**Lebih banyak sumber daya spreadsheet**

Dalam semangat pembelajaran seumur hidup, ada baiknya jika Anda memiliki sumber daya yang dapat digunakan ketika Anda ingin mengetahui lebih banyak tentang penggunaan spreadsheet. Dua platform spreadsheet yang paling terkenal dan digunakan adalah Google Sheets dan Microsoft Excel. Keduanya menyediakan sumber pelatihan online gratis yang dapat Anda akses kapan pun Anda membutuhkannya. Tandai tautan ini jika Anda ingin mengaksesnya nanti.

[**Pelatihan dan Bantuan Google Spreadsheet**](https://support.google.com/a/users/answer/9282959?visit_id=637361702049227170-1815413770&rd=1)

Pelajari lebih banyak cara untuk memindahkan, menyimpan, dan menganalisis data Anda dengan halaman Pelatihan dan Bantuan Google Spreadsheet, yang terdapat di Pusat Pembelajaran Google Workspace. Hub ini menawarkan daftar tips yang diperluas, dari pemula hingga mahir, bersama dengan lembar contekan, templat, panduan, dan tutorial.

[**Lembar Cheat Google Spreadsheet**](https://support.google.com/a/users/answer/9300022)

Ingin mempelajari lebih lanjut tentang Google Spreadsheet? Artikel bantuan online ini menampilkan daftar singkat fungsi terpenting yang akan Anda gunakan, termasuk baris, kolom, sel, dan fungsi.

[**Pelatihan Microsoft Excel untuk Windows**](https://support.microsoft.com/en-us/office/excel-for-windows-training-9bc05390-e94c-46af-a5b3-d7c22f6990bb)

Kenali spreadsheet Excel lebih baik dengan mengunjungi pusat pelatihan online gratis ini. Menawarkan segalanya mulai dari panduan memulai cepat dan pengenalan hingga tutorial dan templat, Anda akan menemukan semua yang perlu Anda ketahui, semuanya di satu tempat.

# Panduan SQL: Memulai

Sebagaimana manusia menggunakan bahasa berbeda untuk berkomunikasi dengan orang lain, komputer juga demikian. **Structured Query Language** (atau **SQL** , sering diucapkan “sekuel”) memungkinkan analis data untuk berkomunikasi dengan database mereka. SQL adalah salah satu alat analis data yang paling berguna, terutama saat bekerja dengan kumpulan data besar dalam tabel. Ini dapat membantu Anda menyelidiki database yang sangat besar, melacak teks (disebut sebagai string) dan angka, dan memfilter jenis data yang Anda perlukan—jauh lebih cepat daripada yang bisa dilakukan spreadsheet.

Jika Anda belum pernah menggunakan SQL sebelumnya, bacaan ini akan membantu Anda mempelajari dasar-dasarnya sehingga Anda dapat memahami betapa bergunanya SQL dan khususnya betapa bergunanya kueri SQL. Anda akan menulis kueri SQL dalam waktu singkat.

## Apa itu pertanyaan?

Query adalah permintaan data atau informasi dari database **.** Saat Anda membuat kueri database, Anda menggunakan SQL untuk mengomunikasikan pertanyaan atau permintaan Anda. Anda dan database selalu dapat bertukar informasi selama Anda berbicara dalam bahasa yang sama.

Setiap bahasa pemrograman, termasuk SQL, mengikuti serangkaian pedoman unik yang dikenal sebagai **sintaksis** . **Sintaks** adalah struktur bahasa yang telah ditentukan sebelumnya yang mencakup semua kata, simbol, dan tanda baca yang diperlukan, serta penempatannya yang tepat. Segera setelah Anda memasukkan kriteria pencarian menggunakan sintaks yang benar, kueri mulai bekerja untuk mengambil data yang Anda minta dari database target.

Sintaks setiap query SQL adalah sama:

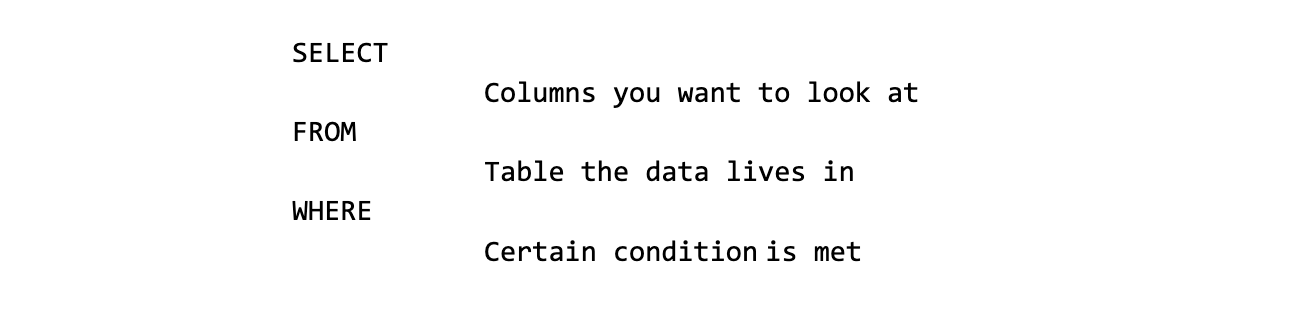
* Gunakan **SELECT** untuk memilih kolom yang ingin Anda kembalikan.
* Gunakan **FROM** untuk memilih tabel tempat kolom yang Anda inginkan berada.
* Gunakan **WHERE** untuk memfilter informasi tertentu.

