**Общее описание задания**

Необходимо загрузить исходные данные, произвести расчет, сохранить результат расчета и показать код решения. Задание состоит из двух частей: аналитической и расчетной. Диалект для SQL скриптов: Postgres и/или ClickHouse

*Дополнительные* задания не обязательны к выполнению.

Исходные данные представлены несколькими [файлами](https://drive.google.com/file/d/1eVeGkxXjZWwUvP4fU_r_9pFIIaCD4muD/view?usp=sharing) в формате csv:

* t\_sales.csv История продаж
* t\_products.csv Описание товара
* t\_cities.csv Список городов
* t\_branches.csv Список складов и магазинов

**Техническая часть**

1. Развернуть локально базу данных и перенести данные из csv-файлов в таблицы при помощи Python;
2. Максимально ускорить время выполнения задач аналитической части, оптимизацией БД и скрипта.

Например:

* Добавить индексы к таблицам для ускорения работы sql запросов;
* Выполнение в несколько потоков.

1. Сохранить результат работы любого из запросов аналитической части в базу данных

**Аналитическая часть**

Требуется подготовить Python скрипты c использованием SQL запросов, которые рассчитывают и записывают результат в таблицы БД:

* 1. десять самых продаваемых товаров для каждого магазина;
  2. десять самых продаваемых товаров для каждого города;
  3. десять первых магазинов по количеству проданного товара с суммой продаж;
  4. рейтинг товаров согласно суммарному количеству проданного товара за всю историю наблюдений со средним количеством продаж за день по убыванию;
  5. Два лучших филиала согласно суммарному количеству проданного товара за всю историю наблюдений в регионе Урал по городу Екатеринбург с суммой продаж за январь
  6. Требуется рассчитать и вывести в какие часы и в какой день недели происходит максимальное количество продаж.

***Дополнительно:***

1. Написать запрос на 2-х диалектах: PostgreSQL и ClickHouse для сохранения результата выполнения задания 1.4 аналитической части и показать отличия между синтаксисом и специфике данных СУБД.
2. Приложить код ddl на 2-х диалектах: PostgreSQL и ClickHouse для создания таблиц, хранящих результат аналитических запросов.

**Будет плюсом, если:**

* Скрипт Python будет фиксировать время выполнения общее и запросов;
* В коде будет присутствовать комментарии.