**LAPORAN PRAKTIKUM**

**BASIS DATA**

**MODUL IV**

**MANIPULASI DAN RETRIEVE DATA (BAGIAN 2)**



Disusun Oleh:

Septiandi Nugraha

21104060

SE05-B

**PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK**

**FAKULTAS INFORMATIKA**

**INSTITUT TEKNOLOGI TELKOM PURWOKERTO**

1. **Tujuan**
2. Dapat menggunakan select statement untuk mengambil data dari satu table
3. Dapat menyeleksi data berdasarkan kondisi tertentu
4. **Dasar Teori**

Select statement digunakan sebagai operasi dalam basis data untuk mengambil sejumlah baris data yang memenuhi predikat yang diberikan. Predikat ini mengacu pada kondisi yang akan dipenuhi dalam operasi seleksi. Berikut perintah-perintahSELECT:

* + 1. Memberikan nama lain pada kolom :

SELECT namakolomlama AS namakolombaru FROM namatabel;

Contoh :

SELECT jenis AS judul\_buku FROM master\_buku;

* + 1. Menggunakan alias untuk nama tabel :

SELECT namaalias.namakolom1, namaalias.namakolom2 FROM namatabel namaalias;

Contoh :

SELECT B.judul\_buku, B.pengarang FROM master\_buku B;

* + 1. Menampilkan data lebih dari dua tabel :

SELECT \* FROM namatabel1, namatabel2, namatabel-n;

Contoh :

SELECT \* FROM master\_buku;

* + 1. Nested Queries / Subquery (IN, NOT IN, EXISTS, NOT EXIXTS)

Subquery yaitu query yang terdiri dari beberapa query. Dengan menggunakan subquery, hasil dari query akan menjadi bagian dari query di atasnya. Subquery terletak di dalam klausa WHEREatau HAVING. Pada klausa WHERE, subquery digunakan untuk memilih baris-baris tertentu yang kemudian digunakan oleh query. Sedangan pada klausa HAVING, subquery digunakan untuk memilih kelompok baris yang kemudian digunakan oleh query.

Contoh penggunaan **IN :**

SELECT \* FROM costumers WHERE country IN (‘Germany’, ‘France’, ‘UK’);

Contoh penggunaan **NOT IN :**

SELECT \* FROM costumers WHERE country NOT IN (‘Germany’, ‘France’, ‘UK’);

Contoh penggunaan **EXIST :**

SELECT SupplierName

FROM Suppliers

WHERE EXISTS (SELECT ProductName FROM Products WHERE SupplierId = Suppliers.supplierId AND Price = 22);

Contoh penggunaan **NOT EXIST :**

SELECT SupplierName

FROM Suppliers

WHERE NOT EXISTS (SELECT ProductName FROM Products WHERE

SupplierId = Suppliers.supplierId AND Price = 22);

Pada contoh di atas : SELECT ProductName FROM Productsdisebut subquery, sedangkan: SELECT SupplierName FROM Suppliersberkedudukan sebagai query.

* + 1. Operator comparison ANY dan ALL

Operator ANY digunakan berkaitan dengan subquery. Operator ini menghasilkan TRUE (benar) jika paling tidak salah satu perbandingan dengan hasil subquery menghasilkan nilai TRUE.

Contoh penggunaan ANY :

SELECT ProductName, Price

FROM Products

WHERE ProductID = ANY (SELECT ProductID FROM OrderDetails

WHERE Quantity = 10);

Operator ALL digunakan untuk melakukan perbandingan dengan subquery. Kondisi dengan ALL menghasilkan nilai TRUE (benar) jika subquery tidak menghasilkan apapun atau jika perbandingan menghasilkan TRUE untuk setiap nilai query terhadap hasil subquery.

Contoh penggunaan ALL :

SELECT ProductName

FROM Products

WHERE ProductID = ALL (SELECT ProductID FROM OrderDetails

WHERE Quantity = 10);

* + 1. Penggunaan ORDER BY

Klausa ORDER BY digunakan untuk mengurutkan data berdasarkan kolom tertentu sesuai dengan tipe data yang dimiliki.

Contoh penggunaan ORDER BY :

SELECT \* FROM Customers

ORDER BY Country;

atau tambahkan ASC untuk pengurutan secara ascending (menaik) :

SELECT \* FROM Customers

ORDER BY Country ASC;

atau tambahkan DESC untuk pengurutan secara descending (menurun) :

SELECT \* FROM Customers

ORDER BY Country DESC;

* + 1. DISTINCT

Distinct adalah kata kunci ini untuk menghilangan duplikasi. Sebagai Contoh: Sebuah tabel pelanggan yang berisi nama dan kota asal dengan beberapa record isi dan beberapa kota asal yang sama. Kemudian ketikkan perintah berikut :

SELECT DISTINCT Country FROM Customers;

Dengan perintah di atas maka nama kota yang sama hanya akan ditampilkan satu saja.

* + 1. UNION, INTERSECT dan EXPECT

UNION merupakan operator yang digunakan untuk menggabungkan hasil query, dengan ketentuan jumlah, nama dan tipe kolom dari masing-masing tabel yang akan ditampilkan datanya harus sama. Berikut ini perintah untuk memperoleh data pada tabel film dimana jenisnya action dan horor :

SELECT City, Country FROM Customers

WHERE Country='Germany'

UNION

SELECT City, Country FROM Customers

WHERE Country=’France’;

Perintah di atas identik dengan :

SELECT City, Country FROM Customers

WHERE Country='Germany' OR Country=’France’;

Namun tidak semua penggabungan dapat dilakukan dengan OR, yaitu jika bekerja pada dua tabel atau lebih.

INTERSECT merupakan operator yang digunakan untuk memperoleh data dari dua buah query dimana data yang ditampilkan adalah yang memenuhi kedua query tersebut dengan ketentuan jumlah, nama dan tipe kolom dari masing-masing tabel yang akan ditampilkan datanya harus sama.

SELECT \* FROM namatabel1 INTERSECT SELECT \* FROM namatabel2 ;

Pada MySQL tidak terdapat operator INTERSECT namun sebagai gantinya dapat menggunakan operator IN seperti pada bagian (d) Nested Queries. EXCEPT / Set Difference merupakan operator yang digunakan untuk memperoleh data dari dua buah query dimana data yang ditampilkan adalah data yang ada pada hasil query 1 dan tidak terdapat pada data dari hasil query 2 dengan ketentuan jumlah, nama dan tipe kolom dari masing-masing tabel yang akan ditampilkan datanya harus sama.

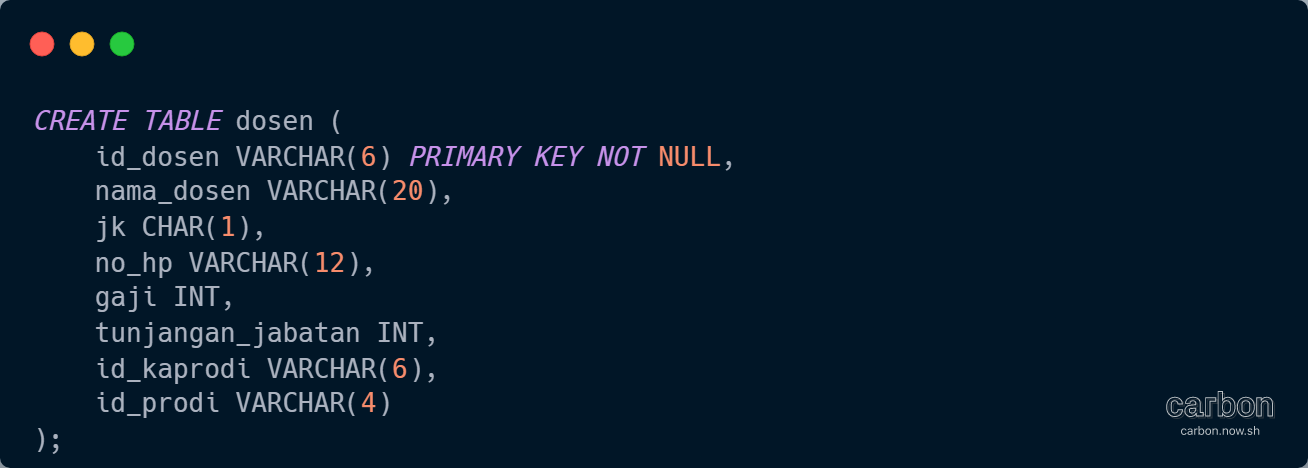
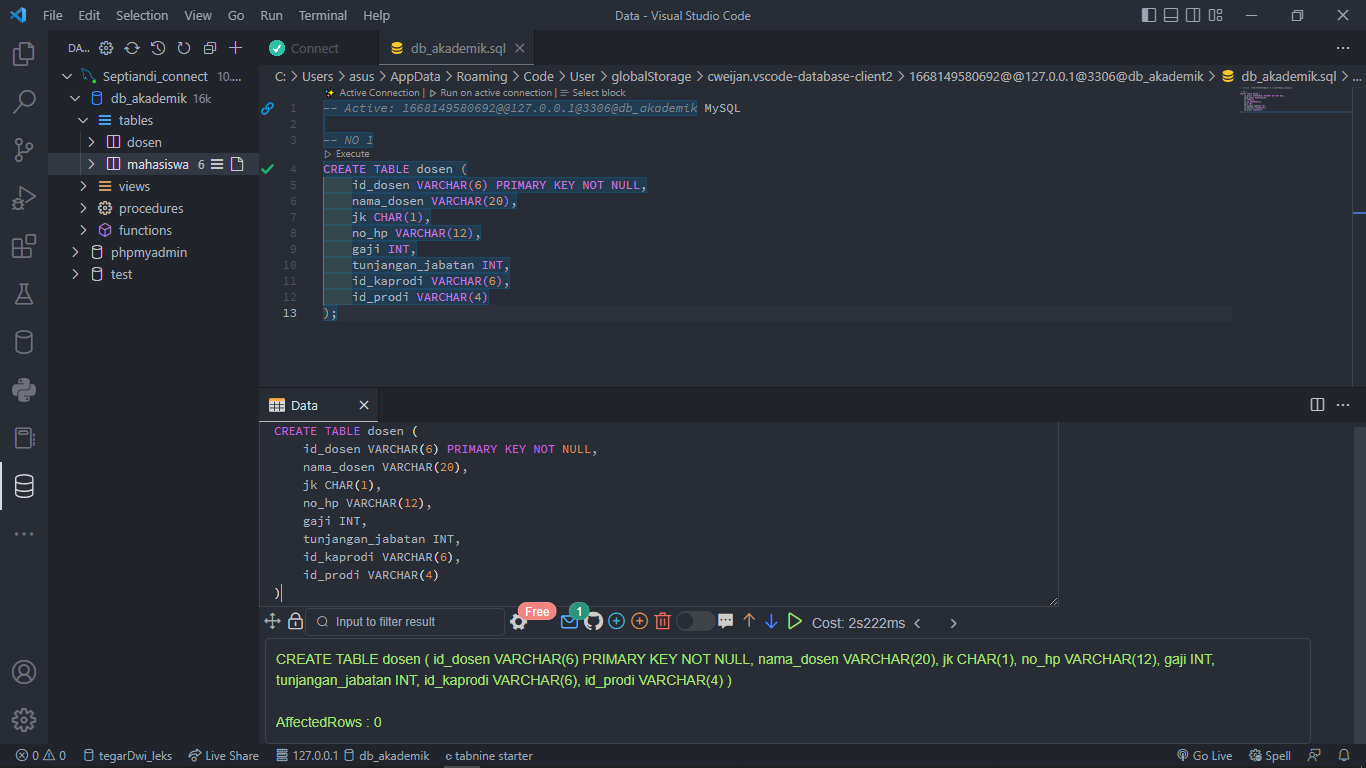
SELECT \* FROM namatabel1 EXCEPT SELECT \* FROM namatabel2;