Kueri SQL seperti mengisi templat. Anda akan menemukan bahwa jika Anda menulis kueri SQL dari awal, akan sangat membantu jika memulai kueri dengan menulis kata kunci SELECT, FROM, dan WHERE dalam format berikut:

A white background with black dots

Description automatically generated

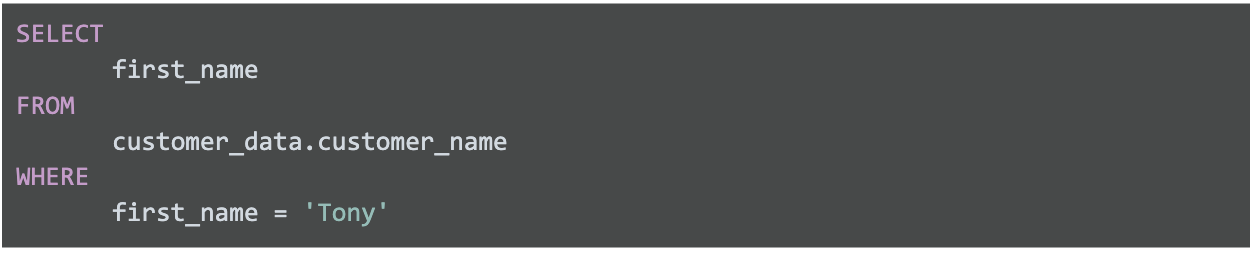
Selanjutnya, masukkan nama tabel setelah FROM; kolom tabel yang Anda inginkan setelah SELECT; dan, terakhir, kondisi yang ingin Anda tempatkan pada kueri Anda setelah WHERE. Pastikan untuk menambahkan baris dan indentasi baru saat menambahkan ini, seperti yang ditunjukkan di bawah ini:



Mengikuti metode ini setiap kali akan mempermudah penulisan kueri SQL. Ini juga dapat membantu Anda mengurangi kesalahan sintaksis.

### **Contoh kueri**

Berikut ini tampilan kueri sederhana di BigQuery, gudang data di Google Cloud Platform.



Kueri di atas menggunakan tiga perintah untuk mencari pelanggan dengan nama depan Tony:

1. **PILIH** kolom bernama **first\_name**
2. **DARI** tabel bernama **customer\_name** (dalam kumpulan data bernama **customer\_data** ) (Nama kumpulan data selalu diikuti dengan titik, lalu nama tabel.)
3. Tapi hanya mengembalikan data **WHERE** first\_name **Tony**

Hasil dari kueri mungkin serupa dengan berikut ini:

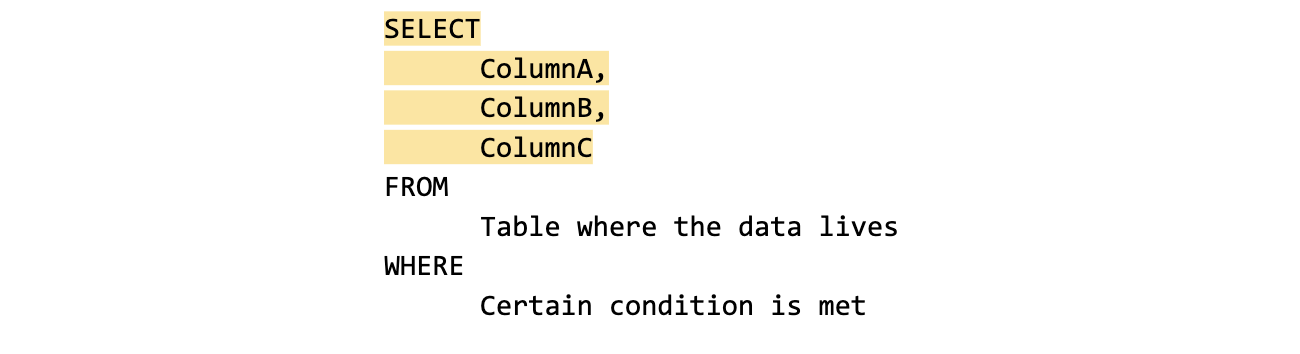
| **nama depan** |
| --- |
| Toni |
| Toni |
| Toni |

Seperti yang dapat Anda simpulkan, kueri ini memiliki sintaksis yang benar, tetapi tidak terlalu berguna setelah data dikembalikan.

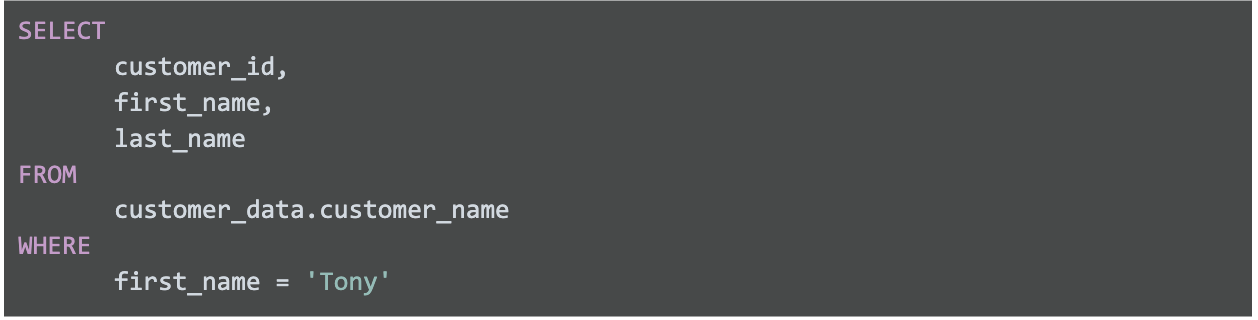
## Beberapa kolom dalam kueri

Dalam kehidupan nyata, Anda perlu bekerja dengan lebih banyak data selain pelanggan bernama Tony. Beberapa kolom yang dipilih dengan perintah SELECT yang sama dapat diindentasi dan dikelompokkan bersama.

