

## **Sistem Informasi Geografis**

### **Tugas 4**



**Disusun Oleh :**

**M. Gymnastiar Syahputra (123140135)**

**Dosen Pengampu :**

**Muhammad Habib Algifari, S.Kom., M.TI**

**Alya Khairunnisa Rizkita, S.Kom., M.Kom**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INDUSTRI**

**INSTITUT TEKNOLOGI SUMATERA**

## Tugas 4

Link Github : [https://github.com/gymnastiarsyahputra/Tugas-SIG\\_123140135.git](https://github.com/gymnastiarsyahputra/Tugas-SIG_123140135.git)

### Deskripsi Tugas :

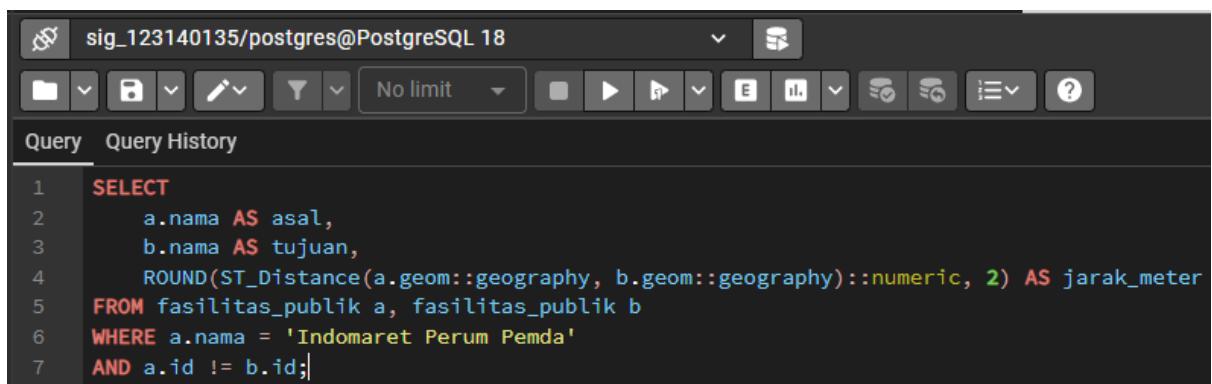
Lakukan analisis spasial menggunakan data dari praktikum sebelumnya. Gunakan fungsi pengukuran dan relasi topologi untuk menjawab pertanyaan analitis

### Ketentuan Tugas:

- Buat minimal 5 query spasial menggunakan fungsi berbeda
- Wajib: ST\_Distance, ST\_Intersects, ST\_Contains/ST\_Within
- Implementasikan K-NN untuk mencari fasilitas terdekat
- Buat query agregasi dengan GROUP BY dan fungsi spasial
- Sertakan screenshot hasil dan interpretas

### Langkah-langkah :

#### 1. Menghitung jarak satu fasilitas ke semua fasilitas lainnya dalam satuan meter (ST\_Distance)



The screenshot shows the pgAdmin 4 interface with a query editor window. The title bar says "sig\_123140135/postgres@PostgreSQL 18". The query window contains the following SQL code:

```
1 SELECT
2     a.nama AS asal,
3     b.nama AS tujuan,
4     ROUND(ST_Distance(a.geom::geography, b.geom::geography)::numeric, 2) AS jarak_meter
5 FROM fasilitas_publik a, fasilitas_publik b
6 WHERE a.nama = 'Indomaret Perum Pemda'
7 AND a.id != b.id;
```

Query ini menggunakan fungsi ST\_Distance dengan casting tipe data ::geography untuk menghitung jarak lurus (Euclidean) aktual secara akurat dalam satuan meter antara Indomaret Perum Pemda dengan fasilitas publik lainnya. Kondisi a.id != b.id ditambahkan agar fasilitas tidak menghitung jarak 0 ke dirinya sendiri.