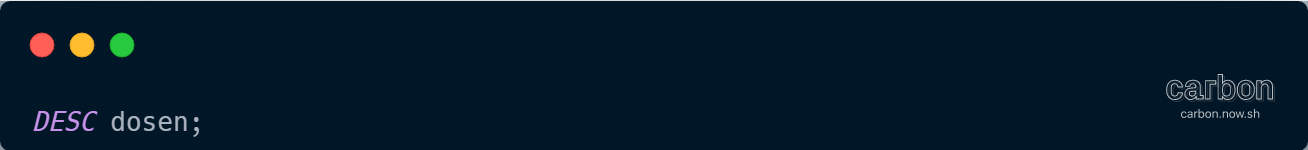
Pada MySQL tidak terdapat operator EXCEPT namun sebagai gantinya dapat

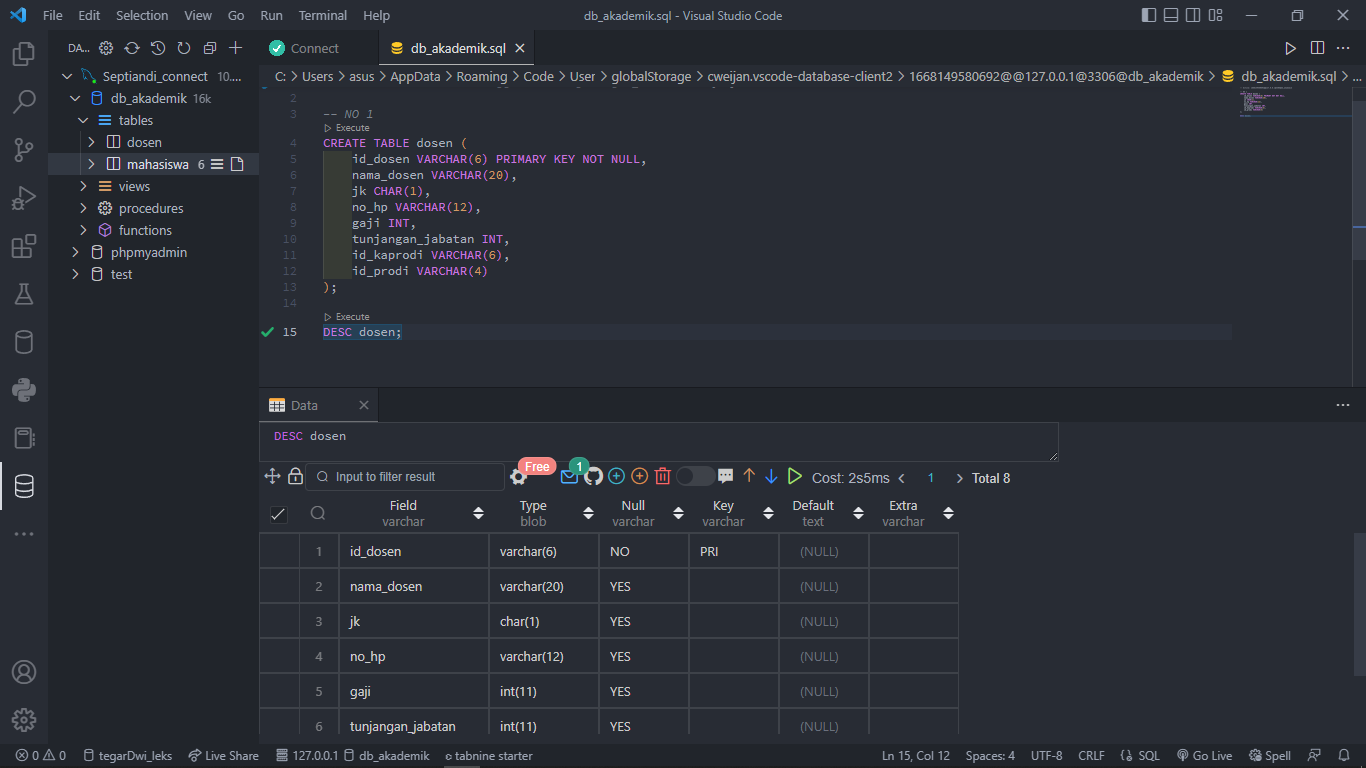
menggunakan operator NOT IN seperti contoh (d) pada bagian Nested Queries.

1. **Praktikum**
   1. Buatlah tabel dosen sebagai berikut

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Ukuran** |
| id\_dosen | Varchar | 6 |
| nama\_dosen | Varchar | 20 |
| jk | Char | 1 |
| no\_hp | Varchar | 20 |
| gaji | Int |  |
| tunjangan\_jabatan | Int |  |
| id\_kaprodi | Varchar | 6 |
| id\_prodi | Varchar | 4 |

