Nama: Zhafira Hasifaturrahmah

NIM: 40011423650271

buatlah excel 20 database perusahaan berupa tabel produksi, persediaan, penjualan, lengkap dengan struktur tabelnya.

**ChatGPT said:**

Berikut adalah langkah-langkah yang bisa saya lakukan untuk Anda:

1. **Membuat File Excel** yang berisi 20 set data perusahaan.
2. **Setiap Perusahaan** akan memiliki 3 sheet (Produksi, Persediaan, Penjualan).
3. **Struktur Tabel** di setiap sheet akan seragam.

Saya akan membuat file Excel sesuai dengan permintaan Anda sekarang. Harap tunggu sejenak!

File Excel dengan 20 set database perusahaan telah berhasil dibuat! Setiap perusahaan memiliki tiga tabel: **Produksi**, **Persediaan**, dan **Penjualan**.

Anda dapat mengunduh file tersebut melalui link berikut:

**Unduh File Excel** ​​

**You said:**

buat tabel excel masing-masing dan primary key, dengan 20 baris

**ChatGPT said:**

Saya akan membuat file Excel yang berisi tiga tabel utama: **Produksi**, **Persediaan**, dan **Penjualan**. Setiap tabel akan memiliki **20 baris data** dan dilengkapi dengan **Primary Key** masing-masing.

Berikut struktur dari masing-masing tabel:

1. **Produksi**:
   * produksi\_id (Primary Key)
   * tanggal\_produksi
   * jumlah\_produksi
   * produk
   * biaya\_produksi
2. **Persediaan**:
   * persediaan\_id (Primary Key)
   * produk
   * jumlah\_stok
   * tanggal\_update
3. **Penjualan**:
   * penjualan\_id (Primary Key)
   * tanggal\_penjualan
   * produk
   * jumlah\_terjual
   * harga\_per\_unit
   * total\_penjualan (dihitung dari jumlah\_terjual × harga\_per\_unit)

Saya akan membuat file Excel dengan tiga sheet terpisah untuk setiap tabel. Tunggu sebentar, saya akan menyusun datanya sekarang.

File Excel telah berhasil dibuat dengan tiga sheet: **Produksi**, **Persediaan**, dan **Penjualan**, masing-masing berisi **20 baris data** dengan **Primary Key**.

Anda dapat mengunduh file tersebut melalui link berikut:

**Unduh File Excel** ​

cleaning data-data ini

**ChatGPT said:**

Baik, saya akan membersihkan data dari tiga file Excel yang Anda unggah:

1. **Persediaan\_id.xlsx**
2. **Penjualan\_id.xlsx**
3. **Produksi\_id.xlsx**

Saya akan melakukan pembersihan data dengan langkah-langkah umum berikut:

* Menghapus duplikat jika ada.
* Mengatasi data yang hilang (missing values).
* Memastikan tipe data sesuai untuk setiap kolom.
* Menghapus atau memperbaiki nilai yang tidak valid.

Saya akan memproses file-file tersebut sekarang.

**Ringkasan Data dari File**

1. **Persediaan\_id.xlsx**:
   * Kolom: persediaan\_id, produk, jumlah\_stok, tanggal\_update
2. **Penjualan\_id.xlsx**:
   * Kolom: penjualan\_id, tanggal\_penjualan, produk, jumlah\_terjual, harga\_per\_unit, total\_penjualan
3. **Produksi\_id.xlsx**:
   * Kolom: produksi\_id, tanggal\_produksi, jumlah\_produksi, produk, biaya\_produksi

**Langkah Pembersihan Data yang Akan Dilakukan**

1. **Menghapus duplikat**.
2. **Mengatasi missing values**.
3. **Memastikan tipe data konsisten**.
4. **Menghapus nilai yang tidak valid atau aneh**.

