

**!! Catatan:** Diharapkan seluruh pengerjaan Assignment tidak sepenuhnya mengandalkan penggunaan AI **!!**

*"Proses belajar ibarat menanam pohon. Jika hanya mengandalkan AI tanpa memahami esensinya, yang berkembang bukan kompetensimu, melainkan ketergantungan yang melemahkan."  
- Learning Design Dibimbing*

# Assignment Guidance: Case Study

## Advanced SQL Queries

Data Science dan Data Analyst Bootcamp

### Periode Pembelajaran

Advanced SQL Queries and Data Analysis  
Specialized SQL Techniques

### Objectives

1. Student mampu menggunakan subquery untuk mendapatkan informasi lebih kompleks dari database.
2. Student mampu memahami dan menerapkan window functions seperti RANK, DENSE\_RANK, dan ROW\_NUMBER.
3. Student mampu mengoptimalkan pengambilan data menggunakan Common Table Expressions (CTE).
4. Student mampu menggunakan CASE WHEN dalam query SQL.

### Deskripsi Assignment

Assignment ini merupakan kelanjutan dari SQL Fundamentals & Query Basics, di mana student akan menerapkan teknik SQL tingkat lanjut dalam PostgreSQL.

Student akan belajar menggunakan subquery, window functions, Common Table Expressions (CTE), dan CASE WHEN untuk menganalisis data lebih kompleks. Fokus utama dari assignment ini adalah mengoptimalkan pengambilan dan manipulasi data untuk mendukung analisis yang lebih mendalam dan efisien.

### Detail Assignment

Link Dataset : [dvdrental](#)

Lakukan langkah-langkah berikut untuk menyelesaikan tugas:

## 1. Menggunakan Subquery

- Tampilkan nama pelanggan yang pernah melakukan transaksi dengan jumlah lebih dari rata-rata transaksi di tabel **payment**.
- Ambil daftar film yang memiliki durasi lebih panjang dibandingkan durasi rata-rata dari semua film dalam tabel **film**.
- Buat query untuk menampilkan aktor yang hanya membintangi satu film dalam database.

**Goal:** Memahami bagaimana subquery digunakan untuk analisis data lebih dalam.

## 2. Menggunakan Window Functions

- Gunakan **RANK()** untuk menentukan peringkat film berdasarkan **rental\_rate**.
- Gunakan **DENSE\_RANK()** untuk menentukan peringkat pelanggan berdasarkan total transaksi yang mereka lakukan.
- Gunakan **ROW\_NUMBER()** untuk memberikan nomor urut pada daftar film berdasarkan **release\_year**.

**Goal:** Mampu menerapkan window functions dalam analisis data yang memerlukan peringkat dan pengelompokan.

## 3. Menggunakan Common Table Expressions (CTE)

- Gunakan **CTE** untuk membuat daftar pelanggan yang melakukan transaksi lebih dari 10 kali.
- Gunakan **CTE** untuk mendapatkan daftar film dengan jumlah rental terbanyak.

**Goal:** Menerapkan Common Table Expressions untuk menyederhanakan query yang kompleks.

## 4. Menggunakan CASE WHEN untuk Klasifikasi Data

- Buat query yang mengelompokkan film berdasarkan **rental\_rate**:
  - Jika **rental\_rate** lebih dari 4, kategori "Premium"
  - Jika **rental\_rate** antara 2 dan 4, kategori "Regular"
  - Jika **rental\_rate** kurang dari 2, kategori "Budget"
- Buat query yang mengelompokkan pelanggan berdasarkan total transaksi mereka:
  - Pelanggan dengan total transaksi lebih dari \$100 sebagai "High Value Customer"
  - Pelanggan dengan transaksi antara \$50–\$100 sebagai "Medium Value Customer"
  - Pelanggan dengan transaksi di bawah \$50 sebagai "Low Value Customer"

**Goal:** Menggunakan CASE WHEN untuk mengkategorikan data berdasarkan kondisi tertentu.

### Ekspektasi Output

Student harus mengumpulkan:

- **Dataset** dalam bentuk **csv** dan mengandung **nama kolom**
- **Query dan screenshot output** dikumpulkan jadi **1 dalam gdocs**, **dikerjakan** dan **diberi nomor sesuai dengan soal**
- Untuk **output yg besar** dan **tidak memungkinkan untuk ss**, maka seluruh datanya di **export** dalam **gsheet** dengan format file= **Nama Lengkap\_Nomor Soal\_Partisi(if any)**
- Semua **file gdocs** dan **gsheet dimasukkan** dalam **satu folder Google Drive**.
- Link yang **dikumpulkan** adalah **link folder Google Drive** tersebut, **bukan link file terpisah**.
- Pastikan **akses view (dapat dilihat)** diberikan **untuk seluruh file di dalam folder**.

### Tools

PostgreSQL

### Pengumpulan Assignment

**Deadline :**

Maksimal H+7 Kelas (Pukul 23.30 WIB)

**Details :**

Dikumpulkan dalam bentuk PDF Query dan screenshot hasilnya, secara INDIVIDU, di LMS

### Indikator Penilaian

No.	Aspek Penilaian	Parameter	Bobot Maksimal
1	Subquery	Mampu menggunakan subquery dalam analisis data	25
2	Window Functions	Mampu menerapkan RANK, DENSE_RANK, dan ROW_NUMBER	25
3	Common Table Expressions	Mampu menyusun query menggunakan CTE	25
4	CASE WHEN	Mampu mengelompokkan data berdasarkan kondisi tertentu	25

## Referensi/Template

1. <https://dqlab.id/belajar-fungsi-group-by-and-aggregate-di-sql>
2. <https://medium.com/analytics-vidhya/sql-for-data-science-259298a10b0d>

### Sanksi Penggunaan AI:

Apabila student terdeteksi 100% menggunakan AI, maka hasil assignment akan diberikan skor 0.

### Ketentuan Pencapaian Nilai:

Nilai minimum Lulus Penyaluran Kerja: 75

Nilai minimum Lulus Bootcamp: 65

### Ketentuan Penilaian:

Mengumpulkan Assignment tepat waktu: Sesuai dengan nilai yang diberikan mentor

Mengumpulkan Assignment 12 jam setelah deadline: - 3 dari nilai yang diberikan mentor

Mengumpulkan Assignment 1 x 24 Jam setelah deadline: - 6 dari nilai yang diberikan mentor

Mengumpulkan Assignment 2 x 24 Jam setelah deadline: - 12 dari nilai yang diberikan mentor

Mengumpulkan Assignment 3 x 24 Jam setelah deadline: - 18 dari nilai yang diberikan mentor

Mengumpulkan Assignment 4 x 24 Jam setelah deadline: - 24 dari nilai yang diberikan mentor

Mengumpulkan Assignment 5 x 24 Jam setelah deadline: - 30 dari nilai yang diberikan mentor

Mengumpulkan Assignment 6 x 24 Jam setelah deadline: - 36 dari nilai yang diberikan mentor

Mengumpulkan Assignment 7 x 24 Jam setelah deadline: - 42 dari nilai yang diberikan mentor

Mencontek tugas siswa lain mendapatkan nilai 0