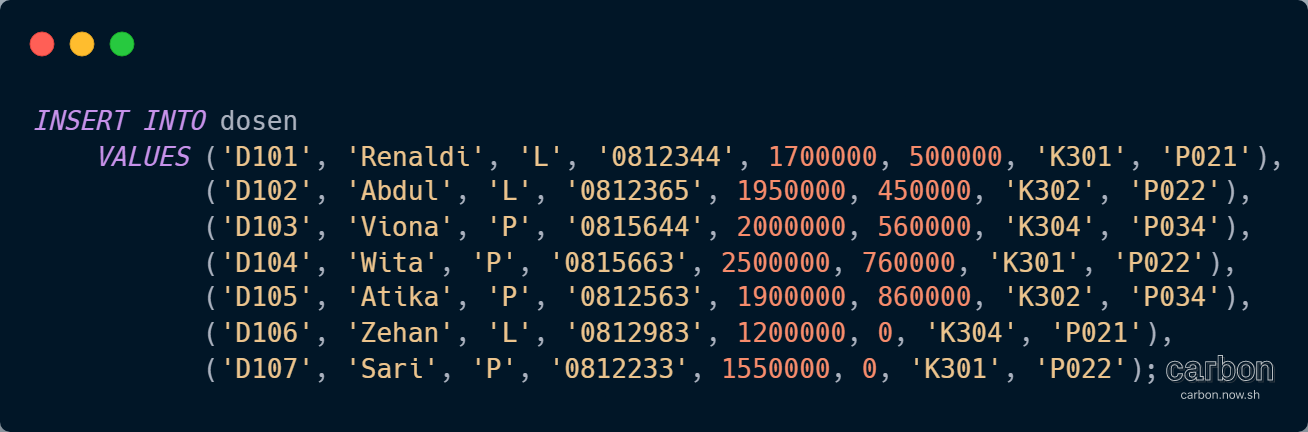
 

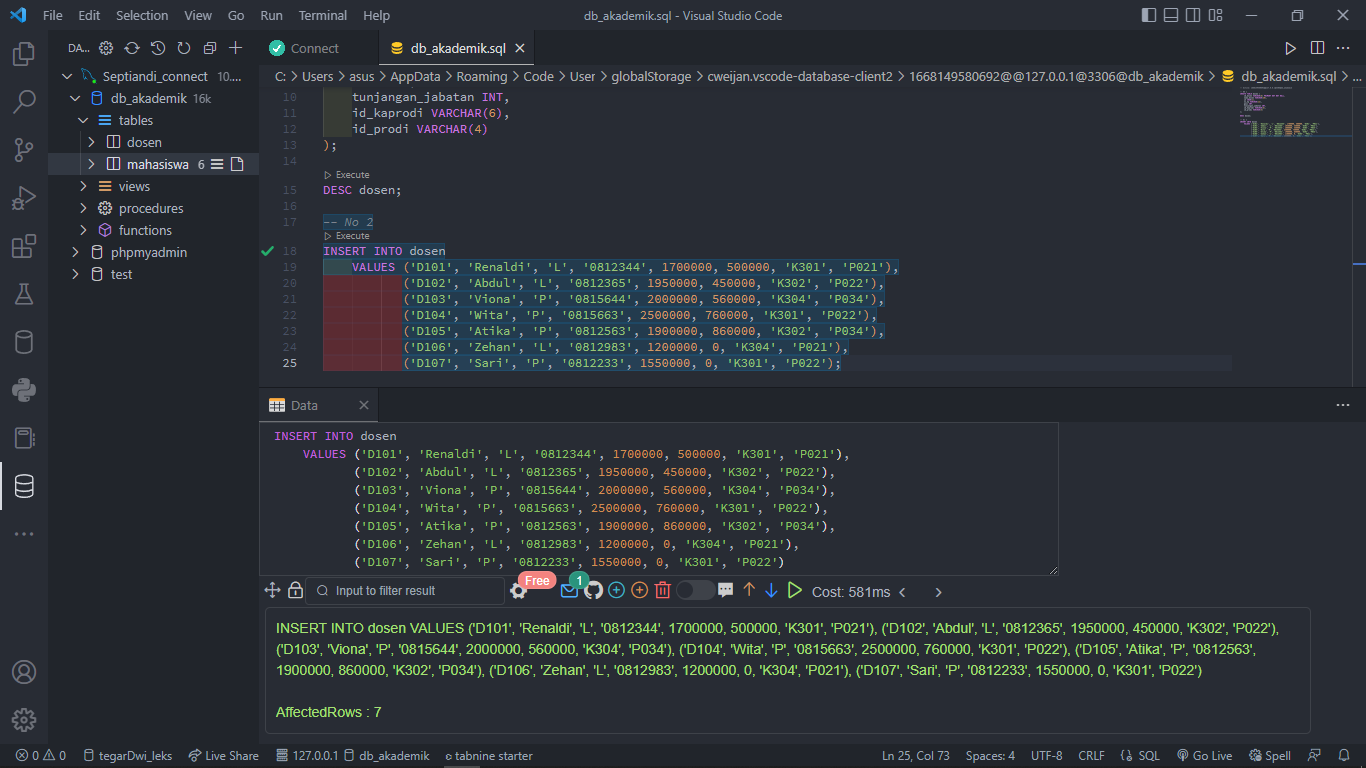




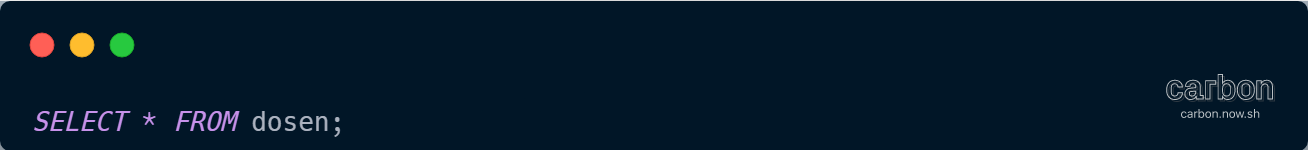
* 1. Isi data tabel dengan data-data berikut!

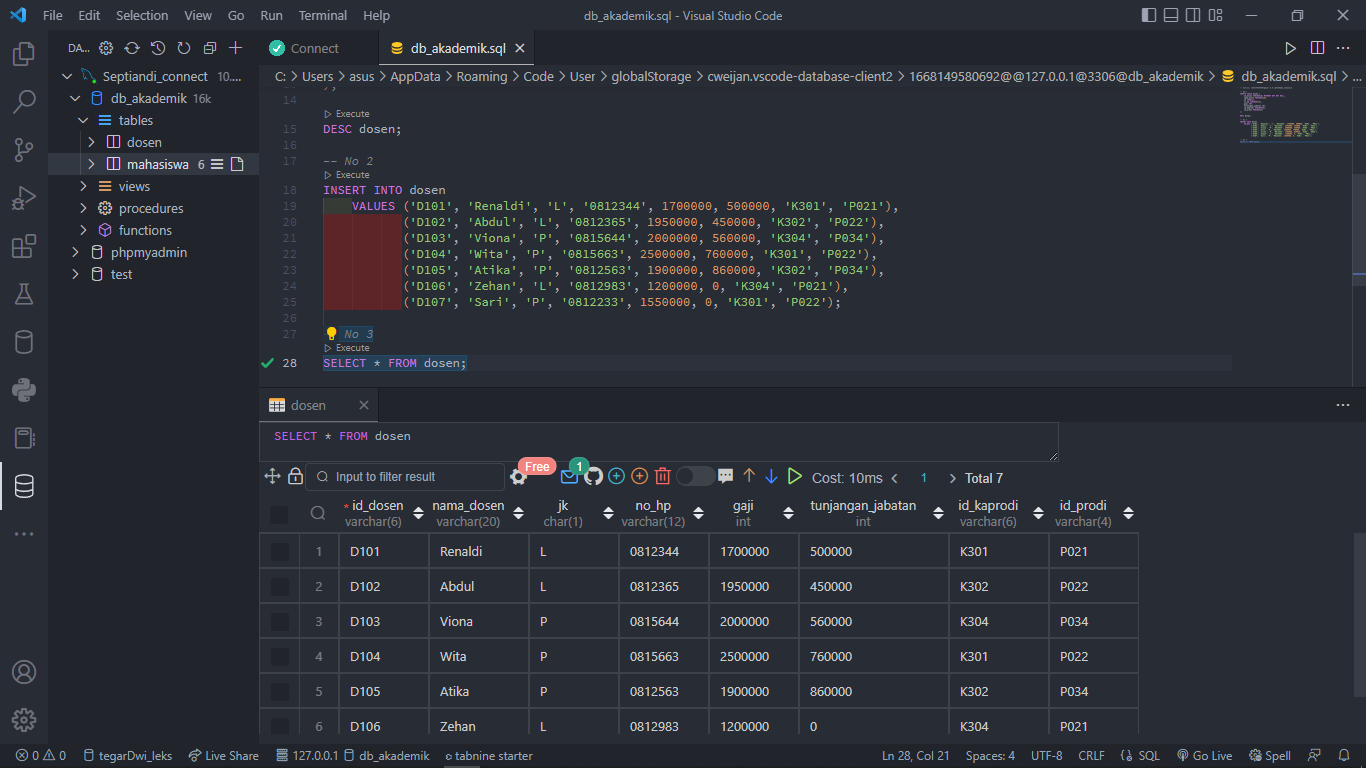
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| id\_dosen | nama\_dosen | jk | no\_hp | gaji | tunjangan\_jabatan | id\_kaprodi | id\_prodi |
| D101 | Renaldi | L | 0812344 | 1700000 | 500000 | K301 | P021 |
| D102 | Abdul | L | 0812365 | 1950000 | 450000 | K302 | P022 |
| D103 | Viona | P | 0815644 | 2000000 | 560000 | K304 | P034 |
| D104 | Wita | P | 0815663 | 2500000 | 760000 | K301 | P022 |
| D105 | Atika | P | 0812563 | 1900000 | 860000 | K302 | P034 |
| D106 | Zehan | L | 0812983 | 1200000 | 0 | K304 | P021 |
| D107 | Sari | P | 0812233 | 1550000 | 0 | K301 | P022 |



