**Verifikasi pembersihan data: Daftar periksa**

Bacaan ini akan memberi Anda daftar masalah umum yang dapat Anda rujuk saat melakukan verifikasi pembersihan data, apa pun alat yang Anda gunakan. Terkait verifikasi pembersihan data, tidak ada pendekatan universal atau daftar periksa tunggal yang dapat diterapkan secara universal pada semua proyek. Setiap proyek memiliki organisasi dan persyaratan datanya sendiri yang mengarah pada daftar unik hal-hal yang harus dijalankan untuk verifikasi.



Perlu diingat, saat Anda menerima lebih banyak data atau pemahaman yang lebih baik tentang tujuan proyek, Anda mungkin ingin meninjau kembali beberapa atau semua langkah berikut.

**Perbaiki masalah yang paling umum**

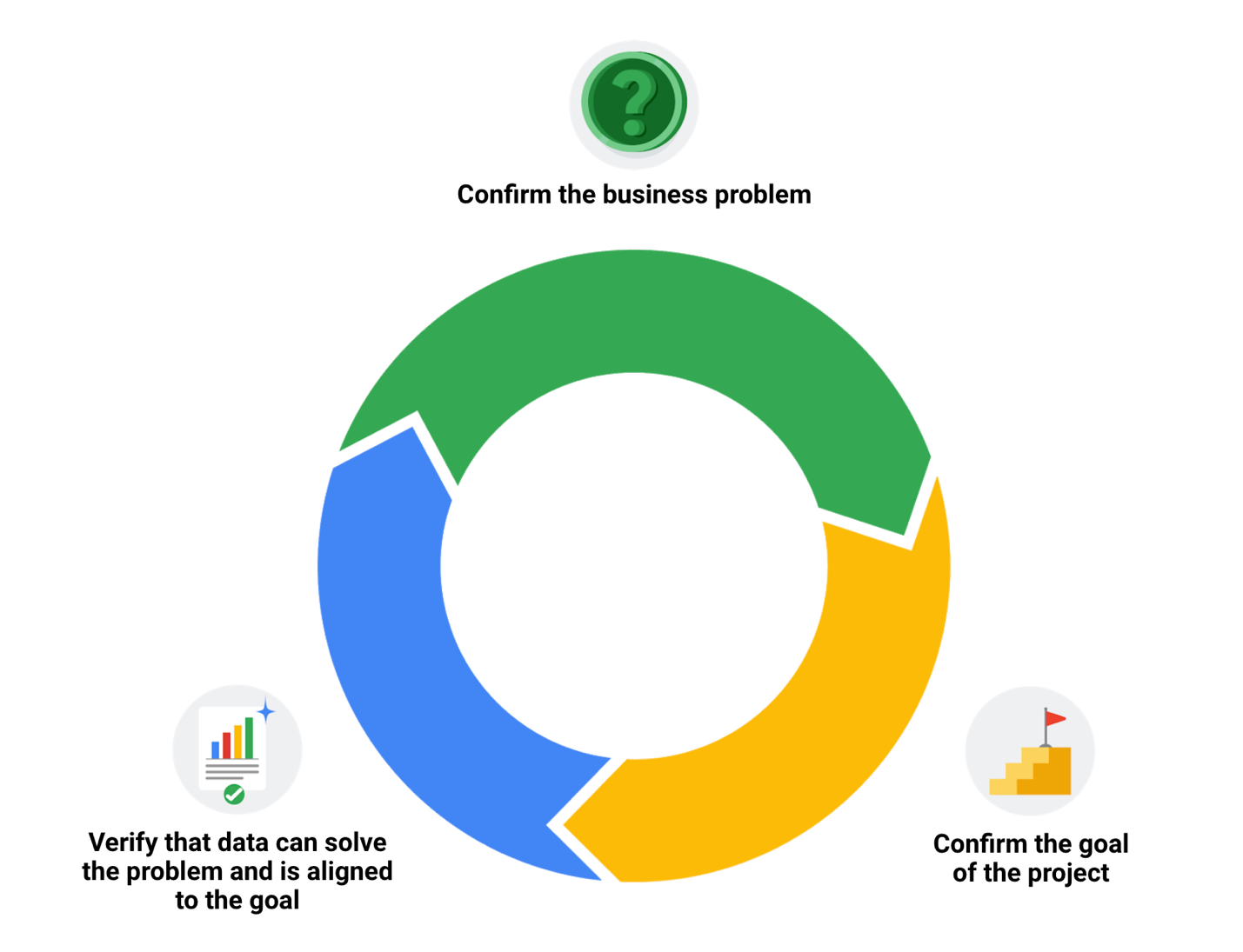
Pastikan Anda mengidentifikasi masalah paling umum dan memperbaikinya, termasuk:

* **Sumber kesalahan** : Apakah Anda menggunakan alat dan fungsi yang tepat untuk menemukan sumber kesalahan dalam kumpulan data Anda?
* **Data nol** : Apakah Anda mencari NULL menggunakan pemformatan dan filter bersyarat?
* **Kata-kata yang salah eja** : Apakah Anda menemukan semua kesalahan eja?
* **Nomor yang salah ketik** : Apakah Anda memeriksa ulang apakah data numerik Anda telah dimasukkan dengan benar?
* **Spasi dan karakter tambahan** : Apakah Anda menghapus spasi atau karakter tambahan menggunakan fungsi **TRIM** ?
* **Duplikat** : Apakah Anda menghapus duplikat di spreadsheet menggunakan fungsi **Hapus Duplikat** atau **DISTINCT** di SQL?
* **Tipe data tidak cocok** : Apakah Anda memeriksa apakah data numerik, tanggal, dan string sudah diketik dengan benar?
* **String berantakan (tidak konsisten)** : Apakah Anda memastikan bahwa semua string Anda konsisten dan bermakna?
* **Format tanggal yang berantakan (tidak konsisten)** : Apakah Anda memformat tanggal secara konsisten di seluruh kumpulan data Anda?
* **Label variabel (kolom) yang menyesatkan** : Apakah Anda memberi nama pada kolom Anda dengan bermakna?
* **Data terpotong:** Apakah Anda memeriksa data terpotong atau hilang yang perlu diperbaiki?
* **Logika Bisnis** : Apakah Anda memeriksa apakah datanya masuk akal berdasarkan pengetahuan Anda tentang bisnis tersebut?

**Tinjau tujuan proyek Anda**

Setelah Anda menyelesaikan tugas pembersihan data ini, sebaiknya tinjau tujuan proyek Anda dan konfirmasikan bahwa data Anda masih selaras dengan tujuan tersebut. Ini adalah proses berkelanjutan yang akan Anda lakukan sepanjang proyek Anda-- namun berikut adalah tiga langkah yang dapat Anda ingat saat memikirkan hal ini:

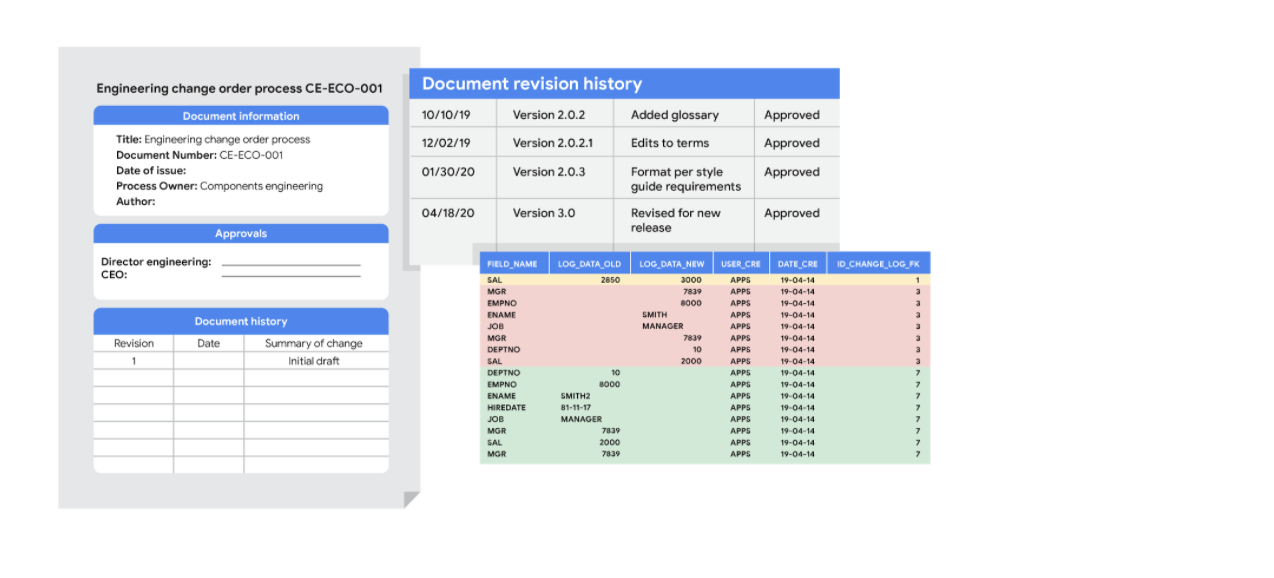
* Konfirmasikan masalah bisnis
* Konfirmasikan tujuan proyek
* Verifikasi bahwa data dapat menyelesaikan masalah dan selaras dengan tujuan