Jika Anda meminta beberapa bidang data dari sebuah tabel, Anda perlu menyertakan kolom ini dalam perintah SELECT Anda. Setiap kolom dipisahkan dengan koma seperti yang ditunjukkan di bawah ini:



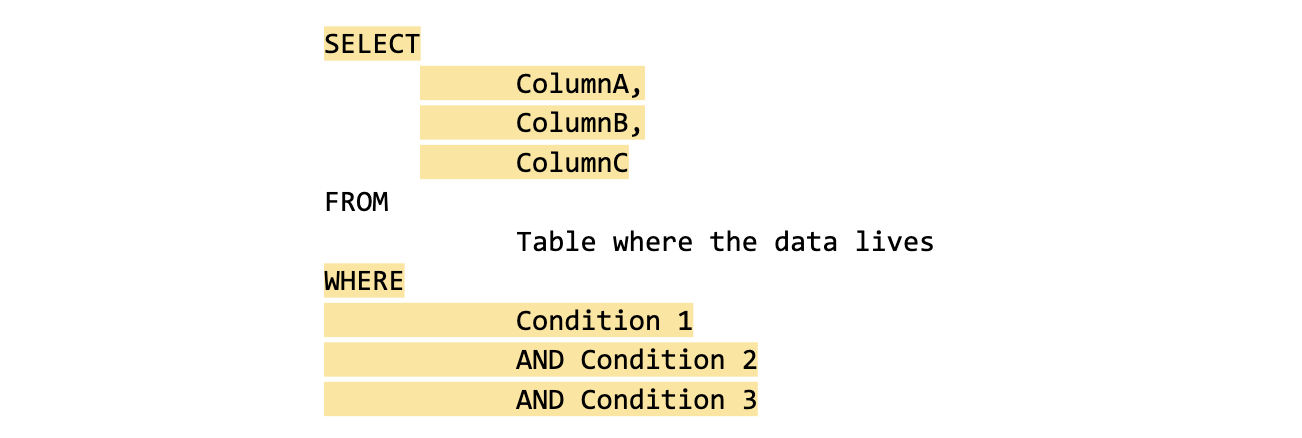
Berikut ini contoh tampilannya di BigQuery:



Kueri di atas menggunakan tiga perintah untuk mencari pelanggan dengan nama depan Tony.

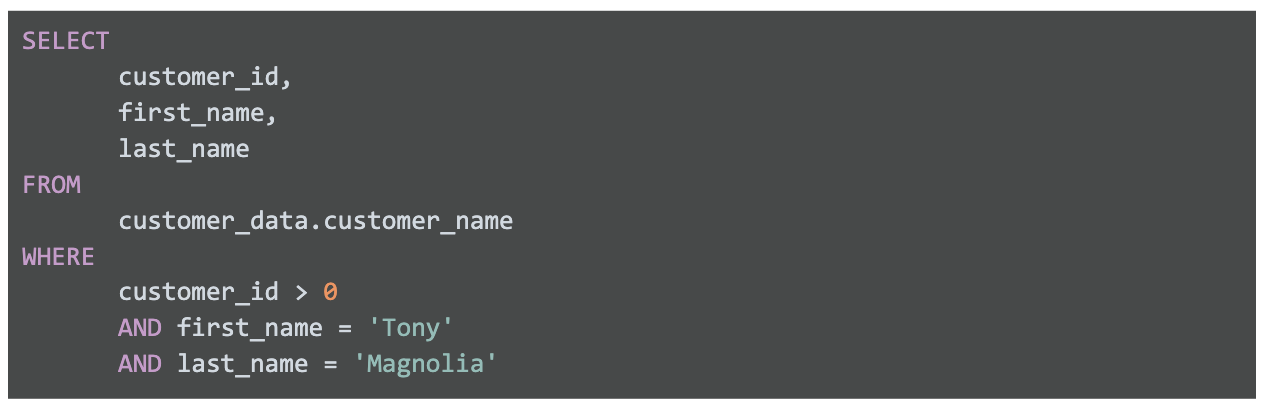
1. **PILIH** kolom bernama **customer\_id** , **first\_name** , dan **last\_name**
2. **DARI** tabel bernama **customer\_name** (dalam kumpulan data bernama **customer\_data** ) (Nama kumpulan data selalu diikuti dengan titik, lalu nama tabel.)
3. Tapi hanya mengembalikan data **WHERE** first\_name **Tony**

Satu-satunya perbedaan antara kueri ini dan kueri sebelumnya adalah lebih banyak kolom data yang dipilih. Kueri sebelumnya hanya memilih nama\_depan, sedangkan kueri ini memilih id\_pelanggan dan nama\_belakang selain nama\_depan. Secara umum, penggunaan sumber daya lebih efisien jika hanya memilih kolom yang Anda perlukan. Misalnya, masuk akal untuk memilih lebih banyak kolom jika Anda benar-benar akan menggunakan kolom tambahan di klausa WHERE Anda. Jika Anda memiliki beberapa ketentuan dalam klausa WHERE, ketentuan tersebut dapat ditulis seperti ini:



Perhatikan bahwa tidak seperti perintah SELECT yang menggunakan koma untuk memisahkan bidang/variabel/parameter, perintah WHERE menggunakan pernyataan AND untuk menghubungkan kondisi. Saat Anda menjadi penulis kueri yang lebih mahir, Anda akan menggunakan konektor/operator lain seperti OR dan NOT.

Berikut adalah contoh BigQuery dengan beberapa kolom yang digunakan dalam klausa WHERE:



Kueri di atas menggunakan tiga perintah untuk mencari pelanggan dengan ID pelanggan valid (lebih besar dari 0) yang nama depannya adalah Tony dan nama belakangnya adalah Magnolia.

1. **PILIH** kolom bernama **customer\_id** , **first\_name** , dan **last\_name**
2. **DARI** tabel bernama **customer\_name** (dalam kumpulan data bernama **customer\_data** ) (Nama kumpulan data selalu diikuti dengan titik, lalu nama tabel.)
3. Namun hanya mengembalikan data **WHERE** customer\_id lebih besar dari **0** , first\_name adalah **Tony** , dan last\_name adalah **Magnolia** .

Perhatikan bahwa salah satu kondisi adalah kondisi logis yang memeriksa apakah customer\_id lebih besar dari nol.