Saya akan memproses pembersihan sekarang. ​​

It seems like I can’t do more advanced data analysis right now. Please try again later. If you'd like, I can guide you on how to clean these files manually or provide steps to do it in Excel or Python. Let me know how you'd like to proceed! ​

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 perusahaan di project saya bernama perusahaan\_id. Saya ingin mencari id\_produk dengan jumlah penjualan terendah selama bulan Juni tahun 2024, yaitu dari tanggal 1 Juni hingga 30 Juni 2024.

**ChatGPT said:**

Untuk menganalisis data di **BigQuery** dan mencari id\_produk dengan jumlah penjualan terendah selama bulan Juni 2024 dari dua tabel, yaitu tabel\_produksi dan tabel\_penjualan, Anda bisa menggunakan query SQL berikut:

**Query BigQuery**

SELECT

p.id\_produk,

SUM(pj.jumlah\_terjual) AS total\_penjualan

FROM

`perusahaan\_id.perusahaan.tabel\_penjualan` pj

JOIN

`perusahaan\_id.perusahaan.tabel\_produksi` p

ON

pj.id\_produk = p.id\_produk

WHERE

DATE(pj.tanggal\_penjualan) BETWEEN '2024-06-01' AND '2024-06-30'

GROUP BY

p.id\_produk

ORDER BY

total\_penjualan ASC

LIMIT 1;

**Penjelasan Query**

1. **SELECT p.id\_produk, SUM(pj.jumlah\_terjual) AS total\_penjualan**
   * Memilih kolom id\_produk dari tabel produksi (p.id\_produk).
   * Menghitung total penjualan dengan menjumlahkan jumlah\_terjual dari tabel penjualan (pj.jumlah\_terjual).
2. **FROM perusahaan\_id.perusahaan.tabel\_penjualan pj**
   * Menggunakan tabel penjualan dengan alias pj.
3. **JOIN perusahaan\_id.perusahaan.tabel\_produksi p ON pj.id\_produk = p.id\_produk**
   * Melakukan **JOIN** antara tabel\_penjualan dan tabel\_produksi berdasarkan kolom id\_produk.
4. **WHERE DATE(pj.tanggal\_penjualan) BETWEEN '2024-06-01' AND '2024-06-30'**
   * Menyaring data penjualan hanya untuk bulan Juni 2024.
5. **GROUP BY p.id\_produk**
   * Mengelompokkan hasil berdasarkan id\_produk.
6. **ORDER BY total\_penjualan ASC**
   * Mengurutkan hasil berdasarkan jumlah penjualan terendah terlebih dahulu.
7. **LIMIT 1**
   * Mengambil hanya satu hasil, yaitu id\_produk dengan jumlah penjualan terendah.

**Catatan**

* Pastikan bahwa kolom id\_produk ada di kedua tabel (tabel\_penjualan dan tabel\_produksi).
* Pastikan bahwa kolom tanggal\_penjualan memiliki tipe data yang sesuai (DATE atau DATETIME).
* Ganti nama proyek, dataset, dan tabel sesuai dengan struktur data Anda jika berbeda.

Dengan query ini, Anda akan mendapatkan **id\_produk** dengan penjualan paling sedikit selama bulan Juni 2024.

**You said:**

Invalid project ID 'perusahaan\_id'. Project IDs must contain 6-63 lowercase letters, digits, or dashes. Some project IDs also include domain name separated by a colon. IDs must start with a letter and may not end with a dash. solusi

**ChatGPT said:**

Error tersebut terjadi karena perusahaan\_id tidak sesuai dengan format Project ID di BigQuery. **Project ID** di BigQuery memiliki aturan khusus:

* Panjang antara 6 hingga 63 karakter.
* Hanya boleh berisi huruf kecil, angka, atau tanda hubung (-).
* Harus diawali dengan huruf dan tidak boleh diakhiri dengan tanda hubung (-).