**Gunakan log perubahan**

Apa kesamaan yang dimiliki para insinyur, penulis, dan analis data? Mengubah.

Insinyur menggunakan **engineering change order** (ECO) untuk melacak detail desain produk baru dan usulan perubahan pada produk yang sudah ada. Penulis menggunakan **riwayat revisi dokumen** untuk melacak perubahan aliran dan pengeditan dokumen. Dan analis data menggunakan **log perubahan** untuk melacak transformasi dan pembersihan data. Berikut beberapa contohnya:



**Kontrol versi otomatis membawa Anda semaksimal mungkin**

Sebagian besar aplikasi perangkat lunak memiliki semacam pelacakan riwayat bawaan. Misalnya, di lembar Google, Anda dapat memeriksa riwayat versi seluruh lembar atau sel individual dan kembali ke versi sebelumnya. Di Microsoft Excel, Anda dapat menggunakan fitur yang disebut **Lacak Perubahan** . Dan di BigQuery, Anda dapat melihat histori untuk memeriksa apa saja yang berubah.

Begini cara kerjanya:

|  |  |
| --- | --- |
| Google Spreadsheet | 1. Klik kanan sel dan pilih **Tampilkan riwayat edit** . 2. Klik panah kiri < atau panah kanan > untuk bergerak mundur dan maju dalam riwayat sesuai kebutuhan. |
| Microsoft Excel | 1. Jika Lacak Perubahan telah diaktifkan untuk spreadsheet: klik **Tinjau** . 2. Di bawah **Lacak Perubahan** , klik opsi **Terima/Tolak Perubahan** untuk menerima atau menolak perubahan apa pun yang dilakukan. |
| Kueri Besar | Buka versi sebelumnya (tanpa kembali ke versi tersebut) dan cari tahu apa yang berubah dengan membandingkannya dengan versi saat ini. |

**Changelogs membawa Anda ke tahap terakhir**

Changelog dapat membangun riwayat versi otomatis Anda dengan memberi Anda catatan pekerjaan Anda yang lebih detail **.** Di sinilah analis data mencatat semua perubahan yang mereka lakukan pada data. Inilah cara lain untuk melihatnya. Riwayat versi mencatat *apa* yang dilakukan dalam perubahan data untuk suatu proyek, namun tidak memberi tahu kami *alasannya* . Changelog sangat berguna untuk membantu kami memahami alasan perubahan dilakukan. Changelog tidak memiliki format tetap dan Anda bahkan dapat membuat entri dalam dokumen kosong. Namun jika Anda menggunakan log perubahan bersama, sebaiknya sepakati dengan analis data lain mengenai format semua entri log Anda.

Biasanya, changelog mencatat jenis informasi ini:

* Data, file, rumus, kueri, atau komponen lain apa pun yang diubah
* Deskripsi tentang apa yang berubah
* Tanggal perubahan
* Orang yang melakukan perubahan
* Orang yang menyetujui perubahan tersebut
* Nomor versi
* Alasan perubahan tersebut

Katakanlah Anda membuat perubahan pada rumus di spreadsheet karena Anda mengamatinya di laporan lain dan ingin data Anda cocok dan konsisten. Jika kemudian Anda mengetahui bahwa laporan tersebut sebenarnya menggunakan rumus yang salah, riwayat versi otomatis akan membantu Anda *membatalkan* perubahan. Namun jika Anda juga mencatat alasan perubahan tersebut di log perubahan, Anda dapat kembali menemui pembuat laporan dan memberi tahu mereka tentang rumus yang salah. Jika perubahan terjadi beberapa waktu lalu, Anda mungkin tidak ingat siapa yang harus ditindaklanjuti. Untungnya, log perubahan Anda sudah menyiapkan informasi tersebut untuk Anda! Dengan menindaklanjutinya, Anda akan memastikan integritas data di luar proyek Anda. Anda juga akan menunjukkan integritas pribadi sebagai seseorang yang dapat dipercaya dengan data. Itulah kekuatan dari changelog!

