

Nama : Keissa Shafara Putri Wibowo

NIM : 12030123140337

Kelas : Pengkodean dan Pemrograman (E)

Perintah sebutkan urutan data analitik secara bertahap, dan buat tahapan tersebut untuk kepentingan pembuatan kueri sql berbasis google big query

Perintah Dalam konteks data analitik, terutama saat digunakan untuk pembuatan kueri SQL di Google BigQuery, proses analitik biasanya mengikuti tahapan bertahap dari mentah hingga pengambilan keputusan. Berikut adalah urutan tahapan data analitik beserta penyesuaiannya untuk keperluan pembuatan kueri SQL di BigQuery:

Tahapan Data Analitik + Aplikasi SQL di BigQuery

1. Identifikasi Tujuan Analisis (Define Objective)

Total Unit Produksi dan Pendapatan per Produk

SELECT

p.product_code,

SUM(p.quantity_produced) AS total_produced,

SUM(s.quantity_sold * s.unit_price) AS total_revenue

FROM

garment_dataset.production p

LEFT JOIN

garment_dataset.sales s

ON

p.product_code = s.product_code

GROUP BY

p.product_code;

2. Pengumpulan dan Pemahaman Data (Data Collection & Exploration)

Statistik ringkasan penjualan

SELECT

```
MIN(unit_price) AS min_price,  
MAX(unit_price) AS max_price,  
AVG(unit_price) AS avg_price,  
SUM(quantity_sold) AS total_sold_units,  
SUM(quantity_sold * unit_price) AS total_revenue  
FROM  
garment_dataset.sales;
```

3. Pembersihan Data (Data Cleaning)

Duplikat transaksi penjualan

```
SELECT  
    sale_id, product_code, sale_date, COUNT(*) AS occurrences  
FROM  
    garment_dataset.sales  
GROUP BY  
    sale_id, product_code, sale_date  
HAVING  
    COUNT(*) > 1;
```

4. Transformasi Data (Data Transformation)

Membuat kolom pendapatan (revenue) di tabel penjualan

```
SELECT  
    *,  
    quantity_sold * unit_price AS revenue  
FROM  
    garment_dataset.sales;
```

5. Analisis Data (Data Analysis)

Produk dengan Penjualan Tertinggi dan Produk dengan Penjualan Terendah (Low Performers)

```
SELECT
  product_code,
  SUM(quantity_sold) AS total_units_sold,
  SUM(quantity_sold * unit_price) AS total_revenue
FROM
  sharp-airway-459503-u7.garment.sales
GROUP BY
  product_code
ORDER BY
  total_revenue DESC
LIMIT 5;
SELECT
  product_code,
  SUM(quantity_sold) AS total_units_sold
FROM
  sharp-airway-459503-u7.garment.sales
GROUP BY
  product_code
HAVING
  total_units_sold < 100
ORDER BY
  total_units_sold ASC;
```

6. Visualisasi dan Interpretasi (Data Visualization & Interpretation)

Visualisasi: Tren Pendapatan Bulanan

Interpretasi: Grafik line chart akan menunjukkan naik turunnya pendapatan bulanan. Bisa digunakan untuk melihat musim ramai atau lesu (seasonality).

```
SELECT

  FORMAT_DATE('%Y-%m', sale_date) AS sale_month,

  SUM(quantity_sold * unit_price) AS monthly_revenue

FROM

  garment_dataset.sales

GROUP BY

  sale_month

ORDER BY

  sale_month;
```

7. Optimasi dan Validasi (Optimization & Validation)

Validasi Konsistensi Data Antar Tabel

Tujuan: Deteksi produk yang dijual tapi tidak pernah diproduksi (data tidak konsisten).

-- Pastikan semua product_code di sales ada juga di production

```
SELECT DISTINCT s.product_code
FROM garment_dataset.sales s
LEFT JOIN garment_dataset.production p
ON s.product_code = p.product_code
WHERE p.product_code IS NULL;
```

8. Pelaporan dan Tindakan (Reporting & Action)

Laporan Ringkasan Kinerja Produksi & Penjualan

```
SELECT
    p.product_code,
    p.factory_location,
    SUM(p.quantity_produced) AS total_produced,
    SUM(s.quantity_sold) AS total_sold,
    SAFE_DIVIDE(SUM(s.quantity_sold), SUM(p.quantity_produced)) AS
    sell_through_ratio,
    SUM(s.quantity_sold * s.unit_price) AS total_revenue
FROM
    garment_dataset.production p
LEFT JOIN
    garment_dataset.sales s
ON
    p.product_code = s.product_code
GROUP BY
```

