

LAPORAN PRAKTIKUM
MATA KULIAH PRAKTIKUM BASIS DATA

Dosen Pengampu : Farid Angga Pribadi, S.Kom., M.Kom.

PERTEMUAN 12 : DATA RETRIVAL LANGUAGE (DRL)



Nama : Yonanda Mayla Rusdiaty

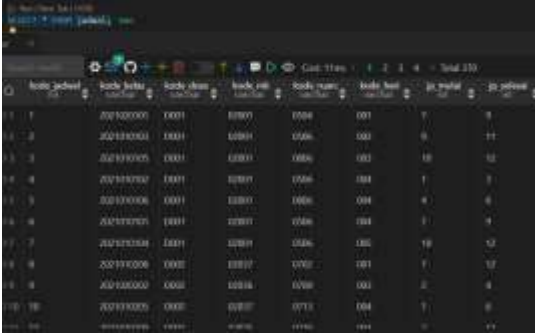
NIM : 2341760184

Prodi : D-IV Sistem Informasi Bisnis

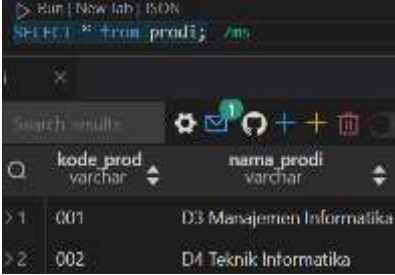
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2024

PRAKTIUM 1 : Membuat Database untuk Percobaan

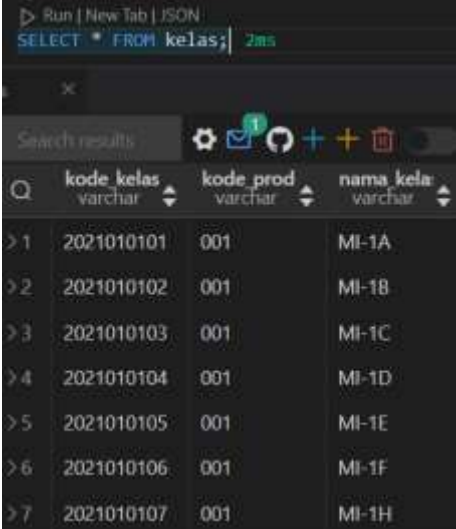
1. Menghapus tabel mahasiswa pada database jadwal perkuliahan, sehingga menghasilkan keseluruhan tabel seperti berikut ini :



	kode_jadwal	kode_kelas	kode_dosen	kode_prodi	kode_hall	jam_masuk	jam_keluar
> 1	1	202101001	0001	0001	0001	7	9
> 2	2	202101001	0001	0001	0001	9	11
> 3	3	202101001	0001	0001	0001	10	12
> 4	4	202101001	0001	0001	0001	7	9
> 5	5	202101001	0001	0001	0001	9	11
> 6	6	202101001	0001	0001	0001	10	12
> 7	7	202101001	0001	0001	0001	7	9
> 8	8	202101001	0001	0001	0001	9	11
> 9	9	202101001	0001	0001	0001	10	12
> 10	10	202101001	0001	0001	0001	7	9



	kode_prodi	nama_prodi
> 1	001	D3 Manajemen Informatika
> 2	002	D4 Teknik Informatika



	kode_kelas	kode_prodi	nama_kelas
> 1	2021010101	001	MI-1A
> 2	2021010102	001	MI-1B
> 3	2021010103	001	MI-1C
> 4	2021010104	001	MI-1D
> 5	2021010105	001	MI-1E
> 6	2021010106	001	MI-1F
> 7	2021010107	001	MI-1H

Run | New Tab | JSON
SELECT * FROM ruang; 3ms

Search results

	kode_ruang varchar	nama_ruang varchar	deskripsi_ruang varchar
> 1	0501	RT01	Ruang Teori 1
> 2	0502	RT02	Ruang Teori 2
> 3	0503	RT03	Ruang Teori 3
> 4	0504	RT04	Ruang Teori 4
> 5	0505	RT05	Ruang Teori 5
> 6	0506	RT06	Ruang Teori 6
> 7	0507	RT07	Ruang Teori 7
> 8	0508	LPY1	Laboratorium Proyek 1
> 9	0615	LSI1	Laboratorium Sistem Inform
> 10	0617	LSI2	Laboratorium Sistem Inform