Jika hanya satu pelanggan bernama Tony Magnolia, hasil kuerinya dapat berupa:

| **ID Pelanggan** | **nama depan** | **nama keluarga** |
| --- | --- | --- |
| 1967 | Toni | Magnolia |

Jika lebih dari satu pelanggan memiliki nama yang sama, hasil kueri dapat berupa:

| **ID Pelanggan** | **nama depan** | **nama keluarga** |
| --- | --- | --- |
| 1967 | Toni | Magnolia |
| 7689 | Toni | Magnolia |

## Pengambilan kunci

Hal yang paling penting untuk diingat adalah bagaimana menggunakan SELECT, FROM, dan WHERE dalam sebuah query. Kueri dengan banyak bidang akan menjadi lebih sederhana setelah Anda berlatih menulis kueri SQL Anda sendiri nanti di program.

# Kemungkinan SQL yang tak ada habisnya

Anda telah mempelajari bahwa kueri SQL menggunakan **SELECT** , **FROM** , dan **WHERE** untuk menentukan data yang akan dikembalikan dari kueri. Bacaan ini memberikan informasi lebih rinci tentang memformat kueri, menggunakan kondisi WHERE, memilih semua kolom dalam tabel, menambahkan komentar, dan menggunakan alias. Semua ini memudahkan Anda memahami (dan menulis) kueri untuk menerapkan SQL. Bagian terakhir dari bacaan ini memberikan contoh tentang apa yang akan dilakukan analis data untuk mengambil data karyawan untuk sebuah proyek.

## Kapitalisasi, indentasi, dan titik koma

Anda dapat menulis kueri SQL dalam huruf kecil semua dan tidak perlu khawatir tentang spasi tambahan di antara kata. Namun, penggunaan huruf besar dan indentasi dapat membantu Anda membaca informasi dengan lebih mudah. Jaga pertanyaan Anda tetap rapi, dan pertanyaan tersebut akan lebih mudah ditinjau atau dipecahkan jika Anda perlu memeriksanya nanti.



Perhatikan bahwa pernyataan SQL yang ditunjukkan di atas memiliki titik koma di akhir. Titik koma adalah terminator pernyataan dan merupakan bagian dari standar SQL-92 American National Standards Institute (ANSI), yang merupakan sintaks umum yang direkomendasikan untuk diadopsi oleh semua database SQL. Namun, tidak semua database SQL mengadopsi atau menerapkan titik koma, jadi mungkin Anda menemukan beberapa pernyataan SQL yang tidak diakhiri dengan titik koma. Jika suatu pernyataan berfungsi tanpa titik koma, tidak masalah.

## kondisi DIMANA

Dalam kueri yang ditampilkan di atas, klausa **SELECT**  mengidentifikasi kolom tempat Anda ingin mengambil data berdasarkan nama, **field1** , dan klausa **FROM** mengidentifikasi tabel tempat kolom tersebut berada berdasarkan nama, **table** . Terakhir, klausa **WHERE** mempersempit kueri Anda sehingga database hanya mengembalikan data dengan nilai yang sama persis atau data yang cocok dengan kondisi tertentu yang ingin Anda penuhi.

Misalnya, jika Anda mencari pelanggan tertentu dengan nama belakang Chavez, klausa WHERE adalah:

**DIMANA bidang1 = 'Chavez'**

Namun, jika Anda mencari semua pelanggan dengan nama belakang yang dimulai dengan huruf “Ch,” klausa WHERE-nya adalah:

**DIMANA bidang1 SEPERTI 'Ch%'**

Anda dapat menyimpulkan bahwa klausa LIKE sangat berguna karena memungkinkan Anda memberi tahu database untuk mencari pola tertentu! Tanda persen (%) digunakan sebagai wildcard untuk mencocokkan satu atau lebih karakter. Dalam contoh di atas, baik **Chavez** maupun **Chen** akan dikembalikan. Perhatikan bahwa di beberapa database, tanda bintang (\*) digunakan sebagai wildcard, bukan tanda persen (%).

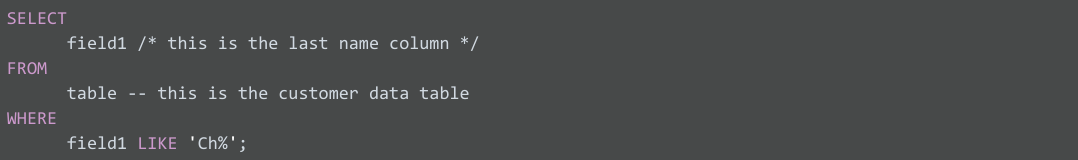
## PILIH semua kolom

Bisakah Anda menggunakan  **SELECT \*** ?

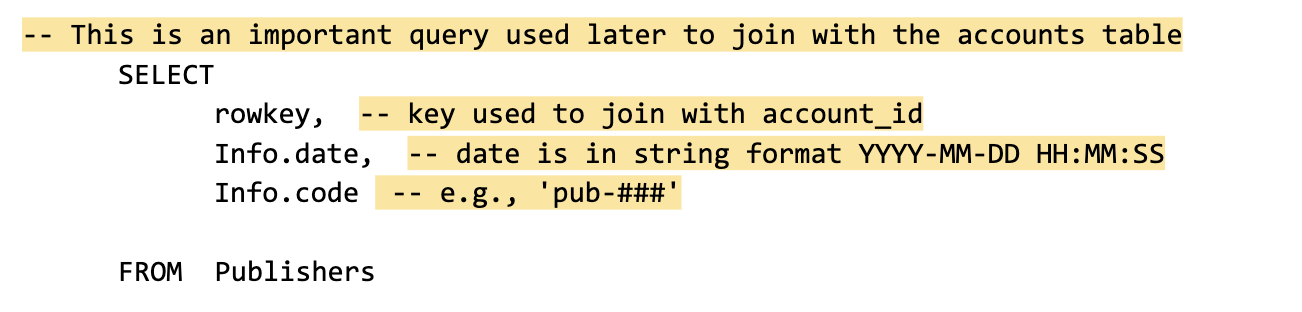
Dalam contoh, jika Anda mengganti **SELECT field1** dengan **SELECT \***  , Anda akan memilih semua kolom dalam tabel, bukan kolom field1 saja. Dari sudut pandang sintaksis, ini adalah pernyataan SQL yang benar, tetapi Anda harus menggunakan tanda bintang (\*) dengan hemat dan hati-hati. Bergantung pada berapa banyak kolom yang dimiliki tabel, Anda mungkin memilih data dalam jumlah besar. Memilih terlalu banyak data dapat menyebabkan kueri berjalan lambat.

