Apa itu SQL?

SQL yang merupakan singkatan dari Structured Query Language, yaitu bahasa komputer standar yang digunakan untuk berinteraksi dengan suatu **sistem database** - atau lebih tepatnya ***sistem manajemen database relasional***. Jadi, user dapat menambahkan, mengubah, mengupdate, mencari dan menghapus data dari suatu sistem database dengan menggunakan SQL.

SQL dilafalkan dengan membaca tiap karakternya S Q L (es kiu el) atau sikuel.

Contoh komunikasi SQL

Terdapat dua kategori dari interaksi SQL:

* ***Data Definition Language*** (DDL), yaitu berbagai perintah yang berfungsi lebih kepada memanipulasi struktur database, seperti Membuat (CREATE), merubah (ALTER), dan menghapus (DROP) struktur penyimpanan data, yaitu database, table, kolom dan tipe data.
* ***Data Manipulation Language*** (DML), yaitu berbagai perintah yang digunakan untuk Menyisipkan data (INSERT), Mengambil data atau *query*(SELECT), Meubah data (UPDATE) dan Menghapus data (DELETE).

Mengapa Perlu Belajar SQL?

 Pada dasarnya, setiap perusahaan memiliki sistem penyimpanan data, khususnya untuk perusahaan yang memiliki sistem IT. Sistem penyimpanan ini bukan di komputer atau laptop dalam bentuk file atau folder, tetapi di suatu sistem database. Nah, sistem database ini biasanya diakses menggunakan SQL. Sebagai analyst, tugas kita tidak hanya menganalisa data yang sudah tersedia tetapi juga mampu untuk mengambil, memodifikasi dan mengakses sendiri data tersebut dari sumber datanya, yaitu dari database.

Penguasaan SQL akan membantu perusahaan pada area berikut:

* Manajemen memerlukan laporan dengan informasi yang semakin beragam, seperti: tren penjualan bulan ke bulan, pertumbuhan pelanggan, apakah perusahaan mencapai target, dan lain-lain. Dan ini membutuhkan keahlian SQL yang mumpuni.
* Programmer yang membangun sistem aplikasi hampir dipastikan selalu bergantung kepada sistem database SQL agar aplikasinya berjalan dengan baik. Dengan demikian, penguasaan SQL adalah hal mutlak.
* Bisa meningkatkan kinerja perusahaan karena informasi yang kaya dapat dihasilkan melalui SQL.

Jadi, untuk beberapa alasan inilah, maka tidak heran SQL menjadi keterampilan utama yang diminta oleh banyak perusahaan.

Dimana saja SQL Digunakan?

Perusahaan – perusahaan yang sudah menerapkan sistem IT pasti memiliki sistem database dan bisa dipastikan menyimpan datanya dalam suatu database. Contohnya perusahaan berbasis teknologi, seperti e-commerce, menyimpan data baik itu data profile user, data transaksi pembelian dan penjualan, data produk dan data *traffic* kunjungan user ke halaman website di sistem database - atau lebih tepatnya sistem manajemen database atau *database management system* (DBMS).

Semua informasi ataupun analisa yang dibutuhkan oleh manajemen, umumnya bersumber dan diolah dari data DBMS ini. Dan di perusahaan, sistem database biasanya tidak hanya satu, bisa dua, tiga bahkan puluhan. Oleh karena itu, SQL sangat berperan disini, karena dengan menggunakan SQL dapat memenuhi kebutuhan manajemen tersebut. Tanpa penguasaan SQL  akan kesulitan memperoleh data yang dibutuhkan, dan akan kesulitan dalam melakukan analisa dan menghasilkan informasi yang dibutuhkan manajemen dan perusahaan.

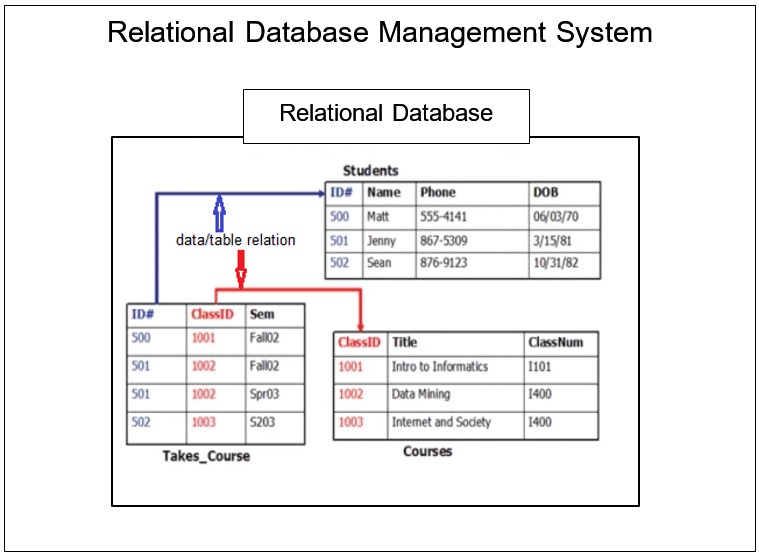
Akan tetapi, perlu diketahui bahwa tidak semua sistem database mendukung SQL. Hanya sistem database berbasis relational database management system (RDBMS) yang mendukung bahasa ini.

**SQL hanya digunakan di sistem database berbasis Relational Database Management System.**

Apa itu RDBMS?

