Тема 5. Data analytics (аналітика даних)

1. Основні поняття в Data Analysis

2. Робота з даними

3. Робота з BigQuery

3.1. Поняття баз даних (БД). Хмарна БД BigQuery

3.2. Опис панелі управління BigQuery

3.3. Основні категорії типів даних, які підтримуються в BigQuery

3.4. База даних Google Trends – International

4. Мова запитів SQL (Structured Query Language)

4.1. Основні категорії елементів SQL

4.2. Агрегатні функції в SQL

4.3. Оператори порівняння

4.4. Робота з типами даних у SQL-коді

4.5. Робота з помилками

5. Візуалізація даних

1. Основні поняття в Data Analysis

Що таке Data Analysis?

Data Analysis (аналіз даних) - це процес перетворення даних у корисну інформацію з метою виявлення закономірностей, отримання інсайтів та прийняття обґрунтованих рішень. Ця діяльність включає в себе використання різних методів, технік та інструментів для аналізу даних і вивчення їхньої структури.

У сфері IT аналіз даних майже завжди пов'язаний із data-driven decision making, тобто з ухваленням рішень на основі даних. Цей підхід відрізняється від ухвалення рішень "на досвіді" чи "за відчуттям".

Важливість Data Analysis у сучасних реаліях

Data Analysis досліджує елементи та аспекти цілого для отримання змістовних висновків. Можна аналізувати, що завгодно — вираз обличчя, зміст літератури, політику, продукт, програмне забезпечення.

Data Analysis є ключовою частиною загальної аналітики даних і однією з основних дисциплін у галузі даних, яка використовує передові аналітичні методи для пошуку корисної інформації в наборах даних.

Ефективний аналіз даних допомагає в різних аспектах плануванню бізнес-стратегії та управління операціями. Сюди входять функції, що стосуються клієнтів, як-от маркетинг, реклама, продажі та підтримка клієнтів, а також виробництво, управління ланцюгом постачання, фінанси та кадри. Аналіз даних підтримує виявлення шахрайства, управління ризиками, планування кібербезпеки та багато інших критичних бізнес-випадків використання. Він також відіграє важливу роль в охороні здоров’я, урядових рішеннях, наукових дослідженнях, математиці, спорті тощо.

Цінність даних нічого не означає, якщо вони не забезпечують змістовну інформацію. Саме тут візуалізація даних стає важливою.

Хто такий дата-аналітик і чим він займається

Дата-аналітик (англ. Data Analyst) - це фахівець, який володіє навичками та знаннями для збору, обробки, аналізу та візуалізації даних з метою отримання інсайтів та підтримки ухвалення рішень.

Основна робота дата-аналітика пов'язана з перетворенням зібраних даних у зрозумілу та корисну інформацію, яка допомагає організаціям і бізнесам робити обґрунтовані висновки й удосконалювати свої операції.

Основні Hard Skills та Soft Skills, якими має володіти дата-аналітик

Дата-аналітик повинен мати ряд технічних (Hard Skills) та міжособистісних (Soft Skills) навичок для успішної роботи.

Hard Skills (Технічні навички):

1. *Знання програмування і мов програмування* - володіння мовами програмування, такими як Python, SQL або іншими, для обробки та аналізу даних.
2. *Робота з базами даних* - здатність взаємодіяти з базами даних, створювати запити SQL, отримувати та обробляти дані.
3. *Статистика і математичний аналіз* - розуміння основних концепцій статистики і можливість застосовувати математичні методи для аналізу даних.
4. *Обробка та візуалізація даних* - вміння використовувати інструменти для очищення та візуалізації даних, такі як SQL, Excel, Power BI, Tableau, Pandas, Matplotlib, Seaborn тощо.
5. *Машинне навчання і аналіз даних* - розуміння основ машинного навчання, можливість побудови моделей та аналізу їх результатів.
6. *Робота з інструментами великих даних* - знання інструментів для обробки та аналізу великих обсягів даних, таких як Power BI тощо.
7. *Знання інструментів BI (Business Intelligence)* - розуміння і вміння користуватися інструментами для створення дашбордів і звітів, такими як Tableau, Power BI, Google Looker Studio, QlikView тощо.

Soft Skills (Міжособистісні навички):

1. *Аналітичне мислення* - здатність аналізувати складні питання та робити висновки на основі даних.
2. *Креативність* - здатність думати креативно і знаходити нестандартні шляхи розв'язання завдань.
3. *Комунікаційні навички* - здатність чітко і зрозуміло комунікувати за результатами аналізу про інсайти з іншими членами команди і стейкхолдерами.
4. *Ініціативність* - здатність висувати нові ідеї, пропозиції та самостійно шукати нові завдання, вдосконалювати свої навички.
5. *Співпраця* - здатність працювати в команді та співпрацювати з іншими членами команди.
6. *Гнучкість і адаптивність* - готовність адаптуватися та підлаштовуватися до змін у завданнях і технологіях.
7. *Етика й конфіденційність* - здатність обробляти дані з урахуванням етичних стандартів та дотримуватися конфіденційності.
8. *Можливість презентації* - здатність ефективно представляти результати своєї роботи перед аудиторією.

Поєднання таких технічних і міжособистісних навичок допомагає дата-аналітику виконувати завдання з обробки та аналізу даних, а також ефективно комунікувати і співпрацювати у бізнес-середовищі.

Роль дата-аналітика у бізнесі

Роль дата-аналітика у бізнесі дуже важлива, і вона може впливати на різні аспекти діяльності компанії.

Розглянемо деякі ключові аспекти, у яких дата-аналітик відіграє важливу роль:

- *Ухвалення рішень на основі даних.* Дата-аналітики допомагають бізнесу ухвалювати обґрунтовані рішення, спираючись на факти та інсайти, які вони отримують під час аналізу даних. Це може стосуватися стратегічних рішень, таких як розширення ринків або запуск нових продуктів, або таких щоденних питань, як, наприклад, ціноутворення.
- *Покращення ефективності операцій.* Дата-аналітика дозволяє бізнесам оптимізувати свої операційні процеси. Наприклад, вона може допомогти виявити ефективні шляхи доставлення, розподілу ресурсів або управління запасами.
- *Розвиток маркетингових стратегій*. Дата-аналітика грає ключову роль у розробці маркетингових кампаній. Вона дозволяє визначити цільову аудиторію (ЦА), оцінити ефективність рекламних кампаній, виявити нові ринкові можливості та змінити стратегії на основі реакції клієнтів.

- *Зростання прибутковості.* Дата-аналітика може допомогти виявити можливості для збільшення прибутковості. Це може бути шлях зниження витрат або збільшення продажів через більш точно спрямовані маркетингові кампанії.
- *Управління клієнтським досвідом*. Аналіз даних допомагає розуміти потреби і побажання клієнтів, що дозволяє покращити обслуговування та створювати продукти та послуги, які відповідають очікуванням.
- *Виявлення ризиків і мінімізація збитків.* Дата-аналітика допомагає виявляти потенційні ризики й небезпеки для бізнесу. Це дозволяє компаніям вживати заходів для запобігання та уникнення таких небезпек.

Загалом дата-аналітика допомагає бізнесам бути більш інформованими та конкурентоспроможними, розвиватися та збільшувати прибутковість.

2. Робота з даними

Основні етапи аналізу даних

Дата-аналіз - це складний процес, який включає в себе кілька ключових етапів для ефективного використання даних у бізнесі або дослідженнях.

Основні етапи Data Analysis:

1. *Визначення завдання.* Спочатку треба розуміти з якою метою проводиться аналіз, які дані потрібні та що саме потрібно аналізувати.
2. *Збір даних.* Етап, на якому сирі дані отримуються з різних джерел, як-от баз даних, вебсервісів, датчиків, опитування тощо.
3. *Очищення і підготовка даних***.** Очищення включає в себе видалення помилкових даних, заповнення пропусків і перетворення даних у зручний для аналізу формат. Цей етап часто займає багато часу, адже якість вихідних даних важлива для точності результатів.
4. *Аналіз даних.* На цьому етапі використовуються різні методи аналізу, як-от статистика, машинне навчання, візуалізація тощо. Метою є виявлення корисних залежностей, трендів та закономірностей в даних.
5. *Візуалізація даних.* Важливий аспект аналізу даних, де створюються графіки, діаграми для наглядного представлення результатів аналізу. Візуалізація допомагає зрозуміти дані швидше і ефективніше.
6. *Інтерпретація результатів і ухвалення рішень.* Після аналізу та візуалізації даних аналітик ухвалює рішення на основі знайдених відомостей. Це може включати в себе розробку стратегій, оптимізацію процесів, прогнозування результатів тощо.
7. *Звітність.* Результати аналізу даних можуть бути представлені у вигляді звіту або документації для спільноти чи стейкхолдерів, які ухвалюють рішення.

Розглянемо більш детально декілька етапів дата-аналізу.

Етап 2. Збір даних. Після визначення завдання в аналізі даних наступним кроком є збір даних. Для того, щоб вирішити поставлене завдання, необхідно мати доступ до даних, які можуть допомогти виявити причини проблеми та запропонувати рішення.

Перш ніж приступити до виконання завдання, переконайтеся, що:

1. Маєте достатньо даних для виконання завдання.
2. Ці дані мають задовільну якість.

Джерела даних можна поділити на п’ять категорій:

- *Внутрішні бази даних (БД) та технічні документи компанії.* Наприклад, БД вашого сайту чи логи вебсервера.
- *Дані з сервісів, які використовує ваша компанія.* Більшість серверів дозволяє забирати звідти дані автоматично, наприклад, через API. Приклади сервісів: CRM, Ads (Google, Facebook), Email marketing, тощо. Інтеграцію з цими сервісами та отримання даних можуть виконати розробники, або ви можете використовувати для цього сторонні сервіси (Fivetran, Stitch)
- *Програмне забезпечення (ПЗ) для аналітики та відстежування поведінки користувачів.* Це може бути, наприклад, Google Analytics, Amplitude, та інші.

- *Таблиці Excel та Google Sheets,* які завжди будуть майже в будь-якій компанії.
- *Дані із зовнішніх сервісів,* наприклад, з реєстрів чи з соціальних мереж. Хоча вони не так часто використовуються саме в аналізі даних, але все ще можуть бути використані.

Якість даних відіграє важливу роль в аналізі даних, оскільки будь-які неточності або помилки в даних можуть призвести до неправильних висновків та рішень. Крім того, неправильні дані можуть призвести до непотрібних витрат на додаткові дослідження та виправлення помилок.

Перед початком роботи з даними важливо переконатися, що вони відповідають **критеріям якості:**

- *Точність (Accuracy)* — дані є точними. Наприклад, дані про оплати не округлюються, а відображаються з точністю до копійки.
- *Повнота (Completeness)* — дані є повними. Наприклад, всі дані про продажі присутні.
- *Консистентність (Consistency)* — дані є логічно пов'язаними між собою та взаємодіють без протиріччя. Наприклад, усі дати продажів зберігаються в єдиному форматі дати DD.MM.YYYY.
- *Вчасність (Timeliness)* — ми отримуємо дані вчасно для того, щоб зробити висновки. Наприклад, дані про продажі через місяць після продажів нам не допоможуть оптимізувати рекламу.
- *Валідність (Validity)* — дані є валідними в бізнес-контексті. Наприклад, дані про продажі містять тільки найменування товарів, які присутні в CRM або на складі.
- *Унікальність (Uniqueness)* — дані не дублюються. Наприклад, один продаж не записується кілька разів.

Етап 3. Очищення даних

Data cleaning (очистка даних) — це спосіб підвищення якості даних, коли ми їх уже отримали. Ми не покращимо заміри, наприклад, не змінимо accuracy (точність), але можемо:

- прибрати дублікати;
- зробити дані консистентними у своєму форматі;
- зробити дані унікальними;
- прибрати невалідні записи;
- покращити повноту даних (completeness) через певний метод інтерполяції (interpolation).

Існує багато інструментів як для очищення даних, так і для подальшого аналізу. Деякі з найпопулярніших інструментів:

- *Google Sheets* — один із інструментів, який найбільш широко використовується для аналізу даних. Google Sheets дозволяє проводити базовий аналіз даних, зокрема сортування, фільтрацію та групування даних. Він також надає можливість створення графіків і діаграм, що робить його корисним для візуалізації даних.
- *SQL* — мова програмування, яка використовується для роботи з базами даних. За допомогою SQL-запитів можна витягувати дані з бази даних, змінювати їх, а також проводити різні операції, такі як сортування та групування даних. SQL широко використовується в галузях, де важливо обробляти великі обсяги даних, таких як банківська справа та торгівля.
- *Python* — мова програмування, яка використовується в різних галузях, зокрема в таких як наука про дані та штучний інтелект. Python надає багато бібліотек та модулів для обробки даних та аналізу, до прикладу Pandas, NumPy та Matplotlib.
- *Tableau або Power BI* — програми для візуалізації даних, які дозволяють створювати інтерактивні діаграми та графіки, а також створювати звіти та панелі керування. Tableau може використовуватись для аналізу даних з різних джерел, зокрема баз даних, Google Sheets та вебсервісів.

Який інструмент обрати — залежить і від завдання, і від того, де і як зберігаються дані. Часто їх використовують у комплексі, наприклад:

Обробляємо дані в Pandas → компонуємо через SQL → Візуалізуємо в Tableau

Етап 4. Аналіз даних

Після збору та очищення даних можна переходити до аналізу. Тут аналітик даних використовує інструменти, про які ми вже розповідали в процесі очищення даних. Тому сконцентруємося на тому, які типи аналізу даних існують.

Як ми вже знаємо, аналіз даних — це процес вилучення корисної інформації з великого обсягу даних, що дозволяє виявити приховані закономірності, зрозуміти причини та наслідки явищ, оцінити результати та передбачити майбутнє.

