Buatlah database perusahaan yang terdiri dari 3 tabel yaitu tabel produksi, tabel persediaan dan tabel penjualan.

**You said:**

buatlah database perusahaan yang terdiri dari 3 tabel yaitu tabel produksi, tabel persmediaan dan tabel penjualan yang ada primary keynya

**ChatGPT said:**

ChatGPT

Berikut adalah contoh struktur database perusahaan yang terdiri dari tiga tabel: tabel Produksi, tabel Persediaan, dan tabel Penjualan. Setiap tabel memiliki primary key yang unik.

**1. Tabel Produksi**

sql

Copy code

CREATE TABLE Produksi (

id\_produksi INT PRIMARY KEY AUTO\_INCREMENT,

nama\_produk VARCHAR(100) NOT NULL,

jumlah INT NOT NULL,

tanggal\_produksi DATE NOT NULL

);

**2. Tabel Persediaan**

sql

Copy code

CREATE TABLE Persediaan (

id\_persediaan INT PRIMARY KEY AUTO\_INCREMENT,

id\_produksi INT,

jumlah\_stok INT NOT NULL,

lokasi VARCHAR(100),

FOREIGN KEY (id\_produksi) REFERENCES Produksi(id\_produksi)

);

**3. Tabel Penjualan**

sql

Copy code

CREATE TABLE Penjualan (

id\_penjualan INT PRIMARY KEY AUTO\_INCREMENT,

id\_produk INT,

jumlah\_terjual INT NOT NULL,

tanggal\_penjualan DATE NOT NULL,

FOREIGN KEY (id\_produk) REFERENCES Produksi(id\_produksi)

);

**Penjelasan Struktur:**

* **Tabel Produksi**: Menyimpan informasi tentang produk yang diproduksi, termasuk ID unik, nama produk, jumlah yang diproduksi, dan tanggal produksi.
* **Tabel Persediaan**: Menyimpan informasi tentang jumlah stok produk yang ada di lokasi tertentu, terkait dengan tabel Produksi melalui foreign key.
* **Tabel Penjualan**: Menyimpan informasi tentang transaksi penjualan, termasuk jumlah yang terjual dan tanggal penjualan, juga terkait dengan tabel Produksi.

Dengan struktur ini, Anda dapat mengelola data produksi, persediaan, dan penjualan secara efektif.

4o mini

**You said:**

primary key apa yang bisa digunakan untuk 3 tabel tersebut berinteraksi

**ChatGPT said:**

ChatGPT

Agar ketiga tabel—Produksi, Persediaan, dan Penjualan—dapat berinteraksi dengan baik, kita perlu menggunakan foreign key yang mengacu pada primary key yang sama di tabel Produksi. Berikut adalah penjelasan tentang primary key dan foreign key untuk interaksi antar tabel:

**1. Tabel Produksi**

* **Primary Key**: id\_produksi
  + Ini adalah identifikasi unik untuk setiap produk yang diproduksi.

**2. Tabel Persediaan**

* **Primary Key**: id\_persediaan
* **Foreign Key**: id\_produksi
  + Mengacu pada id\_produksi di tabel Produksi, menghubungkan stok persediaan dengan produk tertentu.

**3. Tabel Penjualan**

* **Primary Key**: id\_penjualan
* **Foreign Key**: id\_produk
  + Mengacu pada id\_produksi di tabel Produksi, menghubungkan setiap penjualan dengan produk yang diproduksi.

**Interaksi Tabel**

* Ketika Anda ingin mengetahui stok produk tertentu di tabel Persediaan, Anda dapat melakukan join menggunakan id\_produksi.
* Untuk melacak penjualan dari produk tertentu, Anda juga dapat menggunakan id\_produk yang mengacu ke id\_produksi.

**Contoh Query untuk Interaksi**

Misalnya, untuk mendapatkan data penjualan berserta stok persediaan untuk produk tertentu:

sql

Copy code

SELECT

p.nama\_produk,

ps.jumlah\_stok,

j.jumlah\_terjual

FROM

Produksi p

JOIN

Persediaan ps ON p.id\_produksi = ps.id\_produksi

JOIN

Penjualan j ON p.id\_produksi = j.id\_produk

WHERE

p.id\_produksi = ?; -- Gantilah ? dengan id\_produksi yang diinginkan

Dengan cara ini, semua tabel dapat berinteraksi dengan baik menggunakan id\_produksi sebagai kunci penghubung.

4o mini

**You said:**

buatlah database perusahaan yang terdiri dari 3 tabel yaitu tabel produksi, tabel persediaan dan tabel penjualan yang ada primary keynya

**ChatGPT said:**

ChatGPT

Berikut adalah struktur database perusahaan dengan tiga tabel: Produksi, Persediaan, dan Penjualan, lengkap dengan primary key di masing-masing tabel.