## Komentar

Beberapa tabel tidak dirancang dengan konvensi penamaan yang cukup deskriptif. Dalam contoh, **field1** adalah kolom untuk nama belakang pelanggan, namun Anda tidak akan mengetahui namanya. Nama yang lebih baik adalah seperti **last\_name** . Dalam kasus ini, Anda dapat menempatkan komentar di samping SQL untuk membantu Anda mengingat apa yang diwakili oleh nama tersebut. Komentar adalah teks yang ditempatkan di antara karakter tertentu, **/\*** dan **\*/** , atau setelah dua tanda hubung ( **--** ) seperti gambar di bawah ini.



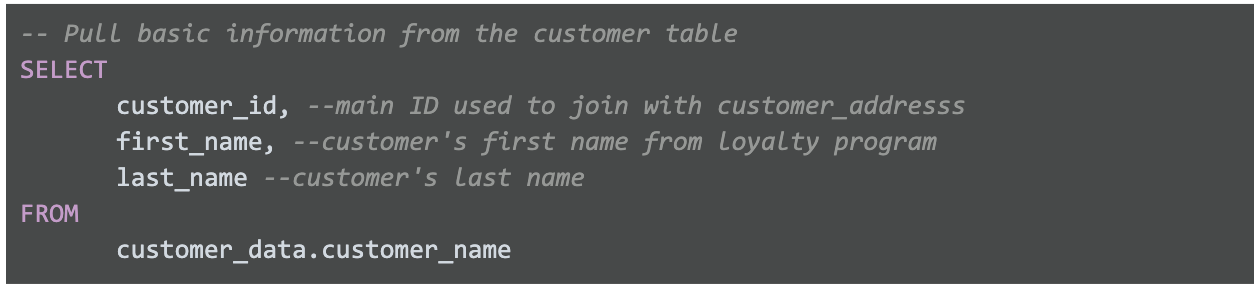
Komentar juga dapat ditambahkan di luar pernyataan maupun di dalam pernyataan. Anda dapat menggunakan fleksibilitas ini untuk memberikan gambaran keseluruhan tentang apa yang akan Anda lakukan, catatan langkah demi langkah tentang cara Anda mencapainya, dan alasan Anda menetapkan parameter/kondisi yang berbeda.



Semakin nyaman Anda dengan SQL, semakin mudah membaca dan memahami kueri secara sekilas. Namun, tidak ada salahnya untuk memiliki komentar dalam kueri untuk mengingatkan diri Anda tentang apa yang Anda coba lakukan. Hal ini juga memudahkan orang lain untuk memahami kueri Anda jika kueri Anda dibagikan. Saat pertanyaan Anda menjadi semakin kompleks, praktik ini akan menghemat banyak waktu dan energi untuk memahami pertanyaan kompleks yang Anda tulis beberapa bulan atau tahun yang lalu.

### **Contoh kueri dengan komentar**

Berikut ini contoh cara menulis komentar di BigQuery:



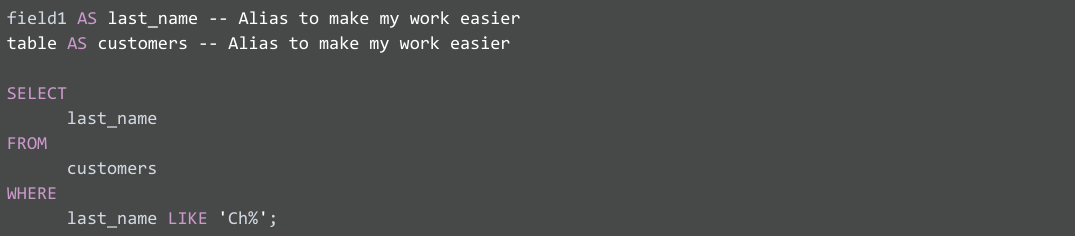
Dalam contoh di atas, sebuah komentar telah ditambahkan sebelum pernyataan SQL untuk menjelaskan apa yang dilakukan kueri. Selain itu, komentar telah ditambahkan di samping masing-masing nama kolom untuk menjelaskan kolom dan penggunaannya. Dua tanda hubung (--) umumnya didukung. Jadi yang terbaik adalah menggunakan -- dan konsisten dengannya. Anda dapat menggunakan # sebagai pengganti -- pada kueri di atas, tetapi # tidak dikenali di semua versi SQL; misalnya, MySQL tidak mengenali #. Anda juga dapat menempatkan komentar di antara /\* dan \*/ jika database yang Anda gunakan mendukungnya.

Saat Anda mengembangkan keterampilan Anda secara profesional, bergantung pada database SQL yang Anda gunakan, Anda dapat memilih simbol pembatas komentar yang sesuai yang Anda sukai dan tetap menggunakannya sebagai gaya yang konsisten. Saat pertanyaan Anda menjadi semakin kompleks, praktik menambahkan komentar yang bermanfaat akan menghemat banyak waktu dan energi untuk memahami pertanyaan yang mungkin Anda tulis beberapa bulan atau tahun sebelumnya.

## Alias

Anda juga dapat mempermudah diri Anda sendiri dengan memberikan nama atau **alias** baru pada nama kolom atau tabel agar lebih mudah digunakan (dan menghindari perlunya komentar). Hal ini dilakukan dengan klausa SQL AS. Pada contoh di bawah, alias **last\_name** telah ditetapkan ke **field1** dan alias **pelanggan** ditetapkan ke **tabel.** Alias ​​ini hanya berlaku selama durasi kueri saja. Alias ​​tidak mengubah nama sebenarnya dari kolom atau tabel dalam database.