Тому, **аналіз даних можна поділити на 4 типи**:

1. *Описова аналітика (Descriptive Analytics)* - розповідає про те, що сталося в минулому. Вона дозволяє перевірити наявні факти, тенденції та зв'язки в даних, що дає можливість відповісти на запитання про те, що сталося, скільки разів це сталося і які фактори сприяли тому, що це сталося.
2. *Діагностична аналітика (Diagnostic Analytics)* - допомагає зрозуміти, чому щось сталося в минулому. Вона дозволяє відслідковувати, які процеси, дії або рішення викликали певні результати або проблеми.
3. *Прогнозна аналітика (Predictive Analytics)* - передбачає, що найбільш імовірно станеться у майбутньому. Вона використовується для прогнозування попиту на товари, фінансових результатів компанії, передбачення можливих проблем у технічному обслуговуванні та багато іншого.
4. *Прескриптивна аналітика (Prescriptive Analytics)* - рекомендує заходи, яких можна вжити для впливу на результати. Вона базується на аналізі попередніх даних та збиранні відомостей з різних джерел, як-от баз даних, датчиків, мережі Інтернет тощо.

Знання цих типів аналізу даних дозволяє аналітику охоплювати всі аспекти аналізу даних і досягати максимальної ефективності у вирішенні різних завдань.

Етап 7. Звітність

Звітність відіграє важливу роль в процесі аналізу даних. Вона надає інформацію про продуктивність бізнесу, цілі та досягнення. Звітність дозволяє виявляти тенденції, прогнозувати майбутні результати та оцінювати ефективність заходів щодо покращення продуктивності. Без правильної звітності рішення, які ухвалюються на основі аналізу даних, можуть бути необґрунтованими та неправильними.

Про це поговоримо в розділі 5.

3. Робота з BigQuery

3.1. Поняття баз даних (БД). Хмарна БД BigQuery

Сучасні компанії зберігають величезну кількість інформації: дані про операції, клієнтів, працівників, інформацію про товари і послуги та інші важливі дані. Інформацію можна зберігати в Google Sheets (Google Таблиці), файлах Excel або навіть у друкованому вигляді в папках. Однак, якщо мова йде про велику компанію, яка здійснює мільйони операцій щодня, то документи не здатні вмістити таку кількість даних. Наприклад, Facebook зберігає особисті дані понад 500 мільйонів користувачів. Без автоматизованого керування численною кількістю даних не обійтися. Для цього необхідні бази даних, які дозволяють зберігати, змінювати та швидко шукати інформацію.

У Microsoft Excel максимально припустима кількість стовпців становить 16 384 і рядків – 1 048 576.

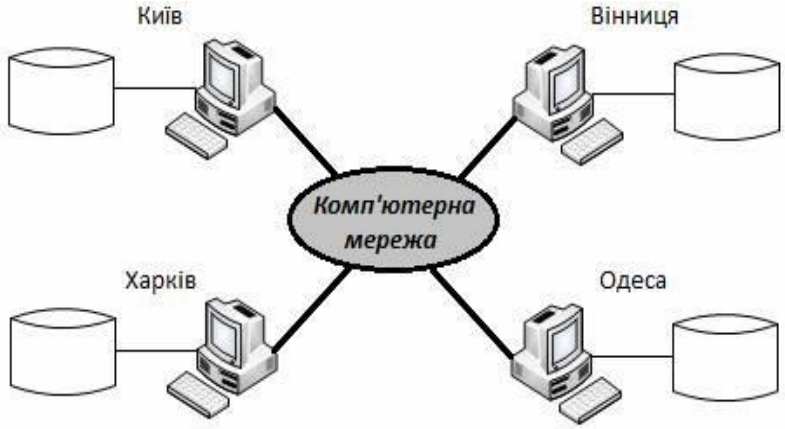
База даних – це набір взаємопов’язаних даних, який спеціально організований для пошуку за певними критеріями. Бази даних забезпечують:

1. Керування великими обсягами даних;
2. Отримання точних даних;
3. Легке оновлення даних;
4. Безпеку даних;
5. Цілісність та коректність даних.

Бази даних бувають різних видів. Найпоширеніші – **реляційні бази даних** (MySQL, PostgreSQL, SQLite, Microsoft SQL Server, інші). Вони влаштовані як набір таблиць (як звичайні таблиці, наприклад, в Excel), які пов’язані між собою. Інформація в таблицях зберігається в рядках (records) і стовпцях (fields).

Бази даних за типом розгортання та зберігання бувають:

1. ***Локальні*** (On-Premise Databases), які розгортаються і зберігаються на власних серверах або комп'ютерах користувача чи організації;



2. ***Хмарні*** (Cloud Databases), які розміщуються у хмарних середовищах і керуються сторонніми

постачальниками послуг.

Одним із популярних варіантів хмарної бази даних є BigQuery в Google Cloud Platform. BigQuery є потужним інструментом для аналітики даних, що дозволяє організаціям швидко отримувати цінну інформацію з великих обсягів даних і ухвалювати обґрунтовані рішення. Google BigQuery є PaaS-сервісом («платформа як послуга»), який підтримує більшість функцій баз даних.

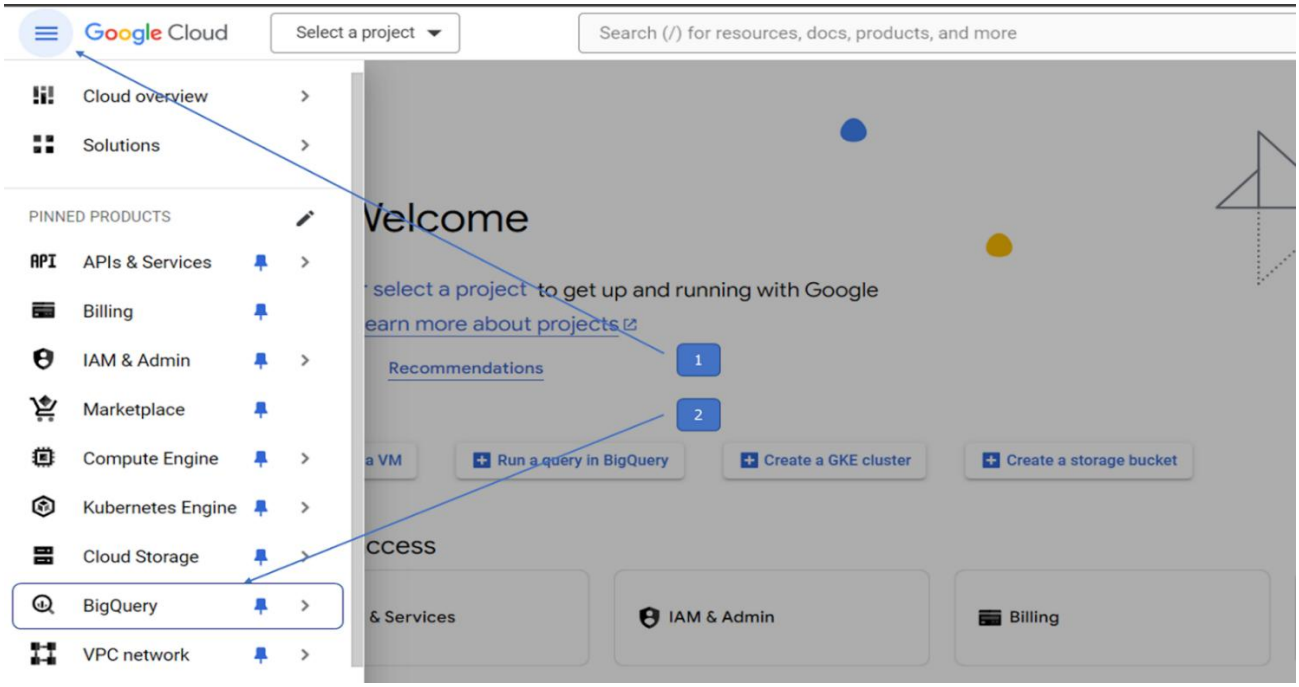
BigQuery дозволяє отримати дані з різних джерел:

- *Local file* (підключення до файлів, які зберігаються на комп'ютері користувача);
- *Google drive* (підключення до файлів, які знаходяться в хмарному сервісі на Google-диску);
- *Public Datasets* (підключення до публічних наборів даних, доступних для безкоштовного використання у вільному доступі);
- *інші* (Analytics Hub, Salesforce DataCloud, Azure Blob Storage і т.д.)

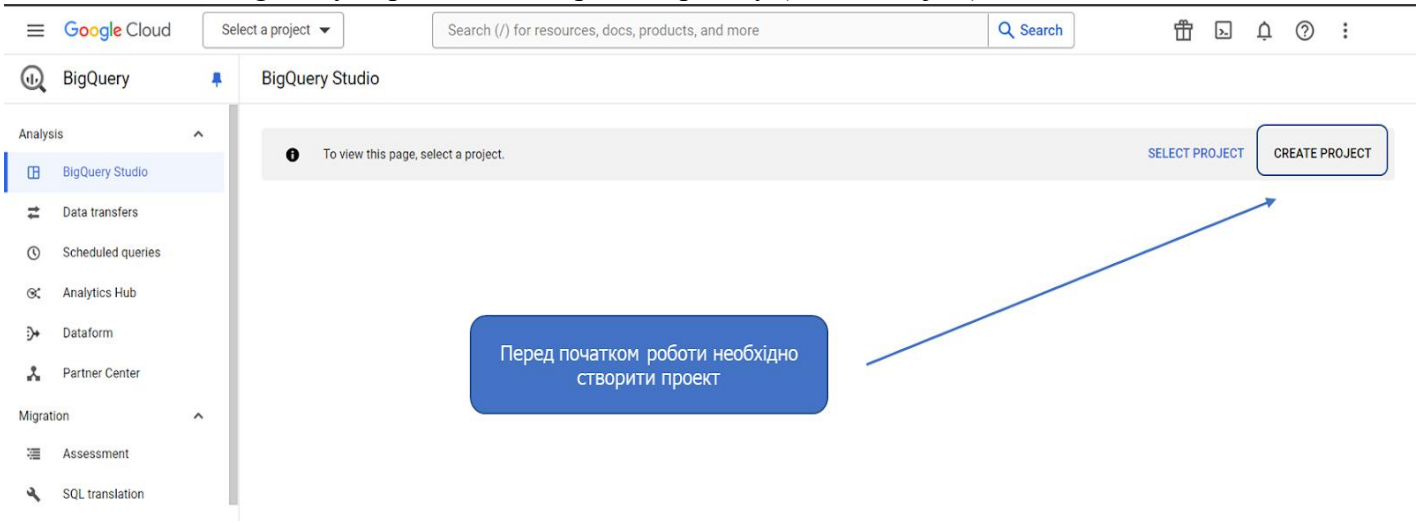
Після підключення до даних у BigQuery ви отримуєте доступ до широкого спектра можливостей для аналізу, опрацювання та використання цих даних.

3.2. Опис панелі управління BigQuery

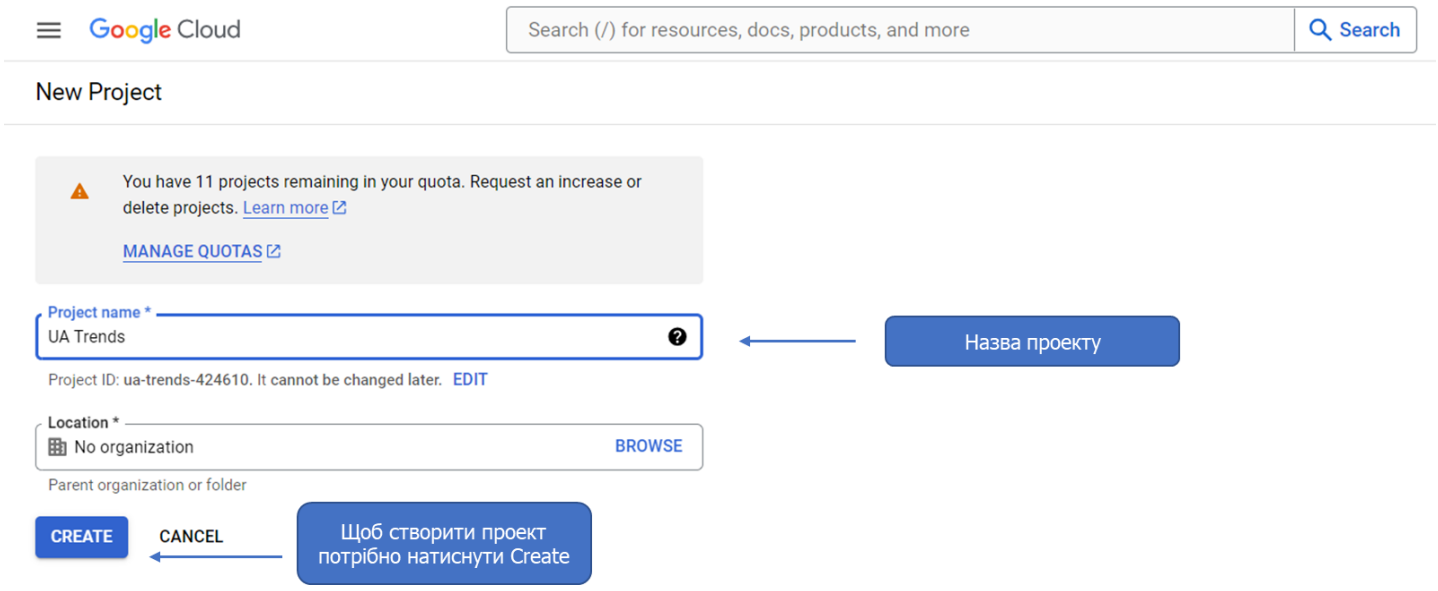
Одним із інструментів Google Cloud Console є BigQuery Studio. Щоб перейти зі стартової сторінки до підключення до даних та написання SQL-коду, потрібно зліва в меню навігації обрати BigQuery (не переходячи до додаткового меню інструментів).