**Solusi**

1. **Pastikan Project ID Valid**: Periksa Project ID yang sebenarnya di Google Cloud Console. Contoh valid:
   * my-project-12345
   * data-analytics-2024
   * company-sales2024
2. **Sesuaikan Query** dengan Project ID yang benar. Misalnya, jika Project ID Anda adalah my-project-12345, maka query-nya menjadi:

SELECT

p.id\_produk,

SUM(pj.jumlah\_terjual) AS total\_penjualan

FROM

`my-project-12345.perusahaan.tabel\_penjualan` pj

JOIN

`my-project-12345.perusahaan.tabel\_produksi` p

ON

pj.id\_produk = p.id\_produk

WHERE

DATE(pj.tanggal\_penjualan) BETWEEN '2024-06-01' AND '2024-06-30'

GROUP BY

p.id\_produk

ORDER BY

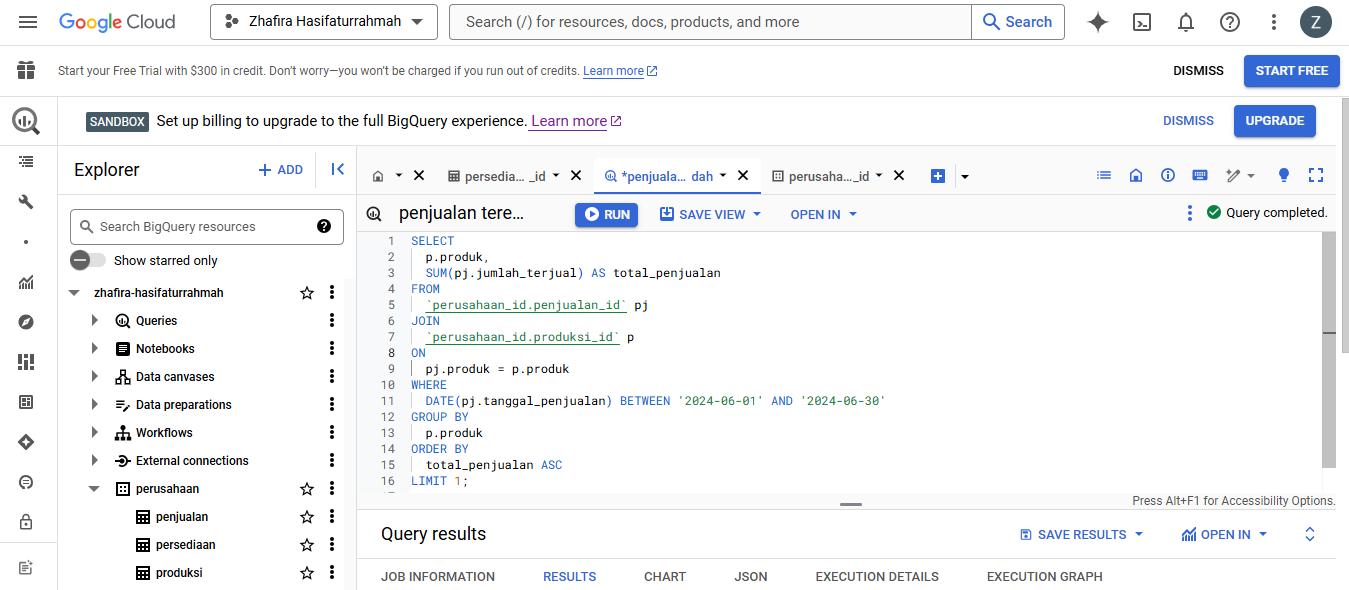
total\_penjualan ASC

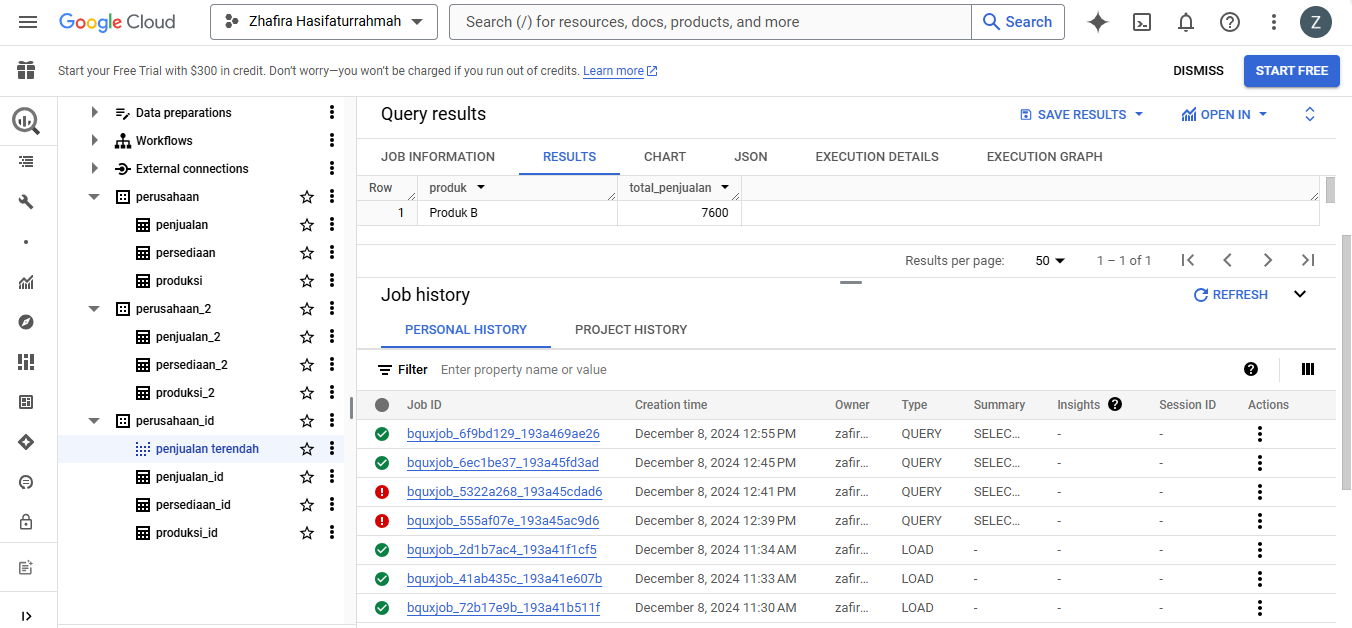
LIMIT 1;