**1. Tabel Produksi**

Tabel ini menyimpan informasi tentang produk yang diproduksi.

sql

Copy code

CREATE TABLE Produksi (

id\_produksi INT PRIMARY KEY AUTO\_INCREMENT,

nama\_produk VARCHAR(100) NOT NULL,

jumlah INT NOT NULL,

tanggal\_produksi DATE NOT NULL

);

**2. Tabel Persediaan**

Tabel ini menyimpan informasi tentang stok produk yang ada.

sql

Copy code

CREATE TABLE Persediaan (

id\_persediaan INT PRIMARY KEY AUTO\_INCREMENT,

id\_produksi INT,

jumlah\_stok INT NOT NULL,

lokasi VARCHAR(100),

FOREIGN KEY (id\_produksi) REFERENCES Produksi(id\_produksi)

);

**3. Tabel Penjualan**

Tabel ini menyimpan informasi tentang transaksi penjualan produk.

sql

Copy code

CREATE TABLE Penjualan (

id\_penjualan INT PRIMARY KEY AUTO\_INCREMENT,

id\_produksi INT,

jumlah\_terjual INT NOT NULL,

tanggal\_penjualan DATE NOT NULL,

FOREIGN KEY (id\_produksi) REFERENCES Produksi(id\_produksi)

);

**Penjelasan Struktur:**

* **Tabel Produksi**:
  + id\_produksi: Primary key yang unik untuk setiap produk.
  + nama\_produk: Nama dari produk.
  + jumlah: Jumlah produk yang diproduksi.
  + tanggal\_produksi: Tanggal ketika produk diproduksi.
* **Tabel Persediaan**:
  + id\_persediaan: Primary key yang unik untuk setiap entri persediaan.
  + id\_produksi: Foreign key yang merujuk ke id\_produksi di tabel Produksi untuk menghubungkan stok dengan produk tertentu.
  + jumlah\_stok: Jumlah produk yang tersedia di persediaan.
  + lokasi: Lokasi penyimpanan stok.
* **Tabel Penjualan**:
  + id\_penjualan: Primary key yang unik untuk setiap transaksi penjualan.
  + id\_produksi: Foreign key yang merujuk ke id\_produksi di tabel Produksi untuk menghubungkan penjualan dengan produk tertentu.
  + jumlah\_terjual: Jumlah produk yang terjual dalam transaksi.
  + tanggal\_penjualan: Tanggal ketika penjualan terjadi.

Dengan struktur ini, Anda dapat dengan mudah mengelola data produksi, persediaan, dan penjualan dalam sistem database perusahaan.

4o mini

**You said:**

contoh tabelnya

**ChatGPT said:**

ChatGPT

Berikut adalah contoh isi tabel untuk setiap tabel dalam database yang telah dibuat: Produksi, Persediaan, dan Penjualan.

**1. Tabel Produksi**

| **id\_produksi** | **nama\_produk** | **jumlah** | **tanggal\_produksi** |
| --- | --- | --- | --- |
| 1 | Produk A | 100 | 2024-01-10 |
| 2 | Produk B | 200 | 2024-01-15 |
| 3 | Produk C | 150 | 2024-01-20 |

**2. Tabel Persediaan**

| **id\_persediaan** | **id\_produksi** | **jumlah\_stok** | **lokasi** |
| --- | --- | --- | --- |
| 1 | 1 | 80 | Gudang 1 |
| 2 | 2 | 150 | Gudang 2 |
| 3 | 3 | 100 | Gudang 1 |

**3. Tabel Penjualan**

| **id\_penjualan** | **id\_produksi** | **jumlah\_terjual** | **tanggal\_penjualan** |
| --- | --- | --- | --- |
| 1 | 1 | 20 | 2024-01-12 |
| 2 | 2 | 50 | 2024-01-16 |
| 3 | 1 | 10 | 2024-01-18 |

**Penjelasan Isi Tabel:**

* **Tabel Produksi**: Menyimpan data tentang produk yang diproduksi beserta jumlah dan tanggal produksinya.
* **Tabel Persediaan**: Menghubungkan setiap produk dengan jumlah stok yang tersedia di lokasi tertentu.
* **Tabel Penjualan**: Menyimpan transaksi penjualan produk, termasuk jumlah yang terjual dan tanggal penjualannya.

Dengan contoh ini, Anda bisa mendapatkan gambaran jelas tentang bagaimana data dikelola di dalam database perusahaan.

4o mini

**You said:**

buatlah promptingnya

**ChatGPT said:**

ChatGPT

Berikut adalah beberapa contoh prompting yang bisa Anda gunakan untuk berinteraksi dengan database perusahaan yang telah dibuat. Prompting ini mencakup berbagai jenis query yang umum dilakukan.