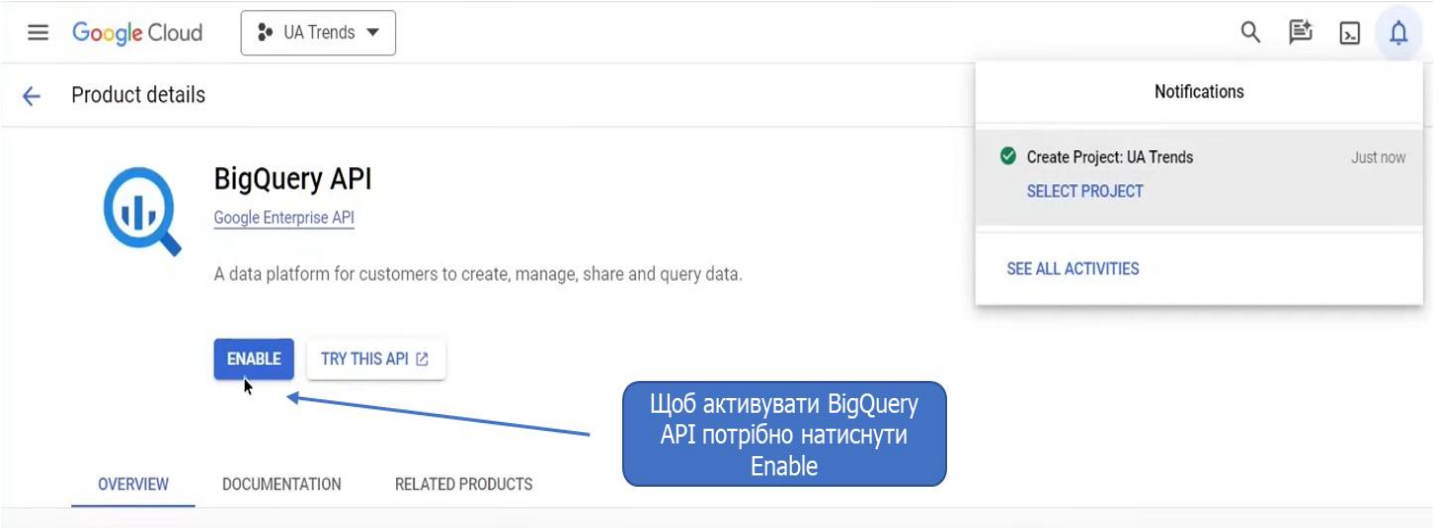
Робота в BigQuery передбачає створення проєкту (Create Project).



Щоб створити проєкт, необхідно надати назву та натиснути Create.

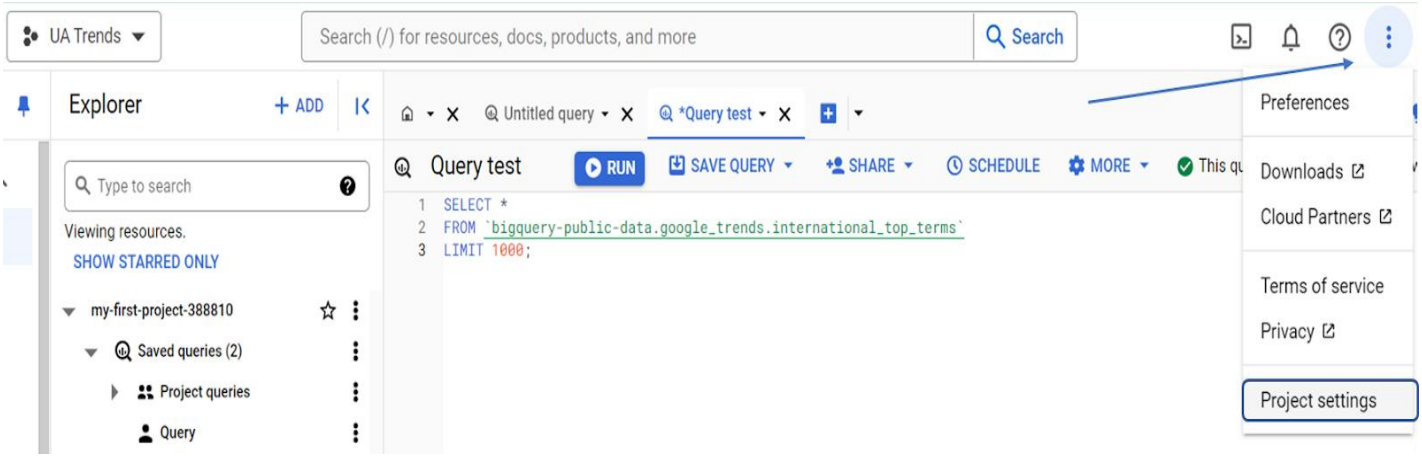


Після створення проєкту необхідно активувати BigQuery API.

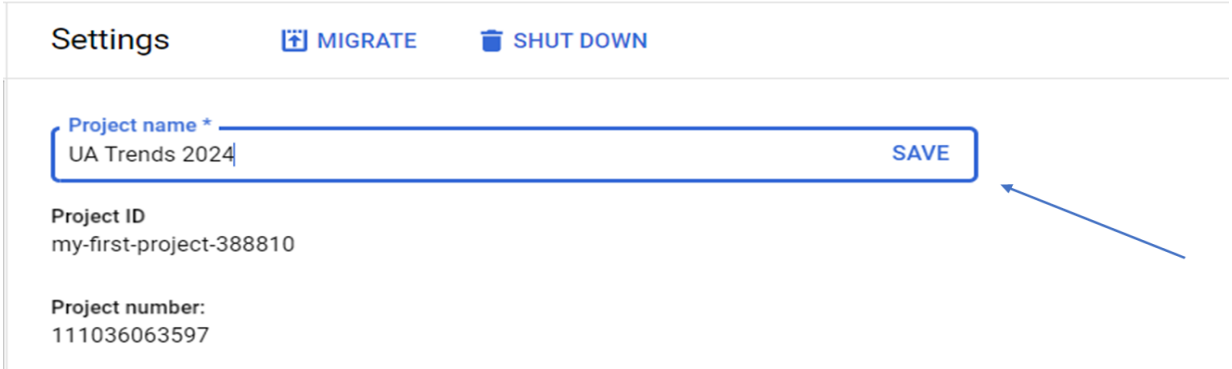


У деяких користувачів BigQuery API вже може бути активованим - це залежить від налаштувань профілю. Якщо у вас не з'явилось меню активації, можете продовжувати працювати.

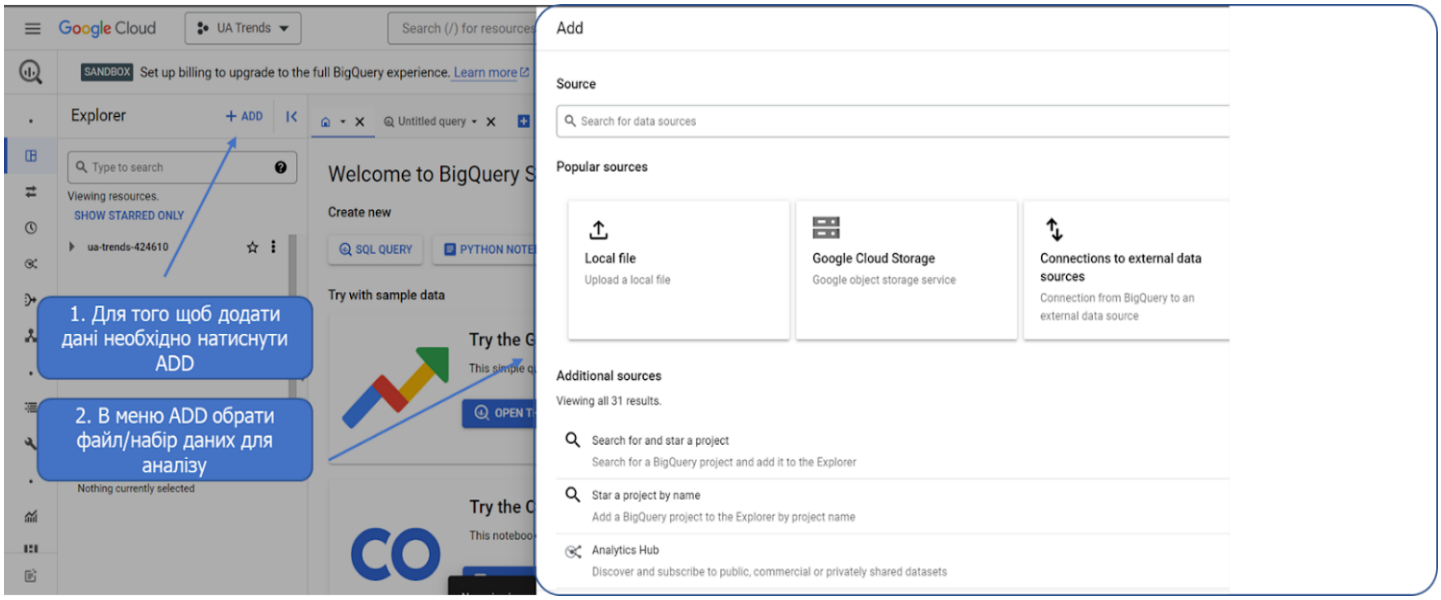
За потреби назву створеного проєкту можна змінити. Для цього необхідно перейти в налаштування справа і обрати Project Settings.



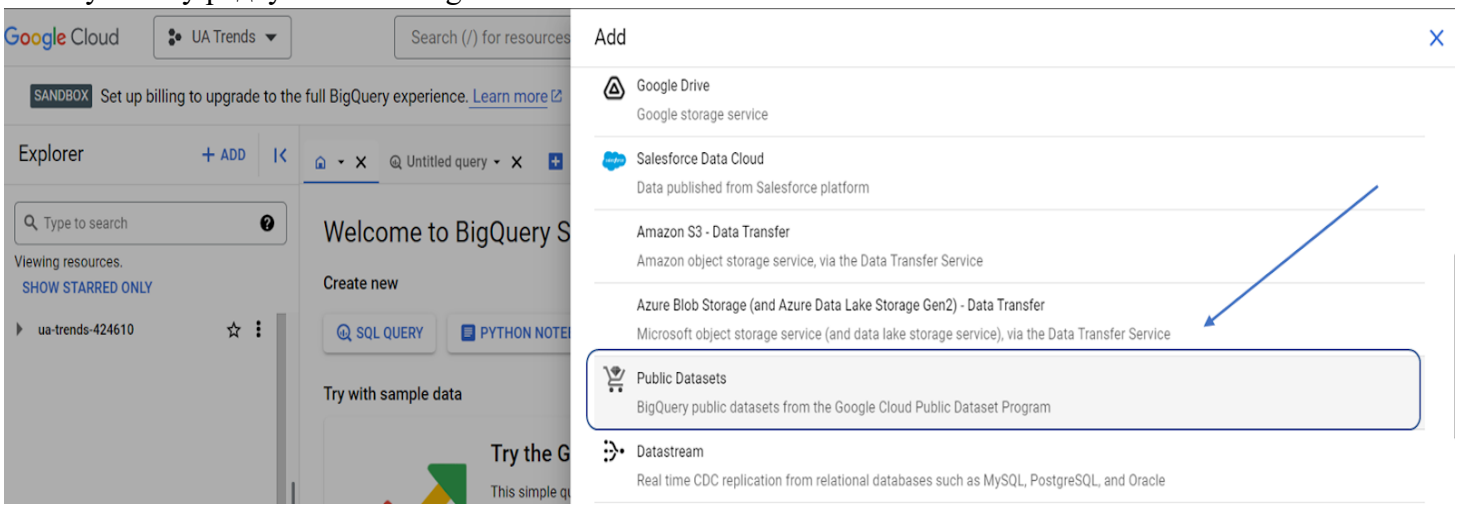
Далі в меню Project name потрібно вказати нову назву і натиснути Save.

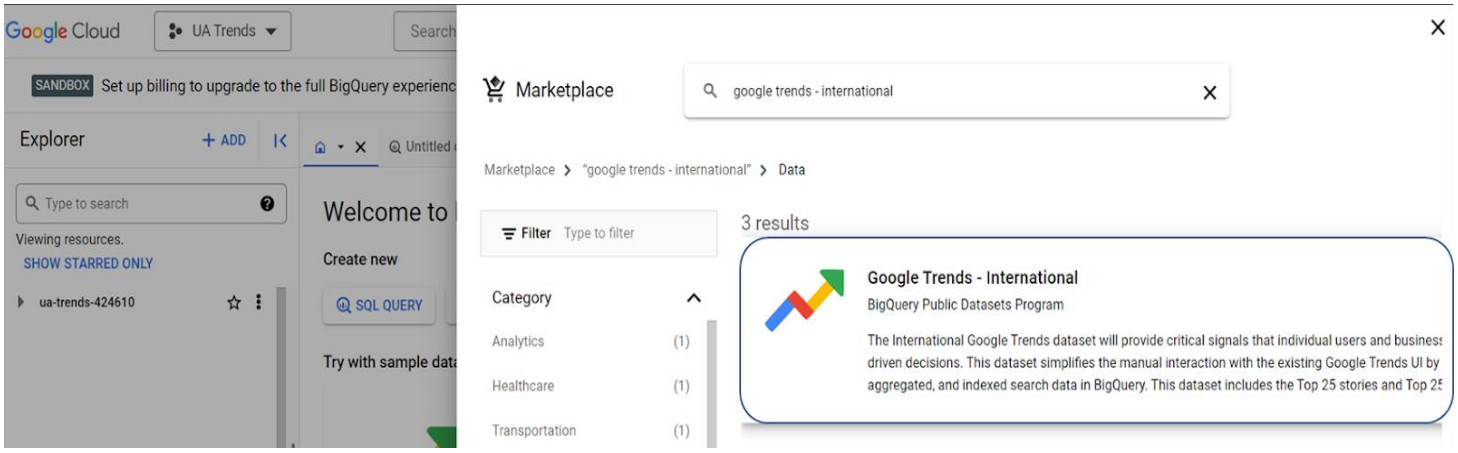


Щоб підключитись до даних, необхідно натиснути ADD та в меню справа обрати потрібний для аналізу файл / набір даних.