Terakhir, log perubahan penting ketika banyak perubahan pada spreadsheet atau kueri telah dilakukan. Bayangkan seorang analis membuat empat perubahan dan perubahan yang ingin mereka kembalikan adalah perubahan #2. Daripada mengklik fitur batalkan tiga kali untuk membatalkan perubahan #2 (dan kehilangan perubahan #3 dan #4), analis dapat membatalkan perubahan #2 saja dan menyimpan semua perubahan lainnya. Sekarang, contoh kita hanya untuk 4 perubahan, tapi coba pikirkan betapa pentingnya log perubahan itu jika ada ratusan perubahan yang harus dilacak.

**Apa yang juga terjadi IRL (dalam kehidupan nyata)**



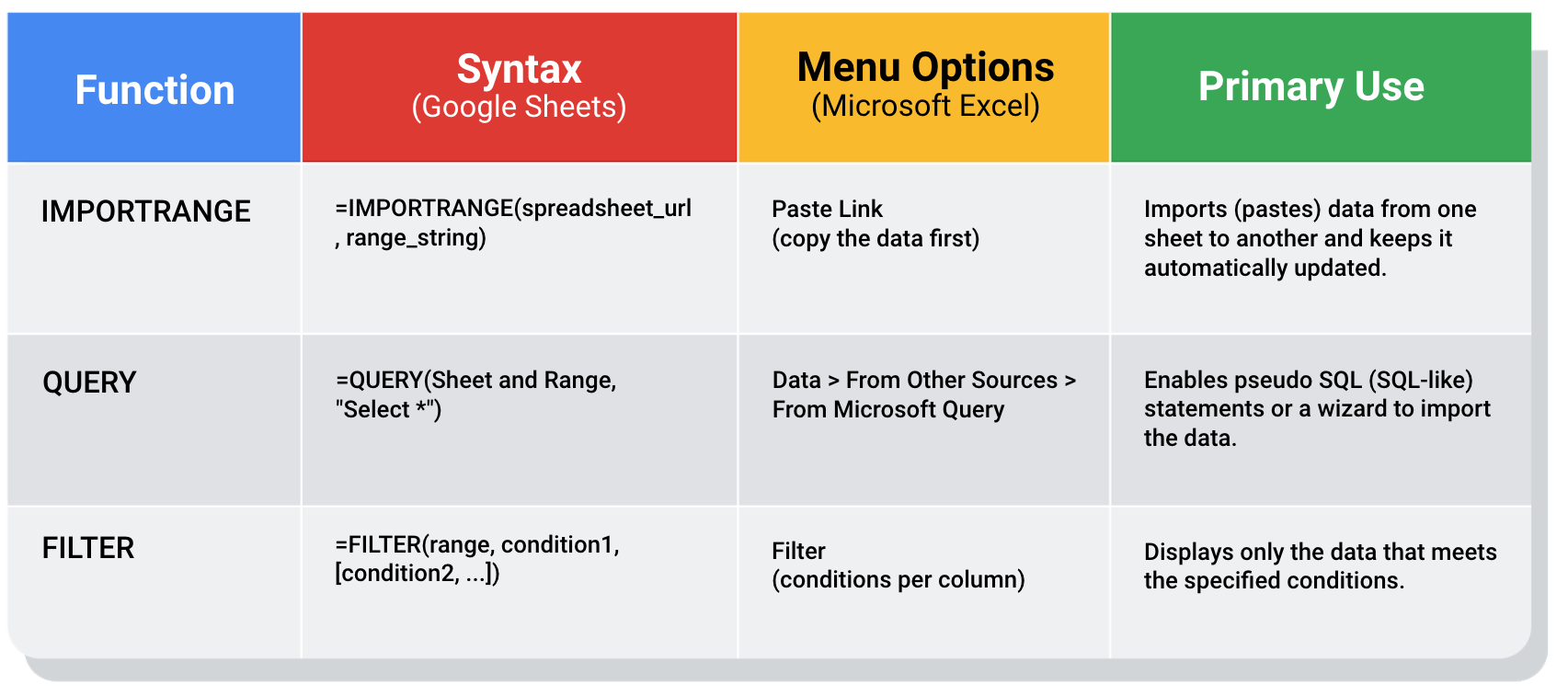
Seorang analis junior mungkin hanya perlu mengetahui hal di atas dengan satu pengecualian. Jika seorang analis membuat perubahan pada kueri SQL yang ada dan dibagikan ke seluruh perusahaan, kemungkinan besar perusahaan tersebut menggunakan apa yang disebut **sistem kontrol versi** . Contohnya adalah kueri yang menarik pendapatan harian untuk membuat dasbor bagi manajemen senior.

