DuckDB

Panduan Praktis SQL DuckDB di GeoLibre.

Belajar ngolah data vektor dari nol sampe analisis spasial



Defani Arman Alfitriansyah

Panduan Praktis SQL DuckDB di GeoLibre.

Belajar ngolah data vektor dari nol sampe analisis spasial

Defani Arman Alfitriansyah

Prakata

Halo temen-temen! Buku ini lahir dari pertanyaan yang sering banget mampir ke kolom komentar: "Bang, ngolah data vektor yang banyak tuh enaknya pake apa?" Jawaban saya sekarang: SQL. Dan tempat paling enak buat belajarnya adalah GeoLibre, karena di dalemnya udah ada DuckDB yang jalan langsung di laptop kamu, tanpa server, tanpa ribet.

Alasan lain buku ini ditulis: buku tutorial SIG berbahasa Indonesia, apalagi yang ngebahas SQL spasial dari nol sampe studi kasus, masih jarang banget. Kebanyakan materi bagus masih berbahasa Inggris, jadi temen-temen yang baru mulai sering mentok duluan di bahasanya. Lewat buku ini saya pengen semua orang bisa belajar pake bahasa sehari-hari kita sendiri.

Gaya bahasanya sengaja santai, kayak lagi ngajarin temen sebelah. Tiap query ditulis lengkap, terus dibedah satu per satu: artinya apa, cara bacanya gimana, dan logikanya kenapa begitu. Datanya dekat sama kita: batas desa dan jalan RBI Kabupaten Kuningan, Open Data Jabar, tuplah dari Kementerian Kehutanan, sampe data biodiversitas dari GBIF dan iNaturalist.

Gak perlu jago coding dulu buat mulai. Yang penting penasaran dan mau nyoba. Kalo ada yang mentok, santai, di tengah bab nanti saya nongol buat ngingetin atau nyemangatin.

Selamat belajar, jeng jeng jeng!

Defani Arman Alfitriansyah
Kuningan, 2026

Daftar Isi

BAGIAN I · KENALAN DULU

1 Kenalan Dulu sama GeoLibre 3

1.1 Pengantar Teori: GeoLibre Itu Apa Sih? 4
1.2 Praktikum Bagian 1: Memasang dan Membuka GeoLibre 6
1.3 Praktikum Bagian 2: Tur Antarmuka, Ini Apa, Buat Apa 7
1.4 Praktikum Bagian 3: Jenis-Jenis Layer 11
1.5 Pertanyaan 11
1.6 Sintesis 12
1.7 Kesimpulan 12
1.8 Bacaan Lanjut 12

2 Operasional Dasar GeoLibre 13

2.1 Pengantar Teori: Data Kecil, Data Besar 14
2.2 Praktikum Bagian 1: Memasukkan Data 17
2.3 Praktikum Bagian 2: Mengelola Layer 17
2.4 Praktikum Bagian 3: Simbologi Dasar 20
2.5 Praktikum Bagian 4: Tabel Atribut 21
2.6 Praktikum Bagian 5: Milih Fitur 22
2.7 Praktikum Bagian 6: Ekspor Data 27
2.8 Praktikum Bagian 7: Nyimpen Proyek 28
2.9 Pertanyaan 28
2.10 Sintesis 29
2.11 Kesimpulan 29

3 Kenapa SQL, dan Kenapa di GeoLibre? 31

3.1 Pengantar Teori 32
3.2 Praktikum: Membuktikan Sendiri 35
3.3 Kenapa SQL di GeoLibre Itu Enak 36
3.4 Kapan SQL Bukan Pilihan Terbaik 38
3.5 Pertanyaan 38
3.6 Sintesis 39
3.7 Kesimpulan 39
3.8 Bacaan Lanjut 39

4 Data Vektor dan Format-formatnya 41

4.1 Vektor dan Raster 42
4.2 Tiga Tipe Geometri 43

4.3 Data Spasial Itu Excel yang Punya Koordinat 43
4.4 Gimana Koordinat Disimpen: WKT dan WKB 44
4.5 Format File Vektor 44
4.6 Kenapa Shapefile Filenya Banyak? 45
4.7 Data Vektor di Buku Ini 47
4.8 Pertanyaan 47
4.9 Kesimpulan 47

5 Tur Lengkap SQL Workspace 49
5.1 Pengantar: Satu Jendela, Banyak Fungsi 50
5.2 Praktikum: Bedah Setiap Bagian 51
5.3 Praktikum: Memanfaatkan History 54
5.4 Praktikum: Lima Error yang Pasti Kamu Temui 55
5.5 Pertanyaan 59
5.6 Sintesis 59
5.7 Kesimpulan 59