**R**elational **D**atabase **M**anagement **S**ystem yang biasa disingkat dengan RDMBS adalah suatu program yang memungkinkan untuk Membuat, Memperbarui, dan Mengelola suatu basis data relasional (*Relational Database)*. Nah, Umumnya RDMBS ini menggunakan SQL untuk mengakses database.

Basis data relasional sendiri merupakan suatu jenis database dimana data – data umumnya disimpan dalam bentuk yang terstruktur berupa tabel (baris dan kolom) dan setiap tabel/ data yang terdapat dalam database memiliki relasi (*relational*) satu sama lain. Seperti terlihat pada gambar berikut



Basis data relasional sangat popular dan banyak digunakan oleh perusahaan – perusahaan karena jenis database ini mudah dikelola terlebih jika memiliki banyak data atau informasi yang perlu disimpan, scalable dan flexibel.

* Basis data rasional cukup mudah dikelola. Setiap tabel/data dapat diupdate atau dimodifikasi tanpa mengganggu tabel/data yang lain.
* Flexible : jika perlu memperbarui data, hanya perlu melakukannya sekali saja - jadi tidak perlu lagi mengubah banyak file satu per satu. Selain itu, basis data rasional juga cukup mudah untuk di-extend. Misalnya saat data sudah semakin banyak, dapat dengan mudah memperbesar kapasitas dari database yang dimiliki.

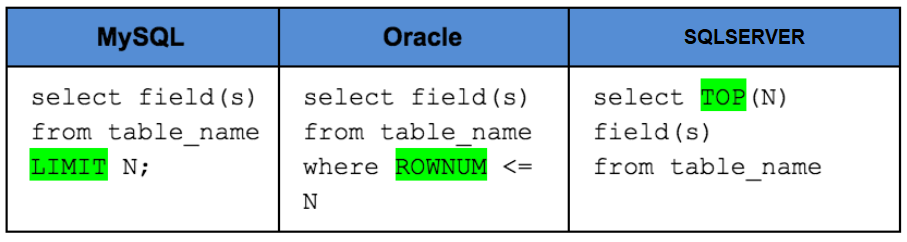
Produk-produk RDBMS di Pasaran

Selain MySQL, masih ada produk lain RDBMS, baik yang berbayar (proprietary) maupun open source. Berikut adalah sebagian produk yang cukup populer di pasaran :

1. **MySQL**Open-source SQL database yang cukup populer. Umumnya digunakan untuk pengembangan aplikasi web.
2. **PostgreSQL**Open-source RDBMS product, dan juga umumnya digunakan untuk pengembangan aplikasi web. Akan tetapi secara kinerja, postgreSQL lebih lambat dibandingkan MySQL.
3. **Oracle DB**Produk RDBMS yang dimiliki oleh Oracle Corporation dan produk ini bersifat proprietary atau tidak open source. Oracle DB umumnya digunakan di industri perbankan.
4. **Microsoft SQL Server**   
   SQL Server adalah produk RDBMS yang dimiliki oleh Microsoft dan sama seperti Oracle DB, SQL Server bersifat proprietary atau tidak open source, SQL Server umumnya digunakan di perusahaan skala besar yang juga menggunakan produk keluaran Microsoft lainnya.
5. **SQLite**Open source RDBMS, umumnya digunakan sebagai database di handphone, MP3 player, and perangkat lainnya.

Selain itu, juga ada MariaDB yang juga gratis atau open source, IBM DB2, Microsoft Access, dan masih banyak lainnya.

Umumnya RDBMS menggunakan SQL untuk mengakses database dan produk RDBMS tidak hanya satu macam saja tetapi ada berbagai macam produk, maka SQL syntax pun bisa jadi sedikit berbeda untuk setiap produk tersebut. Berikut contoh perbandingan MySQL, Oracle, dan SQLSERVER untuk menampilkan beberapa baris data dari suatu tabel :



Struktur Penyimpanan RDBMS

Sebagai penyimpan data, sistem database relasional memiliki struktur hirarki objek penyimpanan sebagai berikut:

* Database
* Tabel (table)
* Kolom (column) atau Field

Dari sini aku belajar informasi menarik yaitu setiap database bisa berisi beberapa tabel, dan setiap tabel bisa terdiri dari beberapa kolom. Di setiap database, tabel dan kolom memiliki nama sendiri sebagai identitas mereka. Tabel dan kolom inilah yang akan diisi data yang kemudian membentuk row (baris data).



Jika diperhatikan, struktur tabel **ms\_produk** terdiri dari empat kolom (column), masing-masing dengan nama berikut:

* no\_urut
* kode\_produk
* nama\_produk
* harga

Dan dalam tabel tersebut terdapat 10 baris data (row) dengan isi data yang bervariasi, contoh isi data untuk kolom "**nama\_produk**" pada baris kelima adalah "**Gift Voucher DQLab 250rb**".