Berikut adalah bagaimana sistem kontrol versi memengaruhi perubahan pada kueri:

1. Sebuah perusahaan memiliki versi resmi dari pertanyaan penting dalam **sistem kontrol versi** mereka .
2. Seorang analis memastikan versi kueri terbaru adalah versi yang akan mereka ubah. Ini disebut **sinkronisasi**
3. Analis membuat perubahan pada kueri.
4. Analis mungkin meminta seseorang untuk meninjau perubahan ini. Ini disebut **peninjauan kode** dan dapat dilakukan secara informal atau formal. Tinjauan informal dapat dilakukan dengan cara sederhana seperti meminta analis senior untuk melihat perubahannya.
5. Setelah peninjau menyetujui perubahan, analis mengirimkan versi kueri yang diperbarui ke repositori di sistem kontrol versi perusahaan. Ini disebut **komit kode** . Praktik terbaiknya adalah mendokumentasikan secara pasti perubahan apa yang terjadi dan alasan perubahan tersebut dilakukan di area komentar. Kembali ke contoh kueri yang memperoleh pendapatan harian, komentarnya mungkin: *Pendapatan yang diperbarui untuk menyertakan pendapatan yang berasal dari produk baru, Calypso* .
6. Setelah perubahan **dikirimkan** , semua orang di perusahaan akan dapat mengakses dan menggunakan kueri baru ini saat mereka **menyinkronkan** ke kueri terbaru yang disimpan di sistem kontrol versi.
7. Jika kueri mengalami masalah atau kebutuhan bisnis berubah, analis dapat ***membatalkan*** perubahan pada kueri menggunakan sistem kontrol versi. Analis dapat melihat daftar kronologis dari semua perubahan yang dilakukan pada kueri dan siapa yang membuat setiap perubahan. Kemudian, setelah menemukan perubahannya sendiri, analis dapat **kembali** ke versi sebelumnya.
8. Kuerinya kembali seperti semula sebelum analis melakukan perubahan. Dan semua orang di perusahaan juga melihat kueri asli yang dikembalikan ini.

# Fungsi lanjutan untuk pembersihan data cepat

Dalam bacaan ini, Anda akan mempelajari beberapa fungsi lanjutan yang dapat membantu Anda mempercepat proses pembersihan data di spreadsheet. Di bawah ini adalah tabel yang merangkum tiga fungsi dan fungsinya:

IMPORTRANGE: Sintaks: =IMPORTRANGE(spreadsheet\_url, range\_string) Opsi Menu: Tempel Tautan (salin data terlebih dahulu) Kegunaan Utama: Mengimpor (menempelkan) data dari satu lembar ke lembar lainnya dan terus memperbaruinya secara otomatis QUERY: Sintaks: =QUERY(Lembar dan Rentang , "Pilih \*") Opsi Menu: Data > Dari Sumber Lain > Dari Microsoft Query Penggunaan Utama: Mengaktifkan pernyataan pseudo SQL (seperti SQL) atau wizard untuk mengimpor data. FILTER: Sintaks: =FILTER(range, condition1, [condition2, ...]) Pilihan Menu: Filter(kondisi per kolom) Penggunaan Utama: Hanya menampilkan data yang memenuhi kondisi yang ditentukan.