**1. Menampilkan Semua Data Produksi**

sql

Copy code

SELECT \* FROM Produksi;

**Prompt**: "Tampilkan semua data produk yang telah diproduksi."

**2. Menampilkan Semua Data Persediaan**

sql

Copy code

SELECT \* FROM Persediaan;

**Prompt**: "Tampilkan semua data persediaan produk."

**3. Menampilkan Semua Data Penjualan**

sql

Copy code

SELECT \* FROM Penjualan;

**Prompt**: "Tampilkan semua data penjualan produk."

**4. Menampilkan Produk Tertentu**

sql

Copy code

SELECT \* FROM Produksi WHERE id\_produksi = ?;

**Prompt**: "Tampilkan data produk dengan ID tertentu."

**5. Menampilkan Stok Persediaan untuk Produk Tertentu**

sql

Copy code

SELECT p.nama\_produk, ps.jumlah\_stok

FROM Produksi p

JOIN Persediaan ps ON p.id\_produksi = ps.id\_produksi

WHERE p.id\_produksi = ?;

**Prompt**: "Tampilkan stok persediaan untuk produk dengan ID tertentu."

**6. Menampilkan Penjualan untuk Produk Tertentu**

sql

Copy code

SELECT p.nama\_produk, j.jumlah\_terjual, j.tanggal\_penjualan

FROM Produksi p

JOIN Penjualan j ON p.id\_produksi = j.id\_produksi

WHERE p.id\_produksi = ?;

**Prompt**: "Tampilkan data penjualan untuk produk dengan ID tertentu."

**7. Menambahkan Data Produksi Baru**

sql

Copy code

INSERT INTO Produksi (nama\_produk, jumlah, tanggal\_produksi) VALUES (?, ?, ?);

**Prompt**: "Tambahkan produk baru ke dalam tabel produksi."

**8. Mengupdate Jumlah Stok di Persediaan**

sql

Copy code

UPDATE Persediaan SET jumlah\_stok = ? WHERE id\_persediaan = ?;

**Prompt**: "Perbarui jumlah stok untuk persediaan dengan ID tertentu."

**9. Menghapus Data Penjualan**

sql

Copy code

DELETE FROM Penjualan WHERE id\_penjualan = ?;

**Prompt**: "Hapus data penjualan dengan ID tertentu."

**10. Menampilkan Total Jumlah Penjualan untuk Setiap Produk**

sql

Copy code

SELECT p.nama\_produk, SUM(j.jumlah\_terjual) AS total\_terjual

FROM Produksi p

JOIN Penjualan j ON p.id\_produksi = j.id\_produksi

GROUP BY p.nama\_produk;

**Prompt**: "Tampilkan total jumlah penjualan untuk setiap produk."

Dengan prompting ini, Anda dapat melakukan berbagai operasi pada database perusahaan dengan mudah.

4o mini

**You said:**

boatman data cleaning dari tabel diatas

**ChatGPT said:**

ChatGPT

Berikut adalah langkah-langkah untuk melakukan data cleaning pada tabel-tabel dalam database Produksi, Persediaan, dan Penjualan. Data cleaning bertujuan untuk memastikan bahwa data yang disimpan adalah akurat, konsisten, dan dapat digunakan untuk analisis lebih lanjut.

**1. Tabel Produksi**

**a. Memeriksa Duplikasi**

sql

Copy code

SELECT nama\_produk, COUNT(\*)

FROM Produksi

GROUP BY nama\_produk

HAVING COUNT(\*) > 1;

**Tindakan**: Hapus atau gabungkan entri duplikat.

**b. Memeriksa Nilai Null**

sql

Copy code

SELECT \*

FROM Produksi

WHERE nama\_produk IS NULL OR jumlah IS NULL OR tanggal\_produksi IS NULL;

**Tindakan**: Isi atau hapus entri yang memiliki nilai null.

**c. Memeriksa Format Tanggal**

Pastikan semua tanggal\_produksi berada dalam format yang benar (YYYY-MM-DD).

**2. Tabel Persediaan**

**a. Memeriksa Duplikasi**

sql

Copy code

SELECT id\_produksi, COUNT(\*)

FROM Persediaan

GROUP BY id\_produksi

HAVING COUNT(\*) > 1;

**Tindakan**: Hapus atau gabungkan entri duplikat.

**b. Memeriksa Nilai Null**

sql

Copy code

SELECT \*

FROM Persediaan

WHERE id\_produksi IS NULL OR jumlah\_stok IS NULL OR lokasi IS NULL;

**Tindakan**: Isi atau hapus entri yang memiliki nilai null.

**c. Memeriksa Ketersediaan Produk**

sql

Copy code

SELECT \*

FROM Persediaan ps

LEFT JOIN Produksi p ON ps.id\_produksi = p.id\_produksi

WHERE p.id\_produksi IS NULL;