Run | New Tab | JSON
SELECT * FROM hari; 8ms

Search results

	kode_hari varchar	nama_hari varchar
> 1	001	Senin
> 2	002	Selasa
> 3	003	Rabu
> 4	004	Kamis
> 5	005	Jumat
> 6	006	Sabtu
> 7	007	Minggu

Run | New Tab | JSON
SELECT * FROM dosen; 2ms

Search results

	kode_dose varchar	nama_dosen varchar
> 1	D001	Abdul Chalim, SAg., MPd.I
> 2	D002	Ade Ismail
> 3	D003	Agung Nugroho Pramudhit
> 4	D004	Ahmadi Yuli Ananta ST., MA
> 5	D005	Ane Fany Novitasari, SH.MK
> 6	D006	Annisa Puspa Kirana MKom
> 7	D007	Annisa Taufika Firdausi ST., I
> 8	D008	Anugrah Nur Rahmanto SSR
> 9	D009	Ariadi Retno Ririd SKom., M
> 10	D010	Arie Rachmad Syulistyo SKo
> 11	D011	Arief Prasetyo SKom., MKom

Run | New Tab | JSON
SELECT * FROM mk; 4ms

Search results

	kode_mk varchar	nama_mk varchar
> 1	02001	Agama
> 2	02002	Aljabar Linier
> 3	02003	Algoritma dan Struktur Data
> 4	02004	Aljabar Linier
> 5	02005	Analisis Dan Desain Berorient
> 6	02006	Bahasa Indonesia
> 7	02007	Bahasa Inggris
> 8	02008	Bahasa Inggris 2
> 9	02009	Bahasa Inggris Persiapan Ke
> 10	02010	Basis Data
> 11	02011	Desain Pemrograman Web

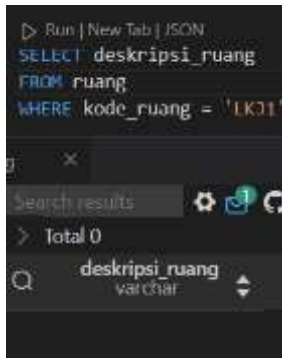
Run | New Tab | JSON
SELECT * FROM jp; files

Search results

	kode jp int	jp mulai time	jp selesai time
1	1	07:00:00	07:50:00
2	2	07:50:00	08:40:00
3	3	08:40:00	09:30:00
4	4	09:40:00	10:30:00
5	5	10:30:00	11:20:00
6	6	11:20:00	12:10:00
7	7	12:50:00	13:40:00
8	8	13:40:00	14:30:00
9	9	14:30:00	15:20:00
10	10	15:30:00	15:30:00
11	11	16:20:00	17:10:00

PRAKTIKUM 2 : Percobaan Select Query

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.



```
Run | New Tab | JSON
SELECT deskripsi_ruang
FROM ruang
WHERE kode_ruang = 'LKJ1'
```

Search results

Total 0

deskripsi_ruang

varchar

2. 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.



```
Run | New Tab | JSON
SELECT DISTINCT kode_hari
FROM jadwal;
```


Search results

Total 5

kode_hari
001
002
003
004
005

varchar

3. 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.



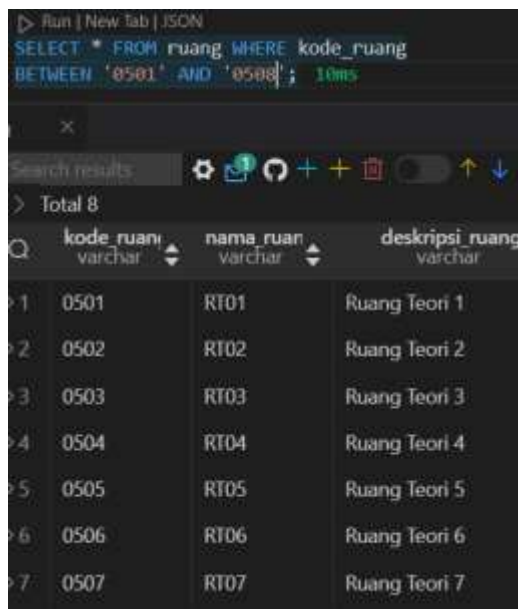
```
Run | New Tab | JSON
SELECT * FROM ruang
WHERE nama_ruang IN ('RT01', 'RT10'); 2ms
```

Search results

Total 2

	kode ruang varchar	nama ruang varchar	deskripsi ruang varchar
> 1	0501	RT01	Ruang Teori 1
> 2	0806	RT10	Ruang Teori 10

4. 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.



