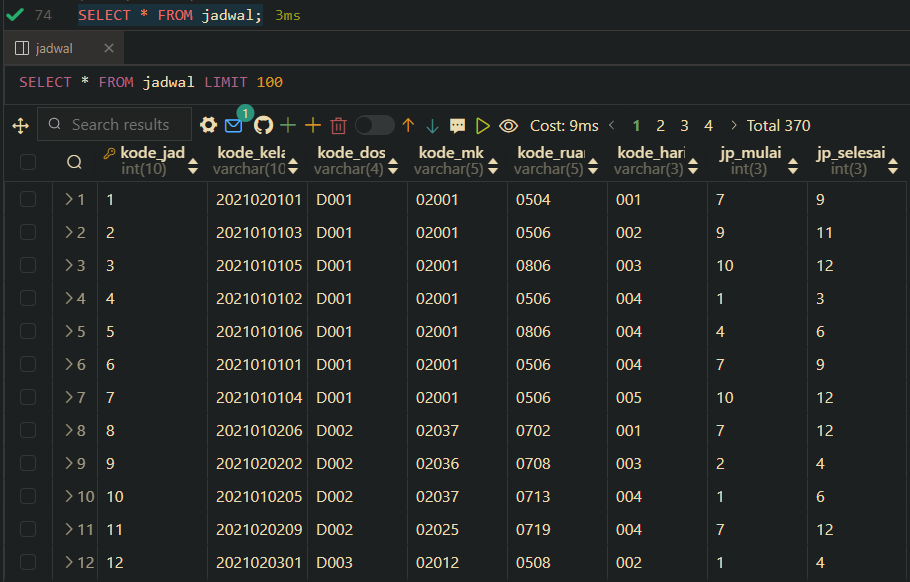
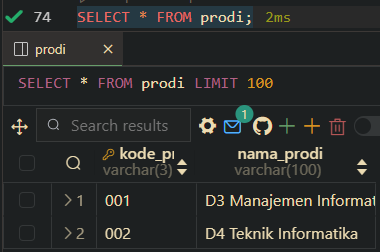
**BASIS DATA**

