЬ ДАНИХ



Мета: Щоб БД фізично не дозволяла зіпсувати дані.

ІНДЕКСИ ТА ПРОДУКТИВНІСТЬ «НА ПАЛЬЦЯХ»



Алфавітний покажчик книги
= **Index** (/Indeks/)

Коли потрібен:

Для полів, де часто йде фільтрація (**WHERE**), пошук або з'єднання (**JOIN**).

↑ + ПЛЮС

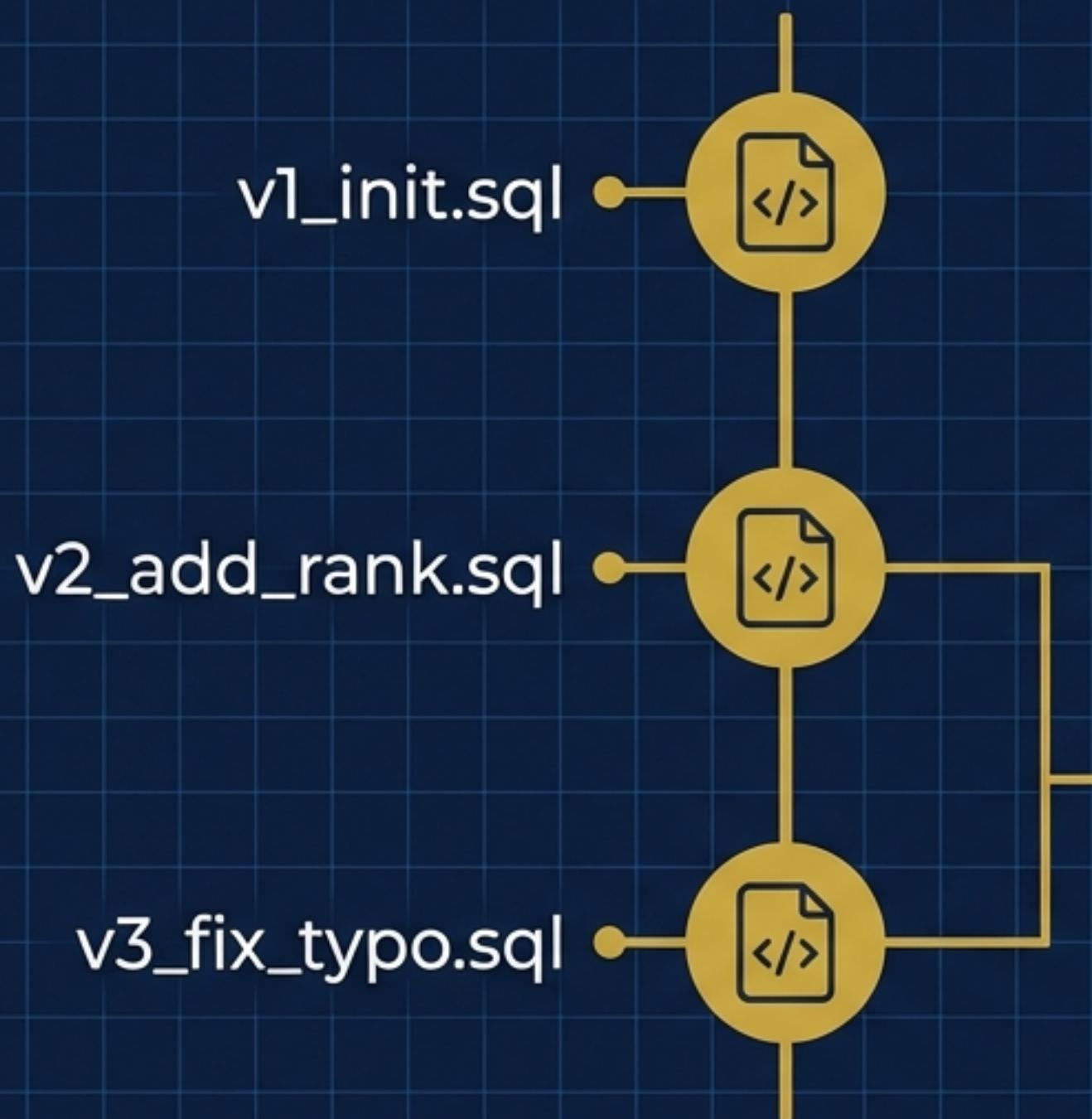
Миттєве читання
(**SELECT**).

↓ - МІНУС

Уповільнення
запису (**INSERT**),
бо індекс треба
оновлювати.

**Не індексуйте все підряд.
Аналізуйте запити.**

МІГРАЦІЙ: КЕРУВАННЯ ЗМІНАМИ СХЕМИ



Migration (/maɪ'greɪʃən/): Скрипт контрольованої зміни.

Замість ручного редагування — код.

Up Script
(Створює / Додає)

Down Script
(Скасовує / Відкочує)

Історія змін (Git) показує: ХТО, КОЛИ і НАВІЩО змінив структуру БД.