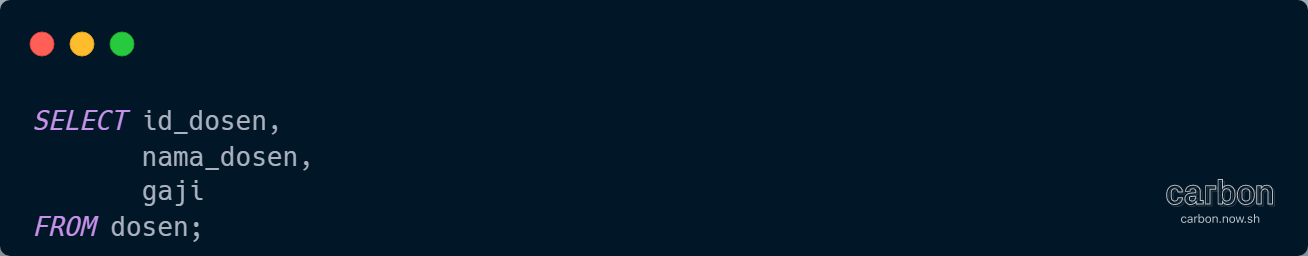


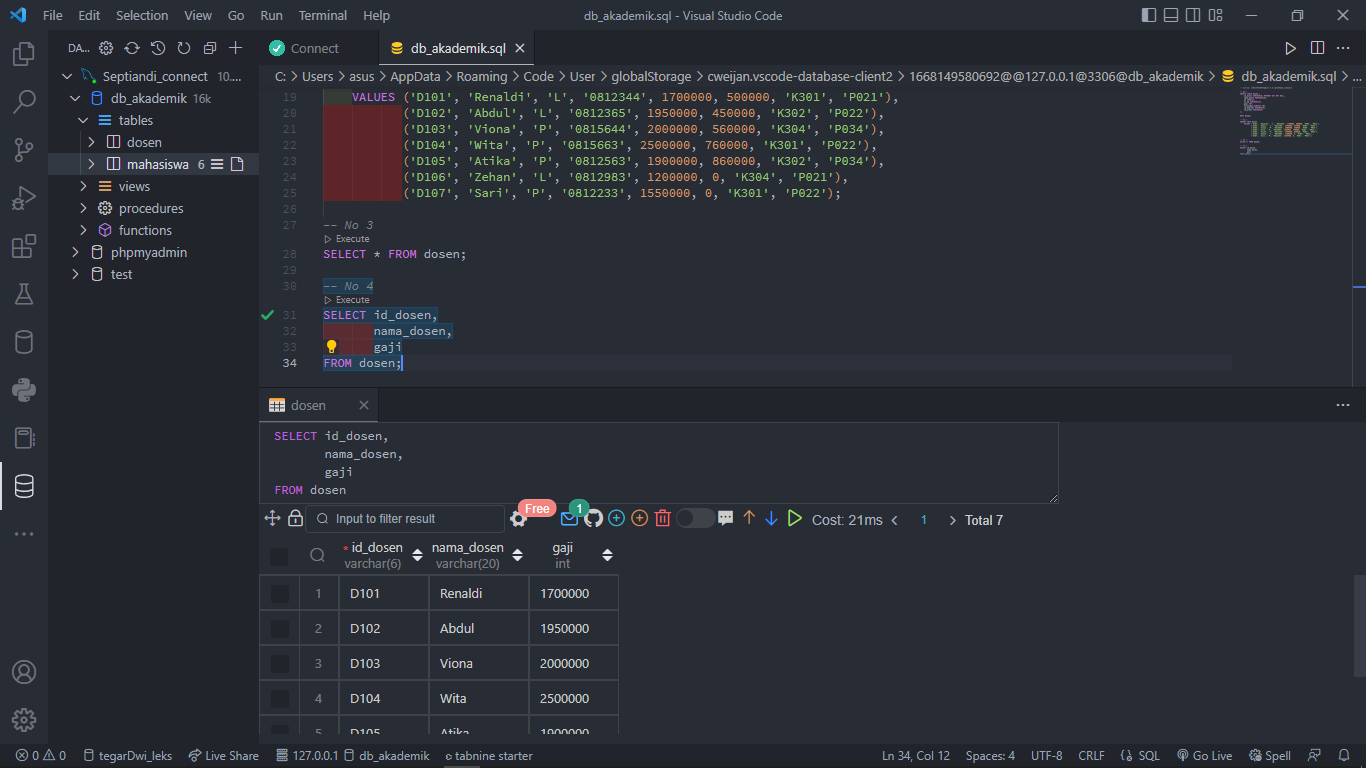
* 1. Tampilkan semua kolom tabel!

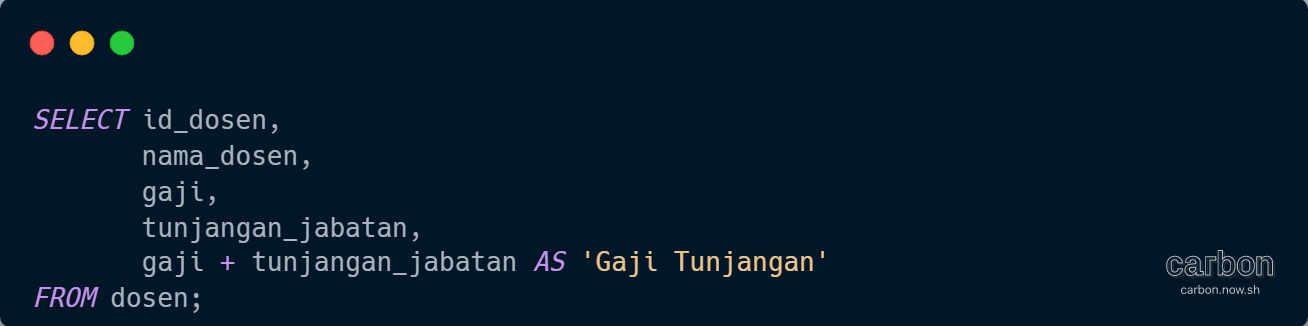


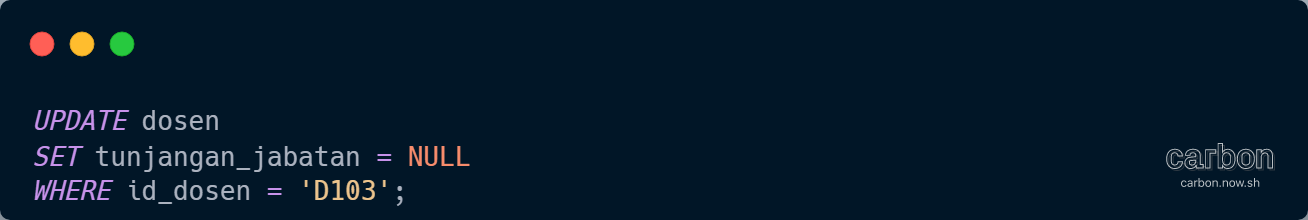
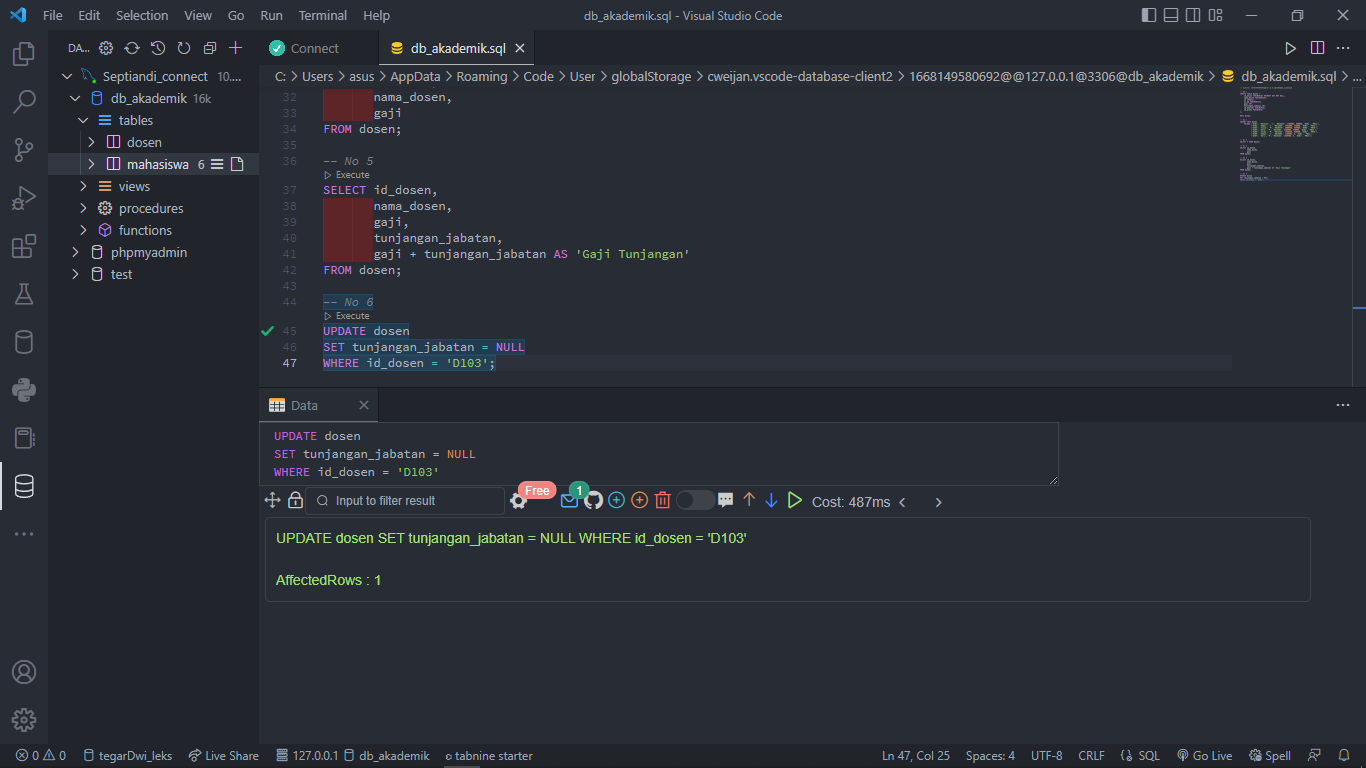


* 1. Tampilkan kolom id\_dosen, nama\_dosen dan gaji saja!

