

# **DATA ANALITIK PERUSAHAAN DENGAN *WEBSITE* BIGQUERY**

Pemenuhan Tugas Sistem Informasi Akuntansi



Dosen Pengampu : Dr. Totok Dewayanto S.E.,M.Si., Akt.

Disusun Oleh :

Adila Putri Maharani

NIM : 40011423650272

Kelas : G

**PROGRAM D4 AKUNTANSI PERPAJAKAN**

**SEKOLAH VOKASI**

**UNIVERSITAS DIPONEGORO**

**2024**

## Proses data analisis sampai dengan menginput data dan menghasilkan data dalam website BigQuery

1. Membuat tiga file excel yang terdiri dari tabel produksi, penjualan dan persediaan dengan masing-masing 20 baris data serta mencari *primary key* dari ketiga tabel tersebut dengan bantuan ChatGPT.

buatlah 3 file excel tabel produksi penjualan persediaan masing masing 20 baris dan struktur mengenai perusahaan dan carikan primary key dari ketiga tabel tersebut untuk menjoinkan tabel tersebut



Berikut struktur dari tiga tabel: Produksi, Penjualan, dan Persediaan, masing-masing berisi 20 baris data. Tabel ini dapat digunakan untuk mencatat informasi tentang barang yang diproduksi, dijual, dan persediaan di perusahaan. Saya akan menjelaskan struktur tabel serta primary key yang bisa digunakan untuk menghubungkan tabel-tabel tersebut.

### Struktur Tabel dan Penjelasan Kolom

#### 1. Tabel Produksi

- ID\_Produksi (Primary Key): Kode unik untuk setiap proses produksi.
- Tanggal\_Produksi: Tanggal barang diproduksi.
- Kode\_Barang: Kode unik untuk setiap barang yang diproduksi.
- Jumlah\_Produksi: Jumlah barang yang diproduksi pada tanggal tertentu.

#### 2. Tabel Penjualan

- ID\_Penjualan (Primary Key): Kode unik untuk setiap transaksi penjualan.
- Tanggal\_Penjualan: Tanggal transaksi penjualan terjadi.
- Kode\_Barang: Kode unik untuk setiap barang yang dijual.
- Jumlah\_Terjual: Jumlah barang yang terjual dalam transaksi.

#### 3. Tabel Persediaan

- ID\_Persediaan (Primary Key): Kode unik untuk setiap pembaruan persediaan.
- Tanggal\_Update: Tanggal ketika persediaan barang diperbarui.
- Kode\_Barang: Kode unik untuk setiap barang dalam persediaan.
- Jumlah\_Persediaan: Jumlah barang yang ada di persediaan pada tanggal tertentu.

### Primary Key untuk Menghubungkan Tabel


Dari struktur di atas, Kode\_Barang dapat digunakan sebagai primary key untuk menghubungkan ketiga tabel karena ini adalah kolom yang umum di ketiga tabel tersebut. Kita bisa membuat relasi antar tabel sebagai berikut:

- Produksi berelasi dengan Penjualan melalui Kode\_Barang untuk melihat berapa banyak barang yang diproduksi dan terjual.
- Penjualan berelasi dengan Persediaan melalui Kode\_Barang untuk mengecek stok setelah barang terjual.

Saya akan membuat file Excel untuk masing-masing tabel sesuai dengan struktur yang telah dijelaskan.

2. Membuat masing-masing tabel produksi, penjualan dan persediaan dalam file excel dengan bantuan ChatGPT.


Buatkan file untuk masing masing tabel tersebut



File Excel yang berisi tabel **Produksi**, **Penjualan**, dan **Persediaan** sudah dibuat. Anda dapat mengunduhnya melalui tautan berikut:

[tabel\\_produksi\\_penjualan\\_persediaan.xlsx](#) [↗]

Buatkan dalam file excel terpisah atas tabel produksi tabel penjualan dan tabel persediaan



Berikut adalah file Excel terpisah untuk masing-masing tabel:


- [Tabel Produksi](#)
- [Tabel Penjualan](#)
- [Tabel Persediaan](#)

Silakan unduh sesuai kebutuhan Anda. [↗]

3. Membuat cleaned data dari file excel yang sudah diberikan dari ketiga tabel yaitu tabel produksi, penjualan dan persediaan.

cleaning file excel diatas

< 1/2 >



Untuk membersihkan file Excel di atas, kita akan melakukan beberapa langkah umum dalam data cleaning, seperti:

1. Menghilangkan data duplikat.
2. Memastikan semua kolom memiliki tipe data yang benar.
3. Menghapus nilai yang kosong atau null (jika ada).
4. Mengatur format tanggal secara konsisten.

Saya akan memeriksa dan membersihkan data pada file **Produksi**, **Penjualan**, dan **Persediaan** sesuai langkah-langkah di atas, lalu menyediakan versi yang sudah dibersihkan.

Berikut adalah file Excel yang sudah dibersihkan:

- [Tabel Produksi \(Cleaned\)](#)
- [Tabel Penjualan \(Cleaned\)](#)
- [Tabel Persediaan \(Cleaned\)](#)

Silakan unduh untuk data versi yang telah dibersihkan. [↗]

4. Membuat SQL dengan masing-masing ketentuan yang diberikan dengan 3 data yang sudah ditetapkan.



## Pertanyaan :

saya sedang menggunakan BigQuery untuk menganalisis data penjualan dan produksi produk dari dua tabel diatas, yaitu produksi dan penjualan, yang ada dalam dataset perusahaan\_2 di project saya bernama adila-putri-maharani. Saya ingin mencari id\_produk dengan jumlah penjualan tertinggi selama 1 bulan pada tahun 2023, yaitu dari tanggal 1 Januari hingga 20 Januari 2023.