### **Contoh query dengan alias**



## Menempatkan SQL untuk bekerja sebagai analis data

Bayangkan Anda adalah seorang analis data untuk bisnis kecil dan manajer Anda meminta beberapa data karyawan dari Anda. Anda memutuskan untuk menulis kueri dengan SQL untuk mendapatkan apa yang Anda perlukan dari database.

Anda ingin menarik semua kolom: **empID** , **firstName** , **lastName** , **jobCode** , dan **Salary** . Karena Anda tahu databasenya tidak terlalu besar, alih-alih memasukkan setiap nama kolom di klausa **SELECT** , Anda menggunakan **SELECT \*** . Ini akan memilih semua kolom dari tabel Karyawan di klausa **FROM** .



Sekarang, Anda bisa lebih spesifik tentang data yang Anda inginkan dari tabel Karyawan. Jika Anda ingin semua data tentang karyawan yang bekerja dalam kode pekerjaan **SFI** , Anda dapat menggunakan klausa **WHERE** untuk memfilter data berdasarkan persyaratan tambahan ini.

Di sini, Anda menggunakan:



Sebagian dari data hasil yang dikembalikan dari kueri SQL mungkin terlihat seperti ini:

| **kosong** | **nama depan** | **nama keluarga** | **Kode pekerjaan** | **gaji** |
| --- | --- | --- | --- | --- |
| 0002 | Homer | Simpson | SFI | 15000 |
| 0003 | Pinggiran | Simpson | SFI | 30000 |
| 0034 | Bart | Simpson | SFI | 25000 |
| 0067 | Lisa | Simpson | SFI | 38000 |
| 0088 | tidak | Flanders | SFI | 42000 |
| 0076 | Barney | Gumble | SFI | 32000 |

Misalkan Anda melihat kisaran gaji yang besar untuk kode pekerjaan **SFI** . Anda mungkin ingin menandai semua karyawan di semua departemen dengan gaji lebih rendah untuk manajer Anda. Karena pekerja magang juga disertakan dalam tabel dan mereka memiliki gaji kurang dari $30.000, Anda ingin memastikan hasil Anda hanya memberikan karyawan penuh waktu dengan gaji $30.000 atau kurang. Dengan kata lain, Anda ingin mengecualikan pekerja magang dengan kode pekerjaan **INT** yang juga berpenghasilan kurang dari $30.000. Klausa **AND** memungkinkan Anda menguji kedua kondisi.

Anda membuat kueri SQL seperti di bawah ini, dengan <> berarti "tidak sama":

A close up of a grey surface

Description automatically generated

Data yang dihasilkan dari kueri SQL mungkin terlihat seperti berikut (pekerja magang dengan kode pekerjaan **INT** tidak dikembalikan):

| **kosong** | **nama depan** | **nama keluarga** | **Kode pekerjaan** | **gaji** |
| --- | --- | --- | --- | --- |
| 0002 | Homer | Simpson | SFI | 15000 |
| 0003 | Pinggiran | Simpson | SFI | 30000 |
| 0034 | Bart | Simpson | SFI | 25000 |
| 0108 | Edna | Krabappel | TUL | 18000 |
| 0099 | Moe | Szyslak | ANA | 28000 |

Dengan akses cepat ke jenis data ini menggunakan SQL, Anda dapat memberikan banyak wawasan berbeda kepada manajer Anda tentang data karyawan, termasuk apakah gaji karyawan di seluruh bisnis adil. Untungnya, kueri menunjukkan hanya dua karyawan tambahan yang mungkin memerlukan penyesuaian gaji dan Anda membagikan hasilnya kepada manajer Anda.

Menarik data, menganalisisnya, dan menerapkan solusi pada akhirnya dapat membantu meningkatkan kepuasan dan loyalitas karyawan. Itu menjadikan SQL alat yang cukup ampuh.

## Sumber daya untuk mempelajari lebih lanjut

Non-pelanggan dapat mengakses sumber daya ini secara gratis, namun jika sebuah situs membatasi jumlah artikel gratis per bulan dan Anda telah mencapai batas tersebut, tandai sumber daya tersebut dan kembali lagi nanti.

* [Tutorial SQL W3School](https://www.w3schools.com/sql/default.asp): Jika Anda ingin menjelajahi tutorial SQL secara mendetail, ini adalah tempat yang tepat untuk memulai. Tutorial ini mencakup contoh interaktif yang dapat Anda edit, uji, dan buat ulang. Gunakan ini sebagai referensi atau selesaikan seluruh tutorial untuk berlatih menggunakan SQL. Klik tombol hijau **Mulai belajar SQL sekarang** atau tombol **Berikutnya** untuk memulai tutorial.
* [Lembar Cheat SQL](https://www.sqltutorial.org/sql-cheat-sheet/): Untuk pelajar tingkat lanjut, baca sumber daya 3 halaman yang berguna ini untuk mendapatkan gambaran umum tentang fungsi dan rumus SQL tambahan. Pada saat Anda selesai membaca lembar contekan, Anda akan mengetahui lebih banyak tentang berbagai teknik SQL dan akan siap menggunakannya untuk analisis bisnis dan tugas lainnya.

# Merencanakan visualisasi data

Sebelumnya, Anda telah mempelajari bahwa **visualisasi data** adalah representasi grafis dari informasi. Sebagai seorang analis data, Anda pasti ingin membuat visualisasi yang membuat data Anda mudah dipahami dan menarik untuk dilihat. Karena pentingnya visualisasi data, sebagian besar alat analisis data (seperti spreadsheet dan database) memiliki komponen visualisasi bawaan, sementara alat lain (seperti Tableau) berspesialisasi dalam visualisasi sebagai nilai tambah utamanya. Dalam bacaan ini, Anda akan mempelajari langkah-langkah yang terlibat dalam proses visualisasi data dan beberapa alat visualisasi data paling umum yang tersedia.