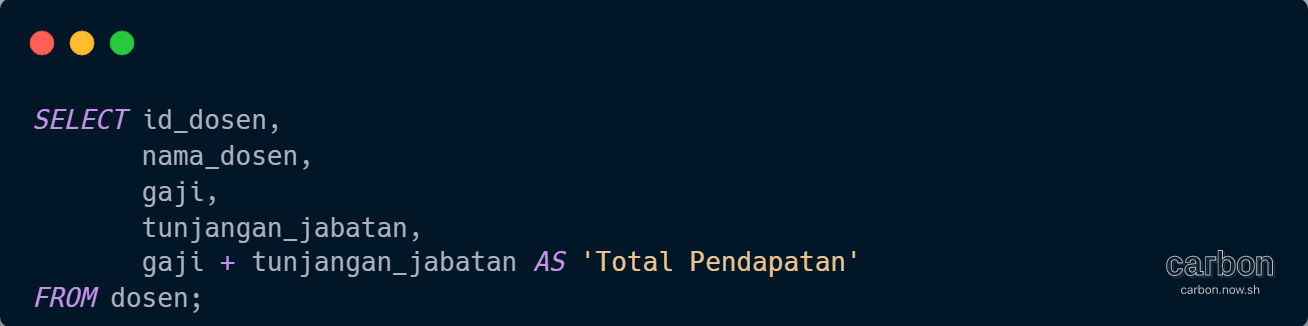


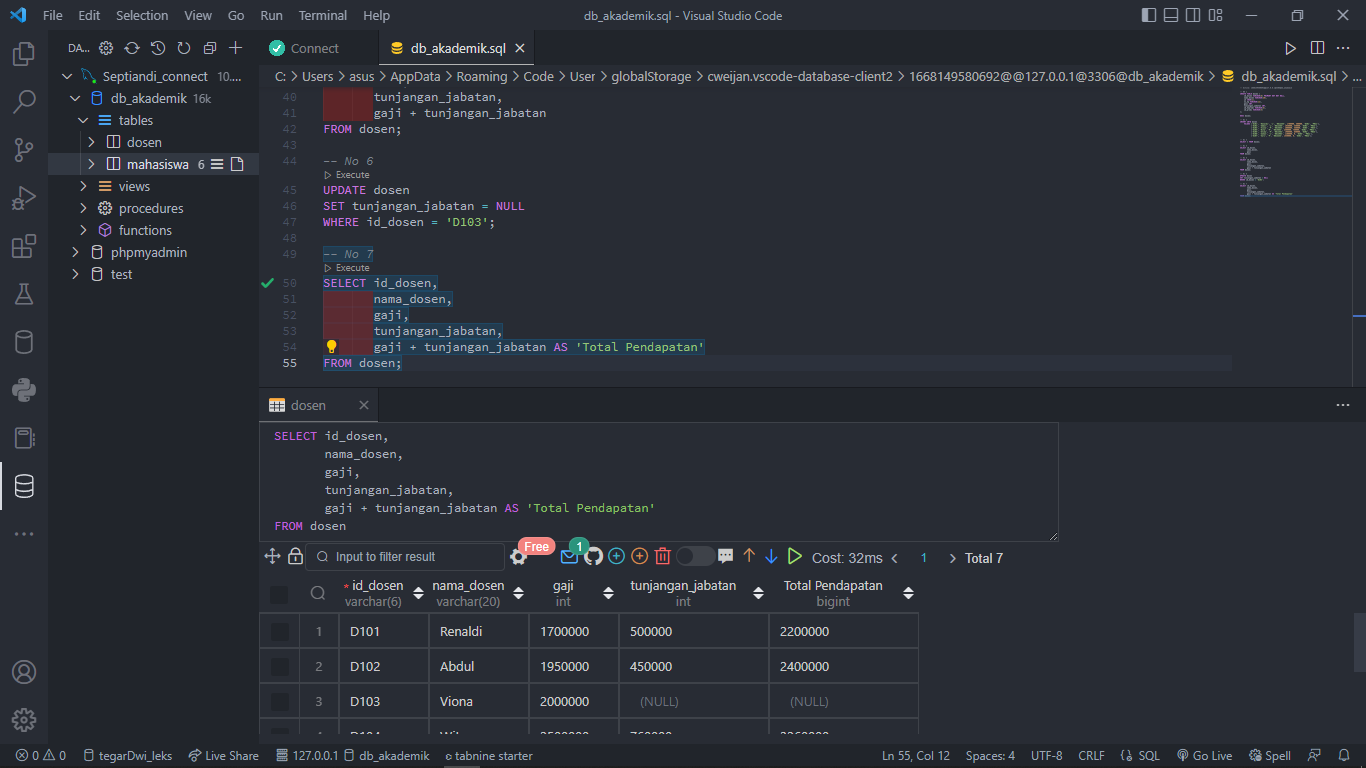


* 1. Tampilkan kolom id\_dosen, nama\_dosen, gaji, tunjangan\_jabatan dan sebuah kolom baru yaitu tunjangan\_jabatan + gaji yang berisi jumlah tunjangan\_jabatan dan gaji !
  2.  Ubah tunjanganJabatan menjadi NULL untuk dosen dengan idDosen = D103. Kemudia lakukan kembali percobaan 5.

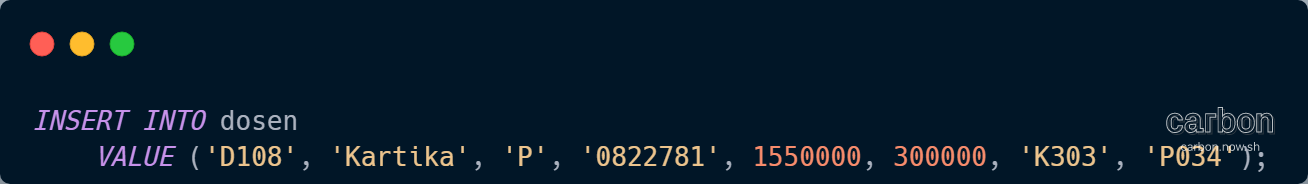
 

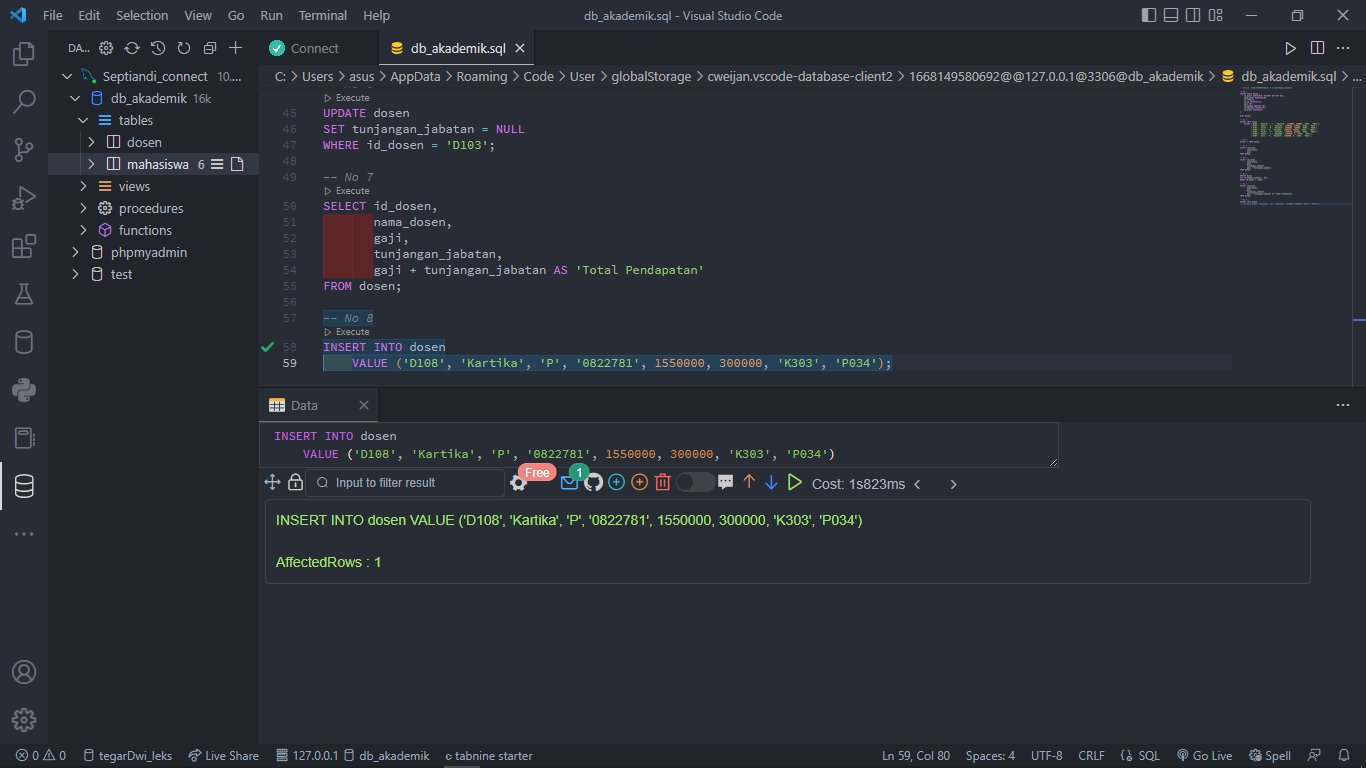
* 1. Seperti percobaan 5, tampilkan kolom idDosen, NamaDosen, gaji, tunjanganJabatan dan sebuah kolom baru (gunakan alias) yaitu total\_pendapatan yang berisi jumlah tunjanganJabatan dan gaji !