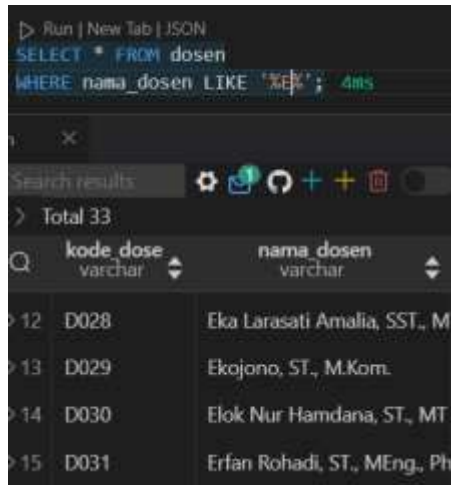
```
Run | New Tab | JSON
SELECT * FROM ruang WHERE kode_ruang
BETWEEN '0501' AND '0508'; 10ms
```

Search results

Total 8

	kode ruang varchar	nama ruang varchar	deskripsi ruang varchar
> 1	0501	RT01	Ruang Teori 1
> 2	0502	RT02	Ruang Teori 2
> 3	0503	RT03	Ruang Teori 3
> 4	0504	RT04	Ruang Teori 4
> 5	0505	RT05	Ruang Teori 5
> 6	0506	RT06	Ruang Teori 6
> 7	0507	RT07	Ruang Teori 7

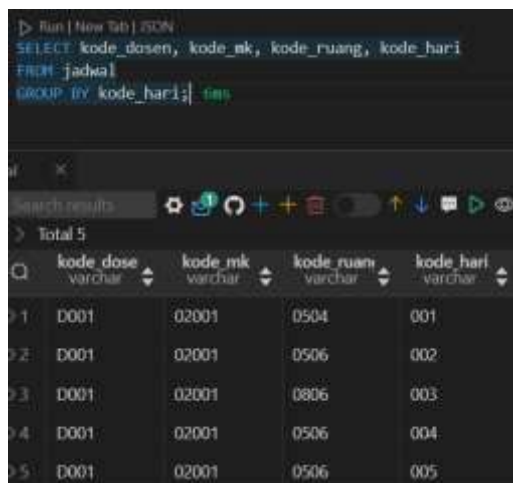
- Untuk menampilkan data yang memiliki kemiripan dengan keyword yang diinginkan digunakan SELECT dengan LIKE. Eksekusi SQL berikut untuk menampilkan kolom kode_dosen, dan nama_dosen, pada tabel dosen yang memiliki nama dengan huruf awal 'E'.



```
Run | New Tab | JSON
SELECT * FROM dosen
WHERE nama_dosen LIKE 'E%'; 4ms
```

	kode_dose varchar	nama_dosen varchar
12	D028	Eka Larasati Amalia, SST., M
13	D029	Ekojono, ST., M.Kom.
14	D030	Elok Nur Hamdana, ST., MT
15	D031	Erfan Rohadi, ST., MEng., Ph

- 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.



```
Run | New Tab | JSON
SELECT kode_dosen, kode_mk, kode_ruang, kode_hari
FROM jadwal
GROUP BY kode_hari; 4ms
```

	kode_dose varchar	kode_mk varchar	kode_ruang varchar	kode_hari varchar
1	D001	02001	0504	001
2	D001	02001	0506	002
3	D001	02001	0806	003
4	D001	02001	0506	004
5	D001	02001	0506	005

7. 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.