6 Muat Data dan Mengenali Isinya 61
6.1 Pengantar Teori 62
6.2 Praktikum Bagian 1: Muat Data Vektor 64
6.3 Praktikum Bagian 2: Muat CSV ala BPS 67
6.4 Praktikum Bagian 3: Ritual Mengenali Data 69
6.5 Praktikum Bagian 4: Kejutan di Data RBI 73
6.6 Pertanyaan 75
6.7 Sintesis 75
6.8 Kesimpulan 75

7 Open Data Jabar: Belanja Data Gratis 77
7.1 Kenalan sama Galura 78
7.2 Praktikum Bagian 1: Nyari Dataset 79
7.3 Praktikum Bagian 2: Muat CSV ke GeoLibre 80
7.4 Praktikum Bagian 3: Langsung dari API 80
7.5 Praktikum Bagian 4: Mapset, Data Spasial dari Galura 81
7.6 Praktikum Bagian 5: Dataset Kab/Kota 82
7.7 Pertanyaan 82
7.8 Sintesis 82
7.9 Kesimpulan 83

BAGIAN II · SQL DASAR BUAT DATA ATRIBUT

8 SELECT: Milih Kolom dan Baris 87
8.1 Pengantar Teori: Anatomi Sebuah Query 88
8.2 Praktikum 89
8.3 Pertanyaan 95

BAGIAN I

01.

Kenalan Dulu

Sebelum ngetik query, kita kenalan dulu sama alatnya: GeoLibre, SQL, DuckDB, dan data vektor. Bagian ini juga ngajak kamu muat data pertama, termasuk dari Open Data Jabar.

1 Kenalan Dulu sama GeoLibre

2 Operasional Dasar GeoLibre

3 Kenapa SQL, dan Kenapa di GeoLibre?

4 Data Vektor dan Format-formatnya

5 Tur Lengkap SQL Workspace

6 Muat Data dan Mengenali Isinya

7 Open Data Jabar: Belanja Data Gratis



1.

Kenalan Dulu sama GeoLibre

BELAJAR APA KITA

Sebelum ngetik query apa pun, kita kenalan dulu sama "rumah" tempat kita bakal kerja sepanjang buku ini: GeoLibre. Di bab ini kita bahas GeoLibre itu apa, siapa yang bikin, kenapa dia ringan banget padahal fiturnya segudang, cara instalasinya, dan tur lengkap semua bagian antarmukanya. Pokoknya setelah bab ini, kamu gak bakal bingung lagi "tombol ini buat apa sih?".

TARGET BAB INI

- Menjelaskan apa itu GeoLibre dan teknologi yang ada di baliknya.
- Memasang GeoLibre Desktop atau membuka GeoLibre versi web.
- Mengenali setiap bagian antarmuka GeoLibre: menu utama, panel **Layers**, panel **Browser**, peta, kontrol peta, panel **Style**, **Comments**, **Live MCP**, dan bilah status.
- Menjelaskan fungsi setiap menu utama, **Project**, **Edit**, **View**, **Add data**, **Processing**, **Controls**, **Plugins**, **Settings**, **Help**, **Layout Composer**.
- Paham perbedaan layer vektor, raster, dan layer hasil query.

SEBELUM MULAI

Belum ada. Ini bab pertama, santai aja. Cukup punya komputer, Windows, macOS, atau Linux, atau sekadar peramban modern kayak Chrome, Edge, atau Firefox.

1.1 Pengantar Teori: GeoLibre Itu Apa Sih?

Definisi singkat

GeoLibre adalah platform SIG, Sistem Informasi Geografis, yang gratis, sumber terbuka, ringan, dan *cloud-native*. Gunanya untuk nampilin, menjelajah, dan menganalisis data geospasial. Proyek ini dikembangkan oleh Prof. Qiusheng Wu bersama komunitas *opengeos*, tim yang juga dikenal lewat pustaka `leafmap`, `geemap`, dan `segment-geospatial`.

Kata kuncinya ada di "*runs everywhere*". GeoLibre bisa jalan di:

- **Desktop**, Windows, macOS, Linux, sebagai aplikasi biasa yang dipasang.
- **Web**, cukup buka alamat `https://geolibre.app` di peramban.
- **Android** sebagai aplikasi seluler.