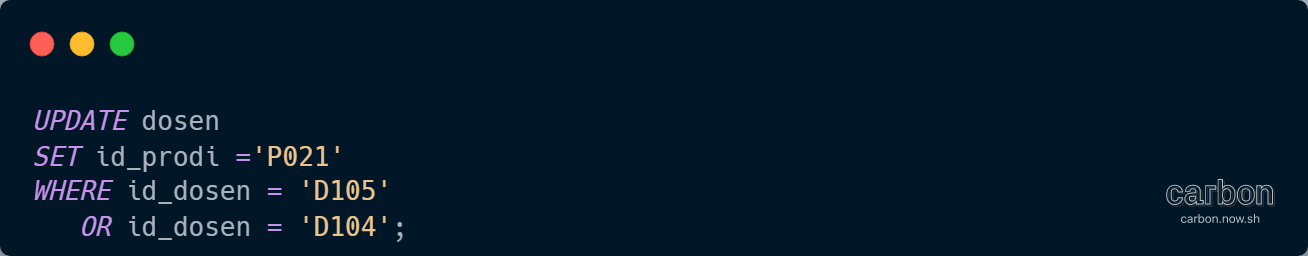


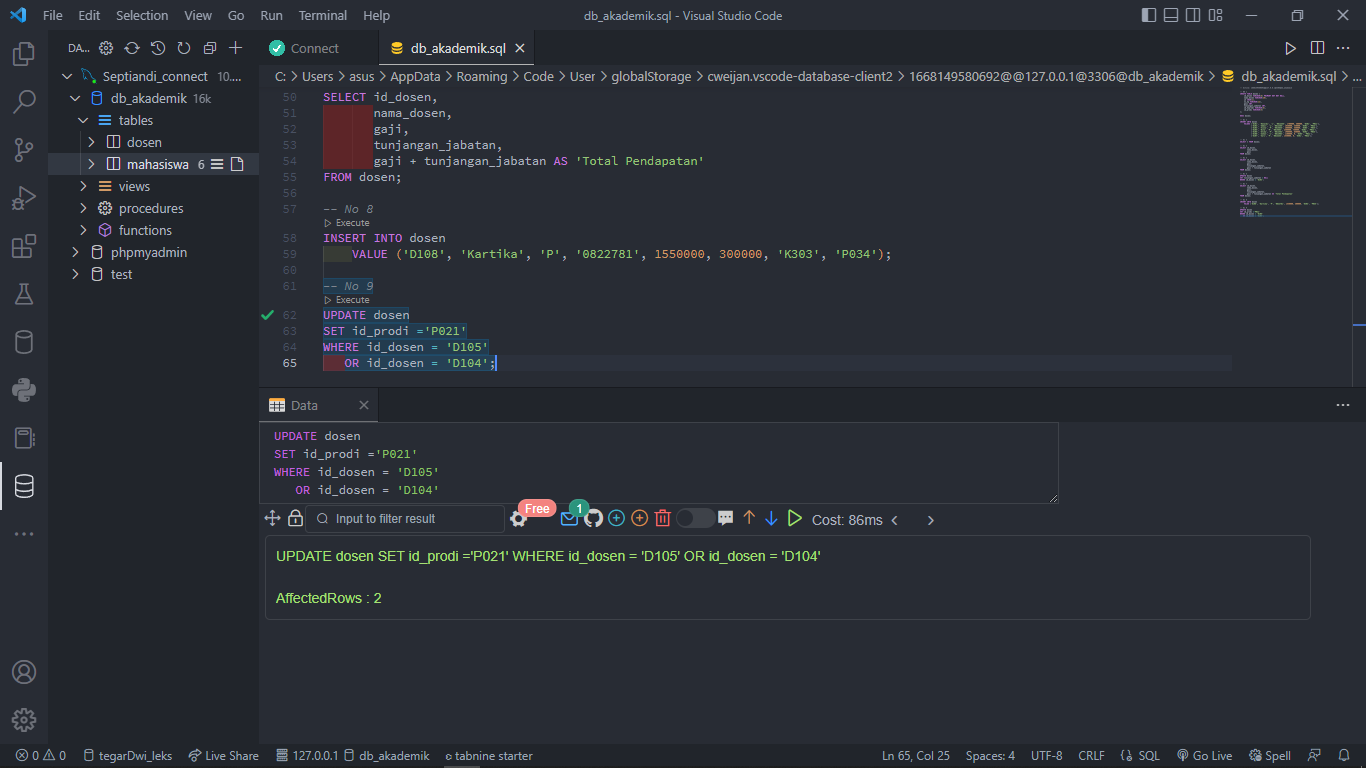
* 1. Tambahkan record baru dengan value: D108, Kartika, P, 0822781, 1550000, 300000, K303, P034 !



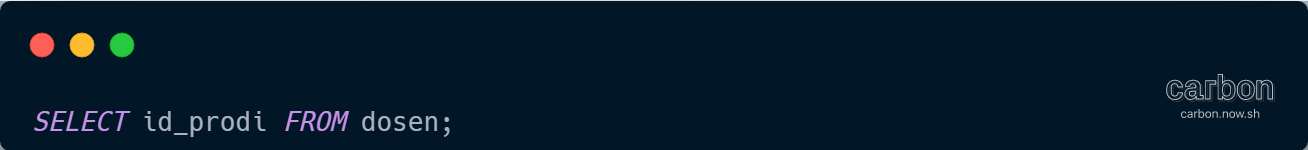


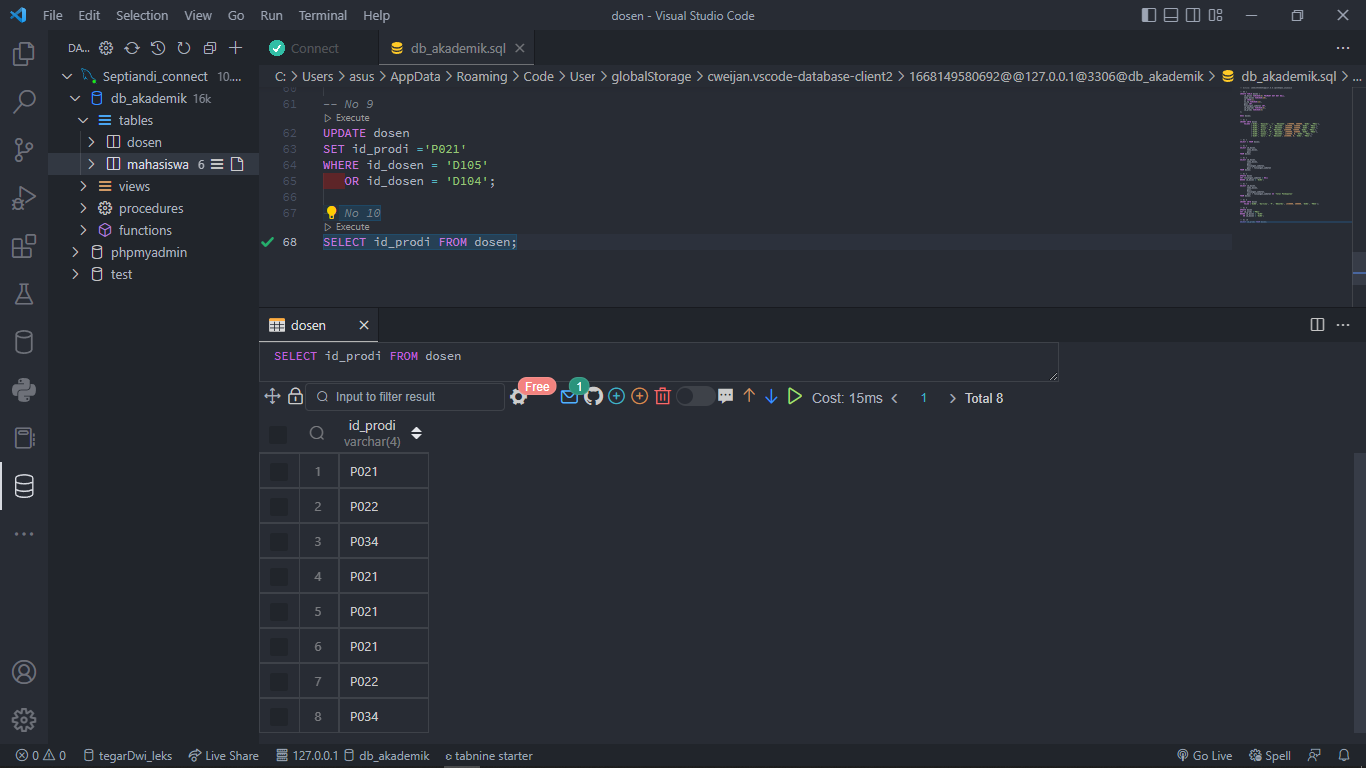
* 1. Untuk dosen yang ber-id D104 dan D105 ubah idProdi menjadi P021!



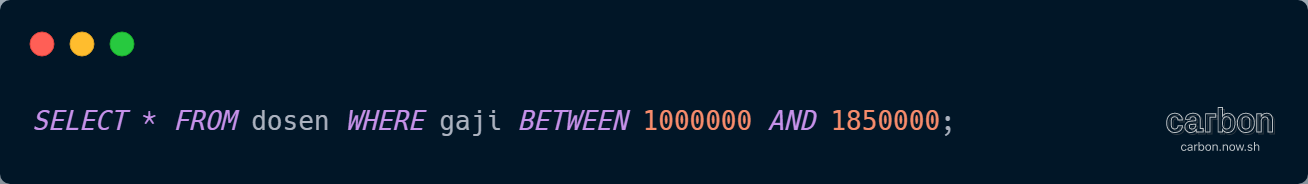
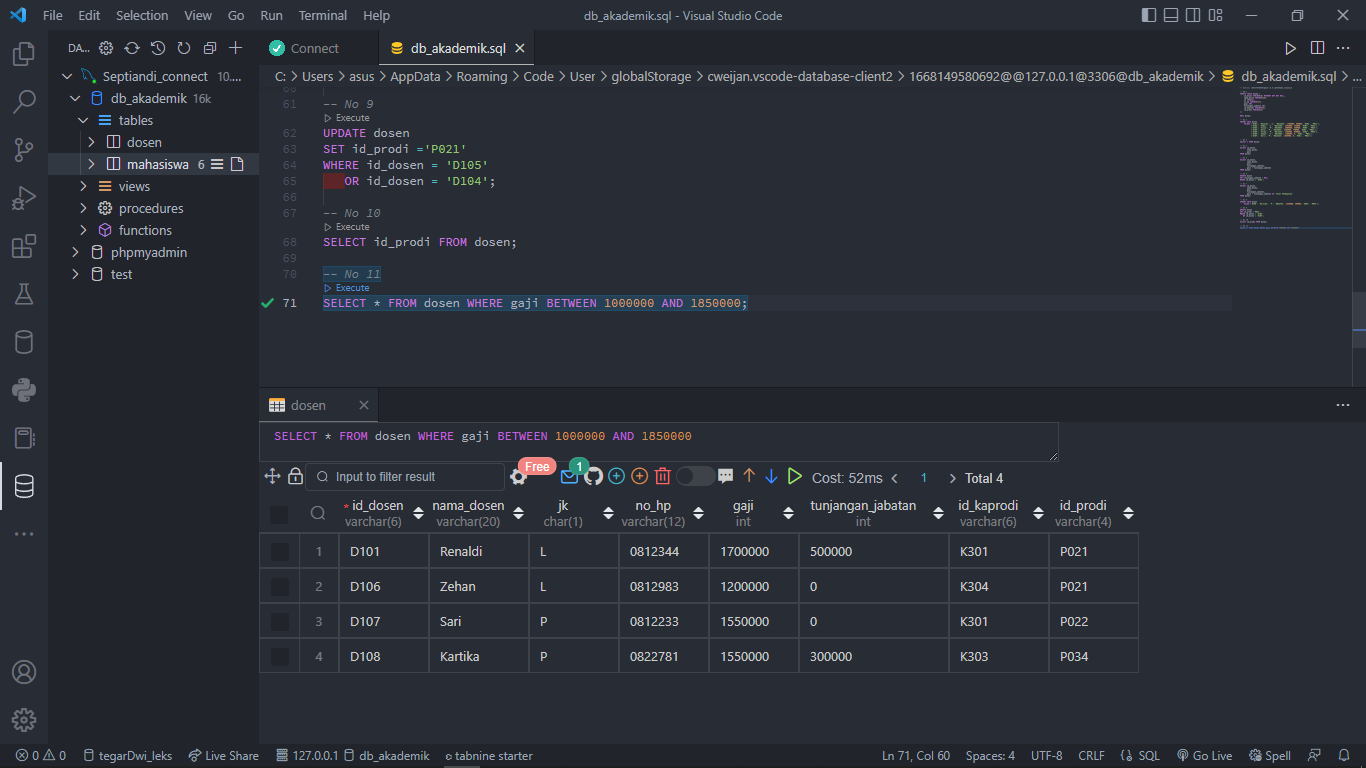


* 1. Sekarang tampilkan kolom idProdi saja !

