



## Project Based Internship

# Import Data & Querying, Sorting, Filtering

**Mastering Data Import and Querying in SQL: The Ultimate Guide to Sorting, Filtering, and Optimizing Your Database**



## Daftar Isi

A. Introduction	3
B. Importing Data in SQL	3
C. Querying, Sorting and Filtering	6
CASE STUDY : E-COMMERCE DATABASE	8
REFERENCE	10

## A. Introduction

Structured Query Language (SQL) adalah alat yang ampuh dan serbaguna untuk mengelola dan menganalisis data dalam berbagai pengaturan. Apakah Anda bekerja dengan database kecil untuk proyek pribadi atau database perusahaan besar untuk tujuan bisnis, mengimpor data dan melakukan kueri secara efektif adalah keterampilan penting. Dalam artikel ini, kita akan menjelajahi proses mengimpor data ke database SQL dan menunjukkan kepada Anda cara melakukan querying, sorting, dan filtering untuk mendapatkan insight tentang data Anda. Dengan teknik ini, Anda akan dapat mengoptimalkan kinerja database, merampingkan manajemen data, dan membuat keputusan berdasarkan informasi yang akurat dan terkini. Jadi, mari selami dunia SQL dan tingkatkan keterampilan manajemen data Anda ke level selanjutnya!

## B. Importing Data in SQL

Importing data dalam SQL adalah tugas umum untuk administrator database dan pengembang yang perlu mengisi atau memperbarui database mereka dengan sejumlah besar informasi. SQL menyediakan beberapa metode untuk mengimpor data, termasuk penggunaan file eksternal, utilitas common-line, dan antarmuka pemrograman. Pada artikel ini, kita akan mengeksplorasi berbagai cara mengimpor data dalam SQL, dengan contoh praktis mengimpor data dari file CSV.

- Mengimpor data dari file CSV menggunakan SQL

File CSV (Comma Separated Values) adalah format umum untuk bertukar data antar aplikasi yang berbeda. Dalam SQL, mengimpor data dari file CSV



relatif mudah, dan dapat dilakukan dengan menggunakan langkah-langkah berikut:

1. Buat tabel baru di database untuk menyimpan data yang diimpor. Anda dapat menggunakan pernyataan `CREATE TABLE` untuk menentukan kolom dan tipe datanya yang cocok dengan struktur file CSV.
2. Buka file CSV di editor teks untuk memastikan formatnya benar dan berisi data yang ingin Anda impor.
3. Gunakan pernyataan `LOAD DATA INFILE` untuk memuat data CSV ke dalam tabel yang baru dibuat. Pernyataan ini khusus untuk MySQL dan memerlukan izin yang sesuai. Kode berikut menunjukkan contoh cara mengimpor data dari file CSV menggunakan pernyataan `LOAD DATA INFILE`:

```
LOAD DATA INFILE '/path/to/csv/file.csv'  
INTO TABLE my_table  
FIELDS TERMINATED BY ','  
ENCLOSED BY '"'  
LINES TERMINATED BY '\n'  
IGNORE 1 ROWS;
```

Pernyataan ini menentukan jalur ke file CSV, nama tabel target, pembatas yang digunakan untuk memisahkan bidang, karakter yang digunakan untuk menyertakan string, dan terminator garis. Parameter `IGNORE 1 ROWS` menunjukkan bahwa baris pertama file CSV berisi tajuk kolom dan harus dilewati selama impor.

4. Verifikasi bahwa data telah diimpor dengan benar dengan menjalankan pernyataan `SELECT` pada tabel yang baru dibuat. Anda juga dapat

menggunakan pernyataan DESCRIBE untuk memeriksa struktur tabel dan memastikannya cocok dengan kolom file CSV.