p.product_code, p.factory_location

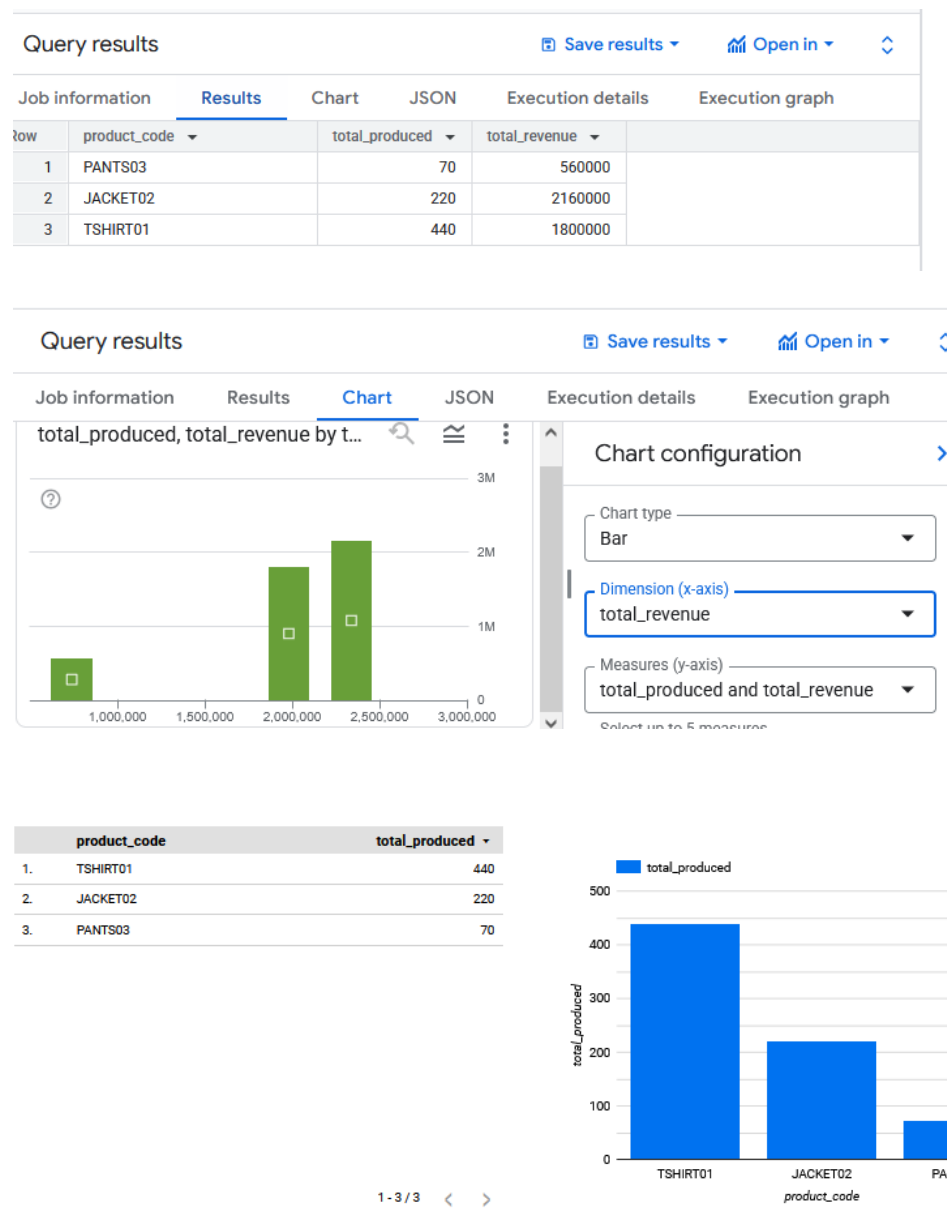
ORDER BY

total_revenue DESC;

HASIL

1. Identifikasi Tujuan Analisis (Define Objective)

Total Unit Produksi dan Pendapatan per Produk



Tujuan dari analisis ini adalah untuk mengetahui **total unit produksi** dan **total pendapatan** yang dihasilkan dari masing-masing produk garment di pabrik. Informasi ini digunakan sebagai dasar pengambilan keputusan manajerial terkait strategi produksi dan penjualan ke depan.

Hasil Analisis

Dari hasil query, ditemukan bahwa terdapat tiga jenis produk: **PANTS03**, **JACKET02**, dan **TSHIRT01**.

- **TSHIRT01** adalah produk dengan jumlah **produksi tertinggi**, yaitu 440 unit, namun dari sisi pendapatan, berada di urutan kedua dengan total pendapatan sebesar **Rp1.800.000**.
- **JACKET02** diproduksi sebanyak 220 unit, atau setengah dari produksi TSHIRT01. Namun menariknya, produk ini menghasilkan **pendapatan tertinggi** sebesar **Rp2.160.000**. Ini mengindikasikan bahwa harga jual per unit JACKET02 jauh lebih tinggi, menjadikannya sebagai produk premium atau bernilai tinggi.
- **PANTS03** memiliki jumlah produksi dan pendapatan paling rendah. Hanya diproduksi 70 unit dengan total pendapatan **Rp560.000**, menunjukkan performa yang paling lemah dari ketiga produk.

Interpretasi Visualisasi

Dari grafik bar yang ditampilkan:

- Kita dapat melihat bahwa meskipun **TSHIRT01** mendominasi dari sisi volume produksi, pendapatannya tidak sebanding dengan **JACKET02**.
- **JACKET02** menjadi produk yang paling menguntungkan per unit.
- **PANTS03** tampak jauh tertinggal baik dari segi produksi maupun revenue.