## Kode SQL :

```
SELECT
  p.Kode_Barang AS id_produk,
  SUM(j.Jumlah_Terjual) AS total_terjual
FROM
  `adila-putri-maharani.perusahaan_2.produksi` AS p
JOIN
  `adila-putri-maharani.perusahaan_2.penjualan` AS j
ON
  p.Kode_Barang = j.Kode_Barang
WHERE
  j.Tanggal_Penjualan BETWEEN '2023-01-01' AND '2023-01-20'
GROUP BY
  id_produk
ORDER BY
  total_terjual DESC
LIMIT 5;
```

## Hasil SQL :

Query results					
Press ALT+1 for Accessibility Options.					
SAVE RESULTS EXPLORE DATA					
JOB INFORMATION RESULTS CHART JSON EXECUTION DETAILS EXECUTION GRAPH					
Row	id_produk	id_produk	tanggal_produk	jumlah_terjual	
1	P018	PRD008	2024-06-26	93	
2	P006	PRD006	2024-04-06	87	
3	P012	PRD002	2024-02-04	47	
4	P020	PRD010	2024-06-15	34	
Results per page: 50 1 - 4 of 4					

JOB INFORMATION RESULTS CHART JSON EXECUTION DETAILS EXECUTION GRAPH			
Row	id_produk	total_terjual	
2	K04	216	
3	K02	208	
4	K00	156	
5	K03	136	
Results per page: 50 1 - 5 of 5			

## Pertanyaan :

saya sedang menggunakan BigQuery untuk menganalisis data penjualan dan produksi produk dari dua tabel diatas, yaitu produksi dan penjualan, yang ada dalam dataset perusahaan\_2 di project saya bernama adila-putri-maharani. Saya ingin mencari id\_produk dengan jumlah penjualan tertinggi selama 1 bulan pada tahun 2023, yaitu dari tanggal 1 Januari hingga 20 Januari 2023.

## Kode SQL :

```
SELECT
  p.Kode_Barang AS id_produk,
  SUM(j.Jumlah_Terjual) AS total_terjual
FROM
  `adila-putri-maharani.perusahaan_2.produksi` AS p
JOIN
  `adila-putri-maharani.perusahaan_2.penjualan` AS j
ON
  p.Kode_Barang = j.Kode_Barang
WHERE
  j.Tanggal_Penjualan BETWEEN '2023-01-01' AND '2023-01-20'
GROUP BY
  id_produk
ORDER BY
  total_terjual DESC
LIMIT 5;
```

## Hasil SQL :

Query results

SAVE RESULTS

OPEN IN

JOB INFORMATION

RESULTS

CHART

JSON

EXECUTION DETAILS

EXECUTION GRAPH

Row	id_produk	total_terjual
1	K01	248
2	K04	216
3	K02	208
4	K00	156

Results per page:

50

1 – 5 of 5

Job history

REFRESH

JOB INFORMATION

RESULTS

CHART

JSON

EXECUTION DETAILS

EXECUTION GRAPH

Row	id_produk	total_terjual
2	K04	216
3	K02	208
4	K00	156
5	K03	136

Results per page:

50

1 – 5 of 5

## Pertanyaan :

Saya sedang menggunakan BigQuery untuk menganalisis data penjualan dan produksi produk dari dua tabel diatas, yaitu produksi dan penjualan, yang ada dalam dataset perusahaan\_3 di project saya bernama adila-putri-maharani. Saya ingin mencari id\_penjualan dengan tren penjualan selama 1 bulan pada tahun 2023, yaitu dari tanggal 1 Januari hingga 20 Januari 2023. Kriteria lainnya adalah id\_penjualan nya bernomor ganjil.

## SQL :

```
-- Query untuk mencari id_penjualan dengan tren penjualan selama 1-20 Januari 2023
SELECT
  id_penjualan,
  SUM(jumlah_terjual) AS total_penjualan
FROM
  `adila-putri-maharani.perusahaan_3.penjualan`
WHERE
  tanggal_penjualan BETWEEN '2023-01-01' AND '2023-01-20'
  AND SAFE_CAST(REGEXP_EXTRACT(id_penjualan, r'\d+$') AS INT64) IS NOT NULL -- Memastikan
hanya nilai numerik
  AND MOD(SAFE_CAST(REGEXP_EXTRACT(id_penjualan, r'\d+$') AS INT64), 2) = 1 -- Memilih ID
bernomor ganjil
GROUP BY
  id_penjualan
ORDER BY
  total_penjualan DESC;
```

## Hasil SQL :

Query results		SAVE RESULTS		OPEN IN		
<	JOB INFORMATION	RESULTS	CHART	JSON	EXECUTION DETAILS	EXECUTE >
Row	id_penjualan	total_penjualan				
1	J007	27				
2	J009	24				
3	J001	20				
4	J011	20				
Results per page: 50 1 – 6 of 6  < < > >						
Job history		REFRESH				

Query results

SAVE RESULTS

OPEN IN

<

JOB INFORMATION

RESULTS

CHART

JSON

EXECUTION DETAILS

EXECUTED

>

Row	id_penjualan	total_penjualan
3	J001	20
4	J011	20
5	J005	18
6	J003	15

Results per page: 501 – 6 of 6<<<>>>

Job history

REFRESH