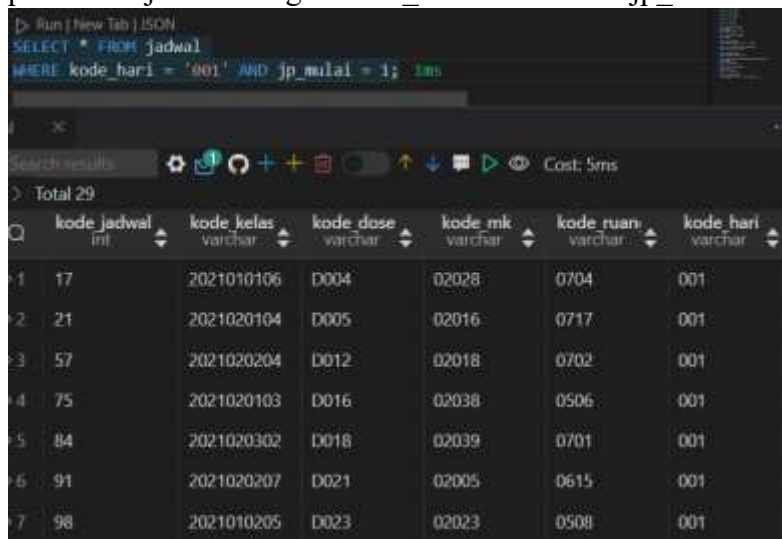
```
SQL> Run [New Tab] | SQL
SELECT kode_jp, jp_mulai
FROM jp
ORDER BY jp_mulai; 3ms
```

Search results

Total 12

	kode_jp int	jp_mulai time
1	1	07:00:00
2	2	07:50:00
3	3	08:40:00
4	4	09:40:00
5	5	10:30:00
6	6	11:20:00
7	7	12:50:00

8. 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.



```
SQL> Run [New Tab] | SQL
SELECT * FROM jadwal
WHERE kode_hari = '001' AND jp_mulai = 1; 5ms
```

Search results

Total 29

	kode_jadwal int	kode_kelas varchar	kode_dose varchar	kode_mik varchar	kode_ruas varchar	kode_hari varchar
1	17	2021010106	D004	02028	0704	001
2	21	2021020104	D005	02016	0717	001
3	57	2021020204	D012	02018	0702	001
4	75	2021020103	D016	02038	0506	001
5	84	2021020302	D018	02039	0701	001
6	91	2021020207	D021	02005	0615	001
7	98	2021010205	D023	02023	0508	001

9. 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.

```
If you are an premium user, you can use the following SQL query to get the data from the 'hari' and 'jadwal' tables using the UNION operator. This query will return the 'kode_hari' column from both tables, removing any duplicate values.
```

```
SELECT kode_hari
FROM hari
UNION
SELECT kode_hari
FROM jadwal;
```

Search results

> Total 7





kode_hari
001
002
003
004
005
006
007

10. 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 ada di tabel penugasan atau departemen.

Run | New Tab | JSON

SELECT kode_hari
FROM hari
UNION ALL
SELECT kode_hari FROM jadwal;

Result(RO) ×

Search results    

2 3 4 > Total 377

kode_hari varchar	
001	
002	
003	
004	
005	
006	
007	

PRAKTIKUM 3 : Percobaan Select Sub-Query

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.

```
SQL> Run [New Tab] SQL
SELECT kode_dosen, kode_mk, kode_hari, jp_mulai, jp_selesai
FROM jadwal
WHERE jp_selesai IN (SELECT MAX (jp_selesai) FROM jadwal);
```

Search results

Total 99

kode dosen varchar	kode mk varchar	kode hari varchar	jp mulai int	jp selesai int
D001	02001	003	10	12
D001	02001	005	10	12
D002	02037	001	7	12
D002	02025	004	7	12
D004	02032	001	10	12
D004	02034	002	7	12
D005	02016	001	10	12

2. 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.

```
SQL> Run [New Tab] SQL
SELECT kode_dosen, kode_mk, kode_hari, jp_mulai, jp_selesai
FROM jadwal
WHERE jp_selesai < ALL
(SELECT jp_selesai FROM jadwal WHERE jp_selesai = 6);
```

