## Wikiracing

Чи ви щось чули про гру, де треба за мінімальну кількість переходів потрапити з однієї статті Вікіпедії на іншу (іноді це стаття про Гітлера). Для виконання цього завдання ми з тобою зіграємо в гру.

Необхідно написати функцію, що приймає як параметр дві назви статтей і повертає список імен сторінок, через які можна потрапити на неї, чи пустий список, якщо такого шляху знайти не вдалось.

Наприклад:

(‘Дружба’, ‘Рим’) -> [‘Дружба’, ‘Якопо Понтормо’, ‘Рим’]

**Умови:**

* Шукаємо статті тільки на українській вікіпедії
* Технічні статті такого типу: <https://uk.wikipedia.org/wiki/%D0%92%D1%96%D0%BA%D1%96%D0%BF%D0%B5%D0%B4%D1%96%D1%8F:%D0%92%D1%96%D0%B9%D0%BD%D0%B0/%D0%A0%D0%B5%D1%81%D1%83%D1%80%D1%81%D0%B8> чи <https://uk.wikipedia.org/wiki/%D0%9E%D0%B1%D0%B3%D0%BE%D0%B2%D0%BE%D1%80%D0%B5%D0%BD%D0%BD%D1%8F:%D0%A8%D0%B5%D0%B2%D1%87%D0%B5%D0%BD%D0%BA%D0%BE_%D0%A2%D0%B0%D1%80%D0%B0%D1%81_%D0%93%D1%80%D0%B8%D0%B3%D0%BE%D1%80%D0%BE%D0%B2%D0%B8%D1%87> парсити не потрібно
* Частоту запитів необхідно обмежити. Ви маєте використовувати параметр - максимальну кількість запитів за хвилину.
* Треба передбачити обробку помилок та retry для запитів
* На кожній сторінці беремо тільки 200 перших посилань
* Отриману інформацію про посилання зі сторінки треба зберігати в базі даних postgres, що працює в контейнері
* При наступному запуску необхідно використовувати зв’язки з бази данних щоб не робити одні й ті ж запити двічі
* На базі, що наповниться після кількох запусків, треба виконати запити, що шукають:
  + Топ 5 найпопулярніших статей (ті що мають найбільшу кількість посилань на себе)
  + Топ 5 статей з найбільшою кількістю посилань на інші статті
  + Для заданної статті знайти середню кількість потомків другого рівня
  + (На додаткові бали) Запит, що має параметр - N, повертає до п’яти маршрутів переходу довжиною N. Сторінки в шляху не мають повторюватись.
* Перевірятись код буде на Python 3.10

**Порядок здачі завдання:**

Необхідно надіслати архів, що містить:

* wikiracing.py з вашою реалізацією а також інші .py файли, якщо ви вирішили їх створити
* wikiracing\_test.py (буде наданий) з тестами, що демонструють роботу вашої програми
* requirements.txt з переліком бібліотек для встановлення через pip install -r
* docker-compose.yaml для запуску постгрес (буде наданий)
* Папку pg\_data для того щоб можна було швидше перевірити завдання (створиться автоматично коли запустите docker compose)
* queries.sql з вашими запитами

Архів із файлами для початку роботи:

<https://drive.google.com/file/d/1MasdP4YFRVMzxOgw8SYUTPQjsFPq-16g>