## Langkah-langkah untuk merencanakan visualisasi data

Mari kita lihat contoh situasi kehidupan nyata ketika seorang analis data mungkin perlu membuat visualisasi data untuk dibagikan kepada pemangku kepentingan. Bayangkan Anda seorang analis data untuk distributor pakaian. Perusahaan membantu toko pakaian kecil mengelola inventaris mereka, dan penjualan meningkat pesat. Suatu hari, Anda mengetahui bahwa perusahaan Anda sedang bersiap untuk melakukan pembaruan besar-besaran pada situs webnya. Untuk memandu keputusan pembaruan situs web, Anda diminta menganalisis data dari situs web dan catatan penjualan yang ada. Mari ikuti langkah-langkah yang mungkin Anda ikuti.

### Langkah 1: Jelajahi data untuk mencari pola

Pertama, Anda meminta manajer Anda atau pemilik data untuk mengakses catatan penjualan terkini dan laporan analisis situs web. Ini mencakup informasi tentang bagaimana pelanggan berperilaku di situs web perusahaan yang ada, informasi dasar tentang siapa yang mengunjungi, siapa yang membeli dari perusahaan, dan berapa banyak yang mereka beli.

Saat meninjau data, Anda melihat pola di antara mereka yang paling sering mengunjungi situs web perusahaan: geografi dan jumlah pengeluaran yang lebih besar untuk pembelian. Dengan analisis lebih lanjut, informasi ini mungkin menjelaskan mengapa penjualan begitu kuat saat ini di wilayah timur laut—dan membantu perusahaan Anda menemukan cara untuk menjadikannya lebih kuat lagi melalui situs web baru.

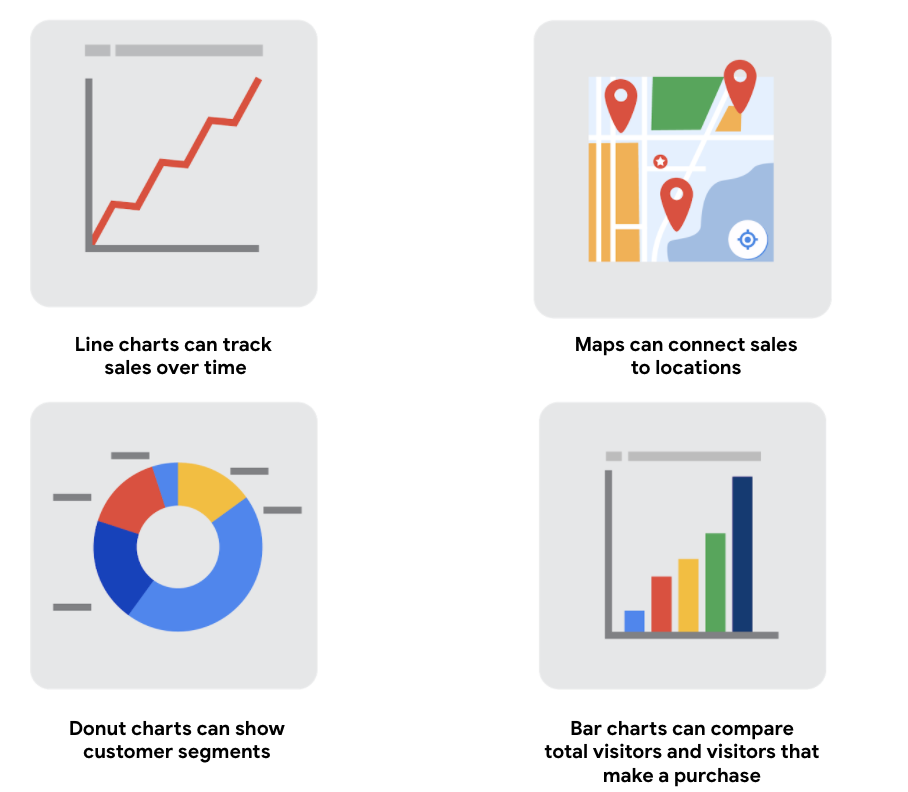
### Langkah 2: Rencanakan visual Anda

Selanjutnya saatnya menyempurnakan data dan menyajikan hasil analisis Anda. Saat ini, Anda memiliki banyak data yang tersebar di beberapa tabel berbeda, dan ini bukan cara ideal untuk membagikan hasil Anda kepada manajemen dan tim pemasaran. Anda ingin membuat visualisasi data yang menjelaskan temuan Anda dengan cepat dan efektif kepada audiens target Anda. Karena Anda tahu audiens Anda berorientasi pada penjualan, Anda sudah tahu bahwa visualisasi data yang Anda gunakan harus:

* Tampilkan nomor penjualan dari waktu ke waktu
* Hubungkan penjualan ke lokasi
* Tunjukkan hubungan antara penjualan dan penggunaan situs web
* Tunjukkan pelanggan mana yang mendorong pertumbuhan

### Langkah 3: Buat visual Anda

Sekarang setelah Anda memutuskan jenis informasi dan wawasan yang ingin Anda tampilkan, sekarang saatnya mulai membuat visualisasi sebenarnya. Ingatlah bahwa menciptakan visualisasi yang tepat untuk presentasi atau untuk dibagikan kepada pemangku kepentingan adalah sebuah proses. Ini melibatkan mencoba format visualisasi yang berbeda dan membuat penyesuaian sampai Anda mendapatkan apa yang Anda cari. Dalam hal ini, perpaduan berbagai visual akan mengkomunikasikan temuan Anda dengan baik dan mengubah analisis Anda menjadi cerita yang paling menarik bagi para pemangku kepentingan. Jadi, Anda dapat menggunakan kemampuan bagan bawaan di spreadsheet Anda untuk mengatur data dan membuat visual Anda.

1) diagram garis dapat melacak penjualan dari waktu ke waktu 2) peta dapat menghubungkan penjualan dengan lokasi 3) diagram donat dapat menunjukkan segmen pelanggan 4) diagram batang dapat membandingkan total pengunjung yang melakukan pembelian

## Bangun perangkat visualisasi data Anda

Ada banyak alat berbeda yang dapat Anda gunakan untuk visualisasi data.