Tabel 7.1 Jenis data di Open Data Jabar

Jenis	Isinya	Bentuk	Contoh
Dataset	Tabel statistik tingkat provinsi	Tabel, bisa didonlot CSV/XLSX dan diakses lewat API	Jumlah penduduk per kabupaten/kota 2010–2025
Dataset Kab/Kota	Tabel statistik dari portal data kabupaten/kota	Tautan ke portal kabupaten, misalnya opendata.kuningankab.go.id	Penduduk per kecamatan dan agama di Kuningan
Mapset	Data spasial	Peta, bisa didonlot SHP atau GeoJSON	Sebaran realisasi investasi per kabupaten/kota
Infografik	Ringkasan data bergambar	Gambar	Kuningan penghasil ubi jalar terbanyak
Visualisasi	Dasbor interaktif	Tableau di dalam halaman	Tren kependudukan dan ekonomi
Artikel dan publikasi	Tulisan dan laporan	Halaman web dan PDF	Rilis berita data

Selain Galura, Jawa Barat juga punya Sapeta Jabar, sapeta.jabarprov.go.id, yang khusus buat peta, dan geoserver.jabarprov.go.id yang ngelayanin WMS. Dari kabupaten, tiap daerah punya portal sendiri. Kuningan punya opendata.kuningankab.go.id.



KATA DEFANI

Kalo kamu nemu data yang cuma ada dalam bentuk infografik, jangan langsung diketik ulang. Cari dulu di bagian Dataset pake kata kunci yang sama. Sering banget angka di infografik itu asalnya dari satu dataset yang bisa didonlot.

7.2 Praktikum Bagian 1: Nyari Dataset

1. Buka galura.jabarprov.go.id, ketik kata kunci di kotak pencarian, misalnya `jumlah penduduk kabupaten kota`.
2. Hasilnya dikelompokin per jenis. Klik dataset *Jumlah Penduduk Berdasarkan Kabupaten/Kota di Jawa Barat*.
3. Halaman dataset nampilin deskripsi, produsen data, satuan, periode, dan tabel pratinjau. Dataset ini produksinya BPS, satuannya ribu orang, isinya 27 kabupaten/kota dari 2010 sampe 2025.

```sql
-- Ambil nama dan geometri aja biar ramping
CREATE OR REPLACE TABLE kab_jabar AS
SELECT wadmkk AS kabupaten, geom
FROM batas_kab_jabar;
```

Kode 12.2 Tabel batas kabupaten/kota

12.3 Praktikum Bagian 2: Join Pertama, dan Kenapa Gagal

Coba gabungin langsung berdasarkan nama:

```sql
-- Berapa pasangan yang ketemu kalo namanya dicocokin apa adanya?
SELECT count(*) AS cocok
FROM kab_jabar AS k
JOIN penduduk_jabar AS b
  ON k.kabupaten = b.nama_kabupaten_kota
 AND b.tahun = 2025;
```

Kode 12.3 Percobaan join pake nama mentah

Hasilnya nol. Gak ada satu pun yang cocok, soalnya BPS nulis pake huruf kapital semua. Oke, samain dulu hurufnya pake `lower()`:

```sql
SELECT count(*) AS cocok
FROM kab_jabar AS k
JOIN penduduk_jabar AS b
  ON lower(k.kabupaten) = lower(b.nama_kabupaten_kota)
 AND b.tahun = 2025;
```

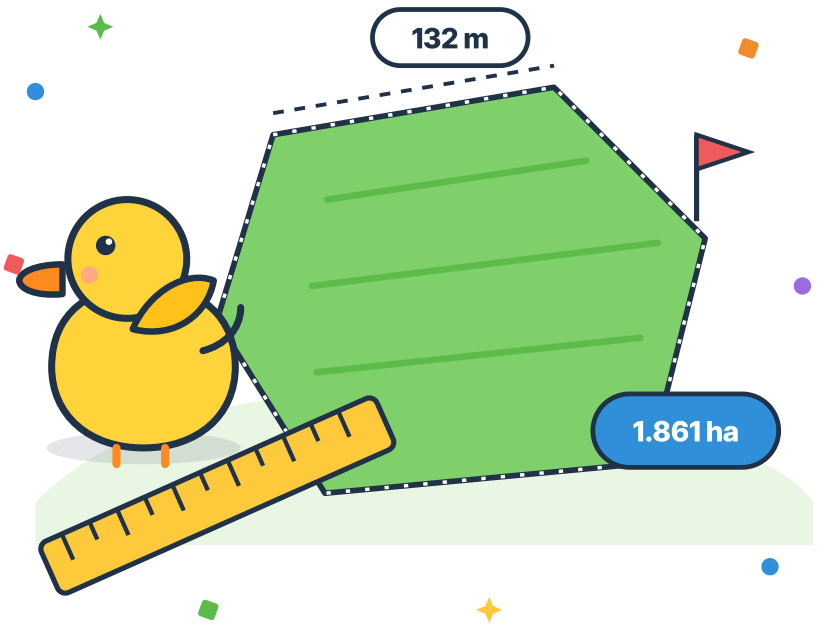
Kode 12.4 Percobaan kedua: huruf disamain

Sekarang 9 pasangan ketemu, dan semuanya kota. “Kota Bandung” sama “KOTA BANDUNG” cocok setelah hurufnya disamain. Tapi 18 kabupaten masih jomblo, soalnya peta nulis “Kuningan” sedangkan BPS nulis “KABUPATEN KUNINGAN”.