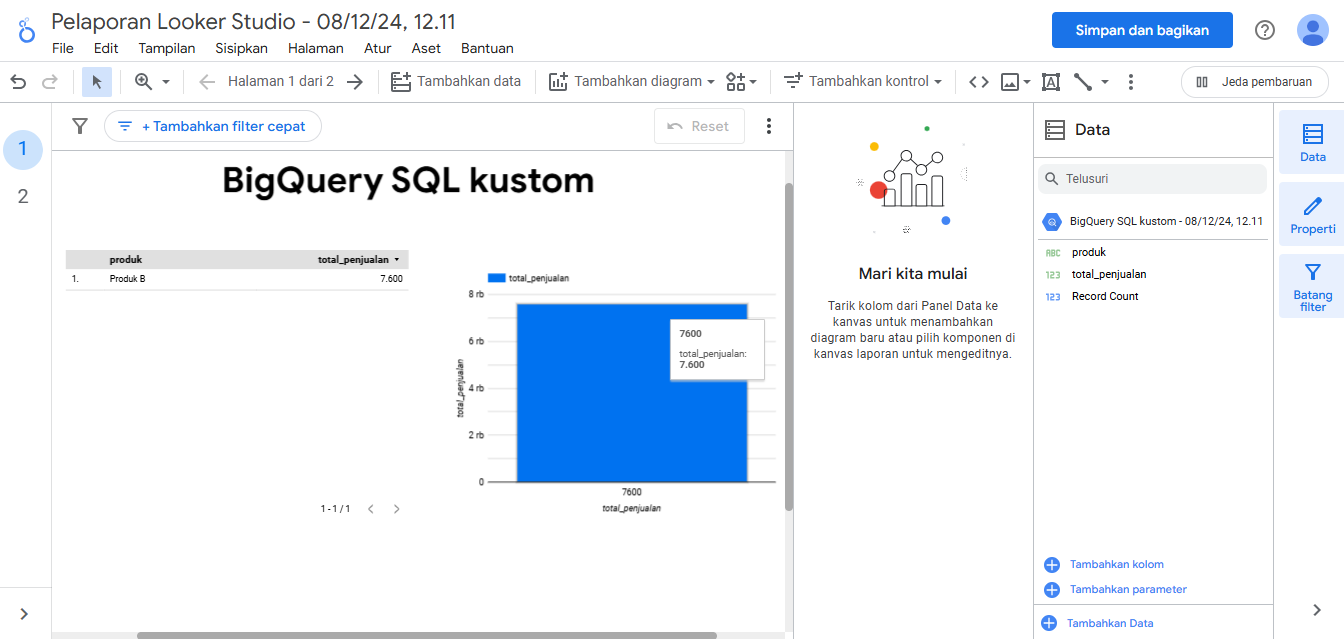
**Langkah Memastikan Project ID**

1. **Google Cloud Console**:
   * Buka Google Cloud Console.
   * Lihat Project ID di pojok kanan atas, di samping nama proyek.
2. **BigQuery UI**:
   * Buka BigQuery UI.
   * Periksa Project ID pada panel navigasi sebelah kiri.

Pastikan mengganti **my-project-12345** dengan Project ID Anda yang sebenarnya, kemudian jalankan kembali query tersebut.







buat analisis database

**ChatGPT said:**

Untuk melakukan **analisis database** yang melibatkan data produksi dan penjualan dari dua tabel dalam **BigQuery**, berikut langkah-langkah yang dapat Anda lakukan untuk menggali wawasan bisnis yang lebih mendalam.

**Langkah-langkah Analisis Database**

1. **Memahami Struktur Data**  
   Pastikan Anda memiliki pemahaman yang jelas tentang kolom yang ada dalam tabel tabel\_produksi dan tabel\_penjualan.
   * **Contoh Struktur Tabel Produksi**:
     + produksi\_id: ID unik produksi.
     + tanggal\_produksi: Tanggal produksi.
     + id\_produk: ID produk.
     + jumlah\_produksi: Jumlah unit yang diproduksi.
     + biaya\_produksi: Biaya produksi.
   * **Contoh Struktur Tabel Penjualan**:
     + penjualan\_id: ID unik penjualan.
     + tanggal\_penjualan: Tanggal penjualan.
     + id\_produk: ID produk.
     + jumlah\_terjual: Jumlah unit yang terjual.
     + harga\_per\_unit: Harga jual per unit.
     + total\_penjualan: Total pendapatan dari penjualan.
2. **Analisis yang Dapat Dilakukan**:
   * **Mengetahui produk dengan penjualan tertinggi dan terendah.**
   * **Mengukur efisiensi produksi dengan membandingkan jumlah produksi dan jumlah penjualan.**
   * **Mengidentifikasi tren penjualan bulanan atau harian.**
   * **Menghitung total pendapatan dan keuntungan berdasarkan penjualan.**
   * **Mengidentifikasi produk yang memiliki kelebihan atau kekurangan stok.**

**Contoh Analisis SQL di BigQuery**

**1. Produk dengan Penjualan Tertinggi dan Terendah Selama Juni 2024**

sql

SELECT

id\_produk,

SUM(jumlah\_terjual) AS total\_penjualan

FROM

`my-project-id.perusahaan.tabel\_penjualan`

WHERE

DATE(tanggal\_penjualan) BETWEEN '2024-06-01' AND '2024-06-30'