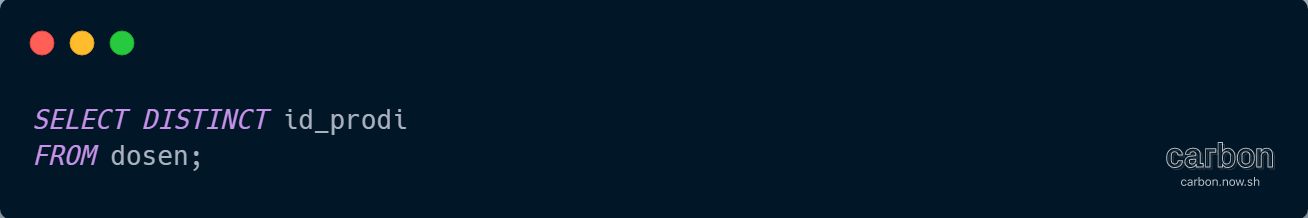


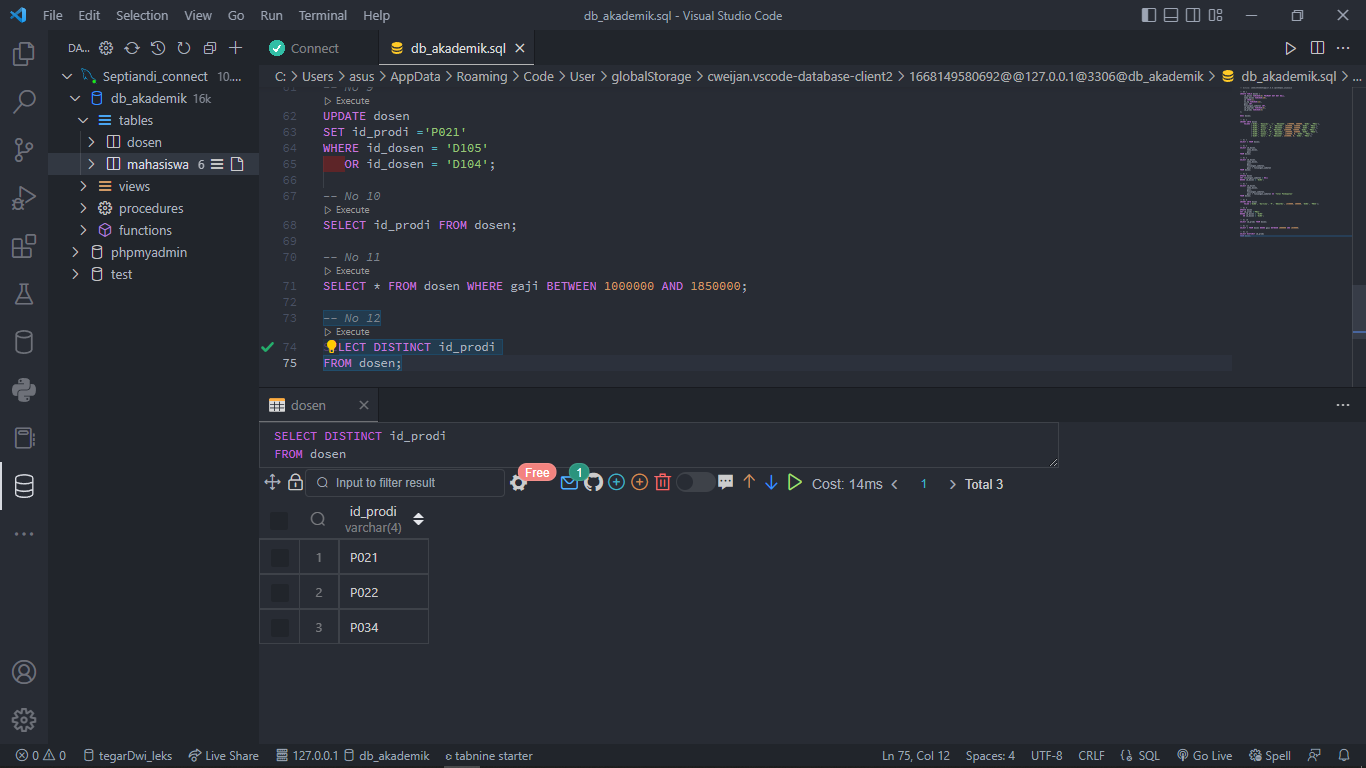


* 1. Tampilkan pegawai yang gajinya antara 1000000 - 1850000 !

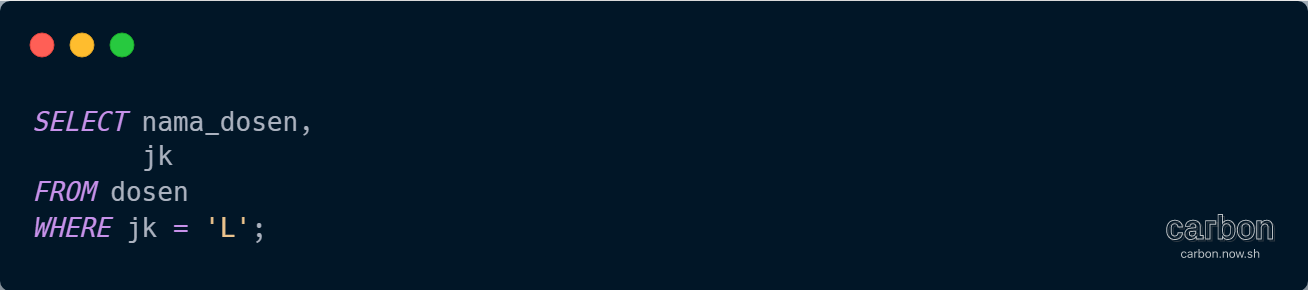
 

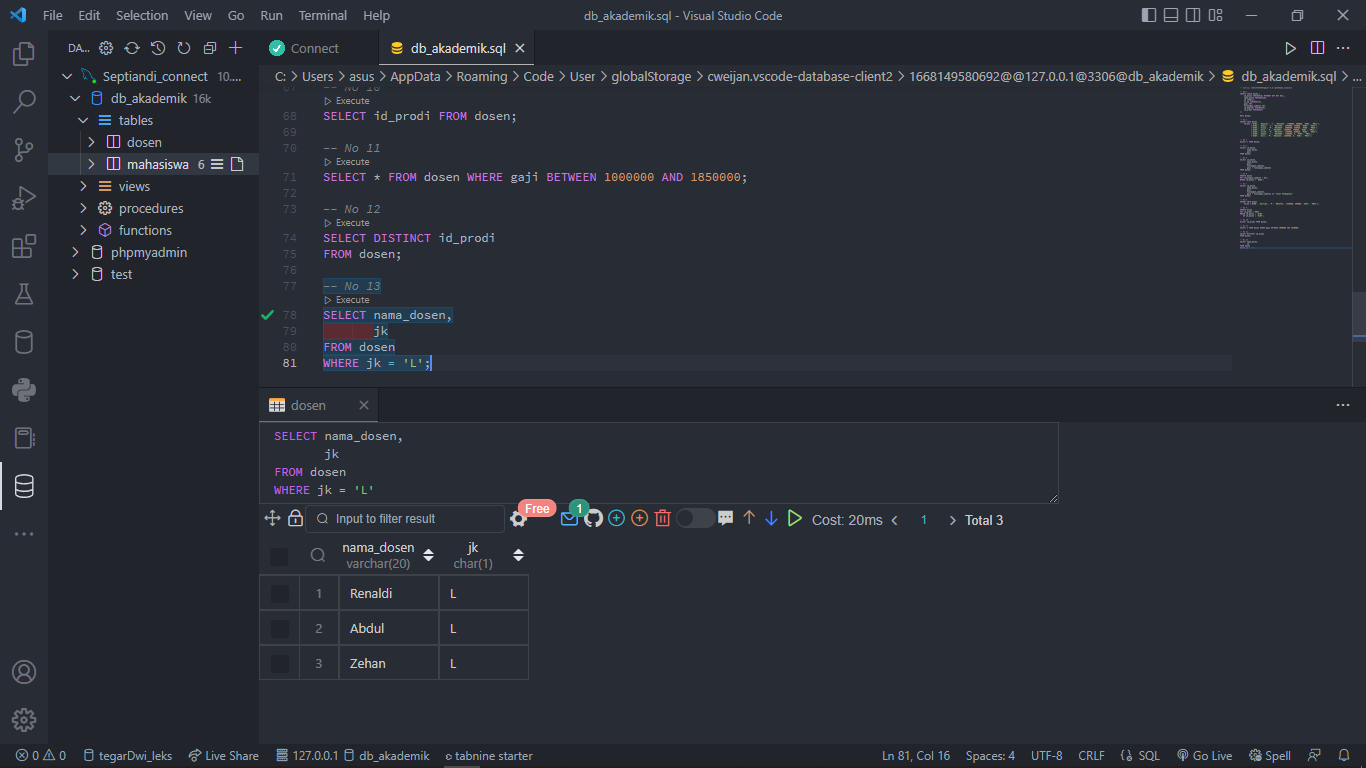
* 1. Tampilkan id\_prodi dengan menghilangkan data id\_prodi yang duplikasi !