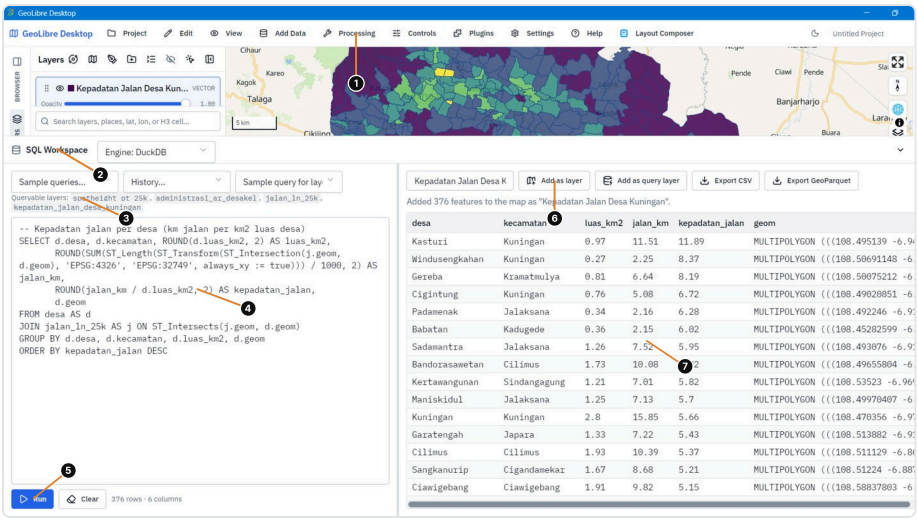
PERTANYAAN BUAT KAMU

Kenapa kata “KABUPATEN” gak boleh asal dihapus dari dua sisi? Coba pikirin Bogor. Ada Kabupaten Bogor, ada juga Kota Bogor. Kalo kata depannya ilang semua, dua-duanya jadi “bogor” dan join-nya bakal dobel.



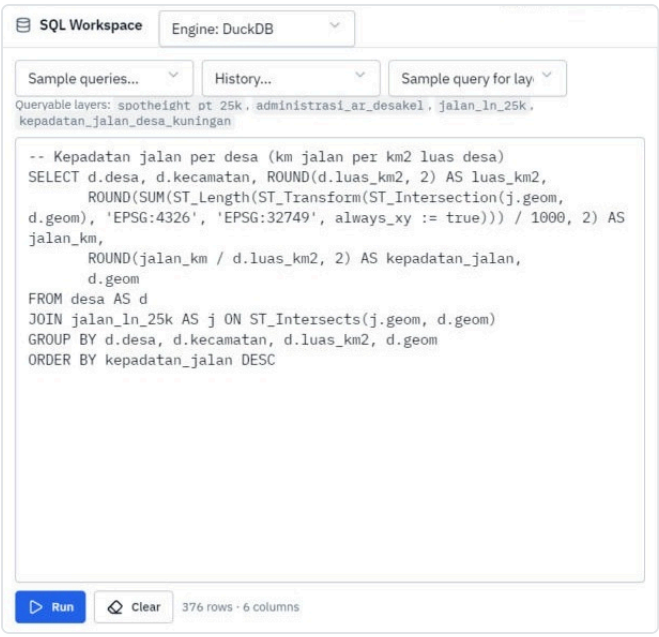
17.

Ngukur: Luas, Panjang, Keliling, dan Kepadatan



Gambar 17.4 Tampilan utuh GeoLibre waktu ngerjain query kepadatan jalan. Nomor 1 menu Processing, 2 panel SQL Workspace, 3 daftar Queryable layers, 4 editor, 5 tombol Run, 6 tombol Add as layer, 7 tabel hasil.

