SQL essential: PostgreSQL + Python

---

программа навчання: \*\*SQL essential\*\*

## **засоби навчання:** Python; встановлений Docker; інтегроване середовище розробки (PyCharm або Microsoft Visual Studio + Python Tools for Visual Studio + можливість використання Юпітер Jupyter Notebook)

## **Тривалість курсу** - 10 годин - 5 лекцій

## **Передумови:** знання основ мови програмування Python (в межаж програм "Python Starter" + "Python Essential" + "Python Advanced 2022" (бажано перед початком вивчення курсу повторення лекції 4 курсу "Python Advanced 2022" - "SQLite: синтаксис і запити")), поверхневе розуміння Docker (в межах курсу "Python&Docker").

---

## **Огляд, мета та призначення курсу**

Вивчити основи побудови реляційних баз даних та SQL.

Ознайомиться з базовим використанням СУБД PostgreSQL.

## **Склад курсу:**

**1) Лекція 1: Створюємо інфраструктуру.**

    1) Введення в реляційні бази даних.

        - моделі даних - ієрархічна, мережева, реляційна

        - що таке SQL, складові частини мови

        - огляд існуючих SQL СУБД

    2) Інфраструктура і розгортання БД

        - пишемо docker-compose.yaml, нюанси і особливості (відмінності від того, що робили в курсі Docker&Python)

        - pgAdmin - інструмент для зручної роботи з PostgreSQL. Розгортаємо в контейнері поряд.

        - розгортання БД Nordwind

        - структура БД Nordwind

        - розгортання інфраструктури і БД

        - знайомство з клієнтами і підключення

            - psql

            - pgAdmin

            - DBeawer

        - підключення до БД з python-коду

            - встановлення модулю psycopg2

            - створення з'єднання

            - функція для виконання SQL-запитів

            - aeyrwsz lkz dbrjyfyyz SQL-запитів і отримання результатів

**2) Лекція 2: Основні типи даних PostgreSQL і прості запити.**

    1) Основні типи даних PostgreSQL:

        - числові

        - символьні

        - типи дати\часу\інтервалу

        - логічні типи

    2) Прості запити:

        - SELECT

            - повна виборка

            - виборка конкретного набора атрибутів

            - прості математичні операції

        - DISTINCT - унікальність

        - COUNT - підрахунок кількості отриманих у виборці рядків

        - WHERE - фільтруємо рядкі по умові

        - AND, OR - комбінація умов

        - BETWEEN - "синтаксичний цукор" - ще одна можливість формувати умови для фільтації рядків

        - IN, NOT IN - оператор входження

        - ORDER BY - як впорядкувати результати

        - MIN, MAX, AVG (агрегаційні вирази)

        - LIKE - відповідність шаблону

        - LIMIT - обмежуємо кількість повертаємих рядків у запиті

        - перевірка на NULL

        - GROUP BY - як сгрупувати

        - HAVING - як фільтрувати результати після групування

        - UNION, INTERSECT, EXCEPTION - згадаємо що будь-яка таблиця в SQL (а результат запиту - це знову таблиця!) - це множина. Тому операції над множинами можуть стати у нагоді.

**3) Лекція 3: об'єднання таблиць (JOIN) і підзапити.**

    1) JOIN - поєднання таблиць

        - що таке поєднання і коли воно потрібно

        - INNER JOIN - детальний розбір алгоритму поєднання

        - OUTER JOIN (LEFT JOIN, RIGHT JOIN) - відмінності алгоритма поєднання

        - FULL JOIN

        - CROSS JOIN

        - USING та NATURAL JOIN: знову про "синтаксичний цукор". Як зробити код коротшим і, можливо, заложити основи складного для пошуку бага.

        - AS - псевдоніми

    2) Підзапити

        - структурування запитів буває користним і зручним

        - синтаксис підзапитів

        - WHERE EXIST: чи існує результат підзапиту?

        - синтаксичні конструкції ANY (SOME) \ ALL

**4) Лекція 4: Інші об'єкти баз даних: представлення і функції.**

    1) Представлення VIEW

        - загальна інформація:

            - що таке представлення

            - для чого використовуються представлення

            - які бувають представлення

        - Створення представлень

            - проосте представлення з однієї таблиці

            - представлення шляхом поєднання двох (або більше) таблиць

            - використання підзапитів при формуванні представлень

            - опціональні елементи синтаксису створення представлень

                - OR REPLACE

                - TEMP (або TEMPORARY)

            - Модифікація даних через VIEW

    2) Функції в PostgreSQL.

        - розуміння функцій в SQL

        - коли писати функцію в БД і коли за межами БД

        - деякі вбудовані в PostgreSQL функції

        - створення власних функцій в PostgreSQL

            - існуючи варіанти мов для написання серверних функцій в PostgreSQL (SQL, PL\pgSQL, PL\Python, PL\Tcl, PL\Perl, C)

            - основні правила створення серверних функцій в PostgreSQL

            - SQL: синтаксис визначення функції

                - функція без параметрів

                - функції з аргументами

            - використання процедурних мов для створення функцій

                - огляд процедурних мов

                - для чого це все?

                - введення в PL\pgSQL

                    - синтаксис визначення функції

                    - функції, які повертають скалярне значення

                    - RECORD

                    - як повернути декілька записів

                    - декларування змінних

                    - керуючи конструкції

                        - if\elsif ... then\else\end if;

                        - цикл WHILE

                        - цикл FOR

**5) Лекція 5: Проектування БД. DLL - керуємо базами даних і об'єктами баз даних (таблицями, представленнями, функціями і іншими).**

    1) Проектування БД.

        - методології проектування БД і чому часто це будете робити саме Ви.

        - етапи проектування БД

            - концептуальний

            - логічний

            - фізичний

        - деякі рекомендації щодо проектування БД

    2) Нормалізація БД

        - основні визначення

        - перша нормальна форма (1NF)

        - друга нормальна форма (2NF)

        - третя нормальна форма (3NF)

        - прикдад створення і нормалізації простої БД

    3) DDL - керуємо базами даних і таблицями.

        - CREATE DATABASE - створення нової БД

        - DROP DATABASE - видалення БД

        - CREATE TABLE - створення таблиці

        - ALTER TABLE - модифікація таблиці

            - ADD COLUMN ... - додаємо колонку

            - RENAME TO ... - перейменування таблиці

            - RENAME ... TO ... - перейменування колонки

            - ALTER COLUMN SET DATE TYPE ... - зміна типу даних колонки

        - DROP TABLE ... видалення таблиці

        - TRUNCATE TABLE ... - видалення даних в таблиці

        - DROP COLUMN ... видалення колонки. Стратегії видалення.

        - CHECK - перевірка даних перед вставкою в таблицю.

        - DEFAULT - встановлення значень за замовченням.

        - INSERT - вставка даних в таблицю.