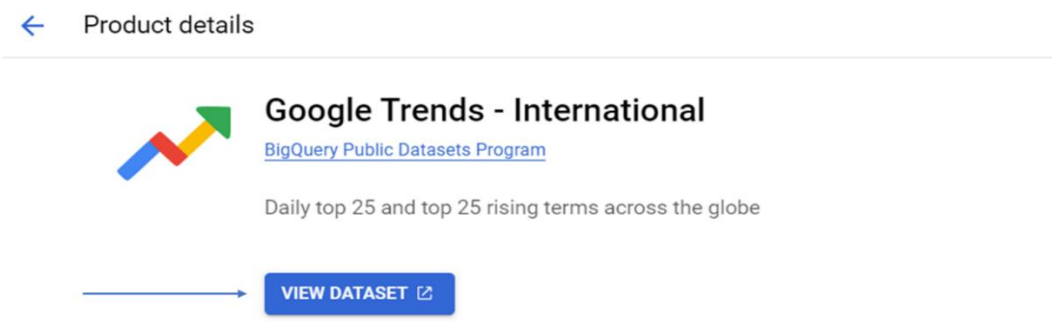


Щоб додати публічний датасет Google Trends – International, потрібно перейти в Public Datasets та в пошуковому рядку ввести Google Trends – International.



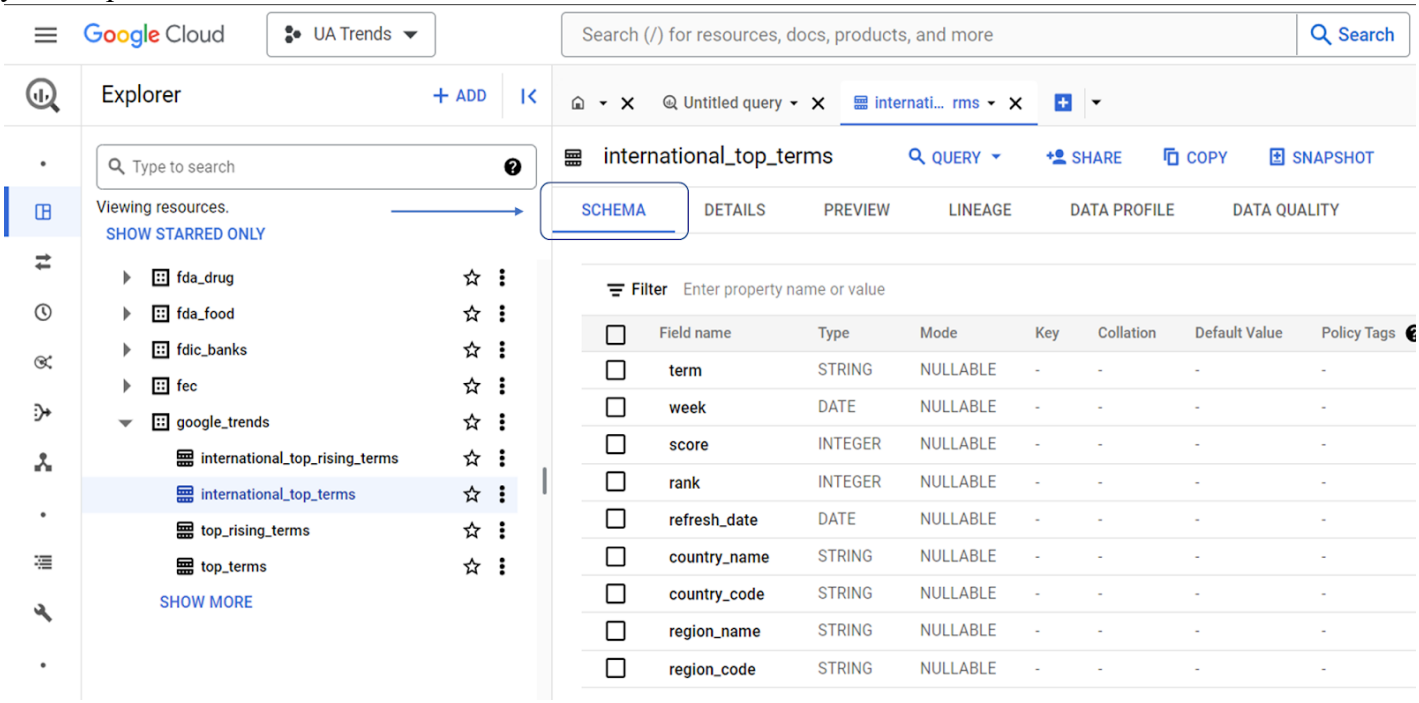


Щоб перейти до роботи з набором даних, необхідно натиснути View Dataset.

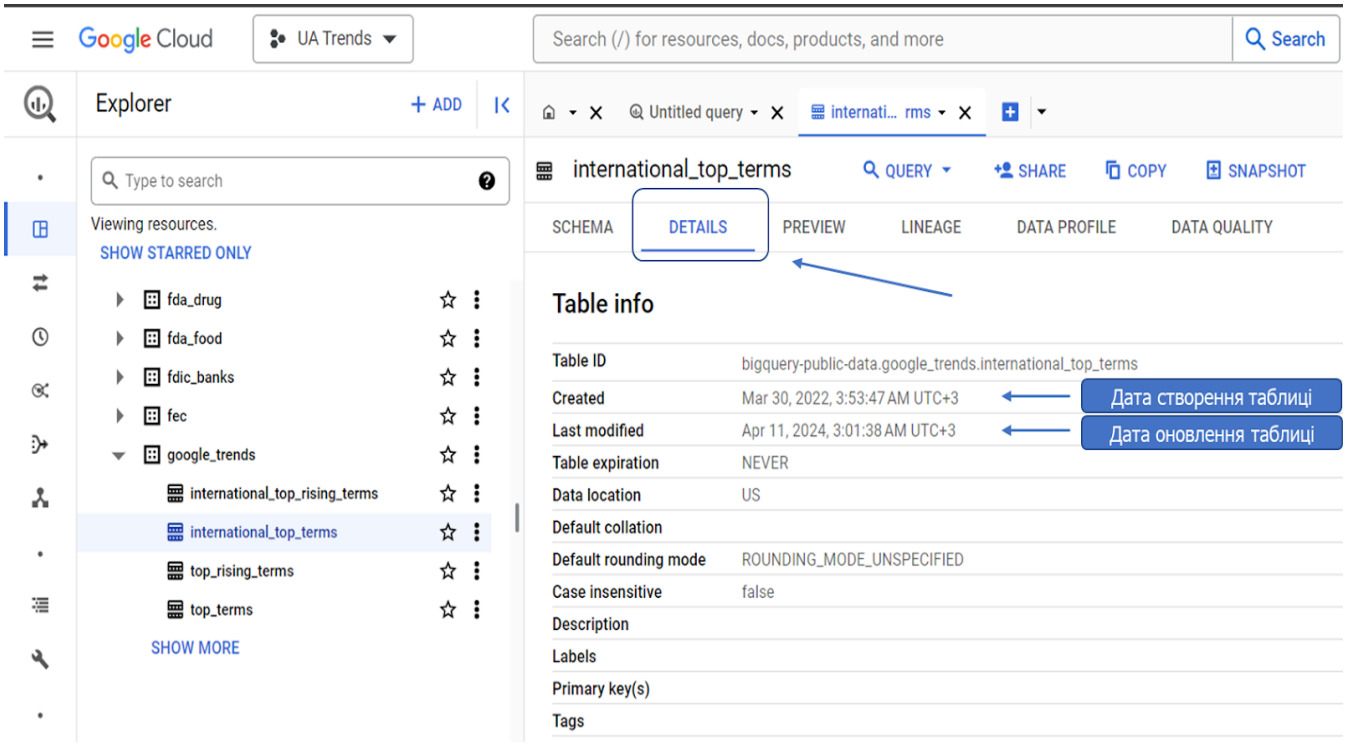


Перед початком роботи з базою даних варто ознайомитися з таблицями, які вона містить, за допомогою полів у Schema, Details, та з даними у Preview.

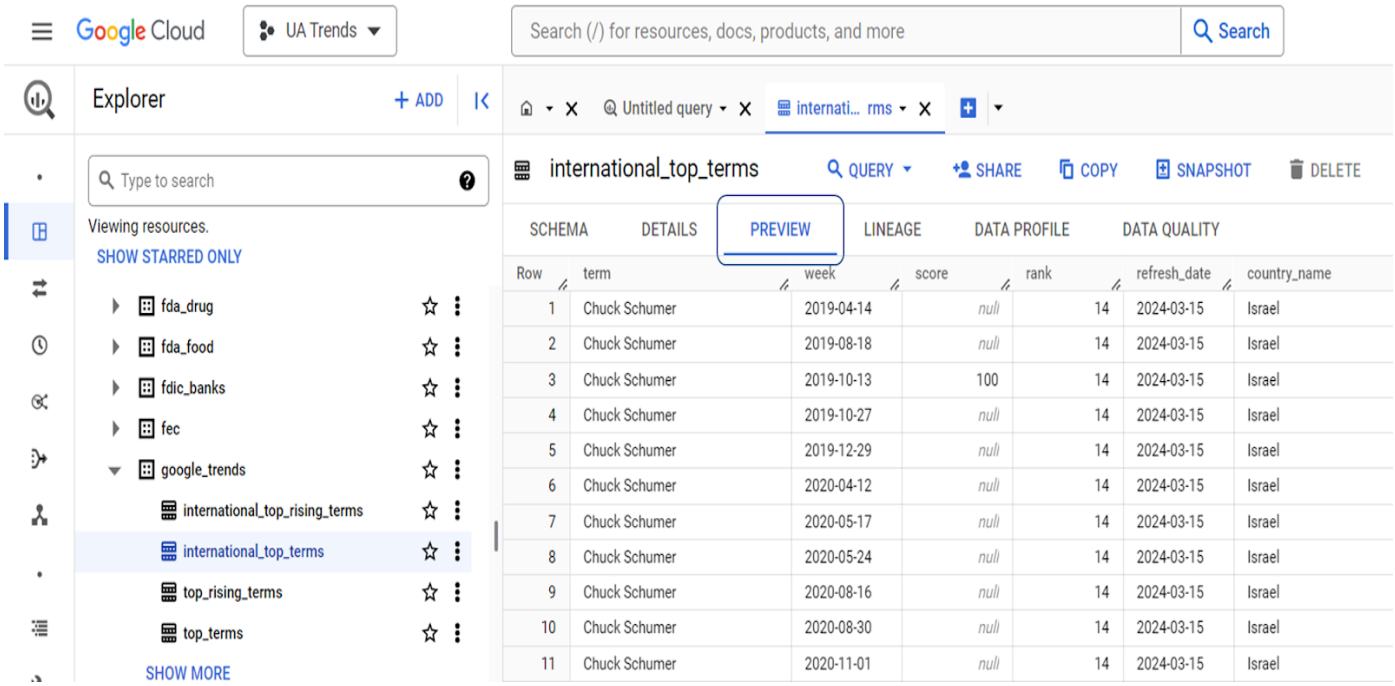
Вкладка **Schema** – тут викладені всі поля таблиці та їх тип даних, від типу залежить те, як ми будемо працювати з полями.



Вкладка **Details** – дає інформацію про дату створення і останнього оновлення таблиці, локацію, розмір тощо.

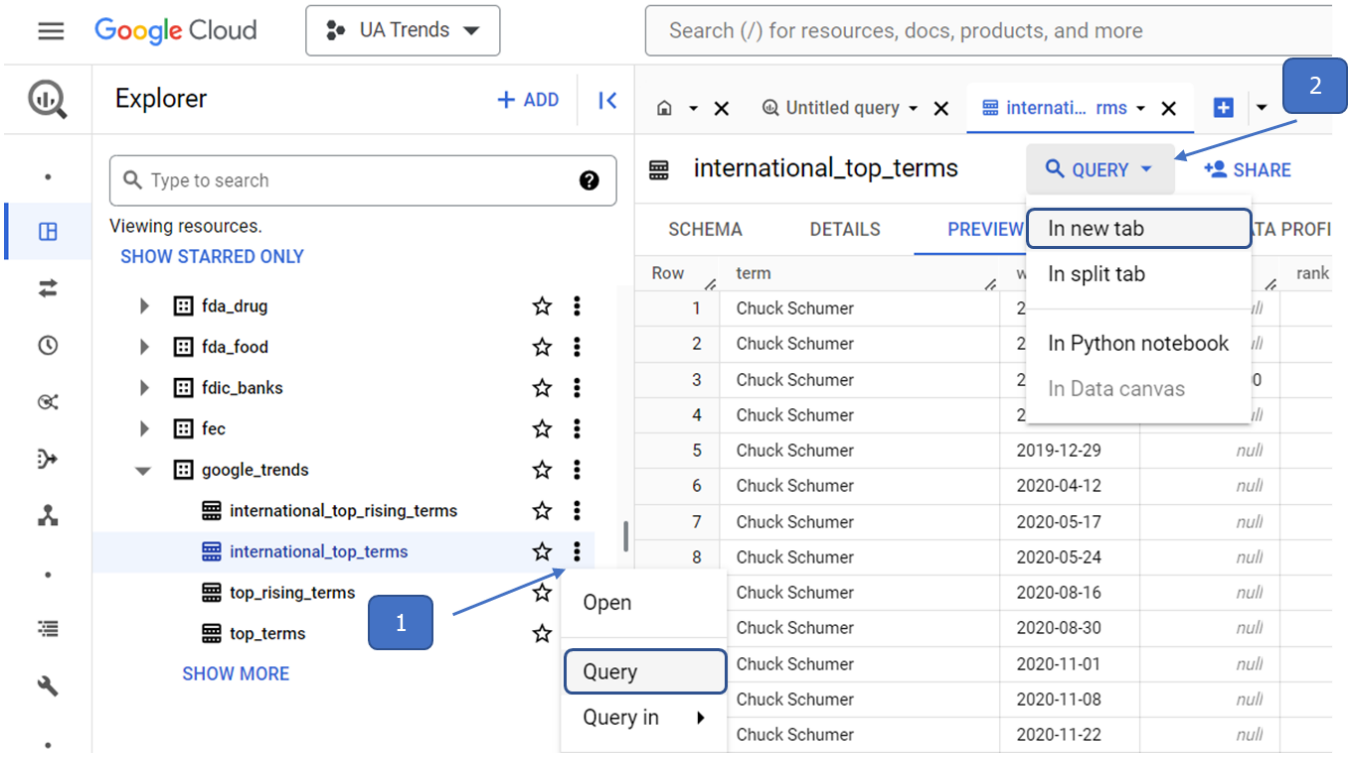


Вкладка **Preview** – тут зібрано дані, які дають уявлення структури таблиці, її наповненості.