## Menjaga data tetap bersih dan sinkron dengan sumbernya

Itu[**PENTING**](https://support.google.com/docs/answer/3093340?hl=en)berfungsi di Google Sheets dan[**Tempel Tautan**](https://professor-excel.com/how-to-paste-cell-links/)fitur (opsi Tempel Spesial di Microsoft Excel) keduanya memungkinkan Anda memasukkan data dari satu lembar ke lembar lainnya. Menggunakan ini pada data dalam jumlah besar lebih efisien daripada menyalin dan menempel secara manual. Mereka juga mengurangi kemungkinan terjadinya kesalahan dengan menyalin dan menempelkan data yang salah. Mereka juga berguna untuk pembersihan data karena Anda dapat “memilih” data yang ingin Anda analisis dan meninggalkan data yang tidak relevan dengan proyek Anda. Pada dasarnya, ini seperti menghilangkan gangguan dari data sehingga Anda dapat fokus pada hal yang paling penting untuk menyelesaikan masalah Anda. Fungsionalitas ini juga berguna untuk pemantauan data sehari-hari; dengan itu, Anda dapat membuat spreadsheet pelacakan untuk berbagi data yang relevan dengan orang lain. Data disinkronkan dengan sumber data sehingga ketika data diperbarui di file sumber, data yang dilacak juga disegarkan.

Di Google Sheets, Anda dapat menggunakan fungsi IMPORTRANGE. Ini memungkinkan Anda menentukan rentang sel di spreadsheet lain untuk diduplikasi di spreadsheet tempat Anda bekerja. Anda harus mengizinkan akses ke spreadsheet yang berisi data saat pertama kali Anda mengimpor data.

**URL yang ditampilkan di bawah ini hanya untuk tujuan sintaksis. Jangan memasukkannya ke dalam spreadsheet Anda sendiri. Ganti dengan URL ke spreadsheet yang Anda buat sehingga Anda dapat mengontrol akses ke spreadsheet tersebut dengan mengklik tombol Izinkan akses.**

A screenshot of a spreadsheet

Description automatically generated

Mengacu kepada[Halaman dukungan Google untuk IMPORTRANGE](https://support.google.com/docs/answer/3093340?hl=en)untuk contoh penggunaan dan sintaksis.

### **Contoh penggunaan IMPORTRANGE**

Seorang analis yang memantau penggalangan dana perlu melacak dan memastikan bahwa dana pendamping didistribusikan. Mereka menggunakan **IMPORTRANGE** untuk menarik semua transaksi yang cocok ke dalam spreadsheet yang berisi semua donasi individu. Hal ini memungkinkan mereka untuk menentukan sumbangan mana yang memenuhi syarat untuk dana pendamping yang masih perlu diproses. Karena jumlah total transaksi yang cocok meningkat setiap hari, transaksi tersebut hanya perlu mengubah rentang yang digunakan oleh fungsi untuk mengimpor data terbaru.

Pada hari Selasa, mereka menggunakan yang berikut ini untuk mengimpor nama donor dan jumlah yang cocok:

=IMPORTRANGE(“https://docs.google.com/spreadsheets/d/abcd123abcd123", "sheet1!A1:C10”, “Dana Sesuai!A1:B4001”)

Pada hari Rabu, 500 transaksi lainnya diproses. Mereka meningkatkan rentang yang digunakan sebesar 500 untuk dengan mudah memasukkan transaksi terbaru saat mengimpor data ke spreadsheet donor individu:

=IMPORTRANGE(“https://docs.google.com/spreadsheets/d/abcd123abcd123”, “Dana Sesuai!A1:B4501”)

**Catatan: Contoh di atas hanya untuk tujuan ilustrasi. Jangan menyalin dan menempelkannya ke dalam spreadsheet Anda. Untuk mencobanya sendiri, Anda perlu mengganti URL Anda sendiri (dan nama sheet jika Anda memiliki banyak tab) bersama dengan rentang sel di spreadsheet yang telah Anda isi dengan data.**