**Tindakan**: Hapus entri persediaan yang merujuk pada produk yang tidak ada di tabel Produksi.

**3. Tabel Penjualan**

**a. Memeriksa Duplikasi**

sql

Copy code

SELECT id\_produk, COUNT(\*)

FROM Penjualan

GROUP BY id\_produk

HAVING COUNT(\*) > 1;

**Tindakan**: Hapus atau gabungkan entri duplikat.

**b. Memeriksa Nilai Null**

sql

Copy code

SELECT \*

FROM Penjualan

WHERE id\_produksi IS NULL OR jumlah\_terjual IS NULL OR tanggal\_penjualan IS NULL;

**Tindakan**: Isi atau hapus entri yang memiliki nilai null.

**c. Memeriksa Ketersediaan Produk**

sql

Copy code

SELECT \*

FROM Penjualan j

LEFT JOIN Produksi p ON j.id\_produksi = p.id\_produksi

WHERE p.id\_produksi IS NULL;

**Tindakan**: Hapus entri penjualan yang merujuk pada produk yang tidak ada di tabel Produksi.

**4. Konsistensi Data**

**a. Memeriksa Rentang Nilai**

Pastikan jumlah produk dan stok tidak negatif:

sql

Copy code

SELECT \*

FROM Produksi

WHERE jumlah < 0;

SELECT \*

FROM Persediaan

WHERE jumlah\_stok < 0;

SELECT \*

FROM Penjualan

WHERE jumlah\_terjual < 0;

**Tindakan**: Hapus atau perbaiki nilai negatif.

**5. Normalisasi**

Jika ada informasi yang berulang (misalnya lokasi penyimpanan), pertimbangkan untuk membuat tabel baru untuk menghindari redundansi data.

**Kesimpulan**

Dengan mengikuti langkah-langkah di atas, Anda dapat memastikan bahwa data dalam tabel Produksi, Persediaan, dan Penjualan bersih, akurat, dan siap untuk analisis lebih lanjut. Pastikan untuk melakukan backup data sebelum melakukan penghapusan atau perubahan besar pada data.

4o mini

**You said:**

boatman tabel data cleaningnya

**ChatGPT said:**

ChatGPT

Berikut adalah contoh tabel yang dapat digunakan untuk mencatat proses data cleaning dari tabel Produksi, Persediaan, dan Penjualan. Tabel ini mencakup informasi tentang entri yang perlu dibersihkan, tindakan yang diambil, dan hasil setelah pembersihan.

**Tabel Data Cleaning**

| **id\_cleaning** | **tabel** | **id\_entri** | **deskripsi\_masalah** | **tindakan\_diambil** | **hasil** | **tanggal\_dibersihkan** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | Produksi | 2 | Duplikat nama produk "Produk A" | Menghapus salah satu entri | Berhasil dihapus | 2024-01-20 |
| 2 | Produksi | 3 | Nilai NULL pada tanggal\_produksi | Mengisi tanggal produksi | Berhasil diperbaiki | 2024-01-20 |
| 3 | Persediaan | 1 | Nilai NULL pada lokasi | Mengisi lokasi penyimpanan | Berhasil diperbaiki | 2024-01-20 |
| 4 | Persediaan | 4 | id\_produksi tidak ada di tabel Produksi | Menghapus entri | Berhasil dihapus | 2024-01-20 |
| 5 | Penjualan | 2 | Nilai NULL pada jumlah\_terjual | Mengisi jumlah terjual | Berhasil diperbaiki | 2024-01-20 |
| 6 | Penjualan | 3 | id\_produksi tidak ada di tabel Produksi | Menghapus entri | Berhasil dihapus | 2024-01-20 |
| 7 | Produksi | 1 | Jumlah negatif | Mengubah jumlah menjadi 0 | Berhasil diperbaiki | 2024-01-20 |
| 8 | Persediaan | 2 | Jumlah\_stok negatif | Mengubah jumlah stok menjadi 0 | Berhasil diperbaiki | 2024-01-20 |
| 9 | Penjualan | 1 | Jumlah terjual negatif | Menghapus entri | Berhasil dihapus | 2024-01-20 |

**Penjelasan Kolom:**

* **id\_cleaning**: ID unik untuk setiap entri proses pembersihan.
* **tabel**: Nama tabel yang dibersihkan (Produksi, Persediaan, Penjualan).
* **id\_entri**: ID entri yang bermasalah dalam tabel.
* **deskripsi\_masalah**: Penjelasan mengenai masalah yang ditemukan.
* **tindakan\_diambil**: Tindakan yang diambil untuk memperbaiki masalah.
* **hasil**: Status hasil dari tindakan yang diambil (misalnya, "Berhasil diperbaiki", "Berhasil dihapus").
* **tanggal\_dibersihkan**: Tanggal ketika pembersihan dilakukan.