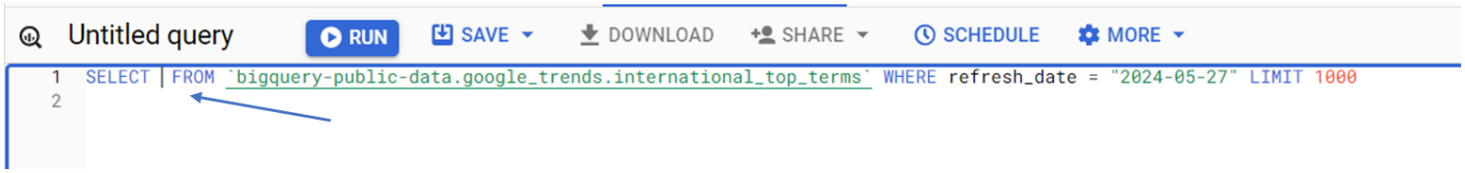


Row	term	week	score	rank	refresh_date	country_name
1	Chuck Schumer	2019-04-14	null	14	2024-03-15	Israel
2	Chuck Schumer	2019-08-18	null	14	2024-03-15	Israel
3	Chuck Schumer	2019-10-13	100	14	2024-03-15	Israel
4	Chuck Schumer	2019-10-27	null	14	2024-03-15	Israel
5	Chuck Schumer	2019-12-29	null	14	2024-03-15	Israel
6	Chuck Schumer	2020-04-12	null	14	2024-03-15	Israel
7	Chuck Schumer	2020-05-17	null	14	2024-03-15	Israel
8	Chuck Schumer	2020-05-24	null	14	2024-03-15	Israel
9	Chuck Schumer	2020-08-16	null	14	2024-03-15	Israel
10	Chuck Schumer	2020-08-30	null	14	2024-03-15	Israel
11	Chuck Schumer	2020-11-01	null	14	2024-03-15	Israel

Щоб перейти до написання запиту, потрібно справа від таблиці обрати **Query (1)** або в меню Query обрати **In new tab (2)**.

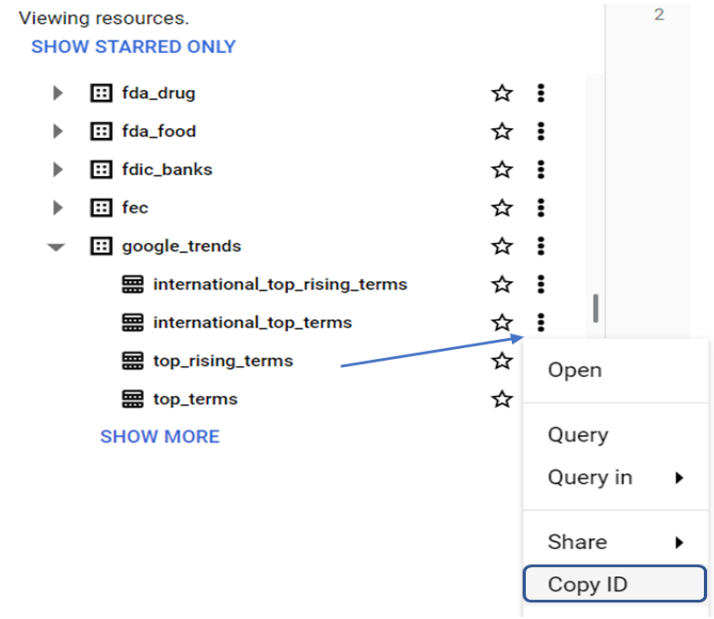


У BigQuery, коли створюється новий запит, після SELECT не вказано перелік колонок. Користувач має визначити, які саме стовпці він хоче вибрати. Цей підхід сприяє точнішому та більш цілеспрямованому формулюванню запитів, особливо у випадку роботи з великими датасетами, де вибір усіх стовпців за допомогою зірочки * може знизити ефективність обробки та спричинити надмірне навантаження на систему.

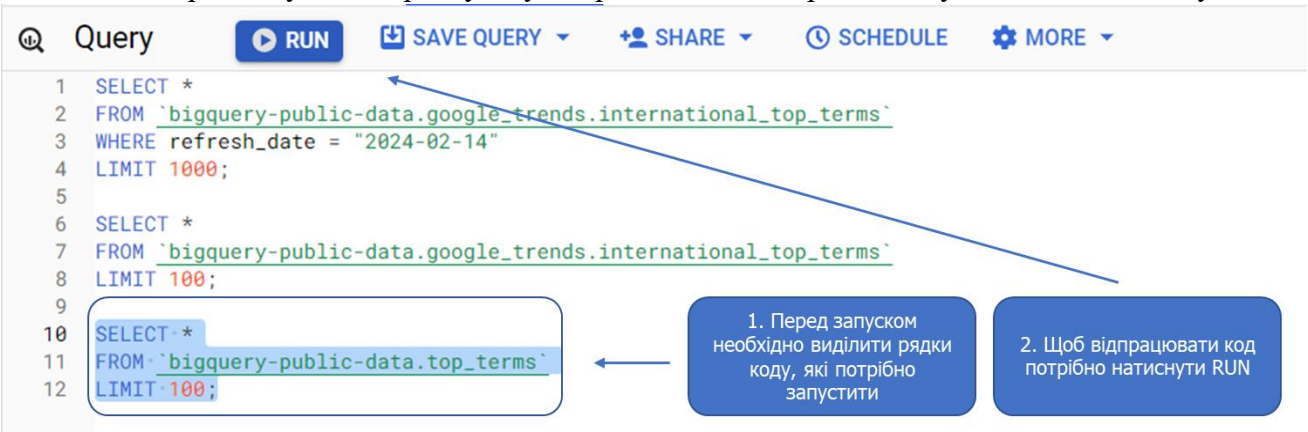


Також запит не містить роздільника в кінці ";". У SQL, символ ";" використовується як роздільник між окремими SQL-запитами. Відсутність або присутність ";" не впливає на результат виконання SQL-запиту. Але він може бути важливим, коли вводити кілька SQL-запитів підряд. У такому випадку ";" використовується для розділення цих запитів і повідомлення базі даних про кінець кожного окремого запиту.

Для зручності назву таблиці можна скопіювати через Copy ID. Потім можете вставити назву (після FROM) за допомогою комбінації клавіш "Ctrl + V" на Windows або "Command (⌘) + V" на Mac.



Для того, щоб відпрацювати (запустити) код, потрібно натиснути RUN. У BigQuery можна запустити запит також за допомогою швидкої комбінації клавіш "Ctrl + Enter" або "Command (⌘) + Enter" на macOS. Перед запуском користувачу потрібно виділити рядки коду, які він бажає запустити:



За потреби рядки коду можна закоментувати. Це означає, що при запуску коду рядок, який є закоментований не буде впливати на результат виконання запиту.

Щоб закоментувати один рядок потрібно на початку рядка прописати дві рисочки --, або виділити рядок та скористатись комбінацією "Ctrl + /" на Windows або "Command (⌘) + /" на MacOS

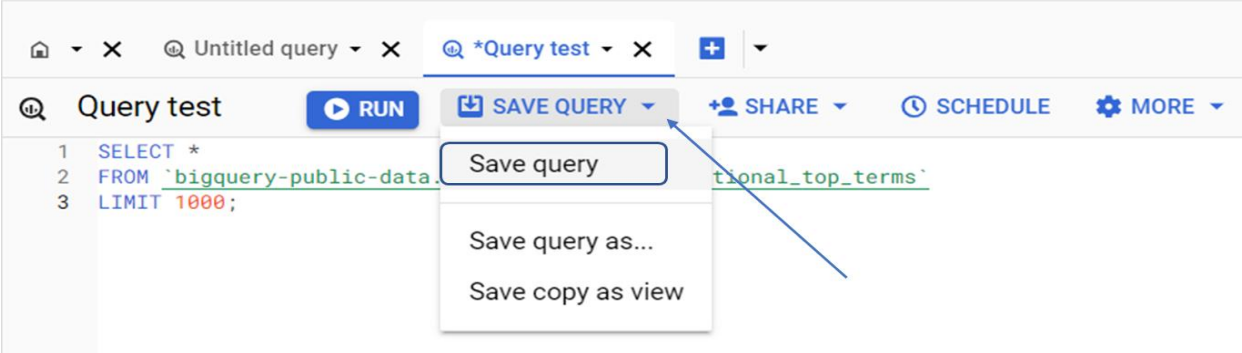
```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
WHERE country_name = "Ukraine"
--AND region_name = "Lviv Oblast"
AND refresh_date = "2024-04-09"
LIMIT 1000;
```

Щоб закоментувати більше рядків чи вставляти великі текстові пояснення, можна скористатися символами /* код */

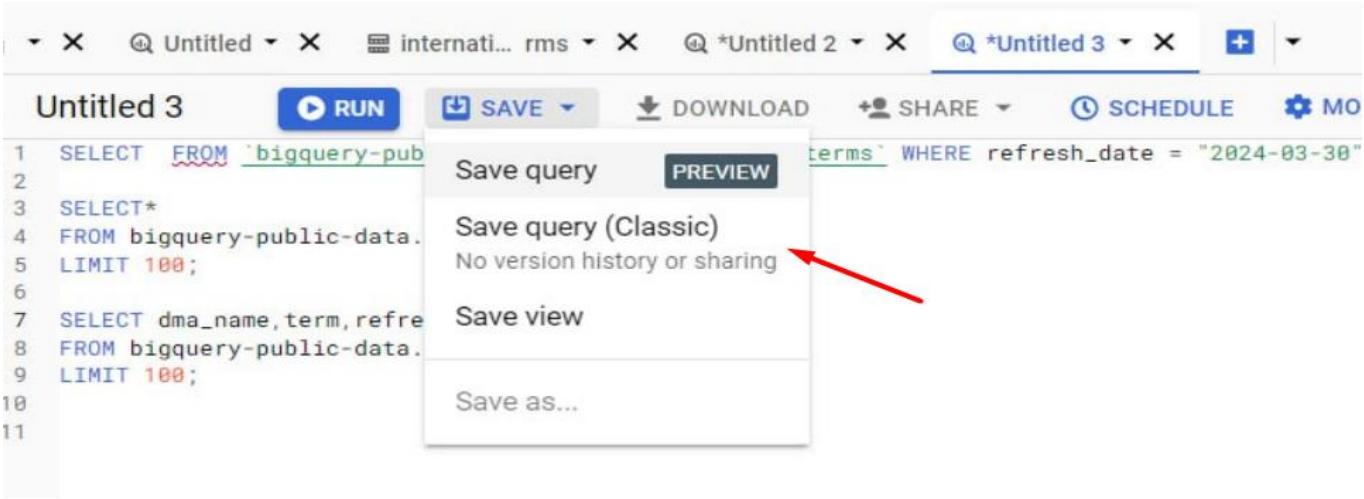
```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
WHERE country_name = "Ukraine"
/*AND region_name = "Lviv Oblast"
AND refresh_date = "2024-04-09"*/
```

LIMIT 1000;

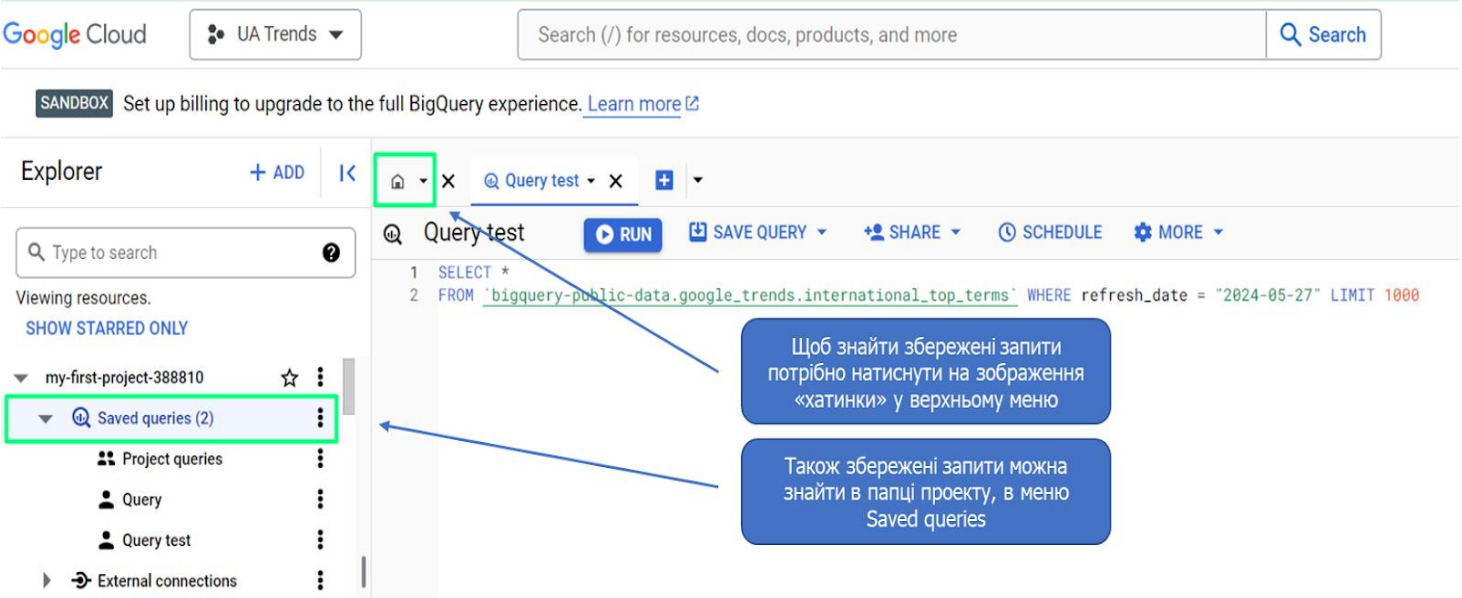
Для того, щоб зберегти запити, використовуйте кнопку **"Save query"**.