GROUP BY

id\_produk

ORDER BY

total\_penjualan DESC;

**2. Membandingkan Produksi dan Penjualan untuk Mengukur Efisiensi**

sql

SELECT

p.id\_produk,

SUM(p.jumlah\_produksi) AS total\_produksi,

COALESCE(SUM(s.jumlah\_terjual), 0) AS total\_penjualan,

(SUM(p.jumlah\_produksi) - COALESCE(SUM(s.jumlah\_terjual), 0)) AS selisih

FROM

`my-project-id.perusahaan.tabel\_produksi` p

LEFT JOIN

`my-project-id.perusahaan.tabel\_penjualan` s

ON

p.id\_produk = s.id\_produk

GROUP BY

p.id\_produk

ORDER BY

selisih DESC;

**Penjelasan**:

* **COALESCE** digunakan untuk mengganti nilai NULL dengan 0 jika tidak ada penjualan.
* **selisih** menunjukkan selisih antara produksi dan penjualan. Nilai positif berarti kelebihan stok, sedangkan nilai negatif berarti kekurangan stok.

**3. Total Pendapatan Per Produk**

sql

SELECT

id\_produk,

SUM(total\_penjualan) AS pendapatan\_total

FROM

`my-project-id.perusahaan.tabel\_penjualan`

GROUP BY

id\_produk

ORDER BY

pendapatan\_total DESC;

**4. Tren Penjualan Harian Selama Juni 2024**

sql

SELECT

DATE(tanggal\_penjualan) AS tanggal,

SUM(jumlah\_terjual) AS total\_penjualan

FROM

`my-project-id.perusahaan.tabel\_penjualan`

WHERE

DATE(tanggal\_penjualan) BETWEEN '2024-06-01' AND '2024-06-30'

GROUP BY

tanggal

ORDER BY

tanggal;

**5. Produk dengan Biaya Produksi Tertinggi**

sql

SELECT

id\_produk,

SUM(biaya\_produksi) AS total\_biaya\_produksi

FROM

`my-project-id.perusahaan.tabel\_produksi`

GROUP BY

id\_produk

ORDER BY

total\_biaya\_produksi DESC;

**Wawasan dari Analisis**

1. **Identifikasi Produk Berkinerja Tinggi**:  
   Mengetahui produk dengan penjualan tertinggi membantu merencanakan produksi agar sesuai dengan permintaan pasar.
2. **Mengelola Stok Lebih Efektif**:  
   Dengan membandingkan produksi dan penjualan, Anda dapat menghindari kelebihan atau kekurangan stok.
3. **Pengendalian Biaya Produksi**:  
   Melacak biaya produksi per produk membantu mengidentifikasi produk yang membutuhkan optimasi biaya.
4. **Menganalisis Tren Penjualan**:  
   Tren penjualan harian atau bulanan dapat membantu merencanakan kampanye pemasaran yang lebih efektif.

Dengan menggunakan langkah-langkah dan query ini, Anda dapat melakukan analisis yang komprehensif terhadap data penjualan dan produksi untuk mendukung pengambilan keputusan bisnis.

**You said:**

Syntax error: Unclosed identifier literal at [9:31]

**ChatGPT said:**

Error **"Syntax error: Unclosed identifier literal"** biasanya disebabkan oleh penggunaan tanda kutip atau backticks yang tidak sesuai di BigQuery. Mari kita perbaiki error tersebut dengan memastikan format dan tanda kutip sudah benar.