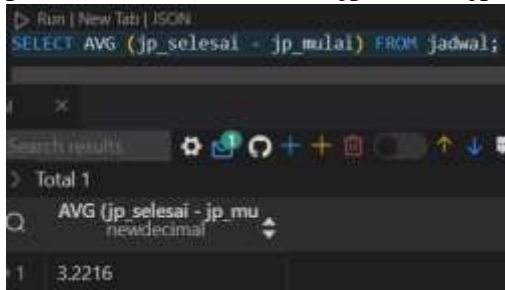
Search results

Total 112

kode dosen varchar	kode mk varchar	kode hari varchar	jp mulai int	jp selesai int
D001	02001	004	1	3
D002	02036	003	2	4
D003	02012	002	1	4
D003	02012	002	1	4
D003	02017	003	2	5
D003	02017	003	2	5
D003	02017	005	2	5

PRAKTIKUM 4 : Percobaan Fungsi Agregasi

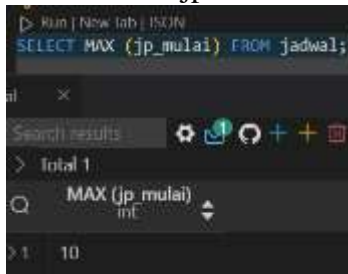
1. Untuk menghitung rata-rata, digunakan tambahan statement AVG. Eksekusi SQL berikut untuk menampilkan rata-rata durasi perkuliahan dalam satu jadwal perkuliahan sesuai kolom jp_mulai, jp_selesai dari tabel jadwal.



The screenshot shows a SQL query editor with the query: `SELECT AVG (jp_selesai - jp_mulai) FROM jadwal;`. Below the query, the search results are displayed, showing a single row with the value 3.2216. The results are summarized as 'Total 1' and the data type is 'newdecimal'.

	AVG (jp_selesai - jp_mu
1	3.2216

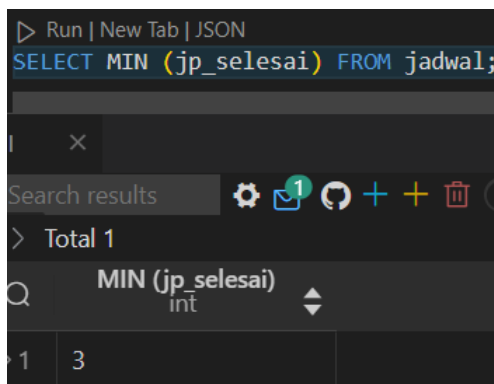
2. Untuk menghitung nilai tertinggi, digunakan tambahan statement MAX. Eksekusi SQL berikut untuk menampilkan dimulainya jam perkuliahan yang paling siang sesuai kolom jp_mulai dari tabel jadwal.



The screenshot shows a SQL query editor with the query: `SELECT MAX (jp_mulai) FROM jadwal;`. Below the query, the search results are displayed, showing a single row with the value 10. The results are summarized as 'Total 1' and the data type is 'int'.

	MAX (jp_mulai)
1	10

3. 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.



The screenshot shows a SQL query editor with the query: `SELECT MIN (jp_selesai) FROM jadwal;`. Below the query, the search results are displayed, showing a single row with the value 3. The results are summarized as 'Total 1' and the data type is 'int'.

	MIN (jp_selesai)
1	3

4. 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.

```
Run | New Tab | JSON
SELECT SUM(jp_selesai - jp_mulai) FROM jadwal WHERE kode_dosen = 'D001';
```

Search results

Total 1

SUM(jp_selesai - jp_mu
newdecimal

1	14
---	----

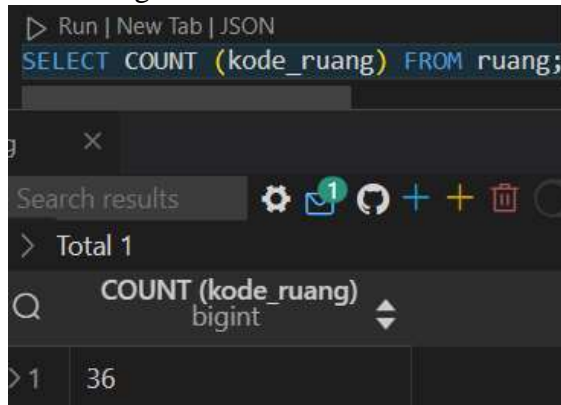
```
Run | New Tab | JSON
SELECT * FROM jadwal WHERE kode_dosen = 'D001';
```

