

Carilah Product_id dengan jumlah penjualan terendah selama Bulan Januari 2024 dan memiliki Production_id bernomor genap!

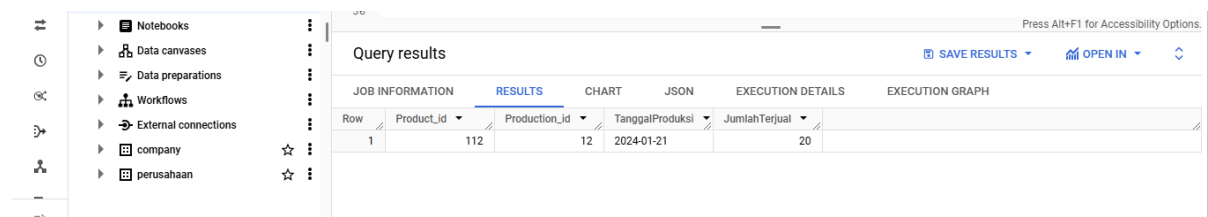
Kode SQL

<https://console.cloud.google.com/bigquery?sq=818780456763:861e49aa5fcf4a5cadb218e080b71fdf>

```
-- Menggabungkan tabel produksi dan penjualan dengan foreign key Product_id
WITH DataGabungan AS (
  SELECT
    p.Product_id,
    p.Production_id,
    p.TanggalProduksi,
    j.JumlahTerjual
  FROM
    `ferdiana-rosyada-selly.company.produksi` p
  JOIN
    `ferdiana-rosyada-selly.company.penjualan` j
  ON
    p.Product_id = j.Product_id
  WHERE
    DATE(p.TanggalProduksi) BETWEEN '2024-01-01' AND '2024-01-31'
    AND MOD(p.Production_id, 2) = 0 -- Hanya Production_id genap
)

-- Mencari Product_id dengan jumlah penjualan terendah
SELECT
  Product_id,
  Production_id,
  TanggalProduksi,
  JumlahTerjual
FROM
  DataGabungan
ORDER BY
  JumlahTerjual ASC -- Mengurutkan dari penjualan terendah
LIMIT 1; -- Mengambil baris dengan penjualan terendah
```

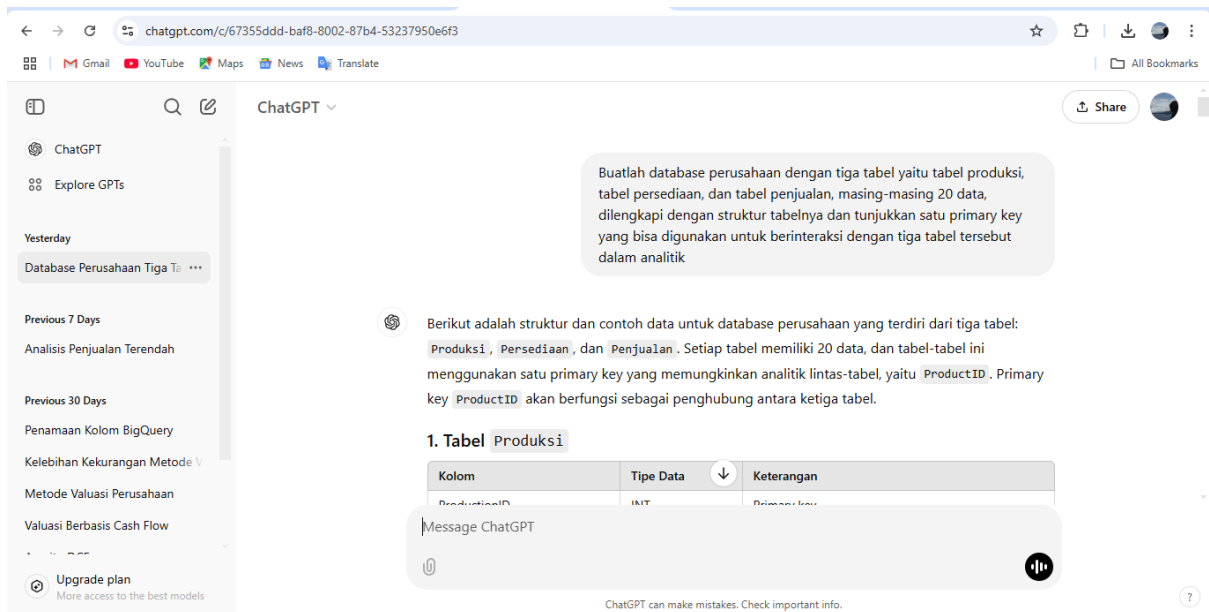
Hasil Kode

The screenshot shows the Google BigQuery interface. On the left is a sidebar with navigation options like Notebooks, Data canvases, Data preparations, Workflows, External connections, company, and perusahaan. The main area displays 'Query results' with tabs for JOB INFORMATION, RESULTS (selected), CHART, JSON, EXECUTION DETAILS, and EXECUTION GRAPH. Below the tabs is a table with 5 columns: Row, Product_id, Production_id, TanggalProduksi, and JumlahTerjual. The first row shows Product_id 112, Production_id 12, TanggalProduksi 2024-01-21, and JumlahTerjual 20.