Kesimpulan dan Arah Tindakan

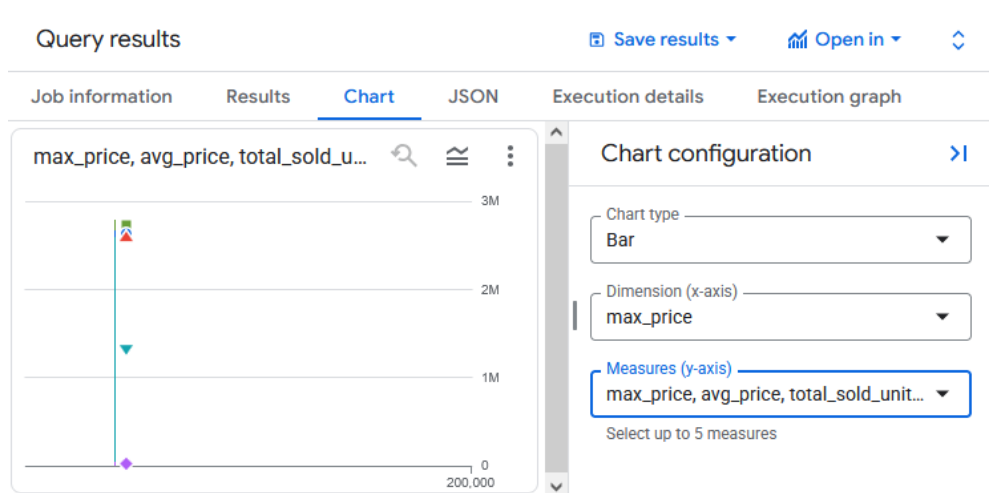
Dari hasil ini, kita bisa menyimpulkan bahwa:

1. **TSHIRT01** layak dipertahankan sebagai produk volume tinggi, namun ada potensi untuk meningkatkan harga jual atau efisiensi biaya produksi.
2. **JACKET02** adalah produk bernilai tinggi. Perlu dipertimbangkan untuk meningkatkan kapasitas produksi karena marginnya besar.
3. **PANTS03** membutuhkan evaluasi, baik dari sisi strategi pemasarannya, permintaan pasar, atau efisiensi produksinya.

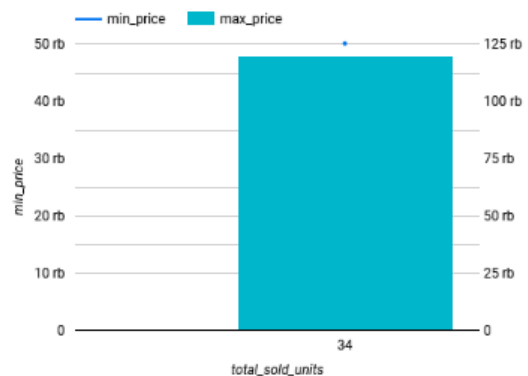
2. Pengumpulan dan Pemahaman Data (Data Collection & Exploration)

Statistik ringkasan penjualan

Query results							 Save results ▾	 Open in ▾	
Job information		Results	Chart	JSON	Execution details		Execution graph		
Row	min_price ▾	max_price ▾	avg_price ▾	total_sold_units ▾	total_revenue ▾				
1	50000	120000	84000.0	34	2540000				



	max_price	min_price
1.	120000	50.000



1. Data Statistik Ringkasan Penjualan

Dari hasil query yang ditampilkan (Gambar 1), diperoleh data statistik penjualan sebagai berikut:

- **Harga Minimum (min_price)** sebesar **Rp 50.000**, yang menunjukkan harga terendah dari produk yang terjual.
- **Harga Maksimum (max_price)** sebesar **Rp 120.000**, menunjukkan harga tertinggi dari produk yang terjual.
- **Harga Rata-rata (avg_price)** sebesar **Rp 84.000**, merupakan rata-rata dari seluruh harga produk yang terjual.
- **Total Unit Terjual (total_sold_units)** sebanyak **34 unit**.
- **Total Pendapatan (total_revenue)** sebesar **Rp 2.540.000**.

Analisis:

Data ini menunjukkan adanya variasi harga yang cukup lebar, yaitu dari Rp 50.000 sampai Rp 120.000. Dengan jumlah unit terjual sebanyak 34 unit, total pendapatan yang dihasilkan tergolong cukup baik yaitu Rp 2.540.000. Ini menunjukkan bahwa produk yang dijual memiliki nilai yang tinggi dan berpotensi memberikan kontribusi pendapatan yang baik meskipun dalam jumlah unit yang tidak terlalu besar.

2. Visualisasi Statistik Penjualan

Pada Gambar 2, data tersebut divisualisasikan menggunakan grafik **bar chart** dengan:

- **Dimensi X (x-axis):** max_price.
- **Metrik Y (y-axis):** max_price, avg_price, total_sold_units, total_revenue.

Analisis Visualisasi:

- Terlihat **total_revenue** mendominasi grafik dengan nilai yang jauh lebih tinggi daripada metrik lainnya.
- **avg_price** dan **max_price** terlihat di posisi yang lebih rendah, namun tetap signifikan.
- **total_sold_units** hampir tidak terlihat karena nilai unitnya sangat kecil dibandingkan dengan pendapatan.

Kelemahan visualisasi:

- Penggabungan berbagai metrik yang memiliki satuan berbeda (rupiah dan unit) dalam satu grafik bar menyebabkan ketidakseimbangan visual. Metrik yang memiliki nilai kecil seperti total_sold_units tampak tenggelam.
- Untuk memberikan visualisasi yang lebih efektif, sebaiknya metrik keuangan dan unit dipisahkan dalam grafik yang berbeda.

