**Фінальний проєкт**

**Частина 1. Building an End-to-End Streaming Pipeline**

Зображення, що містить текст, меню, знімок екрана

Автоматично згенерований опис

Виконано завантаження таблиці olympic\_dataset.athlete\_bio за допомогою spark з mySQL бази даних.

Зображення, що містить текст, знімок екрана, меню

Автоматично згенерований опис

Виконано фільтрацію даних, числові параметри переведені в формат double.

Зображення, що містить текст, меню, знімок екрана, чорно-білий

Автоматично згенерований опис

Зчитано таблицю athlete\_event\_results.

Зображення, що містить текст, знімок екрана, меню, чорно-білий

Автоматично згенерований опис

Дані переведені в формат JSON для подальшого відправлення до kafka топіка.

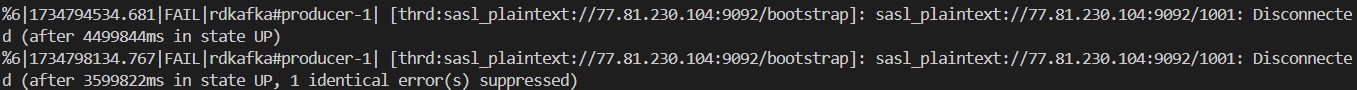
Зображення, що містить текст, знімок екрана, меню, Шрифт

Автоматично згенерований опис

Проведена потокова агрегація даних: середній зріст і вагу атлетів індивідуально для кожного виду спорту, типу медалі або її відсутності, статі, країни (скріншот вище і нижче – приклади батчів).

Зображення, що містить текст, знімок екрана, меню, чорно-білий

Автоматично згенерований опис



Але я так і не впоралася з останнім завданням першої частини: паралельним потоковим записом у два цільових сховища.

**Частина 2. Building an End-to-End Batch Data Lake**

Зображення, що містить текст, знімок екрана, Шрифт, ряд

Автоматично згенерований опис

Скриншот графу відпрацьованого DAGу

Зображення, що містить текст, Шрифт, документ, лист

Автоматично згенерований описЗображення, що містить текст, Шрифт, чорно-білий, знімок екрана

Автоматично згенерований опис

Скачані з FTP та преобразовані в спарк-датафрейм таблиці

Зображення, що містить текст, меню, документ, папір

Автоматично згенерований опис

Зображення, що містить текст, меню, чорно-білий, Шрифт

Автоматично згенерований опис

Таблиці на скріншотах вище – після очистки та дедуплікації

Зображення, що містить текст, документ, чорно-білий, Шрифт

Автоматично згенерований опис

Фінальна таблиця з розрахунками