

Основне завдання Back-end

1. Створити back-end застосунок на зручній платформі (Node.js, Java, C#, Python, etc.)
1. Реалізувати REST API для роботи з даними за викидами CO₂, з можливістю запиту:
 - підсумкових даних по викидах за рік/api/v1/co2/summary;
 - розподілу викидів CO₂ по виробничих галузях за останній рік. Кожна галузь, представлена у відповіді, повинна мати назву і частку від загального. Сума всіх часток повинна скласти 1 (або 100 %)/api/v1/co2/distribution;
 - різниці (у відсотках) викидів CO₂ між попереднім і поточним роком для наявного числа років/api/v1/co2/diffs. Приклад: якщо минулого року було викинуто всього 100 тис. тонн CO₂, а в цьому році 120 тис. тонн, то потрібно повернути «+ 20%» або 1.2 (у вигляді коефіцієнта);
 - списку галузей, які стали піковими щодо викидів CO₂ (для кожного року повернути назву галузі, що стала піковою за викидами CO₂ у той конкретний рік)/api/v1/co2/peaks.
2. Покрити ключовий код тестами.

Додаткове завдання (якщо залишається час та є бажання)

1. Розгорнути застосунок на безкоштовному хостингу (наприклад, Heroku).
2. Передбачити конфігурацію для запуску в різних середовищах (наприклад, development локально та production для хостингу).
3. Додати кешування даних.
4. Передбачити зручне керування та зберігання даних:
 - 1) перекласти зберігання віддаваних величин до бази даних;
 - 2) зробити просту server-side адмін-панель з авторизацією, через яку можна редагувати/заповнювати всі дані, що будуть віддаватися через готовий REST API;
 - 3) у разі, якщо ви додали кешування, розв'язати проблему повернення через REST API закешованих старих даних після оновлення даних у базі (через адмінку).
5. Написати Dockerfile для збірки Docker образу, з якого можна запустити застосунок.