Row	Product_id	Production_id	TanggalProduksi	JumlahTerjual
1	112	12	2024-01-21	20

Product_id dengan penjualan terendah yang memiliki TanggalProduksi di Bulan Januari 2024 dan dengan Production_id yang genap adalah “112” dengan produk terjual yaitu 20 buah dan produksinya dilakukan pada tanggal 21 Januari 2024.

Step by step GPT



Berikut adalah struktur dan contoh data untuk database perusahaan yang terdiri dari tiga tabel: Produksi, Persediaan, dan Penjualan. Setiap tabel memiliki 20 data, dan tabel-tabel ini menggunakan satu primary key yang memungkinkan analitik lintas-tabel, yaitu ProductID. Primary key ProductID akan berfungsi sebagai penghubung antara ketiga tabel.

1. Tabel Produksi

Kolom	Tipe Data	Keterangan
ProductionID	INT	Primary key
ProductID	INT	Foreign key dari Persediaan
TanggalProduksi	DATE	Tanggal produksi
JumlahDiproduksi	INT	Jumlah barang yang diproduksi
BiayaProduksi	DECIMAL	Biaya produksi per unit barang

Contoh Data Produksi (20 Data)

ProductionID	ProductID	TanggalProduksi	JumlahDiproduksi	BiayaProduksi
1	101	2024-01-10	500	50.00
2	102	2024-01-11	300	45.00
3	103	2024-01-15	600	60.00
...
20	120	2024-02-15	400	55.00

2. Tabel Persediaan

Kolom	Tipe Data	Keterangan
ProductID	INT	Primary key
NamaProduk	VARCHAR(50)	Nama produk
Kategori	VARCHAR(30)	Kategori produk
StokAwal	INT	Stok awal
StokAkhir	INT	Stok akhir (update setelah penjualan)
HargaPerUnit	DECIMAL	Harga jual per unit

Contoh Data Persediaan (20 Data)

ProductID	NamaProduk	Kategori	StokAwal	StokAkhir	HargaPerUnit
101	Produk A	Elektronik	500	450	120.00
102	Produk B	Pakaian	300	200	80.00
103	Produk C	Makanan	600	580	20.00
...
120	Produk T	Furnitur	400	370	150.00

3. Tabel Penjualan

Kolom	Tipe Data	Keterangan
SaleID	INT	Primary key
ProductID	INT	Foreign key dari Persediaan
TanggalPenjualan	DATE	Tanggal transaksi penjualan
JumlahTerjual	INT	Jumlah unit yang terjual
Pendapatan	DECIMAL	Pendapatan dari transaksi

Contoh Data Penjualan (20 Data)

SaleID	ProductID	TanggalPenjualan	JumlahTerjual	Pendapatan
1	101	2024-01-15	50	6000.00
2	102	2024-01-16	100	8000.00
3	103	2024-01-18	20	400.00

SaleID ProductID TanggalPenjualan JumlahTerjual Pendapatan

...
20	120	2024-02-20	30	4500.00

Relasi Antar Tabel

- Primary Key di Produksi: ProductionID
- Primary Key di Persediaan: ProductID
- Primary Key di Penjualan: SaleID
- Foreign Key: ProductID di setiap tabel sebagai penghubung antara Produksi, Persediaan, dan Penjualan

Skema Relasi untuk Analitik

Dengan menggunakan ProductID sebagai key, kita dapat membuat analitik lintas-tabel untuk mengetahui:

- Jumlah barang diproduksi, dijual, dan stok yang tersisa.
- Pendapatan dan keuntungan per produk, berdasarkan data dari ketiga tabel.