Dengan tabel ini, Anda dapat melacak proses data cleaning dengan lebih terstruktur dan jelas.

4o mini

**You said:**

buatkan tabel data cleaningnya masing masing dari tabel produksi, tabel persmediaan dan tabel penjualan

**ChatGPT said:**

ChatGPT

Berikut adalah contoh tabel data cleaning yang terpisah untuk masing-masing tabel: Produksi, Persediaan, dan Penjualan.

**1. Tabel Data Cleaning untuk Produksi**

| **id\_cleaning** | **id\_produksi** | **deskripsi\_masalah** | **tindakan\_diambil** | **hasil** | **tanggal\_dibersihkan** |
| --- | --- | --- | --- | --- | --- |
| 1 | 1 | Duplikat nama produk "Produk A" | Menghapus salah satu entri | Berhasil dihapus | 2024-01-20 |
| 2 | 3 | Nilai NULL pada tanggal\_produksi | Mengisi tanggal produksi | Berhasil diperbaiki | 2024-01-20 |
| 3 | 2 | Jumlah negatif | Mengubah jumlah menjadi 0 | Berhasil diperbaiki | 2024-01-20 |

**2. Tabel Data Cleaning untuk Persediaan**

| **id\_cleaning** | **id\_persediaan** | **deskripsi\_masalah** | **tindakan\_diambil** | **hasil** | **tanggal\_dibersihkan** |
| --- | --- | --- | --- | --- | --- |
| 1 | 1 | Nilai NULL pada lokasi | Mengisi lokasi penyimpanan | Berhasil diperbaiki | 2024-01-20 |
| 2 | 3 | id\_produksi tidak ada di Produksi | Menghapus entri | Berhasil dihapus | 2024-01-20 |
| 3 | 2 | Jumlah\_stok negatif | Mengubah jumlah stok menjadi 0 | Berhasil diperbaiki | 2024-01-20 |

**3. Tabel Data Cleaning untuk Penjualan**

| **id\_cleaning** | **id\_penjualan** | **deskripsi\_masalah** | **tindakan\_diambil** | **hasil** | **tanggal\_dibersihkan** |
| --- | --- | --- | --- | --- | --- |
| 1 | 2 | Nilai NULL pada jumlah\_terjual | Mengisi jumlah terjual | Berhasil diperbaiki | 2024-01-20 |
| 2 | 3 | id\_produksi tidak ada di Produksi | Menghapus entri | Berhasil dihapus | 2024-01-20 |
| 3 | 1 | Jumlah terjual negatif | Menghapus entri | Berhasil dihapus | 2024-01-20 |

**Penjelasan Kolom:**

* **id\_cleaning**: ID unik untuk setiap entri proses pembersihan.
* **id\_produksi** / **id\_persediaan** / **id\_penjualan**: ID entri yang bermasalah dalam tabel masing-masing.
* **deskripsi\_masalah**: Penjelasan mengenai masalah yang ditemukan.
* **tindakan\_diambil**: Tindakan yang diambil untuk memperbaiki masalah.
* **hasil**: Status hasil dari tindakan yang diambil (misalnya, "Berhasil diperbaiki", "Berhasil dihapus").
* **tanggal\_dibersihkan**: Tanggal ketika pembersihan dilakukan.

Dengan tabel-tabel ini, Anda dapat memantau dan mendokumentasikan proses data cleaning secara terpisah untuk setiap tabel dalam database.

4o mini

**You said:**

ubah ketiga data menjadi excel

**ChatGPT said:**

ChatGPT

Berikut adalah langkah-langkah untuk mengubah data cleaning yang telah dibuat menjadi format Excel. Anda dapat menyalin data ini ke dalam Excel:

**1. Tabel Data Cleaning untuk Produksi**

| **id\_cleaning** | **id\_produksi** | **deskripsi\_masalah** | **tindakan\_diambil** | **hasil** | **tanggal\_dibersihkan** |
| --- | --- | --- | --- | --- | --- |
| 1 | 1 | Duplikat nama produk "Produk A" | Menghapus salah satu entri | Berhasil dihapus | 2024-01-20 |
| 2 | 3 | Nilai NULL pada tanggal\_produksi | Mengisi tanggal produksi | Berhasil diperbaiki | 2024-01-20 |
| 3 | 2 | Jumlah negatif | Mengubah jumlah menjadi 0 | Berhasil diperbaiki | 2024-01-20 |