- Metode alternatif untuk mengimpor data dalam SQL

Sementara mengimpor data dari file CSV adalah cara sederhana dan efektif untuk mengisi database SQL, metode lain tersedia tergantung pada jenis data yang diimpor dan database target. Beberapa metode yang paling umum meliputi:

- a. Menggunakan tools GUI: Banyak tools GUI SQL, seperti SQL Server Management Studio, Oracle SQL Developer, atau MySQL Workbench, menyediakan wizard impor yang dapat memandu Anda melalui proses mengimpor data dari berbagai sumber, termasuk Excel, Access, dan database lainnya .
- b. Menggunakan utilitas common-line: Beberapa database menyediakan utilitas common-line untuk mengimpor data, seperti utilitas bcp untuk SQL Server atau utilitas sqldr untuk Oracle. Alat ini dapat digunakan untuk mengimpor data dari berbagai format file atau sumber database, dengan opsi yang lebih canggih untuk pemformatan dan transformasi.
- c. Menggunakan antarmuka pemrograman: Untuk impor data yang lebih kompleks atau umpan data real-time, SQL menyediakan antarmuka pemrograman seperti JDBC (Java Database Connectivity), ODBC (Open Database Connectivity), atau ADO.NET (ActiveX Data Objects for .NET). Antarmuka ini memungkinkan Anda menulis kode khusus untuk terhubung ke database, menjalankan kueri, dan mengambil atau memasukkan data dalam berbagai format.

## C. Querying, Sorting and Filtering

Structured Query Language (SQL) adalah bahasa pemrograman yang banyak digunakan untuk mengelola dan memanipulasi database relasional. Salah satu fitur utama SQL adalah kemampuan untuk membuat kueri, mengurutkan, dan memfilter data dalam database. Dalam artikel ini, kita akan membahas dasar-dasar kueri, pengurutan, dan pemfilteran data dalam SQL, serta memberikan contoh dan studi kasus untuk mengilustrasikan penggunaannya.

- Querying Data Pada SQL

Querying data dalam SQL melibatkan pemilihan data spesifik dari database berdasarkan kondisi atau kriteria tertentu. Kueri SQL paling dasar adalah pernyataan SELECT, yang digunakan untuk mengambil data dari satu atau lebih tabel dalam database. Sintaks dasar untuk pernyataan SELECT adalah sebagai berikut:

```
SELECT column_1, column_2, ..., column_n
```

```
FROM table_name
```

```
WHERE conditions;
```

Dalam sintaks ini, SELECT menentukan kolom yang ingin Anda ambil, FROM menentukan tabel tempat Anda ingin mengambil data, dan WHERE menentukan kondisi untuk memfilter data.

- Sorting Data Pada SQL

Pengurutan data dalam SQL melibatkan pengurutan hasil kueri berdasarkan satu atau beberapa kolom. Klausa ORDER BY digunakan untuk mengurutkan baris dalam kumpulan hasil. Misalnya:



```
SELECT column1, column2, ... FROM table_name WHERE condition ORDER BY  
column1, column2, ... ASC/DESC;
```

Dalam contoh di atas, ASC dan DESC menentukan urutan hasil yang harus diurutkan (naik atau turun). Secara default, hasil diurutkan dalam urutan menaik.

- Filtering Data Pada SQL

Memfilter data dalam SQL hanya melibatkan pemilihan baris tertentu berdasarkan kriteria tertentu. Klausa WHERE digunakan untuk memfilter baris berdasarkan satu atau lebih kondisi, seperti yang kita lihat sebelumnya. Selain klausa WHERE, SQL menyediakan beberapa cara lain untuk memfilter data, termasuk operator LIKE, operator IN, dan operator BETWEEN.