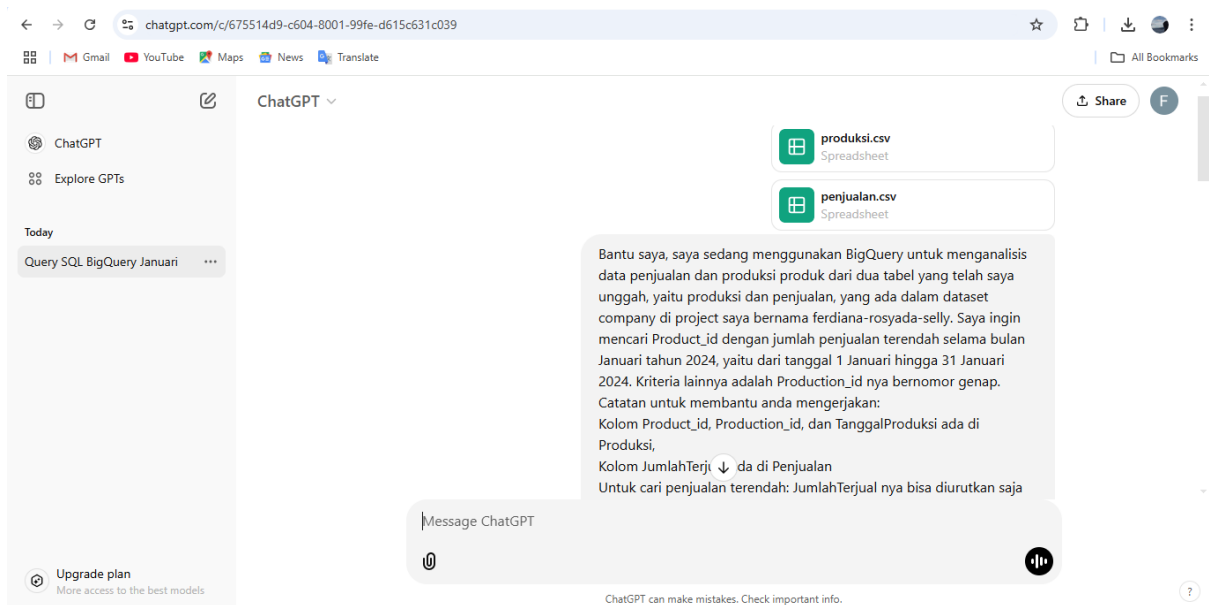
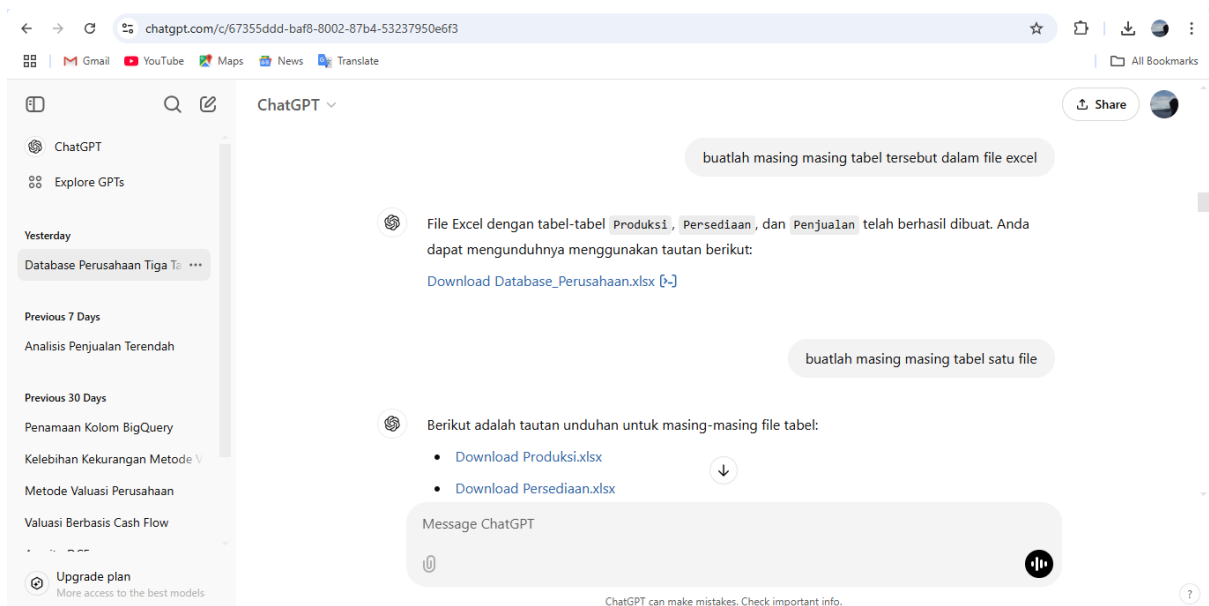
Primary key ProductID di Persediaan berfungsi sebagai foreign key di Produksi dan Penjualan, memungkinkan analitik lintas tabel yang komprehensif.

Note:

Saya mengubah beberapa nama kolom, yaitu:

1. **ProductID** menjadi **Product_id**
2. **ProductionID** menjadi **Production_id**
3. **SaleID** menjadi **Sales_id**

Hal itu dikarenakan penulisan huruf “I” di excel berbeda sehingga menyebabkan tidak terdeteksi di bigquery.