Search results

Total 7

	kode jadw int	kode kelas varchar	kode dose varchar	kode mk varchar	kode ruan varchar	kode ha varchar
> 1	1	2021020101	D001	02001	0504	001
> 2	2	2021010103	D001	02001	0506	002
> 3	3	2021010105	D001	02001	0806	003
> 4	4	2021010102	D001	02001	0506	004
> 5	5	2021010106	D001	02001	0806	004
> 6	6	2021010101	D001	02001	0506	004
> 7	7	2021010104	D001	02001	0506	005

5. Untuk menunjukkan jumlah baris pada kolom yang diinginkan digunakan COUNT(). Eksekusi SQL berikut untuk menampilkan jumlah baris pada kolom kode_ruang dari tabel ruang.

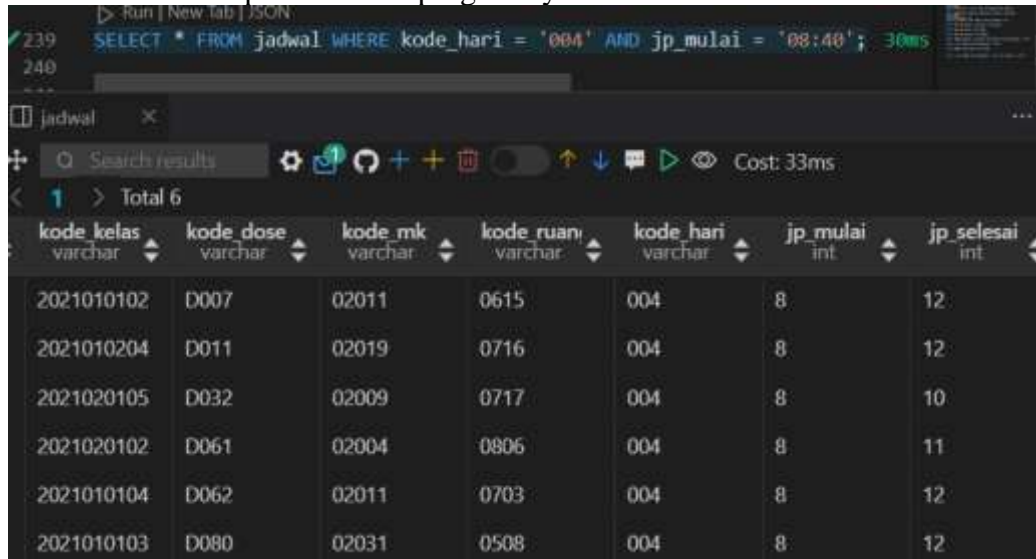


The screenshot shows a SQL IDE interface. At the top, there's a toolbar with 'Run', 'New Tab', and 'JSON' buttons. Below it, the SQL query `SELECT COUNT (kode_ruang) FROM ruang;` is entered. A search bar labeled 'Search results' is visible, along with icons for settings, a notification (1), refresh, and other actions. Below the search bar, it says '> Total 1'. The main area shows the result of the query: a single row with the value '36'. The column header is 'COUNT (kode_ruang)' and the data type is 'bigint'.

```
Run | New Tab | JSON
SELECT COUNT (kode_ruang) FROM ruang;
Search results
> Total 1
COUNT (kode_ruang)
bigint
> 1 | 36
```

TUGAS

1. Himpunlah *screenshot* sintaksis SQL berikut hasil SELECT yang Anda dapat di komputer Anda pada bagian **Praktikum** dalam sebuah laporan dengan format **PDF**!
➔ Screenshot sudah pada bagian atas
2. Bagaimana sintaksis untuk menampilkan semua kolom jadwal pada hari kamis yang dimulai pada pukul 08:40 ? Tunjukkan outputnya!
➔ Berikut adalah output dan kode programnya















```
SELECT * FROM jadwal WHERE kode_hari = '004' AND jp_mulai = '08:40';
```

kode_kelas varchar	kode_dose varchar	kode_mk varchar	kode_ruang varchar	kode_hari varchar	jp_mulai int	jp_selesai int
2021010102	D007	02011	0615	004	8	12
2021010204	D011	02019	0716	004	8	12
2021020105	D032	02009	0717	004	8	10
2021020102	D061	02004	0806	004	8	11
2021010104	D062	02011	0703	004	8	12
2021010103	D080	02031	0508	004	8	12