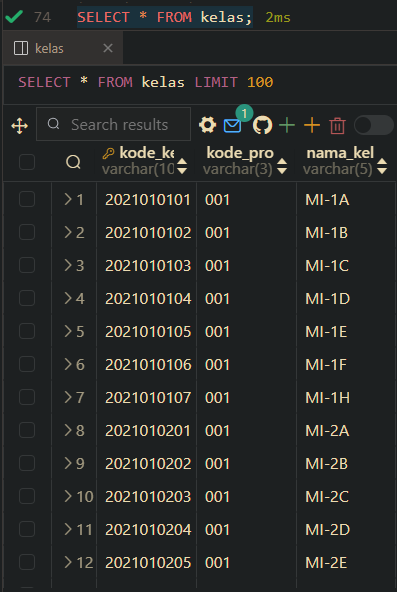
**Dasar MySQL Data Retrieval Language**

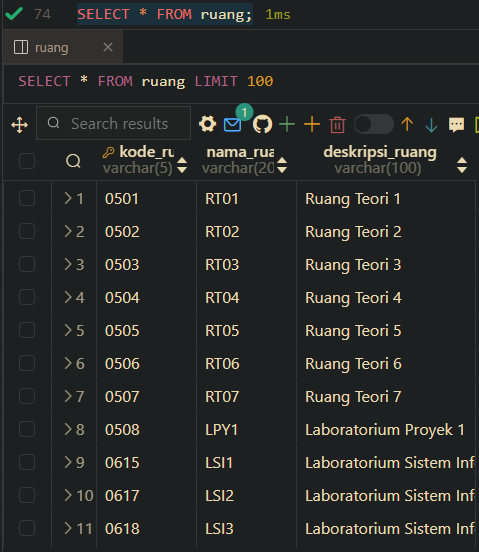
**Lavina 2341760062**

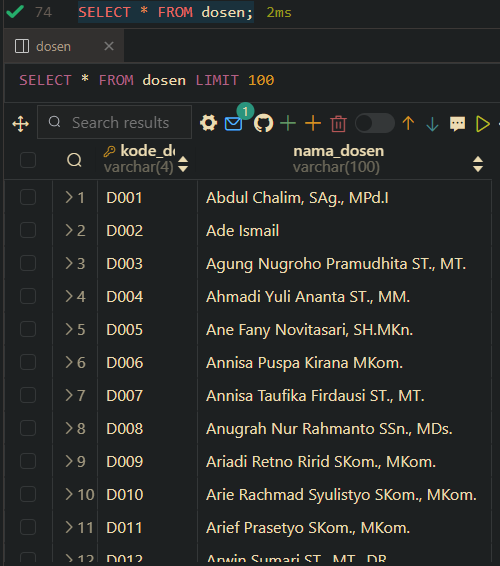
**Praktikum 1**

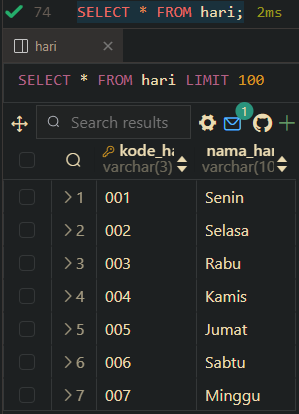
****

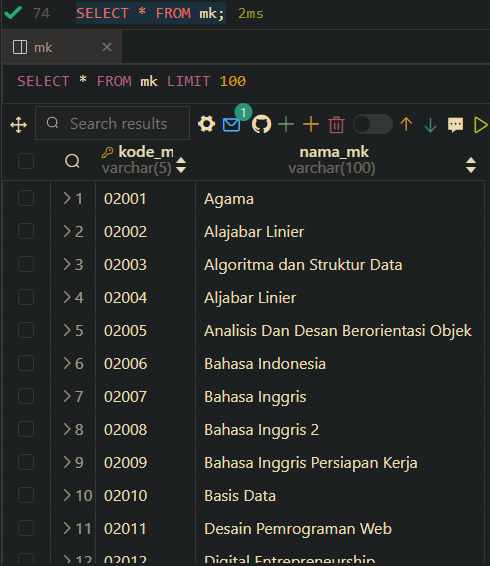
****

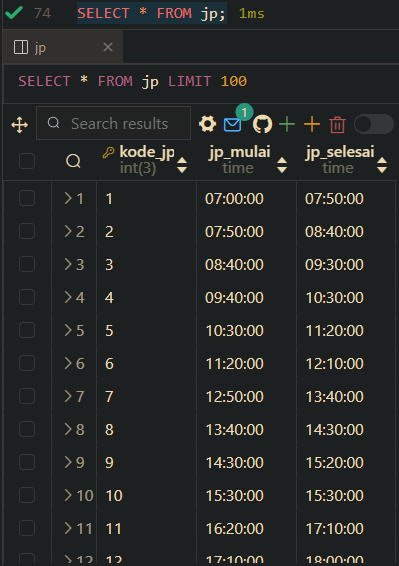
****

****

****

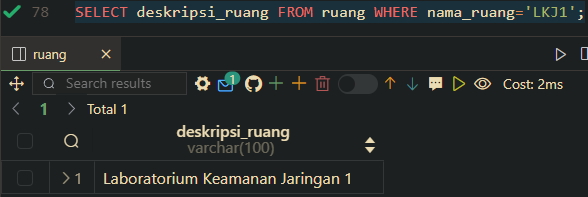
****

****

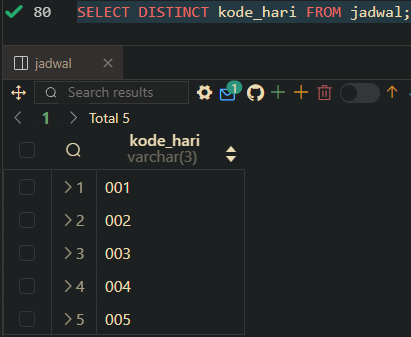
****

**Praktikum 2**

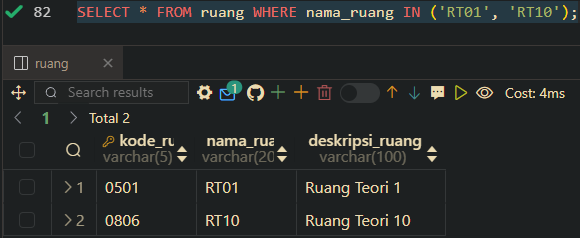
1. Untuk menampilkan baris dengan suatu kondisi tertentu, digunakan statement **SELECT** dengan **WHERE**. Eksekusi SQL berikut untuk menampilkan (record) pada tabel **Ruang** yang memiliki kolom nama\_ruang = LKJ1.