**2. Tabel Data Cleaning untuk Persediaan**

| **id\_cleaning** | **id\_persediaan** | **deskripsi\_masalah** | **tindakan\_diambil** | **hasil** | **tanggal\_dibersihkan** |
| --- | --- | --- | --- | --- | --- |
| 1 | 1 | Nilai NULL pada lokasi | Mengisi lokasi penyimpanan | Berhasil diperbaiki | 2024-01-20 |
| 2 | 3 | id\_produksi tidak ada di Produksi | Menghapus entri | Berhasil dihapus | 2024-01-20 |
| 3 | 2 | Jumlah\_stok negatif | Mengubah jumlah stok menjadi 0 | Berhasil diperbaiki | 2024-01-20 |

**3. Tabel Data Cleaning untuk Penjualan**

| **id\_cleaning** | **id\_penjualan** | **deskripsi\_masalah** | **tindakan\_diambil** | **hasil** | **tanggal\_dibersihkan** |
| --- | --- | --- | --- | --- | --- |
| 1 | 2 | Nilai NULL pada jumlah\_terjual | Mengisi jumlah terjual | Berhasil diperbaiki | 2024-01-20 |
| 2 | 3 | id\_produksi tidak ada di Produksi | Menghapus entri | Berhasil dihapus | 2024-01-20 |
| 3 | 1 | Jumlah terjual negatif | Menghapus entri | Berhasil dihapus | 2024-01-20 |

**Cara Menyalin ke Excel:**

1. **Buka Microsoft Excel**.
2. **Buat lembar kerja baru**.
3. **Salin tabel di atas** (mulai dari header hingga baris terakhir).
4. **Tempel ke dalam lembar kerja** Excel (gunakan Ctrl + V atau klik kanan dan pilih "Paste").
5. **Atur lebar kolom** jika diperlukan agar semua teks terlihat jelas.

Setelah melakukan langkah-langkah di atas, Anda akan memiliki tiga tabel yang terpisah di dalam satu file Excel. Anda juga bisa menyimpan file dengan nama yang sesuai, seperti "Data\_Cleaning\_Produksi\_Persediaan\_Penjualan.xlsx".

4o mini

**You said:**

buatlah tabel produksi, tabel persmediaan dan tabel penjualan sebanyak 20 baris

**ChatGPT said:**

ChatGPT

Berikut adalah contoh tabel untuk Produksi, Persediaan, dan Penjualan dengan masing-masing memiliki 20 baris data.

**1. Tabel Produksi**

| **id\_produksi** | **nama\_produk** | **jumlah** | **tanggal\_produksi** |
| --- | --- | --- | --- |
| 1 | Produk A | 100 | 2024-01-01 |
| 2 | Produk B | 200 | 2024-01-02 |
| 3 | Produk C | 150 | 2024-01-03 |
| 4 | Produk D | 300 | 2024-01-04 |
| 5 | Produk E | 250 | 2024-01-05 |
| 6 | Produk F | 400 | 2024-01-06 |
| 7 | Produk G | 350 | 2024-01-07 |
| 8 | Produk H | 500 | 2024-01-08 |
| 9 | Produk I | 450 | 2024-01-09 |
| 10 | Produk J | 600 | 2024-01-10 |
| 11 | Produk K | 550 | 2024-01-11 |
| 12 | Produk L | 700 | 2024-01-12 |
| 13 | Produk M | 650 | 2024-01-13 |
| 14 | Produk N | 800 | 2024-01-14 |
| 15 | Produk O | 750 | 2024-01-15 |
| 16 | Produk P | 900 | 2024-01-16 |
| 17 | Produk Q | 850 | 2024-01-17 |
| 18 | Produk R | 950 | 2024-01-18 |
| 19 | Produk S | 1000 | 2024-01-19 |
| 20 | Produk T | 1100 | 2024-01-20 |

**2. Tabel Persediaan**

| **id\_persediaan** | **id\_produksi** | **jumlah\_stok** | **lokasi** |
| --- | --- | --- | --- |
| 1 | 1 | 80 | Gudang 1 |
| 2 | 2 | 150 | Gudang 2 |
| 3 | 3 | 100 | Gudang 1 |
| 4 | 4 | 250 | Gudang 3 |
| 5 | 5 | 200 | Gudang 2 |
| 6 | 6 | 300 | Gudang 1 |
| 7 | 7 | 250 | Gudang 3 |
| 8 | 8 | 400 | Gudang 2 |
| 9 | 9 | 350 | Gudang 1 |
| 10 | 10 | 500 | Gudang 3 |
| 11 | 11 | 450 | Gudang 2 |
| 12 | 12 | 600 | Gudang 1 |
| 13 | 13 | 550 | Gudang 3 |
| 14 | 14 | 700 | Gudang 2 |
| 15 | 15 | 650 | Gudang 1 |
| 16 | 16 | 800 | Gudang 3 |
| 17 | 17 | 750 | Gudang 2 |
| 18 | 18 | 900 | Gudang 1 |
| 19 | 19 | 850 | Gudang 3 |
| 20 | 20 | 1000 | Gudang 2 |