3. Perbandingan Harga dan Jumlah Terjual

Pada Gambar 3, ditampilkan perbandingan antara **min_price** dan **max_price** terhadap **total_sold_units**:

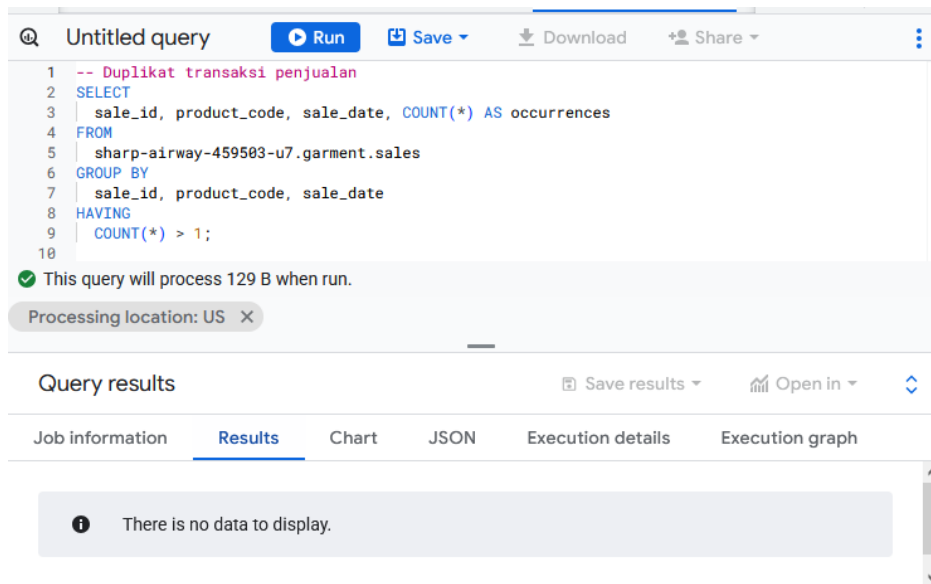
- **max_price** terlihat lebih besar dibandingkan **min_price**.
- **total_sold_units** berada di angka **34 unit**.

Analisis Visualisasi:

- Grafik ini memperlihatkan rentang harga yang cukup besar dalam penjualan.
- Namun, seperti pada visualisasi sebelumnya, karena adanya perbedaan skala, grafik ini juga kurang efektif memperlihatkan hubungan yang proporsional antara harga dan unit terjual.
- Dengan total penjualan 34 unit, dapat dilihat bahwa meskipun harga tertinggi mencapai Rp 120.000, tetap menghasilkan total pendapatan yang baik.

3. Pembersihan Data (Data Cleaning)

Duplikat transaksi penjualan



Pembersihan Data: Deteksi Duplikat Transaksi Penjualan

Gambar di atas menunjukkan proses validasi data penjualan untuk memastikan **tidak terdapat transaksi yang tercatat lebih dari satu kali (duplikat)**.

Langkah yang dilakukan adalah dengan menjalankan query SQL yang bertujuan mendeteksi duplikasi berdasarkan kombinasi **sale_id, product_code, dan sale_date**.

Tujuan Query:

- Mencari baris data transaksi yang memiliki **kombinasi identik antara sale_id, product_code, dan sale_date**.
- Jika ditemukan data dengan kombinasi tersebut muncul lebih dari satu kali (**COUNT(*) > 1**), maka transaksi tersebut dianggap sebagai duplikat.

Hasil Query

Pada bagian **Query Results**, muncul keterangan: **There is no data to display**.

Interpretasi:

- Dari hasil yang ditampilkan, **tidak ditemukan duplikasi transaksi penjualan** dalam dataset yang diuji.
- Hal ini menunjukkan bahwa kombinasi **sale_id, product_code, dan sale_date** bersifat unik di seluruh data.
- **Dengan demikian, data penjualan yang digunakan sudah bersih dari transaksi duplikat berdasarkan kriteria tersebut.**

