

# Assignment Guidance: Case Study

## SQL 2

Data Science Bootcamp

### Periode Pembelajaran

SQL I

SQL II

### Objectives

1. Student mampu membuat database menggunakan PostgreSQL
2. Student mampu membuat Query SQL sesuai dengan Case Study
3. Student mampu menggunakan Aggregate statement
4. Student mampu menggunakan Conditional statement
5. Student mampu menggunakan Join Statement
6. Student mampu menggunakan Union Operation

### Deskripsi Assignment

Solve SQL Query in real case 1

### Detail Assignment

#### Part 1: Create a New Database and Table

1. Buatlah sebuah database dengan nama **dibimbing**.
2. Di dalam database tersebut, buat sebuah tabel bernama **students** dalam skema **public**, dengan struktur kolom sebagai berikut:
  - o **id** (integer)
  - o **nama** (varchar)
  - o **institute** (varchar)
  - o **berat\_badan** (float)
  - o **tinggi\_badan** (float)
3. Isi tabel tersebut dengan minimal **5 baris data**, dan pastikan setiap baris memiliki nilai yang **berbeda-beda**. Nilai data dapat ditentukan secara bebas.

## Part 2: Query on Existing Schema `dvdrental`

Gunakan skema `dvdrental` untuk menjawab pertanyaan berikut:

1. Tampilkan kolom `first_name` dan `last_name` dari tabel `actor` untuk aktor yang memiliki `first_name` **Jennifer**, **Nick**, dan **Ed**.
2. Hitung total `amount` untuk setiap `payment_id` dari tabel `payment`, dan tampilkan hanya baris yang memiliki total `amount` lebih dari **5.99**.  
*Petunjuk: Gunakan klausa `HAVING`.*
3. Dari tabel `film`, tampilkan `film_id`, `title`, `description`, dan `length`. Kelompokkan `length` ke dalam **4 kategori** sebagai berikut:
  - Over 100
  - Between 87 and 100
  - Between 72 and 86
  - Under 72

*Penamaan kategori dapat ditentukan sendiri. Gunakan klausa `CASE WHEN` atau `BETWEEN`.*

4. Gabungkan tabel `rental` dan `payment` untuk menampilkan **10 baris pertama** dengan kolom:
  - `rental_id`
  - `rental_date`
  - `payment_id`
  - `amount`

Urutkan hasil berdasarkan `amount` secara **ascending**.

5. Dari tabel `address`, gabungkan seluruh kolom dari alamat yang memiliki `city_id = 42` dan `city_id = 300` menggunakan `UNION`.

## Tools

PostgreSQL

## Pengumpulan Assignment

### Deadline :

Maksimal H+7 Kelas (Pukul 23.30 WIB)

### Details :

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

## Indikator Penilaian

No.	Aspek Penilaian	Parameter	Bobot Maksimal
1	Pembuatan Database	Student mampu membuat database dan tabel dengan struktur yang benar	20
2	Query SQL	Student mampu menulis query SQL sesuai dengan studi kasus	20
3	Aggregate Statements	Student mampu menggunakan fungsi agregat dengan benar	20
4	Conditional Statements	Student mampu menyaring data menggunakan kondisi yang relevan	10
5	Join Statements	Student mampu menghubungkan tabel dengan teknik JOIN	20
6	UNION Operations	Student mampu menggunakan Union Operation	10

## Referensi/Template

1. <https://www.postgresqltutorial.com/>
2. <https://www.mlq.ai/sql-for-data-science-filtering-sorting-calculating/>
3. <https://medium.com/analytics-vidhya/sql-for-data-science-259298a10b0d>

### 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 1x 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 akan dikenakan penalty (nilai 0)