**Tips Perbaikan:**

1. **Pastikan Backticks untuk Referensi Tabel Lengkap**  
   Gunakan tanda **backtick (`)** untuk menyebutkan proyek, dataset, dan tabel secara lengkap:
   * Format: `project-id.dataset-name.table-name`
2. **Hindari Tanda Kutip yang Tidak Ditutup**  
   Pastikan semua tanda kutip (' atau ") dibuka dan ditutup dengan benar.
3. **Contoh Referensi Tabel yang Valid**:

sql

SELECT \* FROM `my-project-id.perusahaan.tabel\_penjualan`

**Contoh Query yang Telah Diperbaiki**

Berikut adalah contoh query yang diperbaiki untuk memastikan tidak ada kesalahan penulisan:

**1. Produk dengan Penjualan Tertinggi dan Terendah Selama Juni 2024**

sql

SELECT

id\_produk,

SUM(jumlah\_terjual) AS total\_penjualan

FROM

`my-project-id.perusahaan.tabel\_penjualan`

WHERE

DATE(tanggal\_penjualan) BETWEEN '2024-06-01' AND '2024-06-30'

GROUP BY

id\_produk

ORDER BY

total\_penjualan ASC

LIMIT 1;

**2. Membandingkan Produksi dan Penjualan**

sql

SELECT

p.id\_produk,

SUM(p.jumlah\_produksi) AS total\_produksi,

COALESCE(SUM(s.jumlah\_terjual), 0) AS total\_penjualan,

(SUM(p.jumlah\_produksi) - COALESCE(SUM(s.jumlah\_terjual), 0)) AS selisih

FROM

`my-project-id.perusahaan.tabel\_produksi` p

LEFT JOIN

`my-project-id.perusahaan.tabel\_penjualan` s

ON

p.id\_produk = s.id\_produk

GROUP BY

p.id\_produk

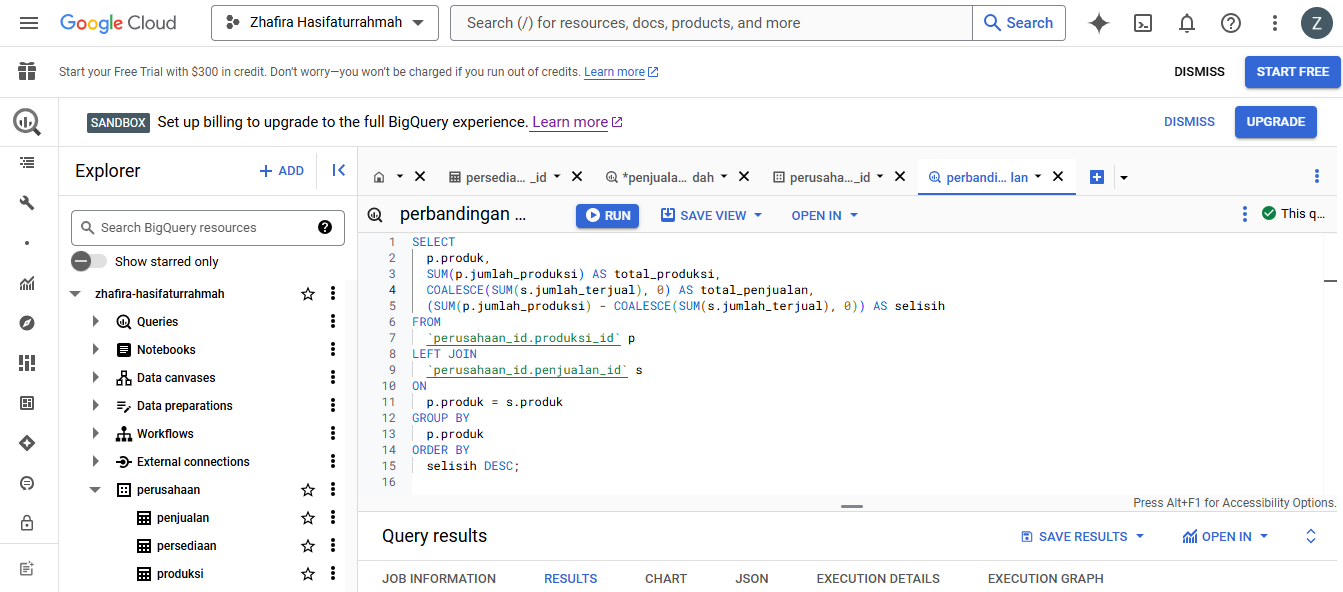
ORDER BY

selisih DESC;