Bantu saya, saya sedang menggunakan BigQuery untuk menganalisis data penjualan dan produksi produk dari dua tabel yang telah saya unggah, yaitu produksi dan penjualan, yang ada dalam dataset company di project saya bernama ferdiana-rosyada-selly. Saya ingin mencari Product_id dengan jumlah penjualan terendah selama bulan Januari tahun 2024, yaitu dari tanggal 1 Januari hingga 31 Januari 2024. Kriteria lainnya adalah Production_id nya bernomor genap.

Catatan untuk membantu anda mengerjakan:

Kolom Product_id, Production_id, dan TanggalProduksi ada di Produksi,

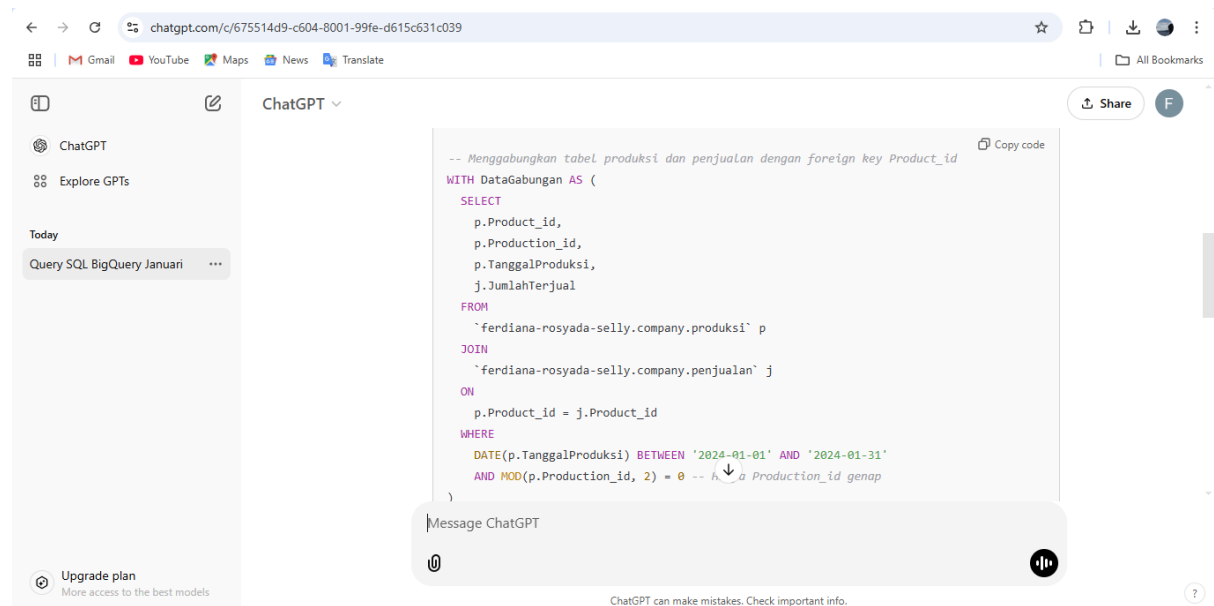
Kolom JumlahTerjual ada di Penjualan

Untuk cari penjualan terendah: JumlahTerjual nya bisa diurutkan saja dari yang terbesar ke terkecil

Kolom yang dipakai hanya Product_id, Production_id, TanggalProduksi, dan JumlahTerjual saja

Untuk menghubungkan 2 tabel tersebut menggunakan foreign key, foreign key nya yaitu kolom Product_id

Dapatkan Anda memberikan query SQL khusus BigQuery untuk menjalankan analisis ini?



Penjelasan:

1. WITH DataGabungan:

- Menggabungkan data dari tabel produksi dan penjualan menggunakan kolom Product_id sebagai foreign key.
- Filter data untuk bulan Januari 2024 (TanggalProduksi dalam rentang 1-31 Januari 2024).
- Menyaring hanya Production_id yang bernomor genap dengan $\text{MOD}(\text{Production_id}, 2) = 0$.

2. Query utama:

- Mengambil semua kolom yang relevan (Product_id, Production_id, TanggalProduksi, JumlahTerjual).
- Mengurutkan data berdasarkan JumlahTerjual dari yang terendah ke tertinggi.
- Menggunakan LIMIT 1 untuk mendapatkan hasil dengan penjualan terendah.

Jika ada penyesuaian atau tabel menggunakan nama kolom yang berbeda, silakan beri tahu saya agar saya dapat memperbaiki query sesuai kebutuhan. Anda juga bisa mengunggah data untuk validasi lebih lanjut.