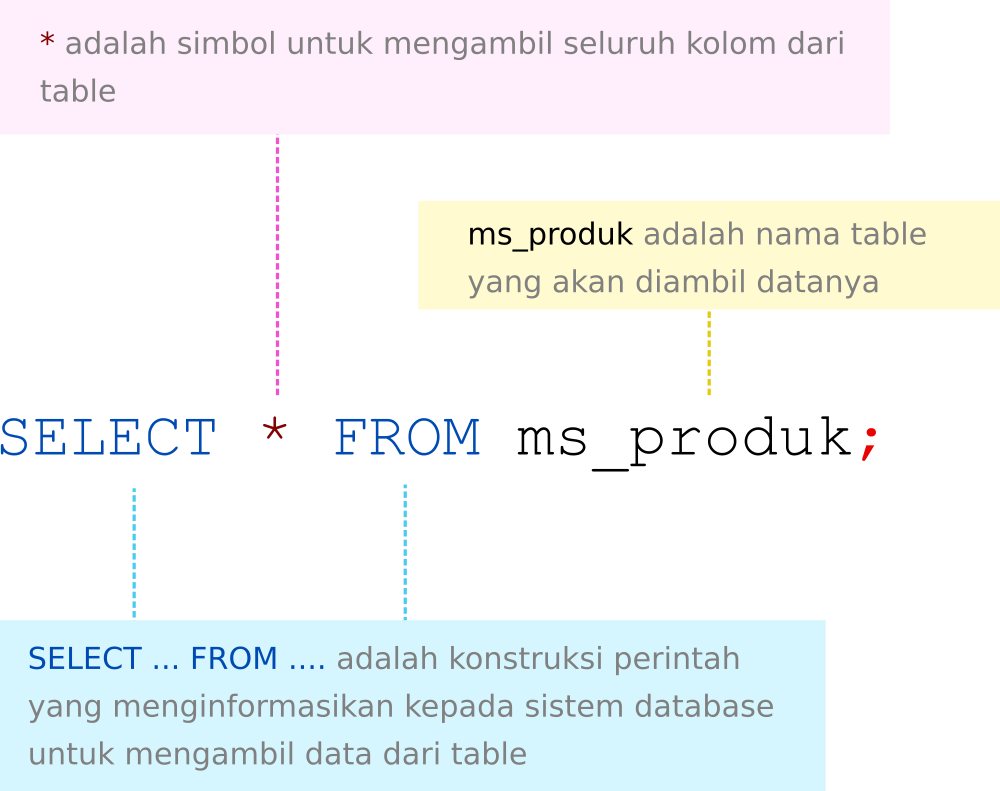
 Kesimpulan

1. Sistem relasional database atau relational database management system (RDBMS) adalah sistem database paling populer di dunia saat ini dan menggunakan bahasa SQL untuk pengolahannya.
2. Beberapa produk RDBMS yang terkenal antara lain adalah Microsoft SQL Server, Oracle, MySQL, PostgreSQL, IBM DB2, dan masih banyak lainnya.
3. Struktur penyimpanan data di RDBMS menggunakan hirarki:
   * Database
   * Tabel (Table)
   * Kolom (Column)
4. Data akan diisi ke dalam table dalam bentuk Baris (Row) data

Mengakses Database

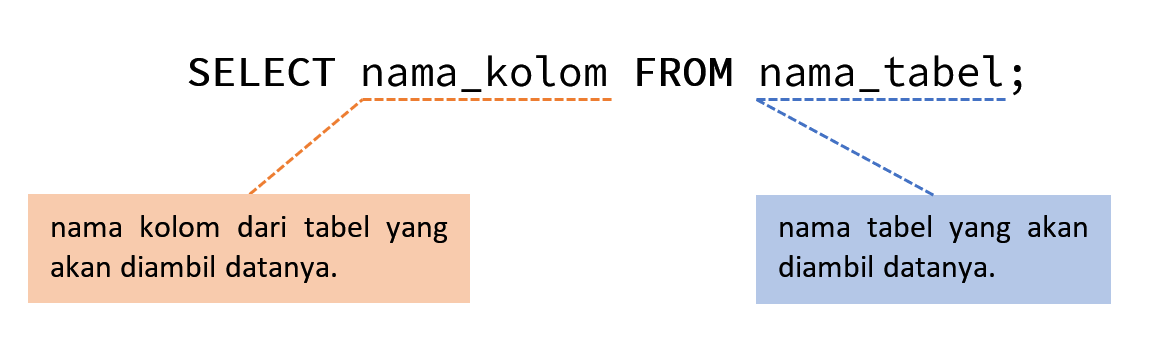
Untuk mengakses data di database, kita dapat menggunakan SELECT statement. Pada SELECT statement kita menyatakan kolom - kolom mana saja yang ingin kita tampilkan dari suatu tabel di database. SELECT statement tidak berdiri sendiri. Setelah menyatakan kolom - kolom yang ingin ditampilkan, kita melanjutkan dengan FROM. Di FROM inilah kita menyatakan dari tabel mana data yang ingin kita tampilkan berada. **SELECT… FROM…** adalah statement paling sederhana di SQL, dan merupakan bagian utama dari query. Kita tidak bisa meng-query data tanpa menggunakan statement ini.

Query dasar dan sederhana perintah SELECT yang berfungsi untuk menampilkan seluruh kolom, sebagai berikut:

* 
* Kata awal, yaitu **SELECT** digunakan untuk menginformasikan kepada sistem bahwa kita ingin mengambil data.
* Tanda **\*** (bintang) artinya seluruh kolom perlu diambil dari tabel yang dirujuk. Tanda ini sering juga disebut sebagai **wildcard**.
* **FROM [NAMA\_TABLE]**, artinya table yang akan diambil datanya.
* Tanda **;** (titik koma) adalah tanda yang menyatakan akhir dari perintah SELECT atau SQL lain.