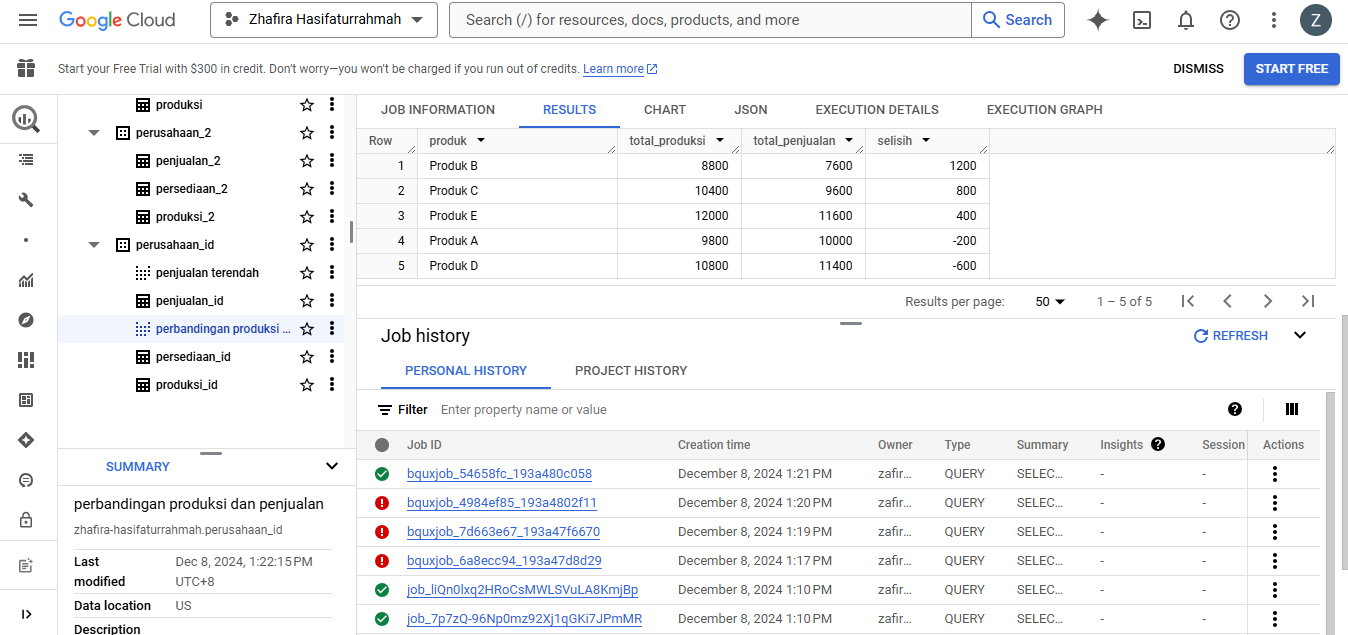
**Langkah Memeriksa dan Menjalankan Query**

1. **Periksa Project ID**:  
   Pastikan my-project-id diganti dengan Project ID Anda yang benar.
2. **Dataset dan Tabel**:  
   Pastikan dataset (perusahaan) dan tabel (tabel\_produksi dan tabel\_penjualan) sesuai dengan yang ada di BigQuery Anda.
3. **Menjalankan Query**:  
   Jalankan query di **BigQuery Console** untuk memastikan tidak ada syntax error.

Jika masih ada masalah, silakan beri tahu saya pesan error lengkapnya untuk analisis lebih lanjut! 😊



.



.