Зверніть увагу, що у деяких користувачів може бути присутня кнопка **"Save query (Classic)"**. У такому випадку скористайтеся нею для збереження вашого запиту, це пов'язано з поступовим поліпшенням інтерфейсу в BigQuery.



Щоб знайти збережені запити, потрібно натиснути на зображення «хатинки» у верхньому меню. Всі збережені запити знаходяться в папці проєкту, в меню **Saved queries**.



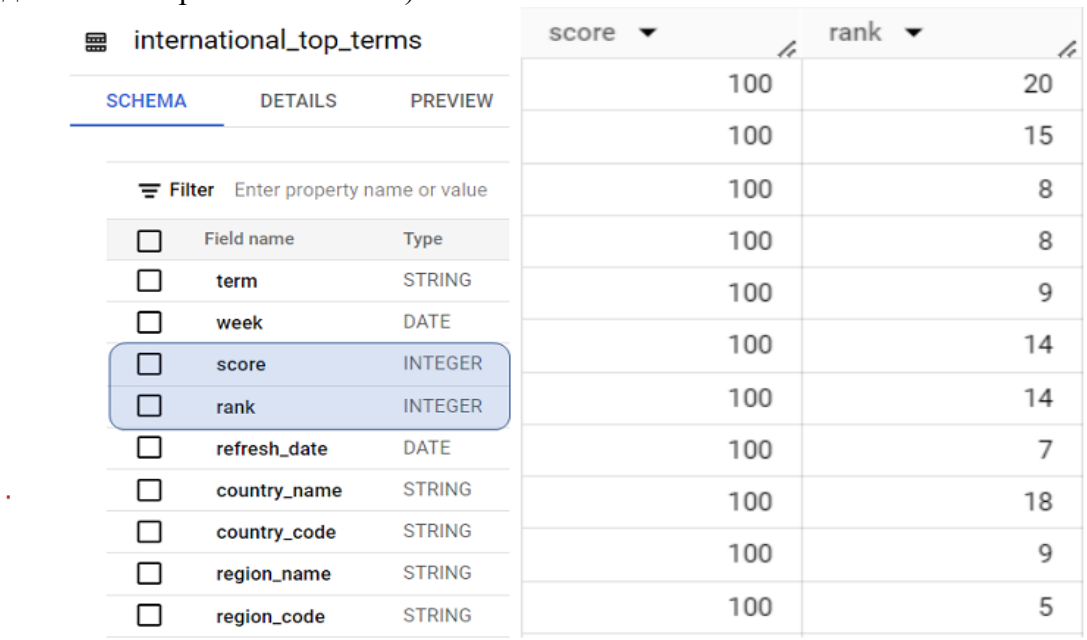
3.3. Основні категорії типів даних, які підтримуються в BigQuery

Кожна СУБД (система управління базами даних) має свої особливості та додаткові типи даних, але основні 3 категорії залишаються схожими:

Числові (Numeric types) типи даних

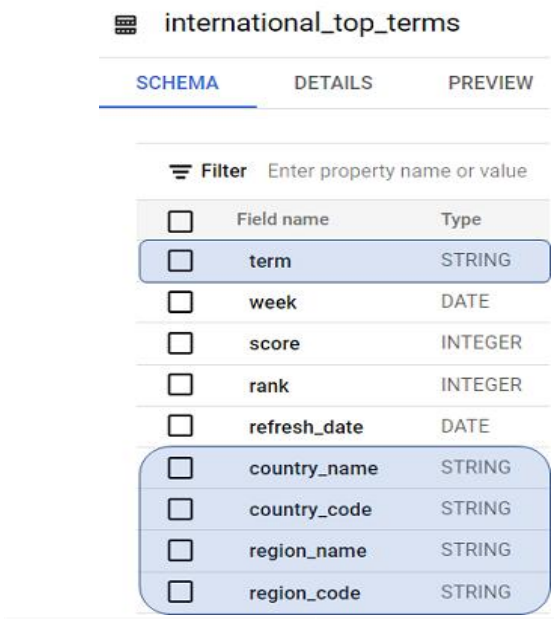
Дозволяють ефективно зберігати і обробляти числові значення, враховуючи вимоги до точності та розміру даних. Числові типи даних включать 3 основних типи:

- *Integer types* (зберігання цілих чисел, які не мають дробової частини). INTEGER є основним і широко використовуваним типом для зберігання цілих чисел у базах даних.
- *Decimal types* (зберігання чисел з фіксованою комою. Ці типи даних особливо важливі для фінансових і комерційних додатків, де необхідна висока точність числових значень, наприклад, для грошових обчислень).
- *Floating point types* (зберігання чисел, що можуть мати дуже широкий діапазон значень з дробовою частиною. Ці типи даних корисні для наукових обчислень, де потрібна підтримка чисел з великим діапазоном і різною точністю).



Символьні/Текстові (String types) типи даних

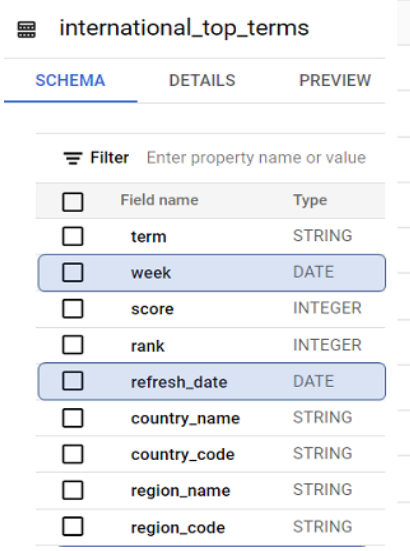
Зберігають текстові значення у базах даних. Ці типи даних дозволяють зберігати як короткі рядки (наприклад, імена), так і довгі тексти (наприклад, пошуковий запит).



term	country_name	country_code	region_name	region_code
ผลบอลแมนยู	Thailand	TH	Pathum Thani	TH-13
PPP	Indonesia	ID	Special Region of Yogyakarta	ID-YO
Will Smith	Indonesia	ID	East Java	ID-JI
Will Smith	Indonesia	ID	Central Java	ID-JT
SkyBox	Denmark	DK	Region Syddanmark	DK-83
พยากรณ์อากาศ	Thailand	TH	Lopburi	TH-16
Swiss Open 2024	Indonesia	ID	Central Sulawesi	ID-ST
גוש עציון	Israel	IL	Jerusalem District	IL-JM
Swiss Open 2024	Indonesia	ID	East Nusa Tenggara	ID-NT
איתן אורנשטיין	Israel	IL	South District	IL-D
ผลบอลแมนยู	Thailand	TH	Phang-nga	TH-82

Дата та час (Data and time types)

Зберігають часові дані, як-от дати народження, дати замовлень, часові мітки та інші часові інтервали.



week	refresh_date
2019-07-21	2024-03-06
2024-03-03	2024-03-06
2020-11-01	2024-03-18
2024-03-03	2024-03-06
2023-12-31	2024-03-06
2023-08-13	2024-03-18
2019-09-01	2024-03-06
2020-02-23	2024-03-18
2022-09-25	2024-03-06
2021-07-25	2024-03-18
2021-10-10	2024-03-06

3.4. База даних Google Trends – International

База даних Google Trends – International є публічним набором даних, який доступний для безкоштовного використання у вільному доступі. База даних містить 4 таблиці:

1. international_top_rising_terms;

2. international_top_terms;
3. top_rising_terms;
4. top_terms.

Таблиці ***international_top_rising_terms*** та ***international_top_terms*** містять дані пошукових запитів понад 40 країн. У межах країн є ідентифікація термінів за регіонами.

Детальніше про сутність колонок таблиці **international_top_terms**:

- **term** – пошуковий запит Google, який потрапив у ТОП-25 популярних запитів;
- **week** – містить дату першого дня тижня для кожного запису. Її застосовують для фільтрації та аналізу даних по тижнях, щоб порівнювати популярність запитів за певний період;
- **score** – бали, вони не всюди є. Якщо термін має бал, то він відображає популярність терміну на певному тижні відповідно до інших тижнів. Score має діапазон від 0 до 100. Термін зі score 100 означає, що він найбільш популярний. Score показує, наскільки популярним був термін протягом поточної дати порівняно з іншими датами в тому ж часовому ряді для цього терміну (у таблиці доступно 260 тижнів, тобто 5 років);
- **rank** – рейтинг, який вказує, наскільки популярним є кожний термін у порівнянні з іншими термінами, які були додані в той самий день. Рейтинг має діапазон від 1 до 25. Найбільш популярний термін має рейтинг 1;
- **refresh_date** – це дата оновлення термінів, яка вказує, чи був термін популярним у цей день. Якщо потрібно знайти термін по всій таблиці, запит датою оновлення обмежувати не потрібно;
- **country_name** – назва країни;
- **country_code** – код країни за ISO 3166 Alpha-2 (стандарт Міжнародної організації зі стандартизації, для представлення країн, залежних територій і особливих географічних територій);
- **region_name** – назва регіону;
- **region_code** – код одиниць адміністративно-територіального поділу країн за ISO 3166-2 (стандарт Міжнародної організації зі стандартизації, для позначення геокодів).

Таблиці ***top_rising_terms*** та ***top_terms*** містять дані лише однієї країни (США) і тому не містять колонок з переліком країн (country_name) та регіонів (region_name), але містять **dma_id** та **dma_name**:

- **dma_id** - цифровий ідентифікатор, який використовується для позначення DMA (Designated Market Area (DMA) в США. DMA - це термін, що використовується в США для опису географічного регіону;
- **dma_name** – назва DMA.

Таблиці в базі даних Google Trends - International містять мільйони рядків даних за 5 років. Якщо запит не обмежений конкретними критеріями пошуку, дані будуть вибиратися випадковим чином кожен раз із зверненням до таблиць.

В описі до набору даних Google Trends – International вказано, що таблиця ***international_top_terms*** містить 25 щоденних пошукових запитів за країною, регіоном, тижнем та вказаною датою оновлення, які були популярними. ТОП-25 запитів у таблиці ***international_top_terms*** можна отримати за допомогою наступного запиту:

```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
WHERE country_name = "Switzerland"
AND region_name = "Geneva"
AND refresh_date = '2024-04-10'
AND week = '2024-04-07';
```

- Логіка набору даних Google Trends – International полягає в тому, що популярність запитів порівнюються з усіма доступними тижнями, які є в таблиці (260 тижнів, тобто 5 років). Тому аналізувати появу певного терміну потрібно за колонкою refresh_date, а не week.
- Якщо не вказати фільтр week, то результуюча таблиця міститиме порівняння ТОП-25 запитів на дату оновлення щодо всіх доступних тижнів, які є в таблиці.

Для сортування отриманих даних за популярністю можна скористатися командою ORDER BY rank у кінці запиту:

```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
WHERE country_name = "Switzerland"
AND region_name = "Geneva"
AND refresh_date = '2024-04-10'
AND week = '2024-04-07'
ORDER BY rank;
```

rank підсвічується синім через те, що ім'я цієї колонки співпадає з одним із ключових слів SQL-коду, як-от SELECT і FROM.

Дату останнього оновлення та тиждень у таблиці international_top_terms можна знайти за допомогою наступного запиту:

```
SELECT MAX(refresh_date) AS last_refresh_day, MAX(week) AS last_week,
FROM `bigquery-public-data.google_trends.international_top_terms`;
```

В описі до набору даних Google Trends – International вказано, що таблиця top_terms містить 25 щоденних пошукових запитів за географічним регіоном DMA, тижнем та вказаною датою оновлення, які були популярними. ТОП-25 запитів у таблиці top_terms можна отримати за допомогою наступного запиту:

```
SELECT *
FROM `bigquery-public-data.google_trends.top_terms`
WHERE dma_name = "Portland-Auburn ME"
AND refresh_date = '2024-05-25'
AND week = '2024-05-19'
ORDER BY rank;
```

Дату останнього оновлення та тиждень у таблиці top_terms можна знайти за допомогою наступного запиту:

```
SELECT MAX(refresh_date) AS last_refresh_day, MAX(week) AS last_week,
FROM `bigquery-public-data.google_trends.top_terms`;
```

Публічні бази даних у відкритому доступі, схожі на Google Trends - International, можна знайти на таких платформах, як Kaggle, GitHub (у репозиторіях даних), а також через відкриті дані від урядових сайтів або міжнародних організацій, які збирають і публікують статистичні дані.

4. Мова запитів SQL (Structured Query Language)

4.1. Основні категорії елементів SQL

Запити в реляційних базах даних формують за допомогою структурованої мови SQL. Абревіатура SQL:

S - Structured (Структурований)

Q - Query (Запит)

L - Language (Мова)

Тобто, SQL - це мова структурованих запитів. SQL також є декларативною мовою запитів. Тобто вона описує, що ми хочемо отримати, а не яким чином.

Зверніть увагу, SQL - це мова запитів, а не програмування. З його допомогою НЕ вийде написати застосунок або сайт. Це інструмент для взаємодії з даними.

BigQuery використовує власний діалект SQL - GoogleSQL, заснований на стандартному ANSI SQL*, але з додатковими розширеннями та особливостями, специфічними для обробки великих даних і

хмарних обчислень. Наприклад, BigQuery підтримує віконні функції, вбудовані функції для обробки JSON і ARRAY типів даних, а також спеціальні оптимізації для швидкого зчитування великих обсягів даних.