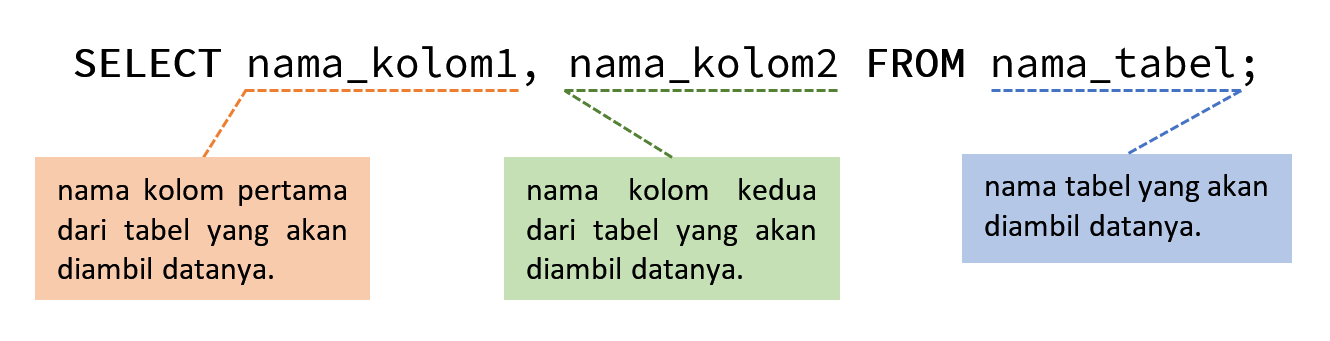
**Mengambil Satu Kolom dari Tabel**

Secara umum penggunaan perintah SELECT untuk mengambil satu kolom dinyatakan oleh sintaks berikut ini:



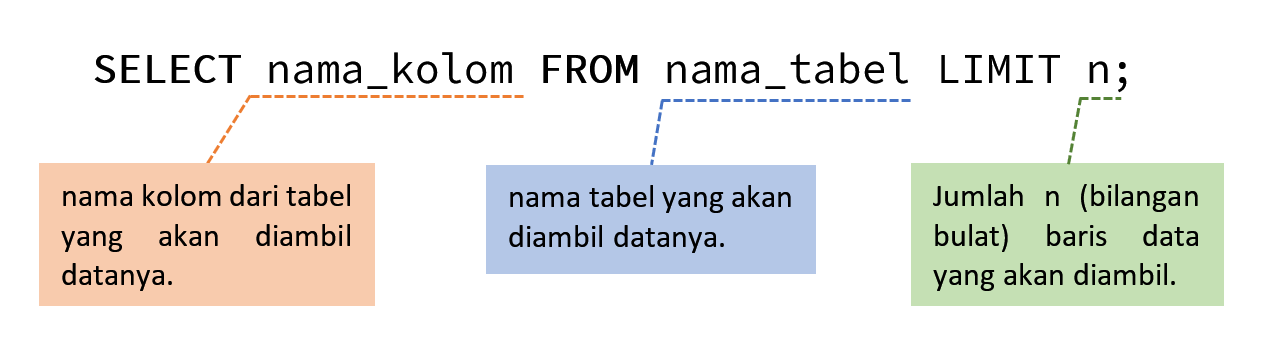
**Mengambil Lebih dari Satu Kolom dari Tabel**

Kalau ingin mengambil kolom lainnya, hanya perlu menuliskan tiap kolom yang ingin ditampilkan dipisahkan dengan tanda koma, seperti contoh berikut untuk dua kolom.



**Membatasi Pengambilan Jumlah Row Data**

Selain pembatasan kolom, bisa juga membatasi jumlah baris data yang diambil. Seperti yang aku pelajari di materi RDMS sebelumnya, bahwa untuk tiap produk RDBMS, caranya agak berbeda-beda. Untuk MySQL dan PostgreSQL, aku dapat menggunakan LIMIT. Secara umum syntaxnya dinyatakan sebagai berikut:

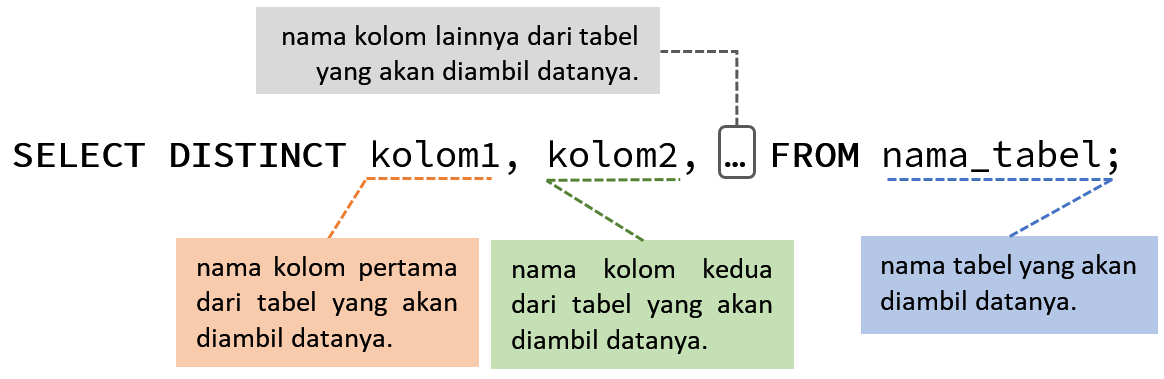


**Penggunaan SELECT DISTINCT statement**

Jika diminta mengambil data dari tabel ms\_pelanggan gunakan perintah yang telah dipelajari. Ternyata, dari data pelanggan, aku menemukan duplikasi data, dalam nama\_customer dan alamat untuk no\_urut 3 & 11, serta 5 & 12 yang sama persis dengan kode\_pelanggan yang berbeda. Tentunya ini akan berdampak pada hasil analisa nantinya.