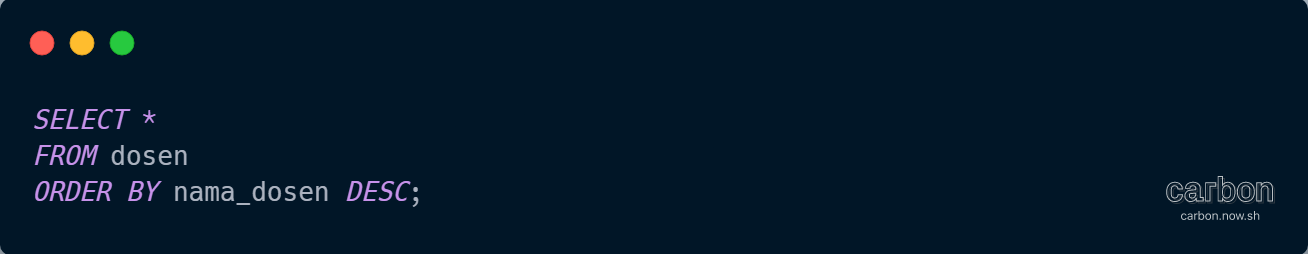


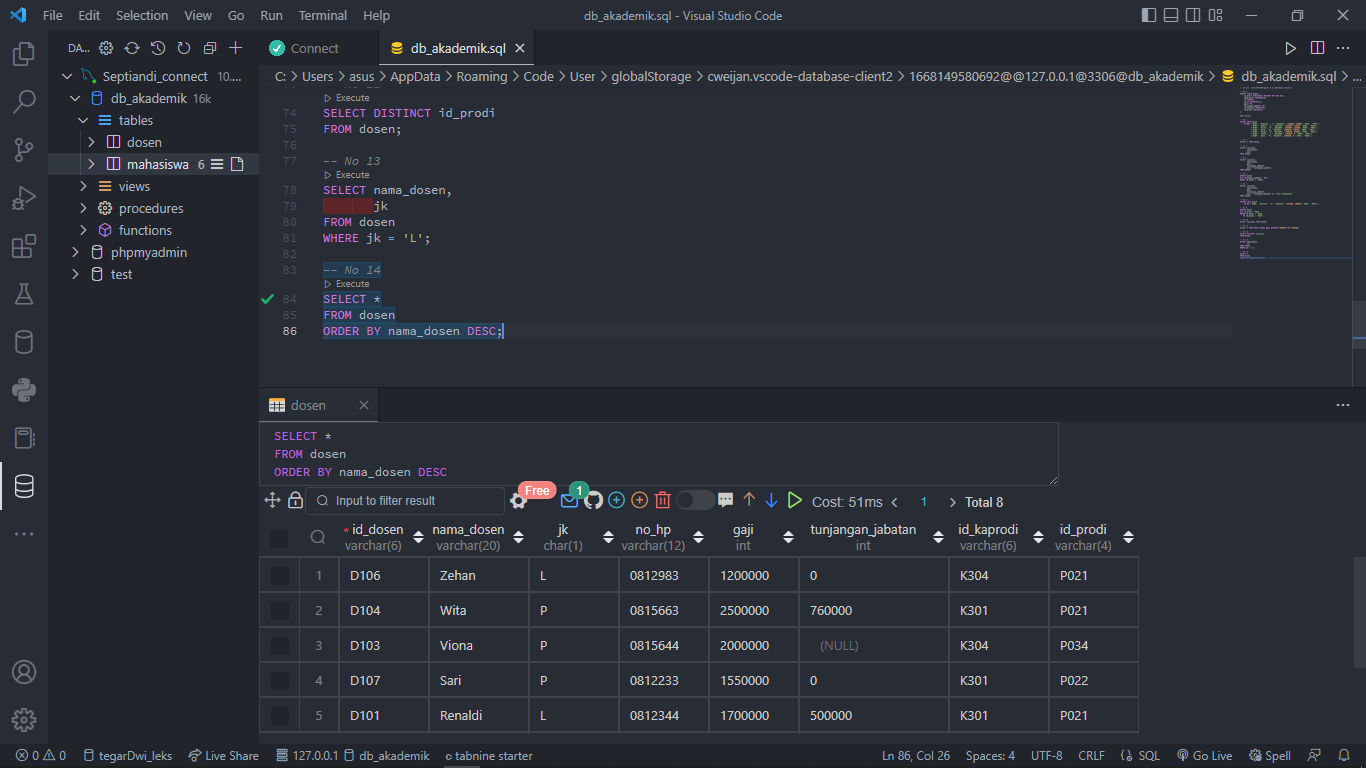
* 1. Tampilkan dosen yang jenis kelamin laki-laki !



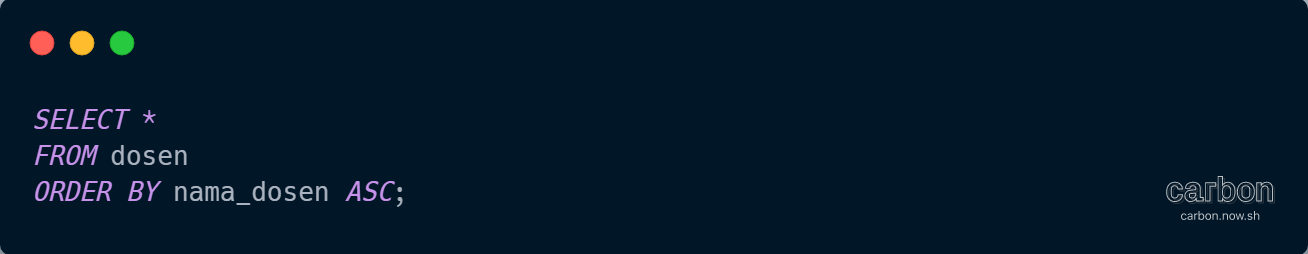


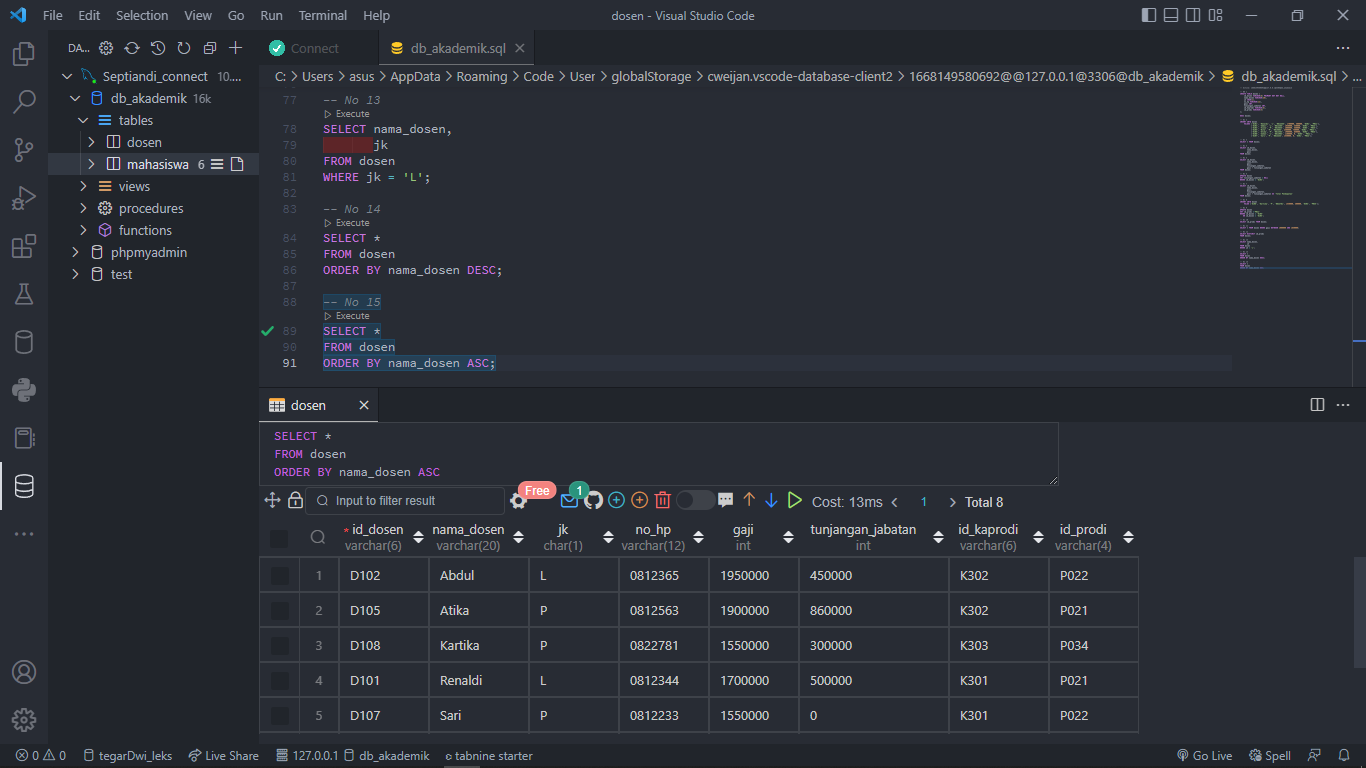
* 1. Tampilkan seluruh dosen menurut abjad dari Z-A !





* 1. Tampilkan seluruh dosen menurut abjad dari A-Z !





1. **Kesimpulan**

Dalam basis data, select statement digunakan sebagai operasi untuk mengambil sejumlah baris data yang memenuhi predikat yang diberikan. Select statement terdiri dari beberapa bagian untuk mengatur dan mengambil data dari suatu tabel, serta dapat menyeleksi data berdasarkan kondisi tertentu.