РОБОЧІ ЗАПИТИ АНАЛІТИКА В VS CODE

The screenshot shows a VS Code interface with a dark theme. A sidebar on the left contains icons for file operations, search, and other tools. The main area is titled 'Result Viewer' and displays a SQL query and its results. The query is:

```
SELECT unit, COUNT(*)  
FROM soldiers  
GROUP BY unit;
```

The results are presented in a table:

unit	COUNT(*)
A	245
B	180
C	310
D	122

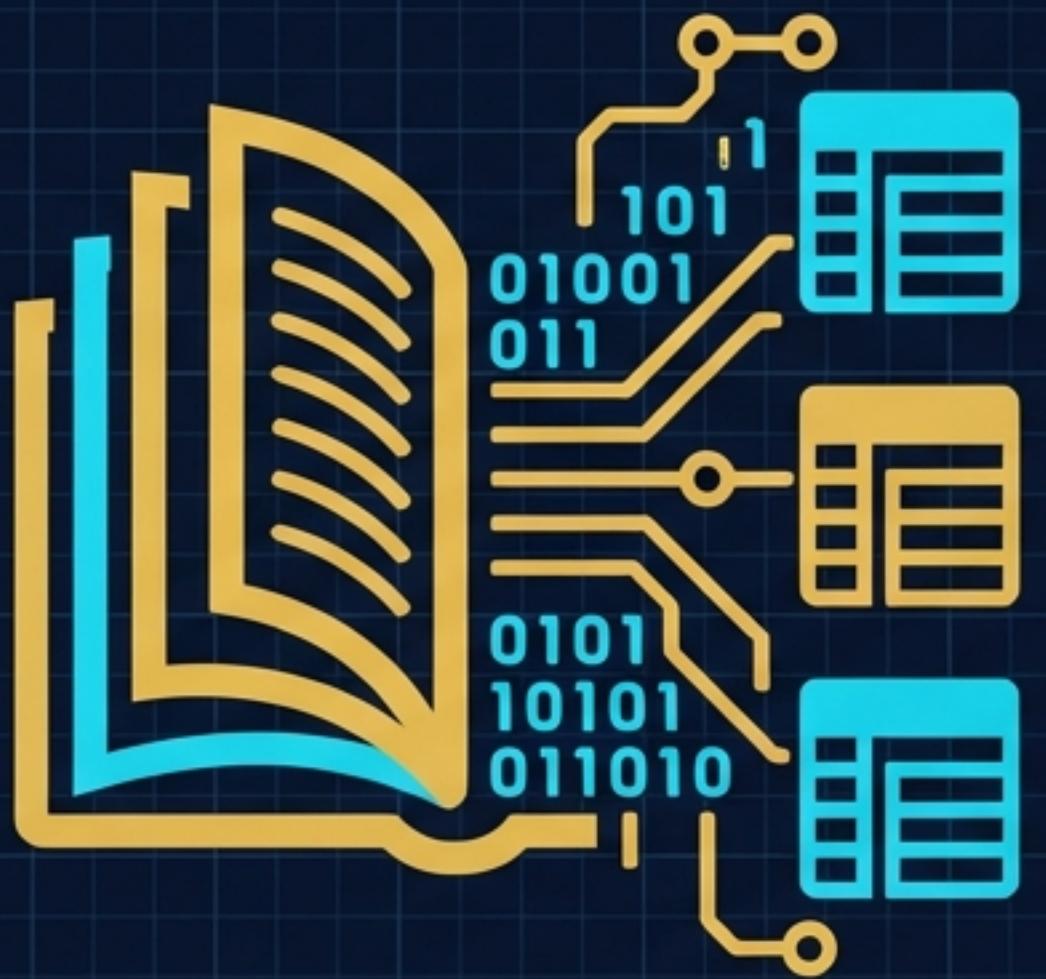
- Шаблони (збережені в `/queries`):
 - Вибірка за певний період.
 - Підрахунок підсумків (`COUNT`, `SUM`).
 - Аналіз за підрозділами або категоріями.

PRO TIP:

Зберігайте вдалі запити як сніпети (snippets) для швидкого доступу.

ДОКУМЕНТАЦІЯ МОДЕЛІ ДАНИХ

DATA DICTIONARY ('deɪtə 'dɪkʃənəri/)



A_

- **Правила найменування:** Чіткий стиль (напр. `snake_case`).
- **Описи (Descriptions):** Що означає поле `status_id`? Допустимі значення?
- **Джерела:** Звідки приходять дані.

?

⌚

Інструменти:

- **dbdiagram.io** (Візуалізація)
- dbt Power User (AI-генерація описів)

Якщо модель не описана — вона не існує.

ПІДСУМОК: РОБОЧИЙ ЦИКЛ У VS CODE



ПОРЯДОК У КОДІ — ДОВІРА ДО ДАНИХ.