****

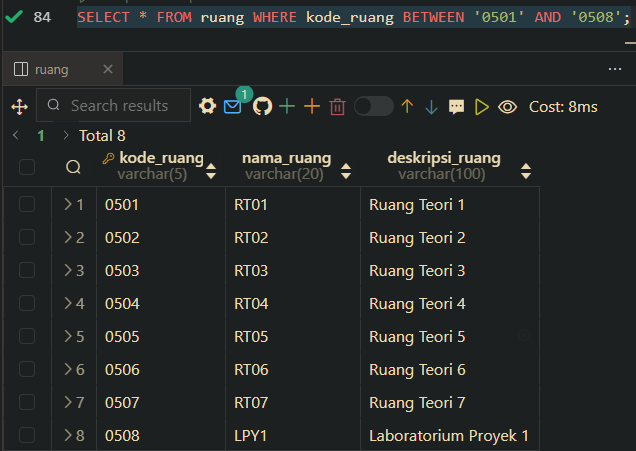
1. Untuk menampilkan data dengan eliminasi data yang sama (duplicate), digunakan statement **SELECT** dengan **DISTINCT**. Eksekusi SQL berikut untuk menampilkan kolom kode\_hari dari tabel jadwal yang bernilai tidak sama.



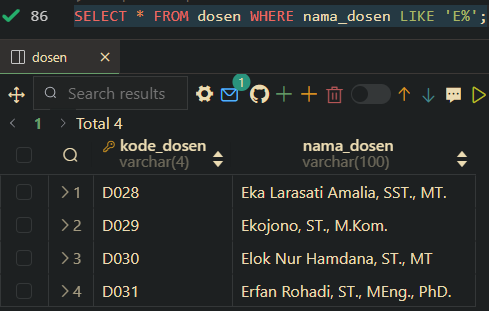
1. Untuk menampilkan data yang spesifik digunakan statement **SELECT** dengan **IN**. Eksekusi SQL berikut untuk menampilkan kolom kode\_ruang, nama\_ruang dan deskripsi\_ruang dari tabel **ruang** yang memiliki nama\_ruang RT01 atau RT10.



1. Untuk menampilkan data pada jarak (range) tertentu digunakan statement **SELECT** dengan **BETWEEN**. Eksekusi SQL berikut untuk menampilkan kolom kode\_ruang dan nama\_ruang dan deskripsi\_ruang dari tabel **ruang** yang memiliki kode\_ruang antara 0501 dan 0508.



1. Untuk menampilkan data yang memiliki kemiripan dengan keyword yang diinginkan digunakan **SELECT** dengan **LIKE**. Eksekusi SQL berikut untuk menampilkan kolom kode\_dosen, dan nama\_dosen, pada tabel **dosen** yang memiliki nama dengan huruf awal ‘E’.



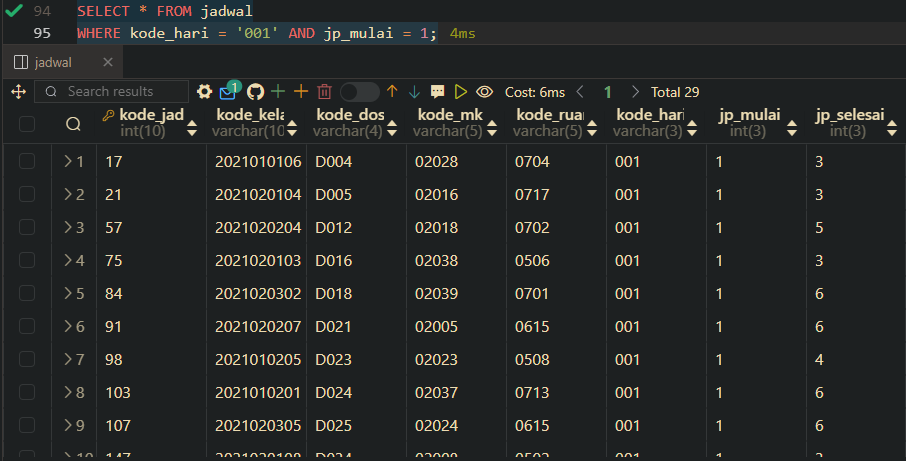
1. Untuk menampilkan susunan data dalam bentuk grup, digunakan **SELECT** dengan **GROUP BY**. Eksekusi SQL berikut untuk menampilkan kolom kode\_dosen, kode\_mk, kode\_ruang dan kode\_hari pada tabel **jadwal** yang dikelompokkan berdasarkan kode\_hari.



