**Prompt ke ChatGPT**

tabel\_penjualan (1).csv

Spreadsheet

tabel\_produksi (1).csv

Spreadsheet

Saya sedang menggunakan BigQuery untuk menganalisis data penjualan dan produksi produk dari dua tabel yang telah saya unggah, yaitu tabel\_produksi dan tabel\_penjualan, yang ada dalam dataset Company di project saya bernama kevin-angga-ramadhan. Saya ingin mencari id\_produk dengan jumlah penjualan tertinggi selama 1 bulan pertama tahun 2023, yaitu dari tanggal 1 februari hingga 20 Februari 2023. Kriteria lainnya adalah id\_produksi nya bernomor genap. Catatan untuk membantu anda mengerjakan: Kolom kode\_produk, id\_produksi, dan tanggal produksi ada di tabel\_produksi, Kolom jumlah\_terjual ada di tabel\_penjualan Untuk cari penjualan terbesar: jumlah\_terjual nya bisa diurutkan saja dari yang terbesar ke terkecil Kolom yang dipakai hanya id\_produk, id\_produksi, tanggal\_produksi, dan jumlah\_terjual saja Untuk menghubungkan 2 tabel tersebut menggunakan foreign key, foreign key nya yaitu kolom id\_produk Dapatkah Anda memberikan query SQL khusus BigQuery untuk menjalankan analisis ini?

**Jawaban ChatGPT**

Untuk melakukan analisis yang Anda inginkan, pertama-tama kita perlu memuat data dari kedua file CSV yang telah Anda unggah ke dalam BigQuery. Namun, sebelum memberikan query SQL, saya akan membuat analisis terlebih dahulu mengenai struktur tabel berdasarkan data yang ada di file CSV tersebut. Saya akan membaca file-file tersebut untuk memahami kolom-kolom yang relevan.

Mari kita mulai dengan memeriksa kedua file tersebut.

Berdasarkan data yang ada pada kedua tabel:

1. Tabel Penjualan memiliki kolom-kolom berikut:
   * id\_penjualan: ID transaksi penjualan.
   * kode\_produk: Kode produk yang terjual.
   * nama\_produk: Nama produk yang terjual.
   * tanggal\_penjualan: Tanggal terjadinya penjualan.
   * jumlah\_terjual: Jumlah produk yang terjual.
   * Kolom-kolom lainnya yang tidak diperlukan untuk analisis.
2. Tabel Produksi memiliki kolom-kolom berikut:
   * id\_produksi: ID produksi.
   * kode\_produk: Kode produk yang diproduksi.
   * nama\_produk: Nama produk yang diproduksi.
   * tanggal\_produksi: Tanggal produksi produk.
   * jumlah\_produksi: Jumlah produk yang diproduksi.
   * Kolom-kolom lainnya yang tidak diperlukan untuk analisis.

Langkah-langkah untuk query SQL:

* Gabungkan kedua tabel menggunakan kode\_produk sebagai foreign key.
* Ambil data penjualan antara tanggal 1 Februari 2023 hingga 20 Februari 2023.
* Filter id\_produksi yang bernomor genap.
* Pilih dan urutkan berdasarkan jumlah\_terjual tertinggi.

Berikut adalah query SQL yang sesuai dengan analisis tersebut:

sql

Copy code

SELECT

p.id\_produk,

pr.id\_produksi,

SUM(jumlah\_terjual) AS total\_terjual

FROM

`kevin-angga-ramadhan.Company.tabel\_penjualan` j

JOIN

`kevin-angga-ramadhan.Company.tabel\_produksi` pr

ON j.kode\_produk = pr.kode\_produk

WHERE

j.tanggal\_penjualan BETWEEN '2023-02-01' AND '2023-02-20'

AND MOD(pr.id\_produksi, 2) = 0

GROUP BY

p.id\_produk, pr.id\_produksi

ORDER BY

total\_terjual DESC

* MOD(pr.id\_produksi, 2) = 0 digunakan untuk memfilter ID produksi yang bernomor genap.
* SUM(jumlah\_terjual) digunakan untuk menjumlahkan penjualan berdasarkan produk dan ID produksi.
* ORDER BY total\_terjual DESC mengurutkan berdasarkan jumlah penjualan tertinggi.