Data Output    Messages    Notifications

	asal character varying (100)	tujuan character varying (100)	jarak_meter numeric
1	Indomaret Perum Pemda	Mushola Al-Azzam	87.56
2	Indomaret Perum Pemda	SPORTERA ARENA	160.08
3	Indomaret Perum Pemda	Gereja Tyrannus Ministry	37.45
4	Indomaret Perum Pemda	Apotek Way Huwi	139.31

2. Mencari jalan mana saja yang melewati atau menyentuh area wilayah tertentu (ST\_Intersects)

sig\_123140135/postgres@PostgreSQL 18

```

SELECT
    w.nama_wilayah,
    j.nama_jalan
FROM wilayah w
JOIN jalan j ON ST_Intersects(w.geom, j.geom);

```

Query ini menggunakan fungsi topologi ST\_Intersects untuk mendeteksi apakah geometri jalan (LineString) bersinggungan atau tumpang tindih dengan batas maupun bagian dalam dari geometri wilayah (Polygon). Hasilnya menampilkan daftar jalan yang melintasi area perumahan/wilayah terkait.

Data Output    Messages    Notifications

	nama_wilayah character varying (100)	nama_jalan character varying (100)
1	Perumahan Gg Perwira	Gg. Perwira

3. mencari tahu fasilitas publik apa saja yang lokasinya masuk secara penuh di dalam suatu batas wilayah (ST\_Within)

```

1  SELECT
2      f.nama AS nama_fasilitas,
3      f.jenis,
4      w.nama_wilayah
5  FROM fasilitas_publik f
6  JOIN wilayah w ON ST_Within(f.geom, w.geom);

```

Menggunakan fungsi relasi ST\_Within, query ini mengevaluasi apakah koordinat titik (Point) fasilitas publik berada sepenuhnya di dalam cakupan area (Polygon) suatu wilayah. Jika nilai evaluasinya TRUE, data fasilitas beserta nama wilayah yang menaunginya akan ditampilkan.

	nama_fasilitas character varying (100)	jenis character varying (50)	nama_wilayah character varying (100)
1	Mushola Al-Azzam	Ibadah	Perumahan Gg Perwira
2	SPORTERA ARENA	Olahraga	Perumahan Gg Perwira

4. Mencari 3 fasilitas publik yang paling dekat dari sebuah titik acuan (misalnya lokasi titik kosan) menggunakan operator K-NN <-> (Nearest Neighbor)

```

1  SELECT
2      nama,
3      jenis,
4      ROUND(ST_Distance(geom::geography, ST_GeomFromText(
5          'POINT(105.31473545369832 -5.354297484228476)', 4326)::geography)::numeric, 2) AS jarak_m
6  FROM fasilitas_publik
7  ORDER BY geom <-> ST_GeomFromText('POINT(105.31473545369832 -5.354297484228476)', 4326)
8  LIMIT 3;

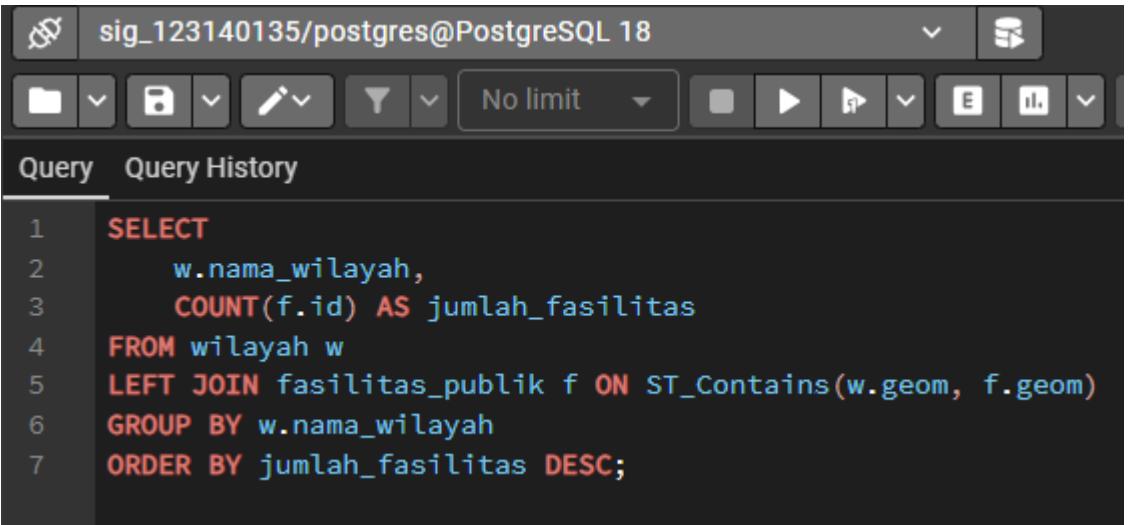
```