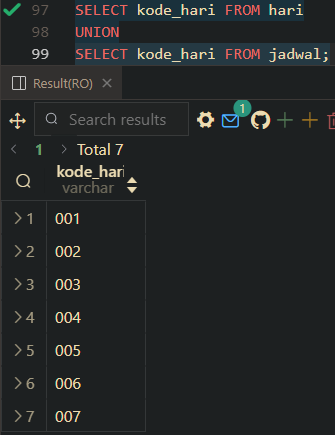
1. Untuk menampilkan baris secara spesifik dan terurut maju atau mundur, digunakan **SELECT** dengan **ORDER** **BY**. Eksekusi SQL berikut untuk menampilkan kolom kode\_jp dan jp\_mulai pada tabel **jp** dari jam yang paling mulai hingga berakhir.



1. Untuk menampilkan data dengan kondisi dan atau atau tidak, digunakan **SELECT** dengan **AND**, **OR** and **NOT**. Eksekusi SQL berikut untuk menampilkan semua kolom pada tabel **jadwal** dengan kode\_hari = ‘001’ dan jp\_mulai = 1.



1. Untuk menampilkan data dari kolom yang terlibat dalam dua tabel dapat digunakan **SELECT** dengan **UNION**. **UNION** secara otomatis akan menghilangkan duplikasi. Eksekusi SQL berikut untuk menampilkan kolom kode\_hari yang ada di tabel hari atau jadwal.

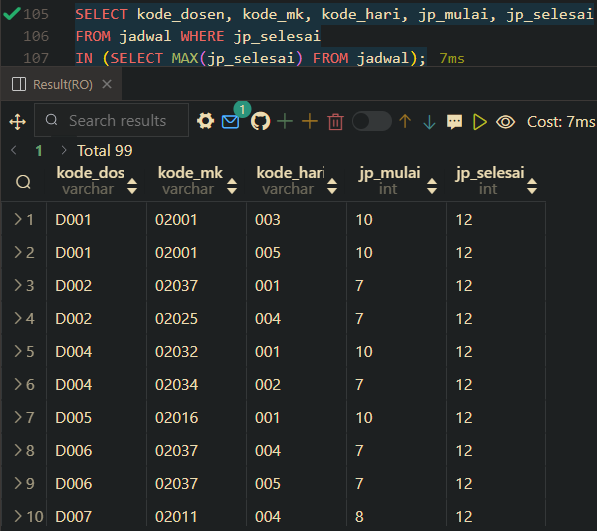


1. Untuk menampilkan data dari kolom yang terlibat dalam dua tabel dapat digunakan **SELECT** dengan **UNION** **ALL**. **UNION** **ALL** akan menampilkan duplikasi data. Eksekusi SQL berikut untuk menampilkan kolom nik yang yang ada di tabel **penugasan** atau **departemen**.

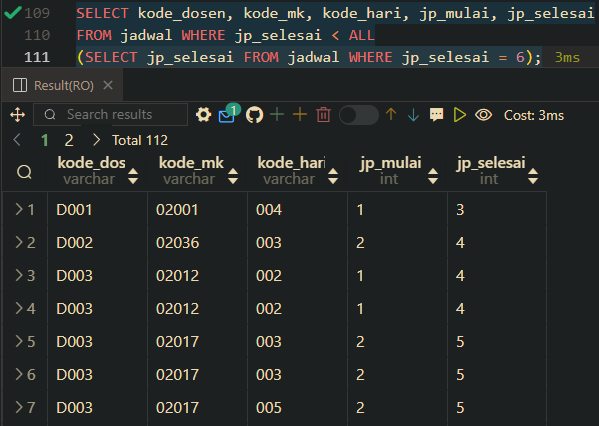


**Praktikum 3**

1. Untuk menampilkan data (test keanggotaan sub-query) yang berasal dari pemilihan tampilan data lain digunakan tambahan statement IN. Eksekusi SQL berikut untuk menampilkan kolom kode\_dosen, kode\_mk, kode\_hari, jp\_mulai dan jp\_selesai pada tabel jadwal dimana jp\_selesai adalah jp\_selesai yang paling lama pada tabel jadwal.