ANSI SQL — це стандартизована мова запитів до реляційних баз даних, розроблена і підтримувана Американським національним інститутом стандартів (ANSI). Метою створення стандарту було забезпечення єдиного синтаксису і семантики для роботи з базами даних незалежно від їх реалізації.

У мові SQL можна виділити кілька основних категорій елементів, які використовуються для створення запитів.

Ключові слова використовуються для визначення структурних елементів запиту. Основні ключові слова включають:

SELECT використовується для вибору даних, **FROM** вказує на таблицю, з якої будуть вибрані дані:

```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`;
```

SELECT DISTINCT повертає унікальні значення, які знаходяться в стовпці:

```
SELECT DISTINCT country_name
FROM `bigquery-public-data.google_trends.international_top_terms`;
```

WHERE використовується для фільтрації записів на основі певних умов:

```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
WHERE country_name = "Ukraine";
```

ORDER BY сортує результати за вказаними стовпцями:

```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
WHERE country_name = "Switzerland"
AND region_name = "Geneva"
AND refresh_date = '2024-04-10'
AND week = '2024-04-07'
ORDER BY rank;
```

ASC / DESC після назви колонки вказує напрямок сортування -**ASC** (за замовчуванням) сортує за зростанням, а **DESC** - за спаданням.

LIMIT у SQL використовується для обмеження кількості рядків, які повертає запит. Це дозволяє отримати лише перші N рядків з результатів запиту, де N - це число, яке ви вказуєте в команді **LIMIT**:

```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`s
LIMIT 10;
```

Різниця між **ORDER BY** та **LIMIT** полягає в тому, що **ORDER BY** використовується для впорядкування результатів запиту за певними стовпцями (за зростанням або спаданням), тоді як **LIMIT** обмежує кількість повернутих рядків. **ORDER BY** впливає на порядок виведення рядків, а **LIMIT** визначає кількість рядків, що повертається.

Є також інші ключові слова SQL, які використовують для отримання комплексних результатів:

- Оператор **JOIN** у SQL використовується для об'єднання рядків з двох або більше таблиць на основі пов'язаних між ними стовпців.

- **GROUP BY** дозволяє групувати рядки на основі значень у стовпцях, що є корисним для агрегатних обчислень.
- Оператор **HAVING** використовується для фільтрації груп, створених оператором **GROUP BY**, на основі певної умови. Він схожий на оператор **WHERE**, але застосовується до груп, а не до окремих рядків.

Логічні оператори

AND використовується в **WHERE** для об'єднання декількох умов. Повертає записи, де всі умови є істинними:

```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
WHERE country_name = "Ukraine"
AND region_name = "Lviv Oblast";
```

За потреби ви можете додавати ще умови через **AND**:

```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
WHERE country_name = "Ukraine"
AND region_name = "Lviv Oblast"
AND refresh_date = "2024-04-10";
```

OR використовують для створення запитів з різними умовами. Наприклад, коли потрібно шукати дані, які відповідають принаймні одній з декількох умов:

```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
WHERE country_name = "France" OR country_name = "Spain";
```

цей запит поверне всі дані з таблиці, де країна є або "France", або "Spain". Якщо в цьому прикладі змінити оператор OR на AND, запит не поверне жодних результатів. Коли ви використовуєте оператор AND між умовами одного стовпця, обидві умови повинні бути правдивими одночасно. Оскільки неможливо, щоб одне значення country_name було одночасно "France" і "Spain", запит з AND не поверне жодних результатів.

Інші оператори

IN дає змогу вибрати рядки з таблиці, які містять одне або кілька значень, зазначених у списку:

```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
WHERE country_name = "Ukraine"
AND region_name IN ("Chernihivs'ka oblast", "Kyiv city");
```

Цей запит SQL поверне всі рядки з таблиці, де значення стовпця country_name дорівнює "Ukraine", і значення стовпця region_name знаходиться в списку, який включає "Chernihivs'ka oblast" або "Kyiv city".

LIKE використовується для пошуку даних за частковими або шаблонними значеннями.

оператор LIKE в SQL зазвичай використовується з типом даних STRING для порівняння текстових рядків.

У поєднанні з оператором **LIKE** часто використовуються два спеціальні символи підстановки:

- знак відсотка (**%**) позначає нуль, один або декілька символів;
- знак нижнього підкреслення (**_**) позначає пошук лише одного символа.

Практичні приклади використання спеціальних символів з LIKE:

Знаходження даних, які починаються на U:

```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
```

```
WHERE country_name LIKE "U%";
```

Знаходження даних, які закінчуються на a:

```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
WHERE country_name LIKE "%a";
```

Знаходження даних, які мають "ai" в будь-якій позиції:

```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
WHERE country_name LIKE "%ai%";
```

Знаходження даних, які мають "r" на другій позиції:

```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
WHERE term LIKE "_r%";
```

Також можна використовувати функцію **LOWER()** (конвертує рядок у нижній регістр "abcd") або **UPPER()** (навпаки, у верхній регістр "ABCD"). Це дозволить вам здійснювати пошук у таблиці, ігноруючи регістр символів.

```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
WHERE LOWER(term) LIKE '%song%';
```

Ви можете використовувати й інші пошукові комбінації, які будуть відповідати вашим потребам.

Дотримання правильного порядку написання запиту SQL є критично важливим для забезпечення коректного виконання коду та отримання очікуваних результатів. Хоча запити можуть виглядати по-різному залежно від складності, загальний порядок написання запиту SQL виглядає наступним чином:

1. **SELECT** - вибирає стовпці, які потрібно отримати з таблиці.
2. **FROM** - вказує таблицю з якої будуть вибрані дані.
3. **JOIN** - об'єднує таблиці на основі певних умов.
4. **WHERE** - фільтрує рядки на основі певних умов.
5. **GROUP BY** - групує рядки, що мають однакові значення в певних стовпцях, для агрегування.
6. **HAVING** - фільтрує групи, створені оператором GROUP BY, на основі певних умов.
7. **ORDER BY** - визначає порядок сортування результатів.
8. **LIMIT** - обмежує кількість рядків, які повертає запит.

4.2. Агрегатні функції в SQL

SQL підтримує агрегатні функції, що виконують обчислення над набором даних та повертають єдине результуюче значення. Вони широко використовуються для виконання різних типів аналізу даних:

- **COUNT()**: підраховує кількість рядків у наборі результатів.

```
SELECT COUNT(*)
FROM `bigquery-public-data.google_trends.international_top_terms`;
```

Цей запит поверне кількість рядків даних, які містить таблиця.

- **SUM()**: обчислює суму значень у стовпці.
- **AVG()**: обчислює середнє значення у стовпці.
- **MIN()**: повертає мінімальне значення у стовпці.

```
SELECT MIN(score)
FROM `bigquery-public-data.google_trends.international_top_terms`;
```

Цей запит поверне найменше значення, яке міститься в стовпці score.

- **MAX()**: повертає максимальне значення у стовпці.
  SELECT MAX(week), MAX(refresh_date)
  FROM `bigquery-public-data.google_trends.international_top_terms`;

  Цей запит повертає дату останнього тижня та дати оновлення даних.

4.3. Оператори порівняння

Для порівняння даних у SQL використовуються оператори порівняння, як-от: =, >, <, >=, <= Ось короткий опис кожного з них:

1. **= (дорівнює)** використовується для порівняння значень на рівність:
   SELECT *
   FROM `bigquery-public-data.google_trends.international_top_terms`
   WHERE rank = 1;

   SELECT *
   FROM `bigquery-public-data.google_trends.international_top_terms`
   WHERE refresh_date = '2024-04-10';

   SELECT *
   FROM `bigquery-public-data.google_trends.international_top_terms`
   WHERE country_name = "Ukraine";

2. **> (більше)** використовується для порівняння значень, щоб знайти ті, що більші за вказане значення:
   SELECT *
   FROM `bigquery-public-data.google_trends.international_top_terms`
   WHERE refresh_date > "2023-12-31";

3. **< (менше)** використовується для порівняння значень, щоб знайти ті, що менші за вказане значення.
   SELECT *
   FROM `bigquery-public-data.google_trends.international_top_terms`
   WHERE score < 50;

4. **>= (більше або дорівнює)** використовується для порівняння значень, щоб знайти ті, що більші або дорівнюють вказаному значенню.
5. **<= (менше або дорівнює)** використовується для порівняння значень, щоб знайти ті, що менші або дорівнюють вказаному значенню.

4.4. Робота з типами даних у SQL-коді

Під час написання запитів важливо враховувати типи даних стовпців, з яких ви вибираєте дані.

Робота з числовими типами даних

У запитах SQL числові дані записуються без подвійних або одинарних лапок, оскільки вони представляють собою прості числа або числові значення. Ви можете використовувати ці значення безпосередньо у вашому SQL-запиті.

SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
WHERE score = 100;

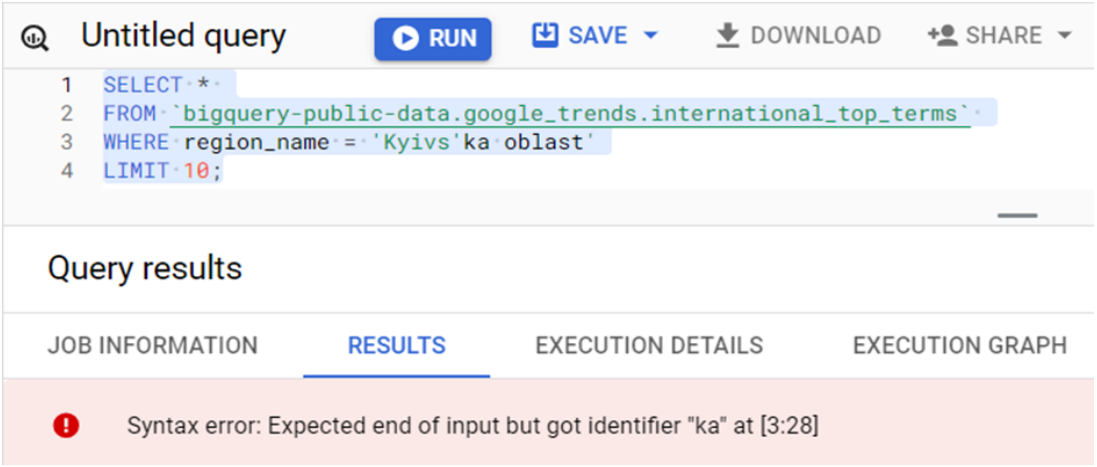
Робота з текстовими даними

У SQL текстові дані потрібно вводити в лапках, щоб база даних правильно їх інтерпретувала. Використання лапок надає інструкції SQL явної інформації про те, які саме дані є текстом, а не ключовим словом або іншим елементом мови SQL.

```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
WHERE country_name = "Ukraine";
```

За наявності в тексті не тільки одинарних лапок, а й подвійних, кожний такий символ у тексті потрібно окремо екранувати знаком похилої риски \' \"

Гарною практикою є використання саме подвійних лапок (") для текстових значень. За наявності у слові апострофа (наприклад, у region_name - "Kyivs'ka oblast") запит з одинарними лапками ('Kyivs'ka oblast') буде видавати помилку.



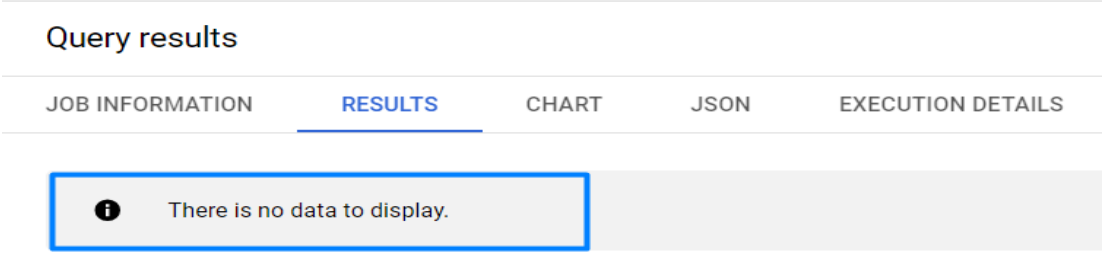
Робота з типом даних дати та часу

1. Дату потрібно вводити в лапках;
2. Важливо дотримуватися порядку написання дати у форматі 'YYYY-MM-DD', тобто на початку вказати рік, далі місяць та день через роздільник -

```
SELECT *
FROM `bigquery-public-data.google_trends.international_top_terms`
WHERE refresh_date = "2024-04-10";
```

4.5. Робота з помилками

Насамперед потрібно зазначити, що результатом виконання запиту не завжди є результуюча таблиця. Бувають випадки, коли за заданими критеріями пошуку в таблиці немає даних, які потрібні користувачу. В такому випадку BigQuery повідомляє: **«There is no data to display»**.

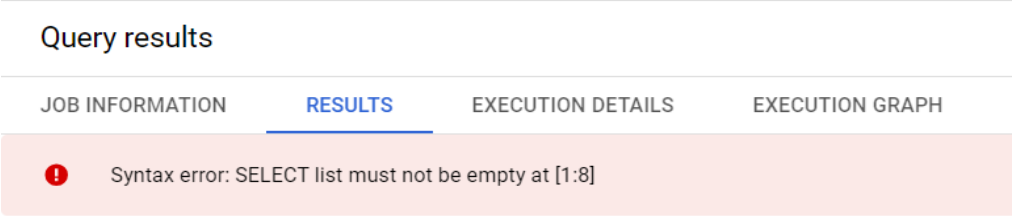


При виконанні запитів у BigQuery помилки зазвичай відображаються в консолі з детальним описом, що допомагає визначити причину проблеми. Повідомлення про помилки містять код помилки, текст помилки та інші діагностичні дані.