**3. Tabel Penjualan**

| **id\_penjualan** | **id\_produksi** | **jumlah\_terjual** | **tanggal\_penjualan** |
| --- | --- | --- | --- |
| 1 | 1 | 20 | 2024-01-05 |
| 2 | 2 | 50 | 2024-01-06 |
| 3 | 3 | 30 | 2024-01-07 |
| 4 | 4 | 60 | 2024-01-08 |
| 5 | 5 | 40 | 2024-01-09 |
| 6 | 6 | 70 | 2024-01-10 |
| 7 | 7 | 80 | 2024-01-11 |
| 8 | 8 | 90 | 2024-01-12 |
| 9 | 9 | 20 | 2024-01-13 |
| 10 | 10 | 50 | 2024-01-14 |
| 11 | 11 | 30 | 2024-01-15 |
| 12 | 12 | 60 | 2024-01-16 |
| 13 | 13 | 40 | 2024-01-17 |
| 14 | 14 | 70 | 2024-01-18 |
| 15 | 15 | 90 | 2024-01-19 |
| 16 | 16 | 20 | 2024-01-20 |
| 17 | 17 | 50 | 2024-01-21 |
| 18 | 18 | 30 | 2024-01-22 |
| 19 | 19 | 60 | 2024-01-23 |
| 20 | 20 | 40 | 2024-01-24 |

Dengan data di atas, Anda memiliki tiga tabel yang lengkap dengan masing-masing 20 baris data. Anda dapat menyalin dan menempelkan data ini ke dalam spreadsheet Excel untuk pengelolaan lebih lanjut.

4o mini

**You said:**

dari tabel penjualan, manakah produksi tertinggi antara tanggal 1 sampai tanggal 20

**ChatGPT said:**

ChatGPT

Untuk menentukan produk dengan penjualan tertinggi antara tanggal 1 hingga 20 Januari, kita akan menjumlahkan jumlah\_terjual dari tabel Penjualan berdasarkan id\_produksi dan kemudian mencari produk dengan total penjualan terbesar.

Berikut adalah langkah-langkah untuk menganalisis data:

**Total Penjualan per Produk**

| **id\_produksi** | **total\_penjualan** |
| --- | --- |
| 1 | 20 |
| 2 | 50 |
| 3 | 30 |
| 4 | 60 |
| 5 | 40 |
| 6 | 70 |
| 7 | 80 |
| 8 | 90 |
| 9 | 20 |
| 10 | 50 |
| 11 | 30 |
| 12 | 60 |
| 13 | 40 |
| 14 | 70 |
| 15 | 90 |
| 16 | 20 |
| 17 | 50 |
| 18 | 30 |
| 19 | 60 |
| 20 | 40 |

**Menghitung Total Penjualan**

Berdasarkan data di atas, kita akan menjumlahkan total penjualan untuk setiap produk:

* Produk A (id 1): 20
* Produk B (id 2): 50
* Produk C (id 3): 30
* Produk D (id 4): 60
* Produk E (id 5): 40
* Produk F (id 6): 70
* Produk G (id 7): 80
* Produk H (id 8): 90
* Produk I (id 9): 20
* Produk J (id 10): 50
* Produk K (id 11): 30
* Produk L (id 12): 60
* Produk M (id 13): 40
* Produk N (id 14): 70
* Produk O (id 15): 90
* Produk P (id 16): 20
* Produk Q (id 17): 50
* Produk R (id 18): 30
* Produk S (id 19): 60
* Produk T (id 20): 40

**Hasil**

Dari total penjualan di atas, **Produk H (id 8)** dan **Produk O (id 15)** memiliki penjualan tertinggi, yaitu **90 unit** antara tanggal 1 hingga 20 Januari.

4o mini

**You said:**

dari tabel penjualan, manakah penjualan tertinggi antara tanggal 1 sampai tanggal 20 di id genap

**ChatGPT said:**

ChatGPT

Untuk menentukan produk dengan penjualan tertinggi antara tanggal 1 hingga 20 Januari dari tabel Penjualan berdasarkan ID genap, kita akan menganalisis produk dengan ID genap dan menjumlahkan jumlah\_terjual mereka.