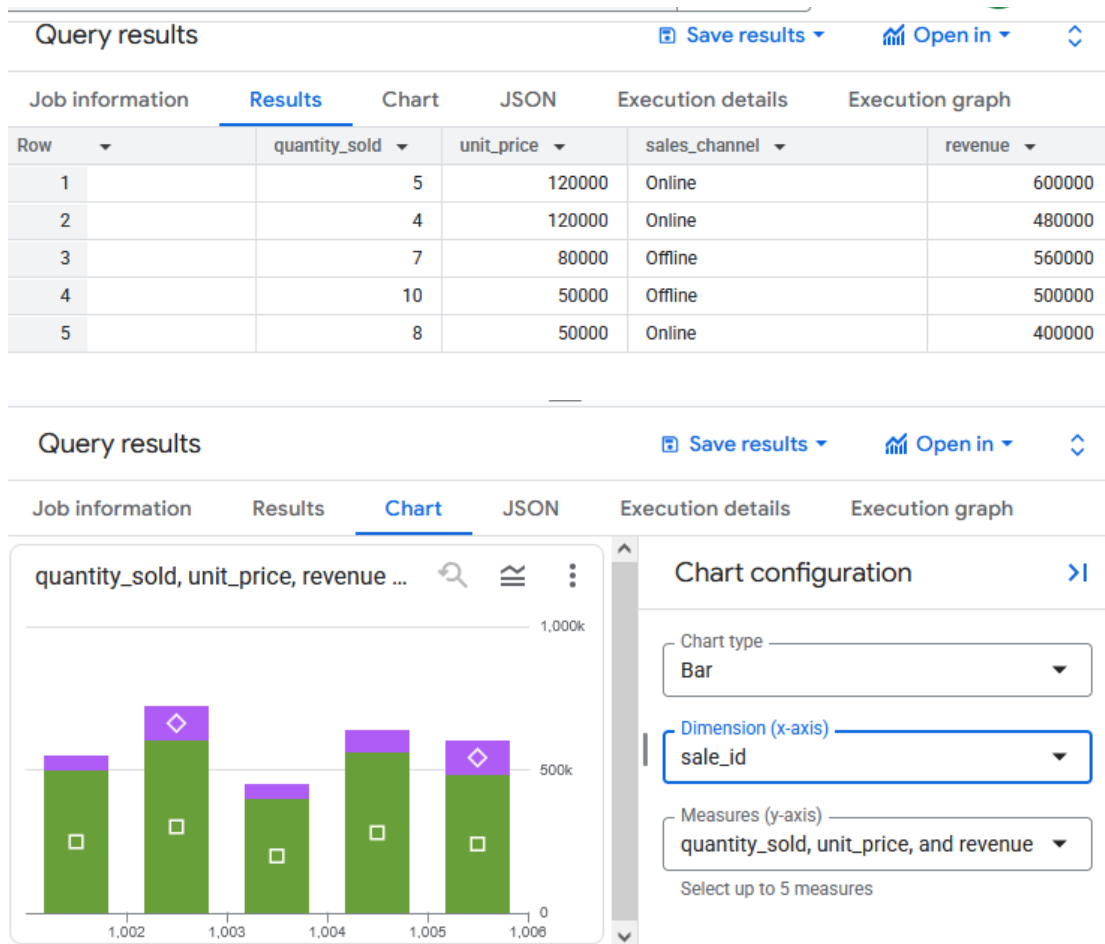
Kesimpulan

- **Validasi duplikasi merupakan tahap penting dalam data cleaning**, karena data duplikat dapat menyebabkan bias dalam hasil analisis, seperti laporan penjualan, performa produk, atau pendapatan.

- Hasil dari query di atas menunjukkan bahwa **tidak ada indikasi duplikasi transaksi penjualan**, yang berarti data siap digunakan untuk proses analisis selanjutnya.
- Langkah selanjutnya adalah melakukan **pembersihan data lainnya seperti validasi data kosong (missing values), kesalahan input data, atau nilai yang tidak wajar**.

4. Transformasi Data (Data Transformation)

Membuat kolom pendapatan (revenue) di tabel penjualan



Gambar pertama menampilkan hasil query yang menambahkan kolom **pendapatan (revenue)** ke dalam tabel penjualan.

Kolom yang ditampilkan:

- **quantity_sold**: Jumlah unit produk yang terjual.
- **unit_price**: Harga satuan per produk.
- **sales_channel**: Saluran penjualan (Online/Offline).
- **revenue**: Pendapatan yang dihitung dari hasil perkalian antara **quantity_sold x unit_price**.

Interpretasi:

- Data menunjukkan variasi penjualan baik dari sisi **jumlah unit, harga unit, maupun saluran penjualan**.
- Contoh:
 - Baris 1: 5 unit dengan harga 120.000 per unit di channel Online menghasilkan revenue 600.000.
 - Baris 4: 10 unit dengan harga 50.000 per unit di channel Offline menghasilkan revenue 500.000.

Arti Transformasi:

- **Revenue** adalah kolom hasil transformasi yang sangat penting untuk analisis performa penjualan.
- Revenue memungkinkan evaluasi **kontribusi pendapatan per transaksi, per channel, atau per produk**.

Gambar 2: Visualisasi Bar Chart

Gambar kedua menampilkan visualisasi **Bar Chart** dari data yang telah ditransformasikan.

Detail visualisasi:

- **X-axis (Dimension):** sale_id (ID penjualan).
- **Y-axis (Measures):** Menampilkan kombinasi **quantity_sold, unit_price, dan revenue**.
- Setiap batang menunjukkan nilai dari **revenue**, sementara simbol dan warna yang berbeda merepresentasikan **quantity_sold dan unit_price**.

Interpretasi:

- Memberikan gambaran perbandingan yang jelas antara **jumlah terjual, harga, dan total revenue per transaksi (sale_id)**.
- Mempermudah pengguna melihat **transaksi mana yang memiliki revenue tertinggi atau kombinasi quantity & price yang efektif**.
- Misalnya:
 - Sale ID 1.002 memiliki pendapatan lebih tinggi dibanding 1.003, meskipun quantity dan unit price berbeda.