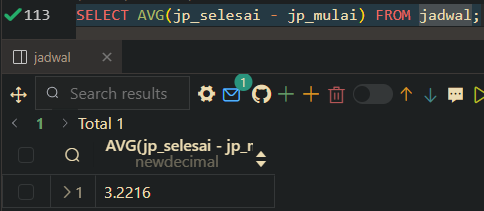


1. Untuk menampilkan data (perbandingan himpunan sub-query) yang berasal dari pemilihan tampilan data lain digunakan tambahan operator. Eksekusi SQL berikut untuk menampilkan kode\_dosen, kode\_mk, kode\_hari, jp\_mulai dan jp\_selesai pada tabel jadwal berdasarkan semua record kolom jp\_selesai harus bernilai kurang dari jp\_selesai yang dimiliki oleh jp\_selesai = 6 dari tabel jadwal.

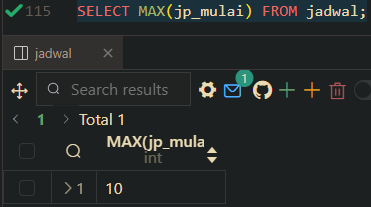


**Praktikum 4**

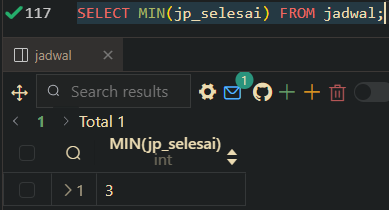
1. Untuk menghitung rata-rata, digunakan tambahan statement AVG. Eksekusi SQL berikut untuk menampilkan rata-rata durasi perkuliahan dalam satu jadwal perkuliahan sesuai kolom jp\_mulai, jp\_selesai dari tabel jadwal.



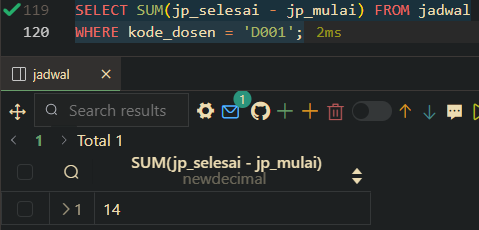
1. Untuk menghitung nilai tertinggi, digunakan tambahan statement MAX. Eksekusi SQL berikut untuk menampilkan dimulainya jam perkuliahan yang paling siang sesuai kolom jp\_mulai dari tabel jadwal.

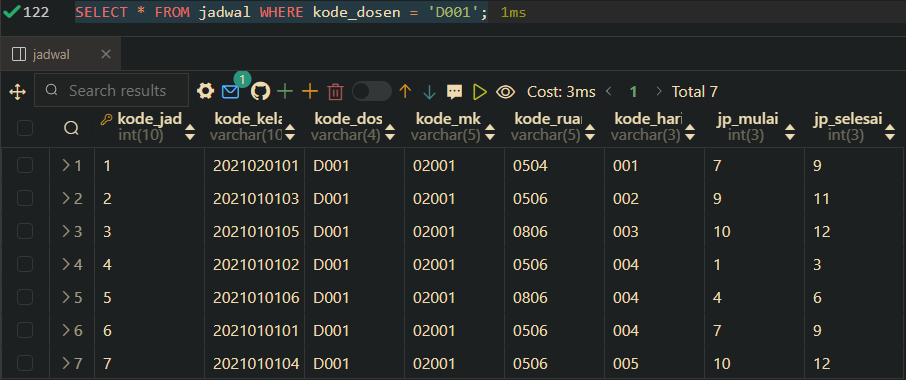


1. Untuk menghitung nilai terendah, digunakan tambahan statement MIN. Eksekusi SQL berikut untuk menampilkan berakhirnya jam perkuliahan yang paling cepat sesuai kolom jp\_selesai dari tabel jadwal.

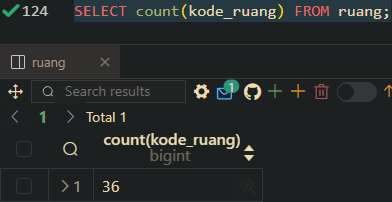


1. Untuk menghitung total jam ajar digunakan tambahan statement SUM. Eksekusi SQL berikut untuk menampilkan total jam ajar dosen dengan kode ‘D001’ sesuai jp\_mulai, jp\_selesai, dan kode\_dosen dari tabel jadwal.