Pastikan Anda menyesuaikan dataset dan project ID sesuai dengan kebutuhan Anda di BigQuery. ​

**Prompt ke ChatGPT**

urutkan berdasarkan data penjualan tertinggi

**Jawaban ChatGPT**

Untuk mengurutkan hasil berdasarkan penjualan tertinggi, kita akan menambahkan ORDER BY total\_terjual DESC pada query SQL, yang sudah saya sertakan sebelumnya. Namun, saya akan menyesuaikan query Anda untuk memastikan data diurutkan berdasarkan total\_terjual yang tertinggi. Berikut adalah query yang sudah diperbaiki:

sql

Copy code

SELECT

j.id\_produk,

pr.id\_produksi,

SUM(j.jumlah\_terjual) AS total\_terjual

FROM

`kevin-angga-ramadhan.Company.tabel\_penjualan` j

JOIN

`kevin-angga-ramadhan.Company.tabel\_produksi` pr

ON j.kode\_produk = pr.kode\_produk

WHERE

j.tanggal\_penjualan BETWEEN '2023-02-01' AND '2023-02-20'

AND MOD(pr.id\_produksi, 2) = 0

GROUP BY

j.id\_produk, pr.id\_produksi

ORDER BY

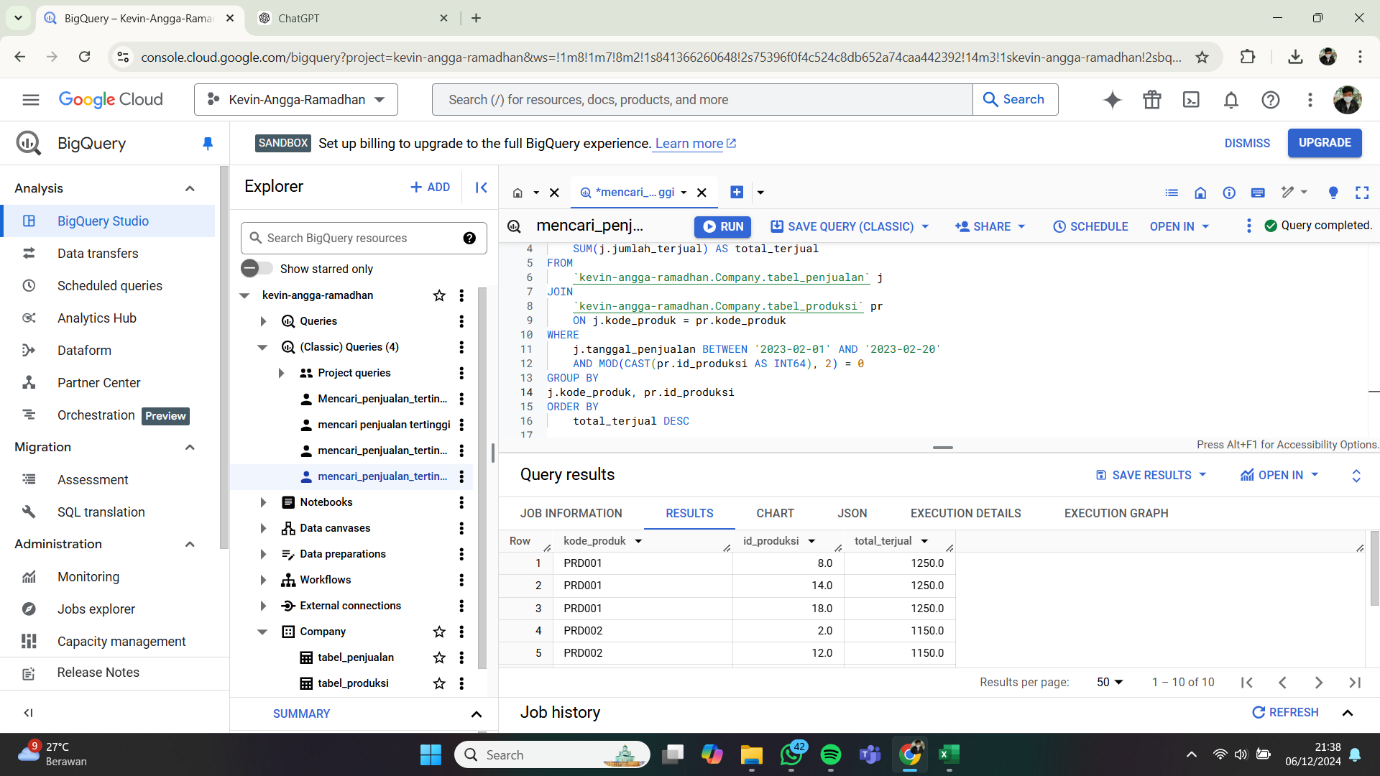
total\_terjual DESC

Dengan query ini:

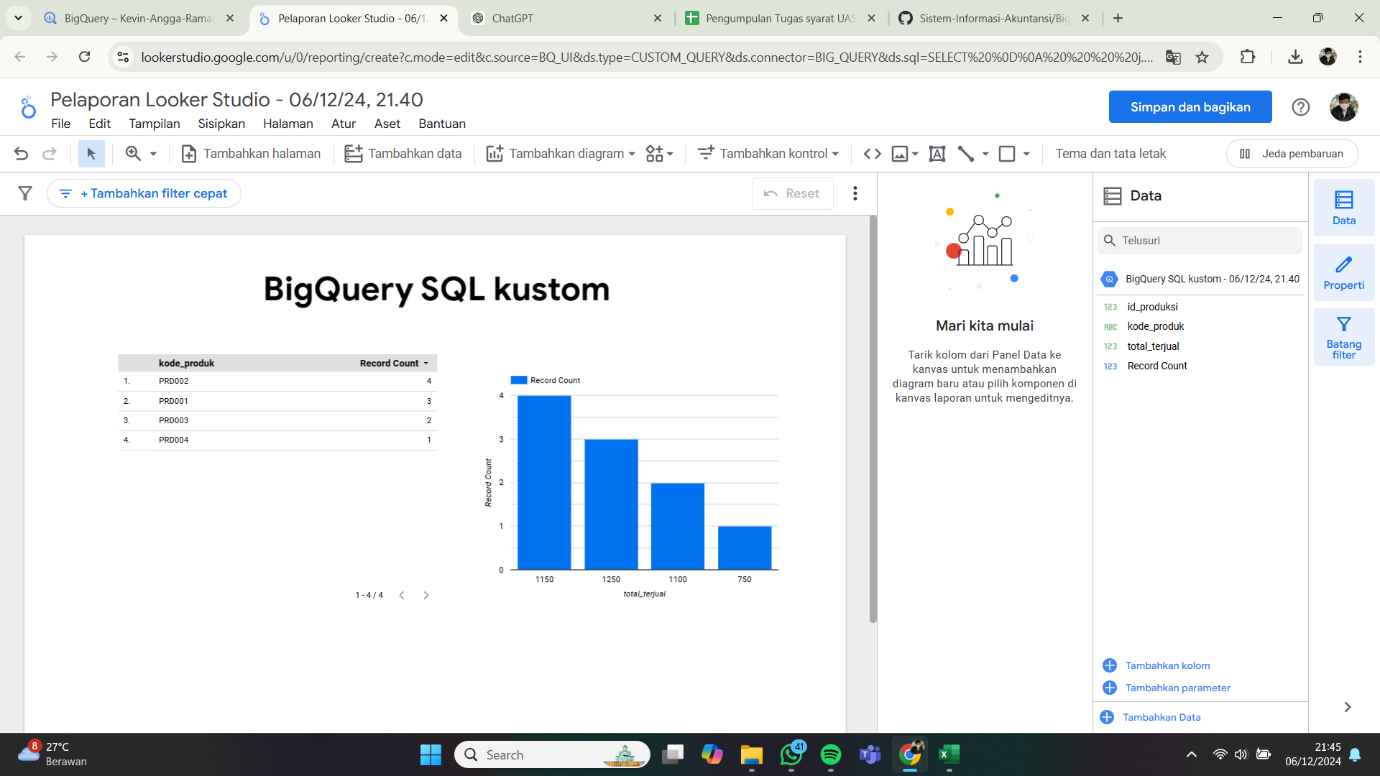
* SUM(j.jumlah\_terjual) menghitung total penjualan berdasarkan produk dan ID produksi.
* ORDER BY total\_terjual DESC memastikan bahwa hasil yang ditampilkan diurutkan dari penjualan tertinggi ke terendah.

Jadi, produk dengan jumlah penjualan tertinggi akan muncul di urutan pertama**.**

SETELAH MENGALAMI BEBERAPA KALI KESALAHAN, KESALAHAN TERDAPAT PADA **ID\_PRODUK** YANG SEHARUSNYA MENJADI **KODE\_PRODUK**

****

**Open in Locker Studio**

****