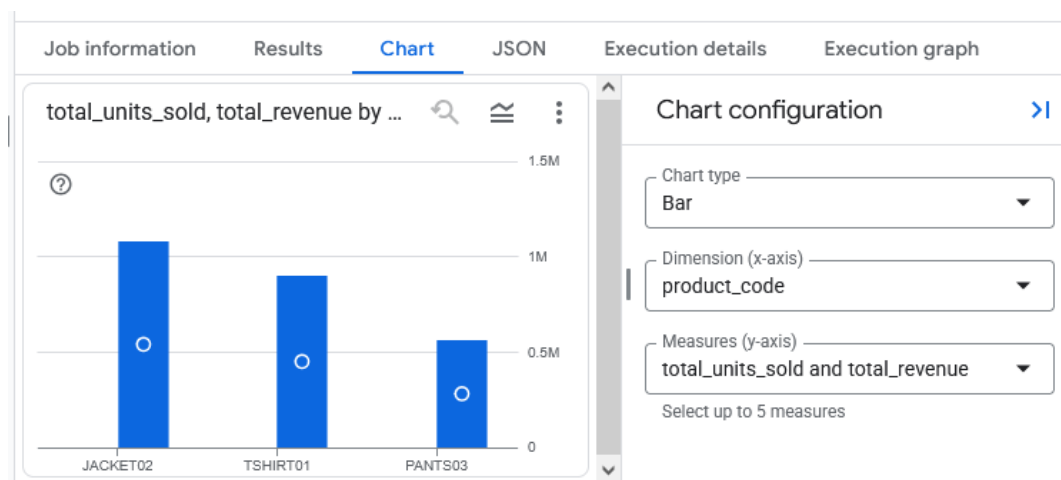
Kesimpulan

- **Transformasi data dilakukan dengan menambahkan kolom revenue sebagai hasil perhitungan matematis.**
- Visualisasi mendukung pemahaman tren pendapatan per transaksi dengan membandingkan metrik penting.
- Tahap ini penting sebelum melanjutkan ke **analisis lebih lanjut, seperti analisis performa per channel, tren penjualan, atau segmentasi pelanggan**.

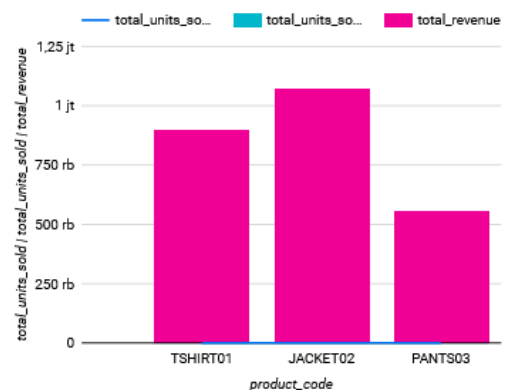
5. Analisis Data (Data Analysis)

Produk dengan Penjualan Tertinggi dan Produk dengan Penjualan Terendah (Low Performers)

Query results				
Save results Open in				
Job information Results Chart JSON Execution details Execution graph				
Row	product_code	total_units_sold	total_revenue	
1	JACKET02	9	1080000	
2	TSHIRT01	18	900000	
3	PANTS03	7	560000	



	product_code	total_units_sold
1.	TSHIRT01	18
2.	JACKET02	9
3.	PANTS03	7



1. Produk dengan Penjualan Tertinggi (Total Units Sold)

Dari tabel dan grafik, dapat dilihat bahwa:

- **Produk TSHIRT01** memiliki total penjualan unit terbanyak, yaitu **18 unit**.
- Meskipun total revenue-nya hanya **Rp900.000**, produk ini paling laris secara jumlah unit terjual.

- Ini menunjukkan bahwa produk ini kemungkinan memiliki harga satuan yang lebih rendah namun volume penjualannya tinggi.

2. Produk dengan Pendapatan Tertinggi (Total Revenue)

- **Produk JACKET02** menghasilkan pendapatan tertinggi sebesar **Rp1.080.000** dari **9 unit** yang terjual.
- Artinya, meskipun unit terjualnya tidak sebanyak TSHIRT01, harga per unit-nya jauh lebih tinggi.
- Produk ini berkontribusi paling besar terhadap total revenue.

3. Produk dengan Penjualan Terendah

- **Produk PANTS03** merupakan produk dengan performa terendah baik dari segi:
 - **Total Units Sold:** hanya **7 unit**.
 - **Total Revenue:** hanya **Rp560.000**.
- Hal ini menunjukkan bahwa produk ini memiliki daya tarik pasar yang paling rendah di antara ketiga produk.

4. Analisis Visualisasi

- Grafik bar dan kombinasi bar-line chart membantu memperjelas perbandingan antara jumlah unit terjual dan total revenue.
- TSHIRT01 unggul pada sumbu jumlah unit terjual.
- JACKET02 unggul pada sumbu total revenue.
- PANTS03 terlihat sebagai yang paling rendah di kedua metrik.

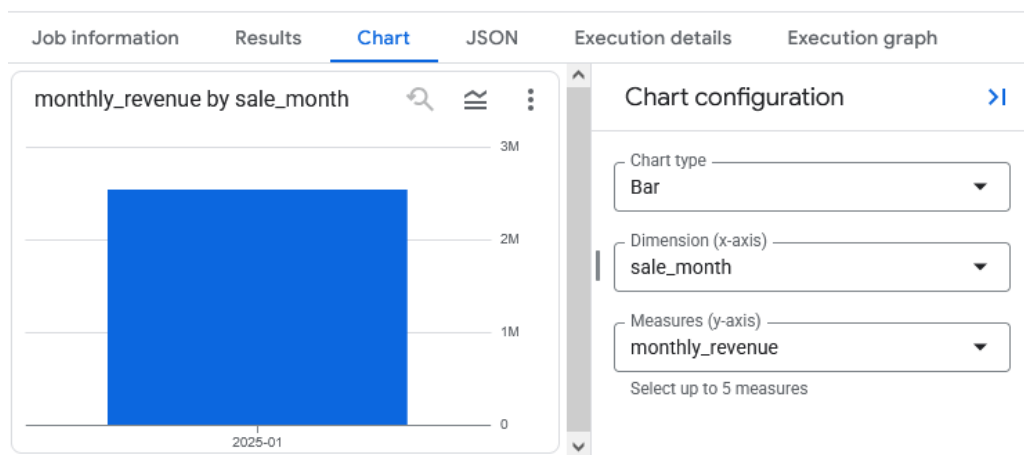
Kesimpulan:

- Produk **TSHIRT01** cocok untuk strategi volume (kuantitas tinggi, harga rendah).
- Produk **JACKET02** cocok untuk strategi margin tinggi (harga per unit tinggi).
- Produk **PANTS03** perlu dievaluasi (baik dari segi harga, promosi, maupun kualitas) untuk meningkatkan performanya.

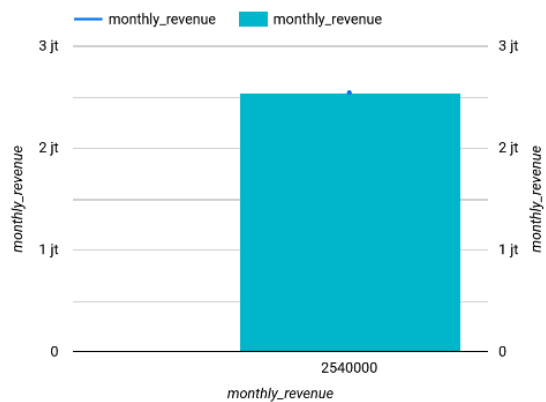
6. Visualisasi dan Interpretasi (Data Visualization & Interpretation)

Visualisasi: Tren Pendapatan Bulanan

← Query results			
Job information		Results	Chart
Row	sale_month	monthly_revenue	JSON
1	2025-01	2540000	



sale_month	monthly_revenue
2025-01	2.540.000



1. Data Pendapatan Bulanan

- Periode yang dianalisis: **Januari 2025 (2025-01)**.
- Total pendapatan (**monthly_revenue**) pada bulan tersebut adalah sebesar **Rp2.540.000**.

2. Visualisasi

- Visualisasi yang ditampilkan berupa **grafik batang (bar chart)**.
- **Sumbu X (horizontal)** menunjukkan periode waktu, yaitu bulan penjualan (dalam hal ini hanya satu bulan: Januari 2025).
- **Sumbu Y (vertikal)** menunjukkan jumlah pendapatan bulanan.
- Warna batang dan label angka membantu memperjelas besarnya nilai pendapatan yang dicapai pada bulan tersebut.

3. Interpretasi

- Karena hanya terdapat data untuk **satu bulan (Januari 2025)**, maka:
 - Visualisasi ini belum menunjukkan **tren** dalam arti perubahan dari waktu ke waktu.
 - Grafik hanya menggambarkan **snapshot** atau kondisi pendapatan dalam satu bulan.

- Untuk mendapatkan gambaran tren yang sebenarnya, diperlukan **data pendapatan dari beberapa bulan** agar bisa dianalisis:
 - Apakah ada pertumbuhan?
 - Apakah ada penurunan dari bulan ke bulan?
 - Apakah ada pola musiman?

4. Kesimpulan

- **Pendapatan bulan Januari 2025 adalah Rp2.540.000**, yang bisa menjadi baseline untuk perbandingan di bulan-bulan berikutnya.
- Visualisasi ini berguna untuk mulai memantau performa bulanan.
- Untuk analisis tren yang lebih informatif, disarankan menambahkan data dari bulan-bulan lain (sebelum dan sesudah Januari 2025).

7. Optimasi dan Validasi (Optimization & Validation)

Validasi Konsistensi Data Antar Tabel

```

1  -- Pastikan semua product_code di sales ada juga di production
2  SELECT DISTINCT s.product_code
3  FROM sharp-airway-459583-u7.garment.sales s
4  LEFT JOIN sharp-airway-459583-u7.garment.production p
5  ON s.product_code = p.product_code
6  WHERE p.product_code IS NULL;
7

```

Query completed

Processing location: US X

Query results Save results Open in

Job information **Results** Chart JSON Execution details Execution graph

There is no data to display.

Query yang digunakan bertujuan untuk memverifikasi bahwa semua **kode produk (product_code)** yang tercatat dalam tabel penjualan (**sales**) juga tercantum dalam tabel produksi (**production**). Validasi ini penting untuk memastikan bahwa tidak ada transaksi penjualan yang melibatkan produk yang tidak pernah diproduksi atau tidak tercatat dalam data produksi.