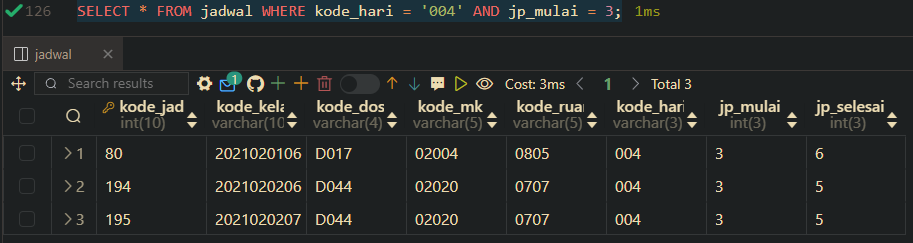


1. Untuk menunjukkan jumlah baris pada kolom yang diinginkan digunakan COUNT(). Eksekusi SQL berikut untuk menampilkan jumlah baris pada kolom kode\_ruang dari tabel ruang.

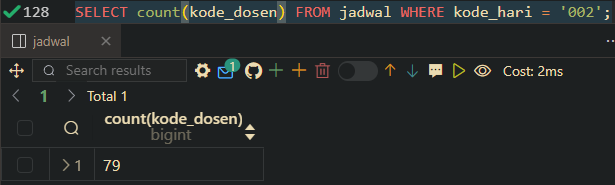


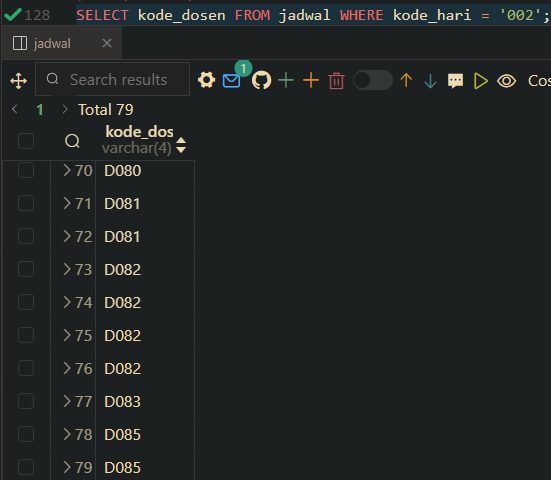
**Tugas**

1. Himpunlah screenshot sintaksis SQL berikut hasil SELECT yang Anda dapat di komputer Anda pada bagian Praktikum dalam sebuah laporan dengan format PDF!
2. Bagaimana sintaksis untuk menampilkan semua kolom jadwal pada hari kamis yang dimulai pada pukul 08:40 ? Tunjukkan outputnya!

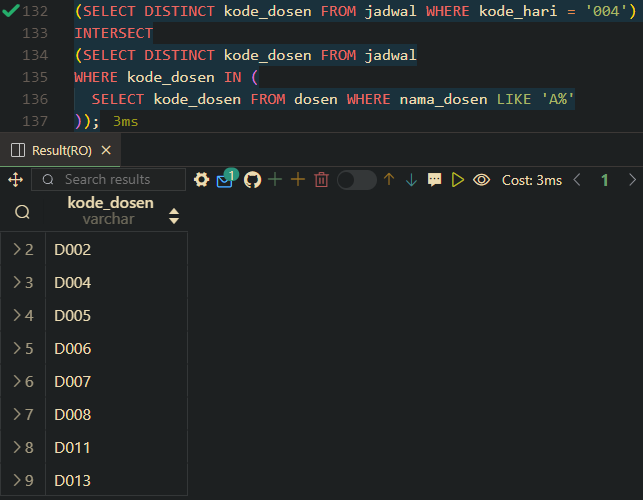


1. Berapakah jumlah dosen yang mengajar pada hari selasa! Tunjukkan sintaksis dan outputnya! Kemudian tampilkan list semua kode dosen yang sesuai dengan kondisi tersebut! Tunjukkan sintaksis dan outputnya!





1. Tampilkan kode dosen yang mengajar pada hari kamis dari tabel jadwal yang nama depanya di awali dengan huruf A. gunakan operasi intersect! Tunjukkan sintaksis dan output dari pencarian tersebut.



1. Tampilkan kode\_ruang tertentu pada hari tertentu dengan diurutkan berdasarkan dengan kode jam yang ter awal.