* Anda dapat menggunakan alat visualisasi di spreadsheet Anda untuk membuat visualisasi sederhana seperti diagram garis dan batang.
* Anda dapat menggunakan alat yang lebih canggih seperti Tableau yang memungkinkan Anda mengintegrasikan data ke dalam visualisasi bergaya dasbor.
* Jika Anda bekerja dengan bahasa pemrograman R, Anda dapat menggunakan alat visualisasi di RStudio.

Pilihan visualisasi Anda akan didorong oleh berbagai faktor termasuk ukuran data Anda, proses yang Anda gunakan untuk menganalisis data Anda (spreadsheet, atau database/kueri, atau bahasa pemrograman). Untuk saat ini, pertimbangkan saja dasar-dasarnya.

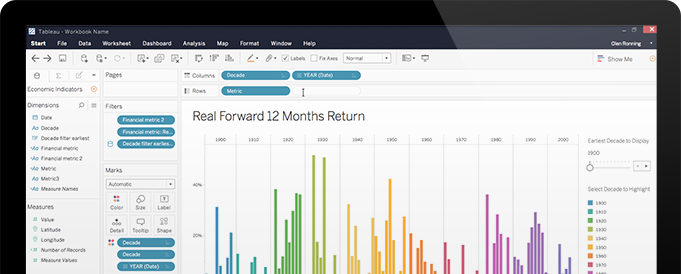
## Spreadsheet (Microsoft Excel atau Google Spreadsheet)

Dalam contoh kita, bagan dan grafik bawaan dalam spreadsheet membuat proses pembuatan visual menjadi cepat dan mudah. Spreadsheet sangat bagus untuk membuat visualisasi sederhana seperti grafik batang dan diagram lingkaran, dan bahkan menyediakan beberapa visualisasi tingkat lanjut seperti peta, serta diagram air terjun dan corong (ditunjukkan pada gambar berikut).

Namun terkadang Anda memerlukan alat yang lebih canggih untuk benar-benar menghidupkan data Anda. Tableau dan RStudio adalah dua contoh platform yang banyak digunakan yang dapat membantu Anda merencanakan, membuat, dan menyajikan visualisasi data yang efektif dan menarik.

## Perangkat lunak visualisasi (Tableau)

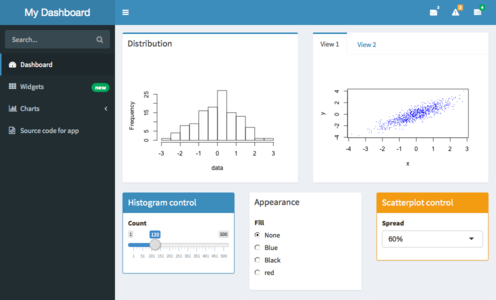
Tableau adalah alat visualisasi data populer yang memungkinkan Anda mengambil data dari hampir semua sistem dan mengubahnya menjadi visual yang menarik atau wawasan yang dapat ditindaklanjuti. Platform ini menawarkan praktik terbaik visual bawaan, yang membuat analisis dan berbagi data menjadi cepat, mudah, dan (yang paling penting) bermanfaat. Tableau berfungsi baik dengan beragam data dan menyertakan dasbor interaktif yang memungkinkan Anda dan pemangku kepentingan mengklik untuk menjelajahi data secara interaktif.



Anda dapat mulai menjelajahi Tableau dari[Video Petunjuk](https://public.tableau.com/en-us/s/resources)sumber daya. Tableau Public gratis, mudah digunakan, dan penuh dengan informasi bermanfaat. Halaman Sumber Daya adalah tempat terpadu untuk video petunjuk, contoh, dan kumpulan data untuk Anda praktikkan. Untuk mempelajari apa yang dibagikan oleh analis data lain di Tableau, kunjungi[Yaitu Hari Ini](https://public.tableau.com/en-us/gallery/?tab=viz-of-the-day&type=viz-of-the-day)halaman di mana Anda akan menemukan visual yang indah mulai dari[Perburuan Planet (Layak Huni).](https://public.tableau.com/en-us/gallery/hunt-habitable-planets?tab=viz-of-the-day&type=viz-of-the-day)ke[Siapa yang Berbicara di Film Populer](https://public.tableau.com/en-us/gallery/whos-talking-popular-films?tab=viz-of-the-day&type=viz-of-the-day).

## Bahasa pemrograman (R dengan RStudio)

Banyak analis data bekerja dengan bahasa pemrograman yang disebut R. Kebanyakan orang yang bekerja dengan R akhirnya juga menggunakan RStudio, sebuah lingkungan pengembang terintegrasi (IDE), untuk kebutuhan visualisasi data mereka. Seperti Tableau, Anda dapat membuat visualisasi data bergaya dasbor menggunakan RStudio.



Kunjungi situs web mereka untuk mempelajari lebih lanjut[RStudio](https://rstudio.com/).

Anda dapat dengan mudah menghabiskan waktu berhari-hari menjelajahi semua sumber daya yang disediakan di RStudio.com, tetapi[Lembar Cheat RStudio](https://rstudio.com/resources/cheatsheets/)dan itu[RStudio Visualisasikan Data Primer](https://rstudio.cloud/learn/primers/3)adalah tempat yang bagus untuk memulai. Jika Anda memiliki lebih banyak waktu, lihat webinar dan video yang menawarkan saran dan perspektif bermanfaat bagi pengguna pemula dan lanjutan.

## Pengambilan kunci

Analis data terbaik menggunakan banyak alat dan metode berbeda untuk memvisualisasikan dan membagikan data mereka. Saat Anda terus mempelajari lebih lanjut tentang visualisasi data sepanjang kursus ini, pastikan untuk tetap penasaran, teliti berbagai opsi, dan terus uji program dan platform baru untuk membantu Anda memaksimalkan data Anda.