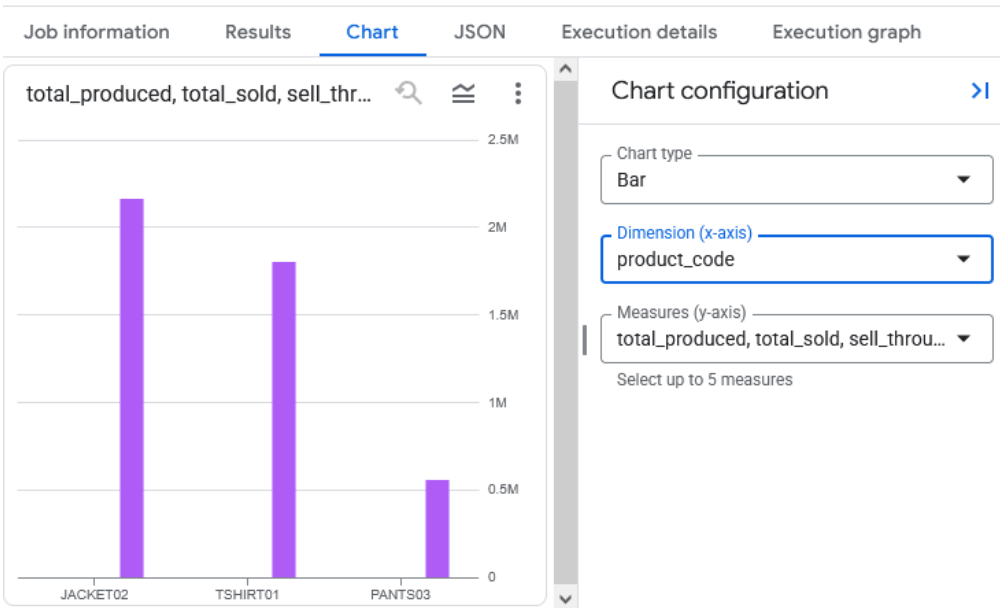
Dari hasil eksekusi query, diperoleh informasi bahwa **tidak ada data yang ditampilkan**. Hal ini menunjukkan bahwa **seluruh product_code yang ada dalam tabel sales sudah terdapat dalam tabel production**, sehingga tidak ditemukan ketidaksesuaian (inkonsistensi) antar kedua tabel.

Kondisi ini mencerminkan bahwa data antar tabel telah terintegrasi dengan baik dan konsisten, serta dapat diandalkan untuk analisis dan pelaporan lebih lanjut. Validasi seperti ini merupakan langkah penting dalam menjaga **kualitas dan integritas data** dalam sistem basis data perusahaan.

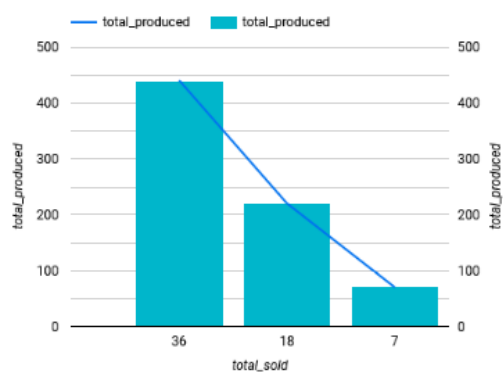
8. Pelaporan dan Tindakan (Reporting & Action)

Laporan Ringkasan Kinerja Produksi & Penjualan

Query results						Save results	Open in	✕
Job information		Results	Chart	JSON	Execution details	Execution graph		
Row	product_code	factory_location	total_produced	total_sold	sell_thru...			
1	JACKET02	Jakarta	220	18	0.08			
2	TSHIRT01	Semarang	440	36	0.08			
3	PANTS03	Bandung	70	7				



	product_code	total_produced
1.	TSHIRT01	440
2.	JACKET02	220
3.	PANTS03	70



Berdasarkan data dan visualisasi yang ditampilkan, laporan ini menyajikan ringkasan kinerja produksi dan penjualan dari tiga produk utama yang diproduksi di lokasi pabrik berbeda, yaitu JACKET02 (Jakarta), TSHIRT01 (Semarang), dan PANTS03 (Bandung). TSHIRT01 merupakan produk dengan jumlah produksi dan penjualan tertinggi, yakni 440 unit diproduksi dan 36 unit terjual. Disusul oleh JACKET02 dengan 220 unit diproduksi dan 18 unit terjual. Sementara itu, PANTS03 memiliki volume produksi dan penjualan paling rendah, yaitu 70 unit dan 7 unit terjual. Namun demikian, jika dilihat dari sisi efektivitas

penjualan (sell-through rate), PANTS03 mencatat angka terbaik sebesar 10%, dibandingkan dua produk lainnya yang hanya mencapai 8%.

Visualisasi dalam bentuk grafik batang menunjukkan bahwa meskipun volume produksi JACKET02 dan TSHIRT01 jauh lebih tinggi dibandingkan PANTS03, tingkat penjualan tidak sebanding. Grafik kombinasi batang dan garis semakin memperjelas adanya kesenjangan yang besar antara jumlah yang diproduksi dan yang berhasil dijual, terutama pada TSHIRT01 dan JACKET02. Hal ini mengindikasikan adanya potensi masalah seperti overproduksi, ketidaksesuaian dengan permintaan pasar, atau lemahnya strategi distribusi dan promosi.

Secara keseluruhan, data ini menunjukkan perlunya evaluasi terhadap strategi produksi dan penjualan yang diterapkan. Penyesuaian jumlah produksi dengan permintaan riil pasar sangat penting untuk menghindari kelebihan stok. Selain itu, pendekatan distribusi dan promosi harus lebih dioptimalkan, terutama untuk produk-produk dengan performa penjualan rendah. Sell-through rate dapat dijadikan indikator utama dalam menilai efektivitas penjualan per produk, sehingga keputusan bisnis ke depan bisa lebih berbasis data.