Implementasi algoritma *K-Nearest Neighbor* menggunakan operator khusus <-> di PostGIS. Operator ini dioptimasi dengan *spatial index* untuk mengurutkan jarak

terdekat dengan sangat efisien dari titik referensi ke data fasilitas. Query dibatasi dengan LIMIT 3 untuk menampilkan top 3 terdekat.

	nama character varying (100) 	jenis character varying (50) 	jarak_m numeric 
1	SPORTERA ARENA	Olahraga	64.33
2	Mushola Al-Azzam	Ibadah	73.38
3	Indomaret Perum Pemda	Minimarket	128.86

## 5. Menghitung total kepadatan fasilitas per wilayahnya (GROUP BY + ST\_Contains)



```

1  SELECT
2      w.nama_wilayah,
3      COUNT(f.id) AS jumlah_fasilitas
4  FROM wilayah w
5  LEFT JOIN fasilitas_publik f ON ST_Contains(w.geom, f.geom)
6  GROUP BY w.nama_wilayah
7  ORDER BY jumlah_fasilitas DESC;

```

Query ini adalah bentuk analisis agregasi yang menggabungkan klausa GROUP BY dengan fungsi relasi ST\_Contains. Query menghitung dan mengelompokkan jumlah fasilitas publik berdasarkan area wilayah yang menampungnya. Penggunaan LEFT JOIN memastikan wilayah yang tidak memiliki fasilitas tetap muncul dengan nilai perhitungan 0.

	nama_wilayah character varying (100) 	jumlah_fasilitas bigint 
1	Perumahan Gg Perwira	2
2	Depan Indomaret	0

## **Analisis 5 Query Operasi Spasial :**

Kelima operasi spasial yang telah dieksekusi merepresentasikan dua kategori utama dalam Sistem Informasi Geografis, yaitu fungsi pengukuran matematis dan pengujian relasi topologi. Pada kelompok pengukuran, fungsi ST\_Distance dan operator K-NN <-> sama-sama berurusan dengan metrik jarak. Perbedaannya, ST\_Distance berfungsi murni untuk menghitung dan menampilkan nilai jarak aktual antar geometri secara terperinci, terutama bila dikonversi ke tipe geography agar menghasilkan satuan meter yang akurat. Sebaliknya, operator K-Nearest Neighbor (<->) tidak sekadar menghitung jarak, tetapi dikhkususkan untuk mencari dan mengurutkan lokasi terdekat secara instan dengan memanfaatkan indeks spasial (spatial index). Pendekatan ini membuat pencarian lokasi terdekat menggunakan operator <-> sangat efisien.

Sementara itu, perbedaan utama dari ketiga fungsi (ST\_Intersects, ST\_Within, dan STContains) ini terletak pada tingkat keketatan syarat perpotongan ruangnya. ST\_Intersects memiliki syarat yang paling longgar karena akan bernilai *TRUE* asalkan kedua geometri saling bersinggungan atau memiliki setidaknya satu titik ruang yang sama, sehingga sangat cocok untuk mendeteksi jalan yang memotong suatu wilayah. Di sisi lain, ST\_Within memiliki syarat ketat di mana sebuah geometri (misalnya fasilitas publik) harus berada sepenuhnya di dalam area geometri lain tanpa melewati batasnya. Berbeda lagi dengan STContains yang merupakan kebalikan logis dari ST\_Within, yakni memastikan area geometri yang lebih besar mengandung geometri yang lebih kecil sepenuhnya. Pembeda paling menonjol pada penerapan STContains dalam analisis ini adalah integrasinya dengan SQL standar, seperti GROUP BY dan fungsi agregasi. Jika ST\_Intersects dan ST\_Within umumnya digunakan untuk pemfilteran data baris per baris, peng gabungan STContains dengan fungsi agregasi bertujuan untuk meringkas relasi topologi tersebut menjadi sebuah luaran analitik statistik, seperti menghitung kepadatan fasilitas di dalam masing-masing wilayah.