- Operator LIKE digunakan untuk mencocokkan baris berdasarkan pola. Sebagai contoh, kita dapat menggunakan operator LIKE untuk memilih semua baris dimana kolom tertentu dimulai dengan huruf tertentu:

```
SELECT column1, column2, ... FROM table_name WHERE column1 LIKE 'A%';
```

- Operator IN digunakan untuk mencocokkan baris berdasarkan daftar nilai. Misalnya, kita dapat menggunakan operator IN untuk memilih semua baris dimana kolom tertentu adalah salah satu dari sekumpulan nilai tertentu:

```
SELECT column1, column2, ... FROM table_name WHERE column1 IN ('value1',  
'value2', ...);
```

- Operator BETWEEN digunakan untuk mencocokkan baris berdasarkan rentang nilai. Sebagai contoh, kita dapat menggunakan operator BETWEEN

untuk memilih semua baris di mana kolom tertentu berada di antara dua nilai:

```
SELECT column1, column2, ... FROM table_name WHERE column1 BETWEEN  
value1 AND value2;
```

## CASE STUDY : E-COMMERCE DATABASE

E-commerce database berisi beberapa tabel, termasuk customers, orders, products, dan reviews. Tabel customers berisi informasi tentang setiap pelanggan, termasuk nama, email, dan alamat pengiriman mereka. Tabel orders berisi informasi tentang setiap pesanan, termasuk pelanggan yang melakukan pembelian, produk yang dibeli, tanggal penjualan, dan status pesanan. Tabel products berisi informasi tentang setiap produk, termasuk nama, deskripsi, harga, dan tingkat inventarisnya. Tabel reviews berisi informasi tentang setiap ulasan, termasuk pelanggan yang menulis ulasan, produk yang diulas, dan konten ulasan.

Misalkan kita ingin mengambil daftar produk dengan peringkat teratas dengan setidaknya 10 ulasan dan peringkat 4 atau lebih tinggi, diurutkan berdasarkan jumlah ulasan dalam urutan menurun. Kita dapat melakukannya dengan menggunakan kueri SQL kompleks yang melibatkan pemfilteran dan penyortiran. Kueri akan terlihat seperti ini:

```
SELECT products.name, AVG(reviews.rating) AS avg_rating,  
COUNT(reviews.product_id) AS review_count  
FROM products  
INNER JOIN reviews ON products.id = reviews.product_id  
WHERE reviews.rating >= 4
```



```
GROUP BY products.id  
HAVING COUNT(reviews.product_id) >= 10  
ORDER BY review_count DESC;
```

Mari uraikan kueri ini langkah demi langkah.

- Pertama, kami memilih nama produk, peringkat rata-rata untuk setiap produk, dan jumlah ulasan untuk setiap produk. Kita menggunakan fungsi AVG untuk menghitung peringkat rata-rata untuk setiap produk dan fungsi COUNT untuk menghitung jumlah ulasan untuk setiap produk. Kita juga memberi masing-masing kolom ini sebuah alias menggunakan kata kunci AS untuk membuat output lebih mudah dibaca.
- Selanjutnya, Kita menggunakan gabungan dalam untuk menggabungkan tabel produk dengan tabel ulasan pada ID produk. Hal ini memastikan bahwa Kita hanya mengambil data untuk produk yang memiliki setidaknya satu ulasan.
- Kemudian, kita menggunakan klausa WHERE untuk memfilter hasil agar hanya menyertakan produk dengan peringkat 4 atau lebih tinggi.
- Setelah itu, kita menggunakan klausa GROUP BY untuk mengelompokkan hasil berdasarkan ID produk. Hal ini memastikan bahwa kita menghitung nilai rata-rata dan jumlah ulasan untuk setiap produk secara terpisah.
- Selanjutnya, kita menggunakan klausa HAVING untuk memfilter hasil agar hanya menyertakan produk dengan setidaknya 10 ulasan. Klausa HAVING mirip dengan klausa WHERE, tetapi berlaku untuk grup daripada baris individual.
- Terakhir, kita menggunakan klausa ORDER BY untuk mengurutkan hasil berdasarkan jumlah ulasan dalam urutan menurun. Ini memastikan bahwa produk dengan ulasan terbanyak dicantumkan terlebih dahulu.



## REFERENCE

<https://learn.microsoft.com/en-us/sql/integration-services/import-export-data/start-the-sql-server-import-and-export-wizard?view=sql-server-ver16>

<https://www.sqlshack.com/learn-mysql-sorting-and-filtering-data-in-a-table/>

<https://www.w3schools.com/sql/>