Untuk menghilangkan data duplikasi, aku bisa menggunakan **SELECT DISTINCT** statement. Dengan **SELECT DISTINCT**, data yang sama atau duplikat akan dieliminasi dan akan ditampilkan data yang unik saja.

Berikut syntax-nya:



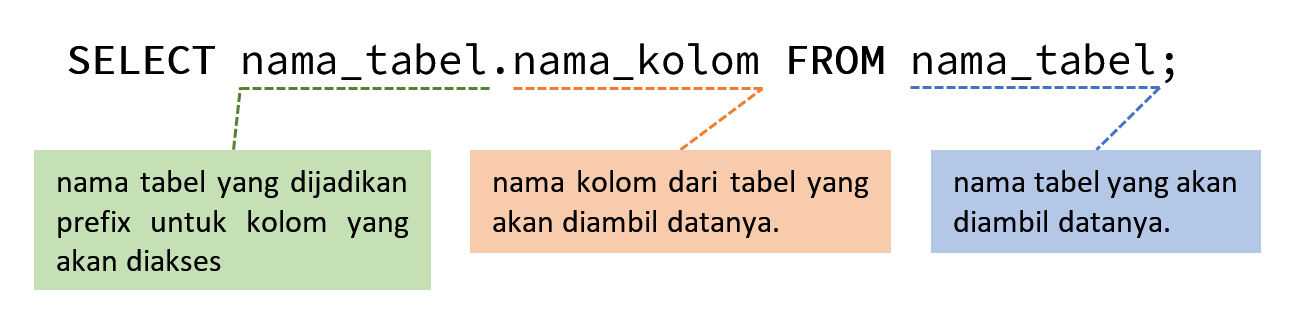
**Prefix dan Alias**

* **Prefix**, dimana kita akan menambahkan nama tabel di depan nama kolom.
* **Alias**, dimana kita memberikan alias atau nama lain untuk tabel maupun kolom.

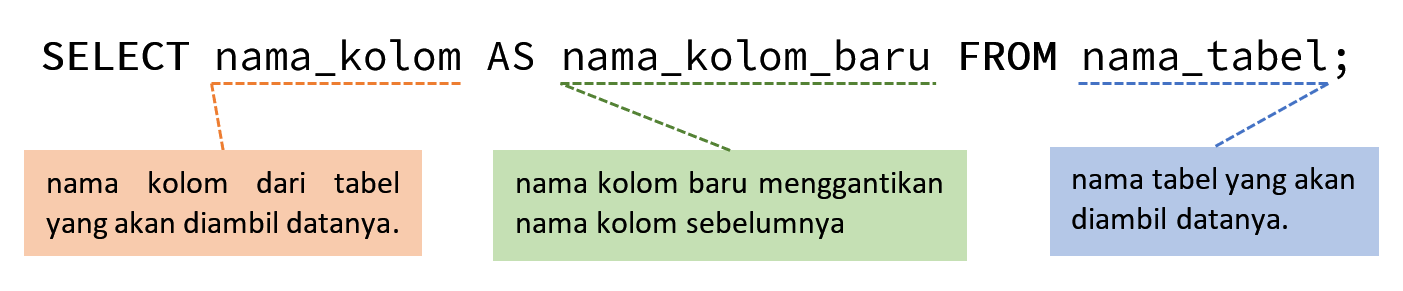
**Menggunakan Prefix pada Nama Kolom**

Pertama, kita mulai dengan menggunakan prefix pada kolom. Pada dasarnya, penulisan nama kolom yang lengkap perlu mencantumkan nama tabel di depan nama kolom tersebut, dengan tanda penyambung berupa tanda titik. Umumnya, jika kita hanya mengambil kolom dari satu tabel, prefix ini jarang digunakan karena sudah jelas dari tabel mana kolom itu berasal. Tetapi ketika kita mengambil data dari dua tabel, misalnya dengan menggabungkan 2 tabel menggunakan JOIN, dan terdapat 2 kolom dengan nama yang sama, maka penggunaan prefix menjadi penting untuk menghindari error karena ambiguitas.

Syntax dasar dari penggunaan prefix pada nama kolom:

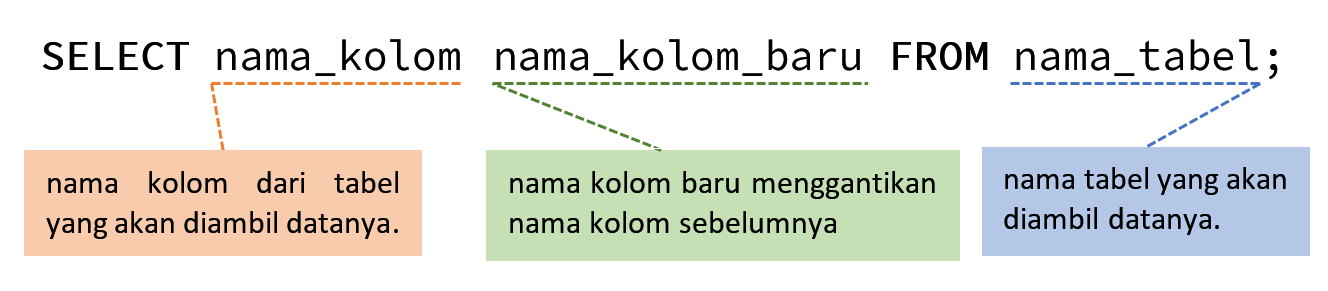


**Menggunakan Alias pada Kolom**

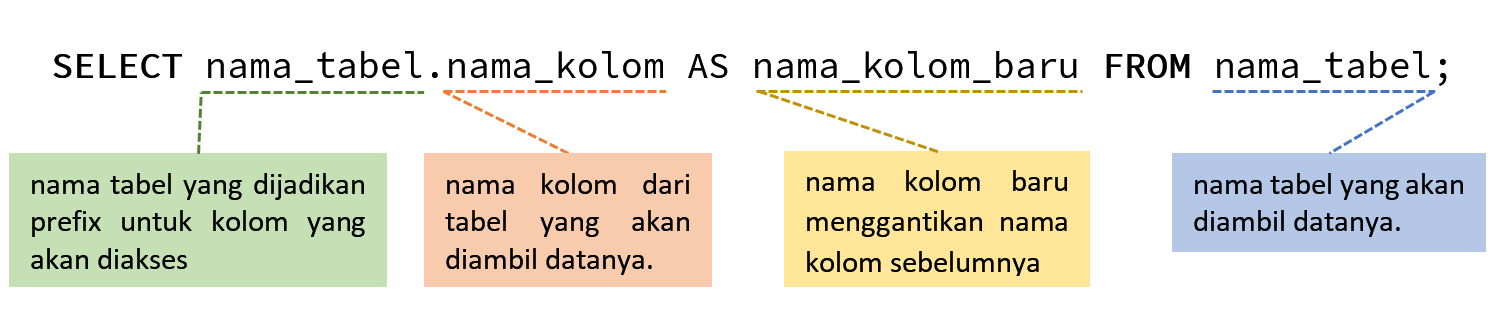
Selain prefix, kita dapat mengubah identitas nama kolom yang diambil dengan SELECT dengan menggunakan *keyword* **AS**. Ini dinamakan alias. Perubahan nama tabel bersifat temporary, artinya hanya berubah ketika mengambil/meng-*query* data, sedangkan nama kolom di tabel dalam database tidak akan berubah. 

**Menghilangkan Keyword 'AS'**

Keyword **AS** yang digunakan sebagai penanda alias pada kolom dapat dihilangkan dengan syntax:



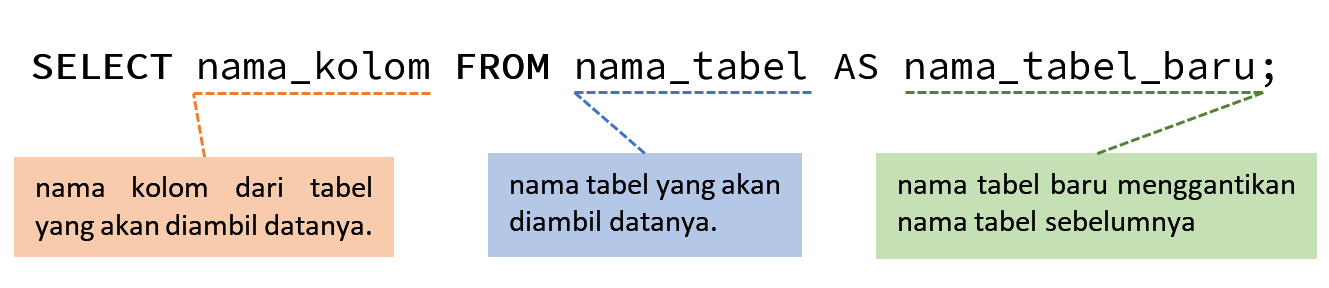
**Menggabungkan Prefix dan Alias**



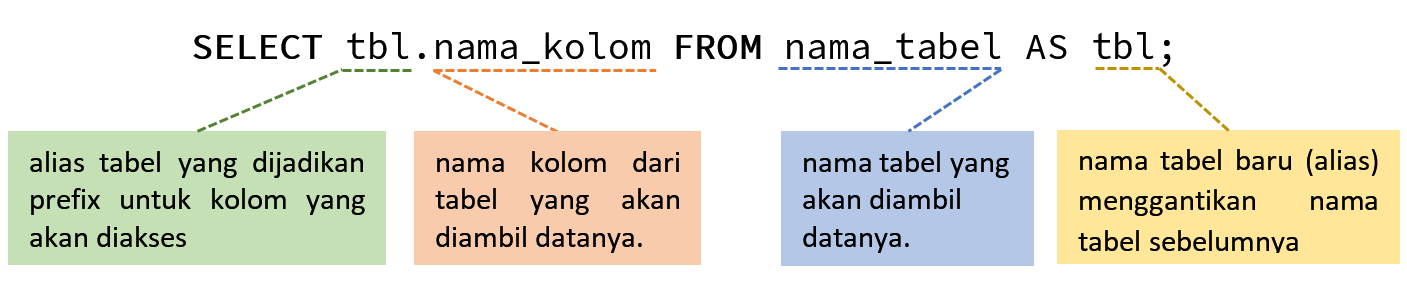
**Menggunakan Alias pada Tabel**

Selain kolom, nama alias juga bisa digunakan untuk tabel dengan menggunakan *keyword***AS** setelah nama tabel. Dan, *keyword* ini juga bisa digunakan atau tidak. Umumnya penggunaan alias pada tabel jika nama tabel tersebut cukup panjang dan muncul atau dirujuk beberapa kali dalam query. Sehingga dengan menggunakan alias pada tabel, dapat menghemat waktu dalam menuliskan query, khususnya untuk query yang cukup rumit, panjang dan melibatkan banyak tabel.