## Menarik data dari sumber data lain

Itu[**PERTANYAAN**](https://support.google.com/docs/answer/3093343?hl=en)Fungsi ini juga berguna ketika Anda ingin mengambil data dari spreadsheet lain. Kemampuan seperti SQL pada fungsi **QUERY** dapat mengekstrak data tertentu dalam spreadsheet. Untuk data dalam jumlah besar, penggunaan fungsi QUERY lebih cepat dibandingkan memfilter data secara manual. Hal ini terutama berlaku bila pemfilteran berulang diperlukan. Misalnya, Anda dapat membuat daftar semua pelanggan yang membeli produk perusahaan Anda pada bulan tertentu menggunakan pemfilteran manual. Namun jika Anda juga ingin mengetahui pertumbuhan pelanggan dari bulan ke bulan, Anda harus menyalin data yang difilter ke spreadsheet baru, memfilter data untuk penjualan selama bulan berikutnya, lalu menyalin hasil tersebut untuk analisis. Dengan **QUERY**fungsinya, Anda bisa mendapatkan semua data untuk kedua bulan tanpa perlu mengubah dataset asli atau menyalin hasil.

Sintaks fungsi **QUERY** mirip dengan **IMPORTRANGE** . Anda memasukkan lembar berdasarkan nama dan rentang data yang ingin Anda kueri, lalu menggunakan perintah SQL **SELECT** untuk memilih kolom tertentu. Anda juga dapat menambahkan kriteria spesifik setelah pernyataan **SELECT** dengan menyertakan pernyataan **WHERE** . Tapi ingat, semua kode SQL yang Anda gunakan harus ditempatkan di antara tanda kutip!

Google Spreadsheet menjalankan Bahasa Kueri API Visualisasi Google di seluruh data. Lembar bentang Excel menggunakan panduan kueri untuk memandu Anda melalui langkah-langkah menyambung ke sumber data dan memilih tabel. Apa pun kasusnya, Anda dapat memastikan bahwa data yang diimpor telah diverifikasi dan bersih berdasarkan kriteria dalam kueri.

### **Contoh penggunaan QUERY**

Lihat[Halaman dukungan Google untuk fungsi QUERY](https://support.google.com/docs/answer/3093343?hl=en)dengan contoh penggunaan, sintaksis, dan contoh dapat Anda unduh di lembar Google.

Tautan untuk membuat salinan lembar:[Contoh QUERY](https://docs.google.com/spreadsheets/d/1815H5TCe91LLT6tD6FmxMHmeJAAkr4o5Q6rNpV6xiFk/copy)

### **Solusi kehidupan nyata**

Analis dapat menggunakan SQL untuk menarik kumpulan data tertentu ke dalam spreadsheet. Mereka kemudian dapat menggunakan fungsi **QUERY** untuk membuat beberapa tab (tampilan) dari kumpulan data tersebut. Misalnya, satu tab dapat berisi semua data penjualan untuk bulan tertentu dan tab lainnya dapat berisi semua data penjualan dari wilayah tertentu. Solusi ini menggambarkan bagaimana SQL dan spreadsheet digunakan bersama-sama dengan baik.

## Memfilter data untuk mendapatkan apa yang Anda inginkan

Itu[**SARING**](https://support.google.com/docs/answer/3093197?hl=en)Fungsi sepenuhnya internal pada spreadsheet dan tidak memerlukan penggunaan bahasa kueri. Fungsi FILTER memungkinkan Anda melihat hanya baris (atau kolom) di data sumber yang memenuhi ketentuan yang Anda tentukan. Ini memungkinkan untuk memfilter data terlebih dahulu sebelum Anda menganalisisnya.

Fungsi **FILTER** mungkin berjalan lebih cepat daripada fungsi **QUERY** . Namun perlu diingat, fungsi **QUERY** dapat digabungkan dengan fungsi lain untuk perhitungan yang lebih kompleks. Misalnya, fungsi **QUERY** dapat digunakan dengan fungsi lain seperti **SUM** dan **COUNT** untuk meringkas data, namun fungsi **FILTER** tidak bisa.

### **Contoh penggunaan FILTER**

Lihat[Halaman dukungan Google untuk fungsi FILTER](https://support.google.com/docs/answer/3093197?hl=en)dengan contoh penggunaan, sintaksis, dan contoh dapat Anda unduh di lembar Google.

Tautan untuk membuat salinan lembar:[Contoh FILTER](https://docs.google.com/spreadsheets/d/1caULJLQvQuzBnCN7rO9utg0xSKrYms7wM0Ph7A2JXY4/copy)