Найпоширенішою помилкою є **Syntax Error** (синтаксична помилка), причиною якої є некоректний SQL-синтаксис. У запиті нижче відсутній перелік колонок після SELECT:

```
SELECT FROM `bigquery-public-data.google_trends.international_top_terms`;
```

Користувач отримає помилку **Syntax Error** та коротку інформацію, що запит після SELECT не може бути порожнім. Також слід звертати увагу на цифри, які вказані в кінці помилки в квадратних дужках. Вони допомагають орієнтуватись, де саме виникає помилка. Перша цифра - це рядок запиту (що виділений), друга - це порядковий номер першого символа в цьому рядку, звідки починається помилка.



5. Візуалізація даних

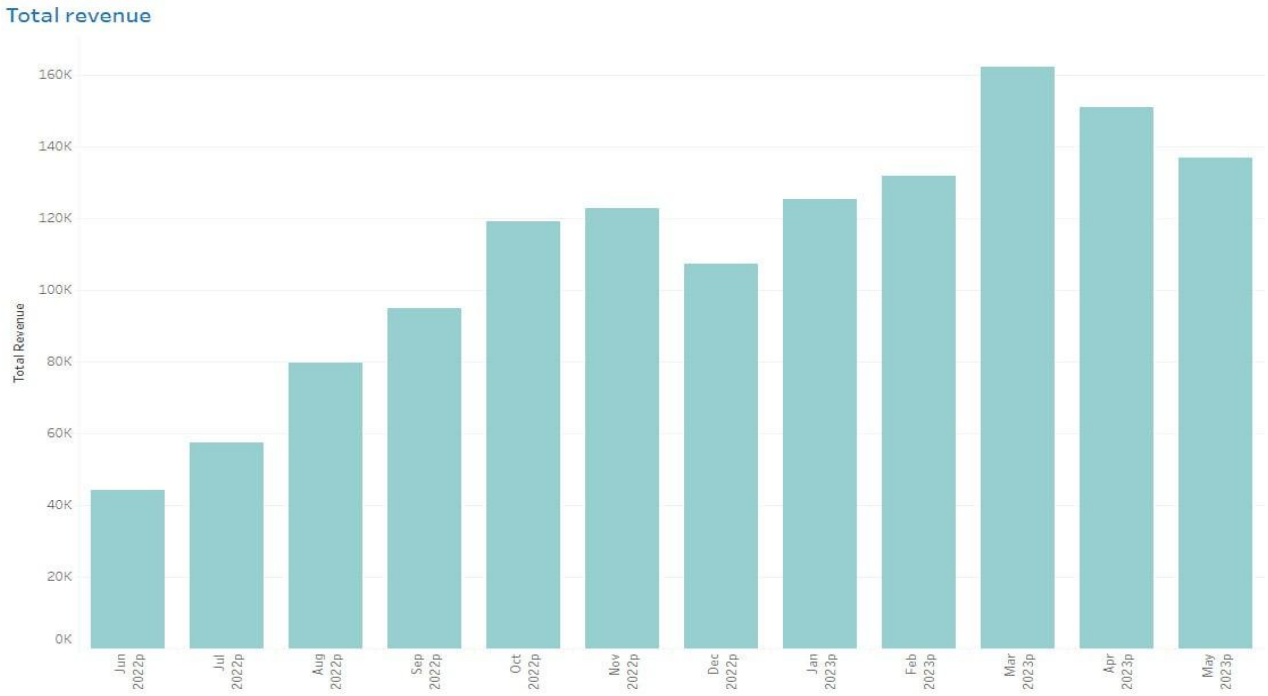
Що таке візуалізація даних?

Візуалізація даних - це процес представлення інформації у вигляді графічних елементів, таких як діаграми, графіки, графічні зображення та інші візуальні об'єкти. Метою візуалізації даних є наглядна інтерпретація і аналіз даних, сприяння розумінню складних зв'язків і закономірностей у даних та використання цієї інформації для прийняття рішень.

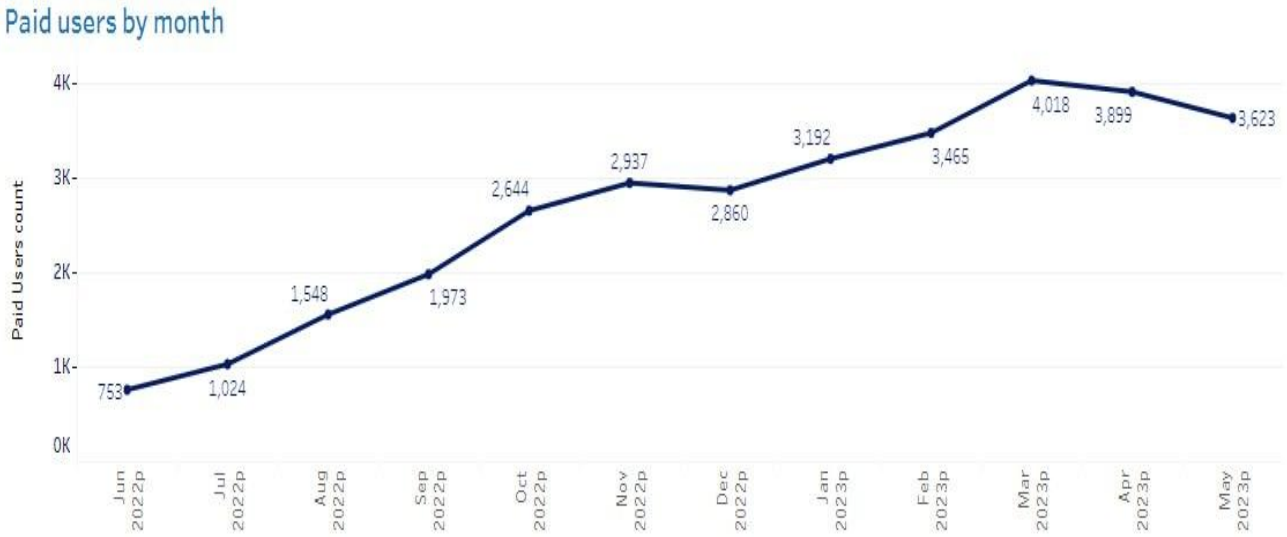
Основні принципи візуалізації даних:

1. **Простота** - візуалізація повинна бути простою і зрозумілою, щоб користувачі могли швидко й точно сприймати інформацію.
2. **Контекст** - важливо збалансувати кількість інформації на візуальному зображенні та забезпечити контекст для зрозуміння.
3. **Чіткість** - графічні елементи повинні бути виразними, а текст та маркери повинні легко читатись.
4. **Вірогідність** - візуалізація повинна точно відображати дані та уникати спотворень.
5. **Контрастність і кольори** - використання кольорів та контрасту може посилити важливість деяких елементів.

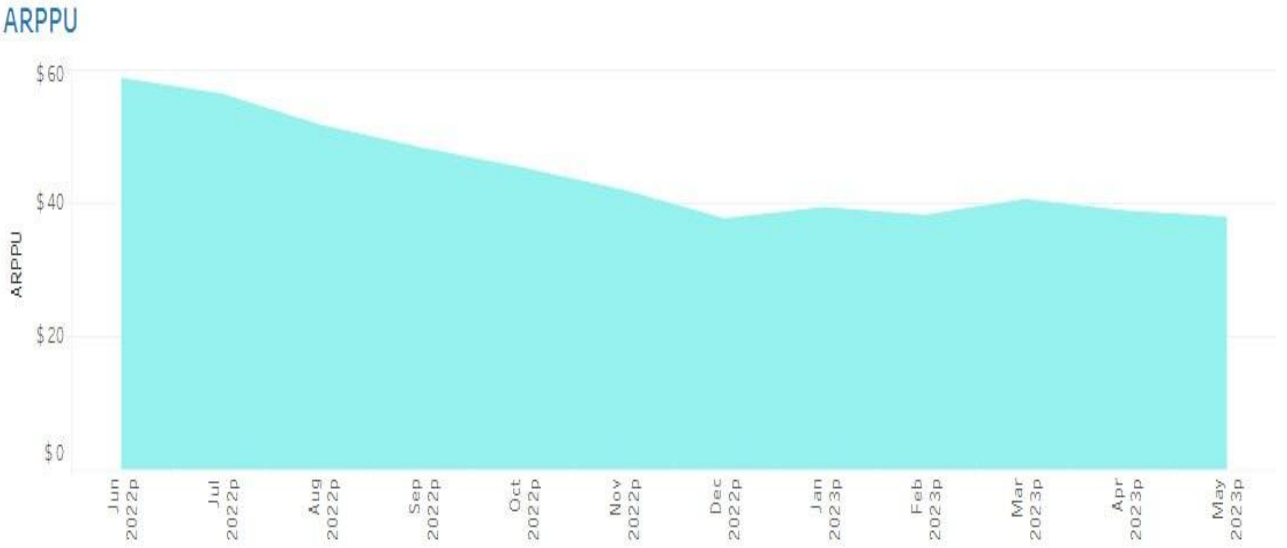
Розглянемо приклади найпоширеніших типів візуалізації даних



Стовпчикова діаграма (Bar Chart) - відображає категорії на одній осі, а значення на іншій. Використовується для порівняння категорій або показників.

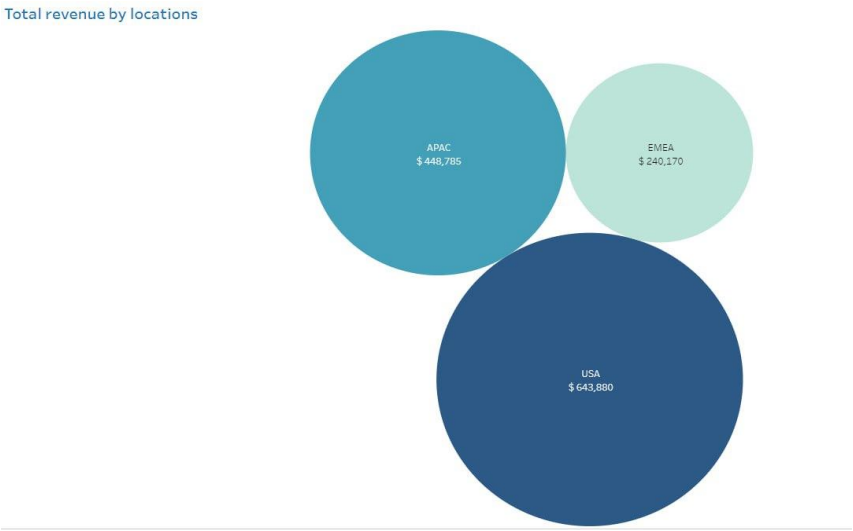


Лінійна діаграма (Line Chart) - відображає зміну значень у часі або інших послідовних періодах. Допомагає показати тренди, залежності та зміни величин.



Графік площин (Area Chart) - схожий на лінійну діаграму, але із зафарбованою зоною під лінією, що показує значення.

Теплова карта (Heat Map) - візуалізує значення змінної за допомогою кольорової шкали на 2D-сітці. Часто використовується для показу інтенсивності або кореляцій.



Бульбашкова діаграма (Packed Bubbles Chart) - відображає дані у вигляді кругів, які мають розмір та розташування, що відповідають значенням.

Графік розподілу (Distribution Plot)- відображає розподіл значень однієї або кількох змінних. Включає графік щільності, гістограму або ящик з вусами.

Окремо слід виділити дашборди, які дозволяють комбінувати різні типи візуалізацій на одному екрані, створюючи комплексний огляд даних та ключових метрик та віджети фільтрів у Tableau, які дозволяють користувачам взаємодіяти з даними та фільтрувати їх за певними критеріями, але про це поговоримо більше у наступних розділах.

Основні засоби візуалізації даних та особливості їх використання

Візуалізація часто використовується у таких галузях та формах:

1. Статистика та звітність – найчастіше для одночасної демонстрації даних за певний період.
2. Довідкова інформація – як додаток до основного тексту, що візуально демонструє те, про що йдеться в тексті.
3. Ілюстрації – дублюють або доповнюють те, про що йдеться в тексті.

Основні засоби візуалізації даних включають в себе різні інструменти та бібліотеки, які дозволяють створювати графічні представлення даних.

Розгляньмо інструменти для створення якісних та зрозумілих візуалізацій даних:

1. **Microsoft Excel або Google Sheets** має вбудовані інструменти для створення різних видів графіків і діаграм, включаючи стовпчасті, кругові, лінійні діаграми тощо.

Підходить для швидкого відображення простих даних, але не найкращий вибір для складних та великих наборів даних.

Особливості: дозволяє працювати у знайомому і зрозумілому інтерфейсі, без додаткового ознайомлення; не потребує жодних навичок програмування чи використання баз даних.

2. **Tableau** - це потужний інструмент для візуалізації даних з великою кількістю можливостей, включаючи можливість створення інтерактивних графіків і дашбордів.

Використовується для великих обсягів даних та для створення складних візуалізацій,особливо в бізнес-аналітиці.

Особливості: можна працювати зі структурованими та неструктурованими даними; допоможе вам вирішити великі проблеми, пов'язані з даними; можливість інтегруватися з великою кількістю додатків.

3. **Power BI** - інструмент для бізнес-аналітики, розроблений Microsoft. Він дозволяє створювати дашборди, звіти і візуалізації даних для бізнес-аналізу.

Ідеальний для підприємств, які хочуть відстежувати та аналізувати свої дані.

Особливості: дозволяє працювати з дуже великими джерелами даних; забезпечує безпечне та надійне підключення до джерел даних, як у хмарі, так і локально; простий у використанні дизайн.

4. **Google Looker Studio** - програма, що допоможе компанії робити інтерактивні та динамічні панелі, використовувати інтегровану аналітику.

Використання цього інструменту дає змогу ділитися з візуалізаціями з багатьма стейкхолдерами, навіть у режимі реального часу.

Особливості: для ілюстрації даних за допомогою проміжних підсумків можна використовувати таблиці; окремі групи можна динамічно фільтрувати; має панель інструментів для поглибленого аналізу даних.

Джерело: https://textbook.edu.goit.global/lms-data-analyst-marathons-v2-textbook/uk/docs/lesson-01/intro/