Penulisannya adalah sebagai berikut.



**Prefix dengan Alias Tabel**

Kalau kita menggunakan alias tabel, maka nama prefix yang digunakan untuk kolom adalah alias tabel dan bukan nama original tabel, seperti yang ditunjukkan berikut ini. Gimana hasilnya? 

Penggunaan nama original tabel sebagai prefix akan menimbulkan error saat query dijalankan karena dengan penggunaan alias, nama tabel secara temporary sudah di-gantikan oleh alias.

**Kesimpulan**

Aku kembali mengambil catatanku dan menuliskan apa yang telah aku pelajari:

1. Perintah SELECT dapat ditulis dengan variasi identitas kolom dan tabel berupa prefix dan alias.
   * Penulisan lengkap untuk nama kolom adalah prefix berupa nama tabel disertai tanda titik sebelum nama kolom itu sendiri.
   * Alias adalah nama lain yang diberikan untuk kolom maupun tabel.
   * Alias dapat digunakan dengan keyword AS atau tanpa keyword AS setelah nama kolom dan tabel.
   * Prefix nama tabel bisa menggunakan alias.

Pemahaman mengenai prefix dan alias akan mendorong kemampuan identifikasi tabel maupun kolom yang terlibat untuk perintah SELECT yang lebih kompleks.

Terkadang kita hanya ingin menampilkan data berdasarkan kondisi tertentu, jadi kita tidak butuh semua data dari tabel. Kalau seperti itu, berarti kita hanya ingin mengambil data produk dengan nama produk tertentu. Nah, di SQL caranya gimana ya?

 Untuk case dimana kita ingin mengambil data berdasarkan kondisi tertentu saja, kita bisa menggunakan filter. SQL memiliki fungsi filter dengan menggunakan klausul WHERE. Jika kondisi WHERE terpenuhi, maka hasil query hanya akan menampilkan data yang sudah terfilter.

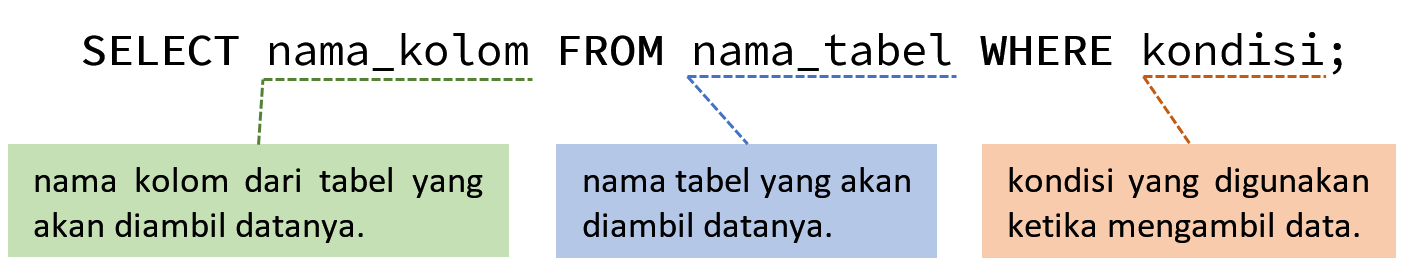
Biar lebih mudah dipahami, selanjutnya kita akan mempraktikkan bagaimana menggunakan klausul WHERE.

Klausul WHERE untuk:

* Filter data dengan kondisi teks tertentu.
* Filter data dengan nilai angka tertentu.
* Filter data dengan dua kondisi menggunakan operator AND dan OR.

**Menggunakan WHERE**

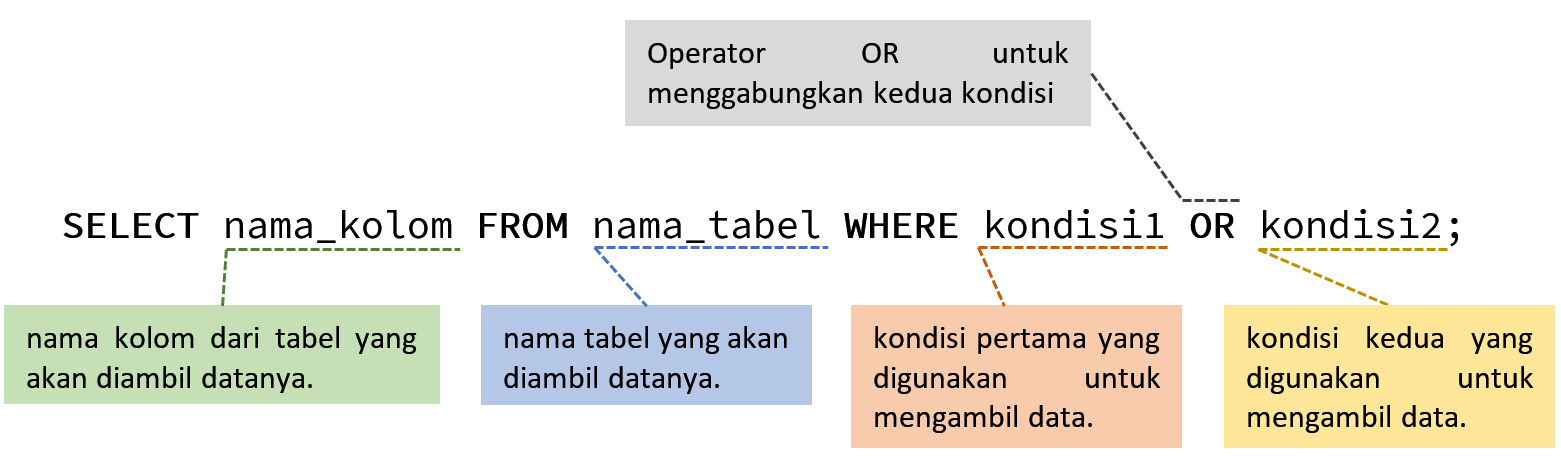
Klausul WHERE dari SELECT digunakan untuk memfilter data berdasarkan kondisi tertentu. Untuk syntax lengkapnya adalah sebagai berikut.