1. Klik **Add Data** → **Vector**, muat `administrasi_ar_desakel` dan `jalan_ln_25k` dari folder RBI Kuningan.
2. Buka **Processing** → **SQL Workspace** (nomor 1 dan 2). Cek dulu di *Queryable layers* (nomor 3), dua nama tabel tadi harus nongol.
3. Bikin tabel `desa` yang udah punya kolom luas, terus ketik query kepadatan di editor (nomor 4). Jalanin pake **Run** atau **Ctrl**+**Enter** (nomor 5).
4. Kalo tabel hasil udah keluar (nomor 7), ganti nama layernya, terus klik **Add as layer** (nomor 6).
5. Di panel Layers, klik **Style** di layer baru, pilih gradasi pake kolom `kepadatan_jalan`.



Gambar 17.5 Fokus ke bagian kiri SQL Workspace: daftar Queryable layers, editor, dan tombol Run. Nama tabel di Queryable layers itulah yang ditulis di FROM dan JOIN.



NGINGETIN NIH

Nama tabel di Queryable layers kamu beda sama punyaku? Santai, tinggal ganti aja di FROM sama JOIN. Yang penting tulisannya persis sama, garis bawahnya juga.

Langkah 3 punya dua query. Yang pertama nyiapin tabel desa:

```sql
CREATE OR REPLACE TABLE desa AS
SELECT NAMOBJ AS desa, WADMKC AS kecamatan,
  ST_Area(ST_Transform(geom, 'EPSG:4326', 'EPSG:32749', always_xy
:= true)) / 1e6 AS luas_km2,
  geom
FROM desa_kng;
```

Kode 17.5 Tabel desa yang udah punya luas

Yang kedua query utamanya, persis kayak yang keliatan di gambar:

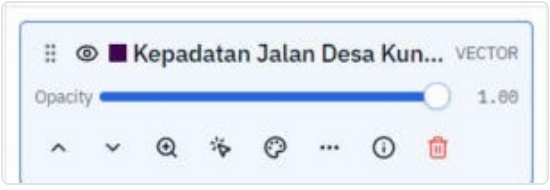
Tabel 17.6 Bedah query kepadatan jalan, bagian bagi, gabung, dan urut

#	Sintaks	Artinya	Cara bacanya
4	`/ 1000`	Meter dibagi 1.000 jadi kilometer.	"... dalam km"
5	`jalan_km / d.luas_km2`	Panjang jalan dibagi luas desa. Alias `jalan_km` boleh dipake lagi di baris yang sama.	"Km jalan tiap km² desa"
7	`FROM desa AS d`	Tabel utama, nama pendeknya d.	"Dari tabel desa, sebut aja d"
8	`JOIN jalan_ln_25k AS j`	Gabungin tabel jalan, nama pendeknya j.	"Gabung sama tabel jalan, sebut j"
8	`ON ST_Intersects(j.geom, d.geom)`	Syarat gabungnya: jalan sama desa harus nyentuh atau motong.	"Pasangin kalo jalan j nyentuh desa d"
9	`GROUP BY d.desa, …`	Satuin baris per desa. Semua kolom di SELECT yang gak dijumlahin wajib ditulis di sini.	"Kelompokin per desa"
10	`ORDER BY … DESC`	Urutin dari gede ke kecil. ASC kebalikannya.	"Urutin dari yang paling rapet"

Kalo dibaca jadi satu kalimat: "Dari tabel desa, pasangin tiap desa sama jalan yang nyentuh desa itu. Potong jalannya sesuai batas desa, ubah ke meter, ukur panjangnya, terus jumlahin per desa. Bagi sama luas desa, lalu urutin dari yang paling rapet."



SEMANGAT!
Kalo udah bisa baca query ini kalimat per kalimat, selamat ya! Bagian paling susah di bab ini udah kelewat. Rehat bentar, ngopi dulu, baru lanjut.



Gambar 17.7 Kartu layer di panel Layers. Ikon palet buat ngatur warna.

Buku ini ngebahas cara ngolah dan ngatur data vektor pake SQL langsung di GeoLibre.

Mulai dari kenalan sama GeoLibre dan SQL Workspace, upload data, simbologi, seleksi, sampe export. Terus struktur SQL DuckDB: SELECT, WHERE, rekap pake GROUP BY, gabungin tabel pake JOIN, CTE, sama window function.

Lanjut ke urusan spasial: sistem koordinat, ngitung luas, panjang, sama jarak yang bener, relasi spasial, buffer, clip, dissolve, dan bersih-bersih data vektor yang berantakan. Ditutup sama studi kasus stok karbon, indeks keanekaragaman, 100 resep SQL sehari-hari, sama kamus sintaks.

Tiap query dijelasin cara bacanya, logikanya, sama bagian mana aja yang bisa diganti buat data sendiri.