**Penjualan untuk ID Genap**

Mari kita lihat data penjualan untuk produk dengan ID genap:

| **id\_penjualan** | **id\_produksi** | **jumlah\_terjual** | **tanggal\_penjualan** |
| --- | --- | --- | --- |
| 2 | 2 | 50 | 2024-01-06 |
| 4 | 4 | 60 | 2024-01-08 |
| 6 | 6 | 70 | 2024-01-10 |
| 8 | 8 | 90 | 2024-01-12 |
| 10 | 10 | 50 | 2024-01-14 |
| 12 | 12 | 60 | 2024-01-16 |
| 14 | 14 | 70 | 2024-01-18 |
| 16 | 16 | 20 | 2024-01-20 |
| 18 | 18 | 30 | 2024-01-22 |
| 20 | 20 | 40 | 2024-01-24 |

**Total Penjualan untuk ID Genap**

* **Produk B (id 2)**: 50
* **Produk D (id 4)**: 60
* **Produk F (id 6)**: 70
* **Produk H (id 8)**: 90
* **Produk J (id 10)**: 50
* **Produk L (id 12)**: 60
* **Produk N (id 14)**: 70
* **Produk P (id 16)**: 20
* **Produk R (id 18)**: 30
* **Produk T (id 20)**: 40

**Hasil**

Dari total penjualan di atas, **Produk H (id 8)** memiliki penjualan tertinggi dengan **90 unit** antara tanggal 1 hingga 20 Januari, di antara produk dengan ID genap.

4o mini

**You said:**

dari database perusahaan diatas buatlah data analitik, penjualan tertinggi nutuk tanggal produksi dennen rentan nama produksi tanggal 01/01/2024 dampai 20/01/2024 dennen id produksi genap, carikan penjualan yang tertinggi dari tanggal 01/01/2024 dampai tanggal 20 januari 2024 yang id produksi genap

**ChatGPT said:**

ChatGPT

Untuk mendapatkan analisis penjualan tertinggi untuk produk dengan ID genap dari tanggal produksi 1 Januari 2024 hingga 20 Januari 2024, kita perlu melihat data penjualan dan memfilter berdasarkan kriteria yang diberikan. Berikut adalah langkah-langkah untuk mendapatkan data tersebut:

**Langkah-langkah Analisis**

1. **Identifikasi ID Genap**:
   * Produk dengan ID genap dari tabel Produksi: 2, 4, 6, 8, 10, 12, 14, 16, 18, 20.
2. **Filter Tanggal Produksi**:
   * Ambil data dari tabel Produksi yang tanggal produksinya berada dalam rentang 01/01/2024 sampai 20/01/2024.
3. **Ambil Data Penjualan**:
   * Gabungkan hasil dari langkah 2 dengan tabel Penjualan untuk mendapatkan jumlah\_terjual untuk setiap produk dengan ID genap.
4. **Hitung Total Penjualan**:
   * Hitung total penjualan untuk setiap produk.
5. **Temukan Penjualan Tertinggi**:
   * Temukan produk dengan total penjualan tertinggi.

**Contoh Data**

**Tabel Produksi (Hanya ID Genap)**

| **id\_produksi** | **nama\_produk** | **jumlah** | **tanggal\_produksi** |
| --- | --- | --- | --- |
| 2 | Produk B | 200 | 2024-01-02 |
| 4 | Produk D | 300 | 2024-01-04 |
| 6 | Produk F | 400 | 2024-01-06 |
| 8 | Produk H | 500 | 2024-01-08 |
| 10 | Produk J | 600 | 2024-01-10 |
| 12 | Produk L | 700 | 2024-01-12 |
| 14 | Produk N | 800 | 2024-01-14 |
| 16 | Produk P | 900 | 2024-01-16 |
| 18 | Produk R | 950 | 2024-01-18 |
| 20 | Produk T | 1100 | 2024-01-20 |

**Tabel Penjualan (ID Genap)**

| **id\_penjualan** | **id\_produksi** | **jumlah\_terjual** | **tanggal\_penjualan** |
| --- | --- | --- | --- |
| 2 | 2 | 50 | 2024-01-06 |
| 4 | 4 | 60 | 2024-01-08 |
| 6 | 6 | 70 | 2024-01-10 |
| 8 | 8 | 90 | 2024-01-12 |
| 10 | 10 | 50 | 2024-01-14 |
| 12 | 12 | 60 | 2024-01-16 |
| 14 | 14 | 70 | 2024-01-18 |
| 16 | 16 | 20 | 2024-01-20 |

**Total Penjualan untuk ID Genap**

* **Produk B (id 2)**: 50
* **Produk D (id 4)**: 60
* **Produk F (id 6)**: 70
* **Produk H (id 8)**: 90
* **Produk J (id 10)**: 50
* **Produk L (id 12)**: 60
* **Produk N (id 14)**: 70
* **Produk P (id 16)**: 20
* **Produk R (id 18)**: 30
* **Produk T (id 20)**: 40

**Penjualan Tertinggi**

Dari data di atas, **Produk H (id 8)** memiliki penjualan tertinggi dengan **90 unit** antara tanggal 01/01/2024 hingga 20/01/2024 dan ID genap.

**Kesimpulan**

* **Produk dengan ID Genap**: Produk H (id 8)
* **Jumlah Terjual**: 90 unit