**Menggunakan Operand OR**

Pada subbab sebelumnya, aku telah menggunakan filter teks sederhana untuk mengeluarkan data masing-masing dengan nama\_produk 'Gantungan Kunci DQLab' dan 'Tas Travel Organizer DQLab'.

Pertanyaannya, bagaimana jika ingin mengeluarkan keduanya sekaligus? Aku bisa menggunakan Operand OR.



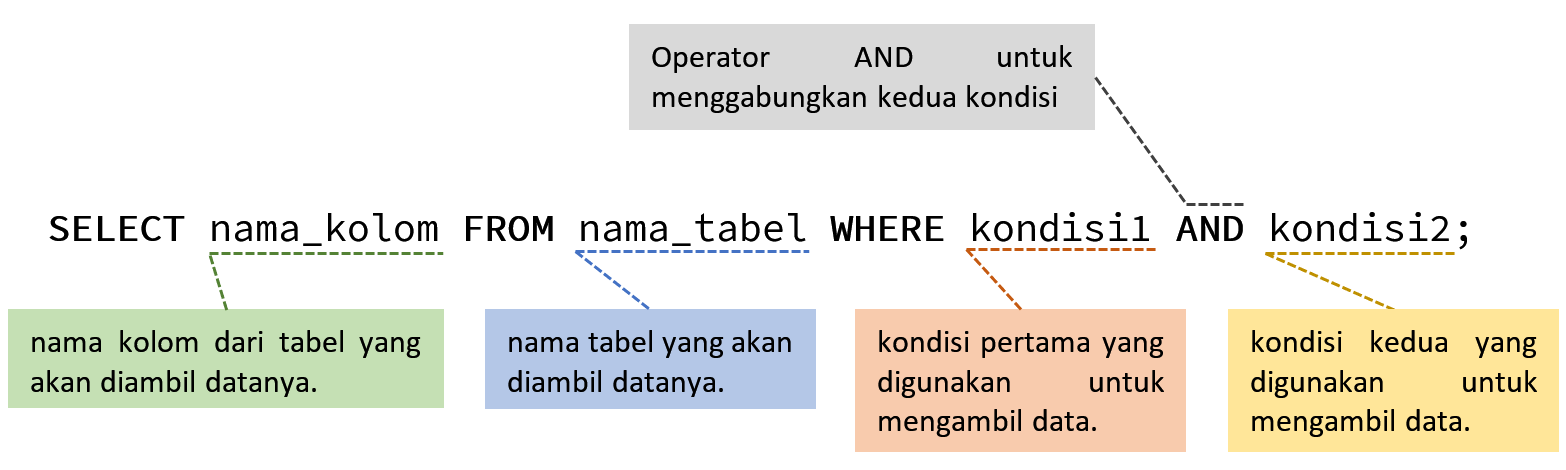
Dengan menggunakan logika **OR**, aku dapat menggabungkan dua atau lebih kondisi untuk memfilter data.

**Filter untuk Angka**

Sebelumnya, aku telah melakukan filtering untuk teks, namun WHERE tidak terbatas untuk tipe data teks saja tapi malah umumnya untuk angka.

**Menggunakan Operand AND**

Jika sebelumnya aku mempelajari Operand OR, aku juga bisa menggunakan operand AND agar dua atau lebih kondisi terpenuhi semuanya. Jika salah satu kondisi tidak terpenuhi, data tidak akan diambil. Secara umum syntaxnya diilustrasikan berikut ini



**Kesimpulan**

Wah, semakin seru saja nih belajar SQL! Sebelum aku lupa, aku mencatat apa yang aku pelajari tadi. Siapa yang punya kebiasaan seperti aku? Dengan mencatat apa yang aku pelajari, aku merasa ilmu yang aku pelajari tadi lebih mudah untuk aku pahami.

1. Filter di SQL diimplementasikan dengan menggunakan WHERE diikuti dengan satu atau lebih kondisi logis.
2. Kondisi logis ini memiliki format nama kolom diikuti dengan nilai yang akan difilter. Untuk teks sederhana bisa menggunakan tanda sama dengan, sedangkan untuk nilai angka bisa menggunakan operator perbandingan matematika.
3. Aku juga bisa menggunakan operand OR dan AND untuk menggabungkan beberapa kondisi menjadi satu kondisi baru yang harus terpenuhi untuk pengambilan data.