3. 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!
➔ Ada 51 dosen yang mengajar pada hari selasa. Berikut adalah kode program dan tampilannya


```
Run | New Tab | JSON
SELECT COUNT(DISTINCT kode_dosen) FROM jadwal WHERE kode_hari = '002';
```

Search results             Cost: 5ms

> Total 1













Q COUNT(DISTINCT kode_dosen) bigint

> 1	51
-----	----

```
Run | New Tab | JSON
SELECT DISTINCT kode_dosen FROM jadwal WHERE kode_hari = '002'; ZMS
```

al X

DISTINCT kode_dosen FROM jadwal WHERE kode_hari = '002' LIMIT 100

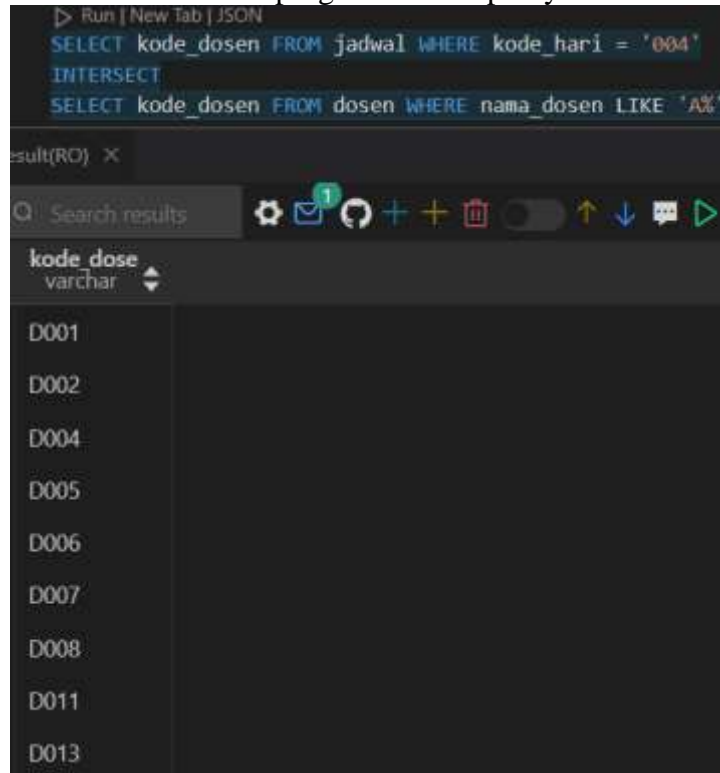
Search results             Cost: 3ms

Q kode_dosen varchar

> 1	D001
> 2	D003
> 3	D004
> 4	D005
> 5	D006
> 6	D007
> 7	D009
> 8	D010
> 9	D011
> 10	D012
> 11	D013
> 12	D015
> 13	D021
> 14	D025

4. Tampilkan kode dosen yang mengajar pada hari kamis dari tabel jadwal yang nama depannya diawali dengan huruf A. gunakan operasi intersect! Tunjukkan sintaksis dan output dari pencarian tersebut.

➔ Berikut adalah kode program dan outputnya :



The screenshot shows a SQL query editor with a dark theme. The query is as follows:

```
SELECT kode_dosen FROM jadwal WHERE kode_hari = '004'
INTERSECT
SELECT kode_dosen FROM dosen WHERE nama_dosen LIKE 'A%'
```

Below the query, the results are displayed in a table. The table has a single column labeled 'kode dose' with a data type of 'varchar'. The results are listed in ascending order: D001, D002, D004, D005, D006, D007, D008, D011, and D013.

kode dose
D001
D002
D004
D005
D006
D007
D008
D011
D013

5. Tampilkan kode_ruang tertentu pada hari tertentu dengan diurutkan berdasarkan dengan kode jam yang ter awal.

➔ Berikut adalah kode program dan outputnya :

Run | New Tab | JSON
SELECT kode_ruang
FROM jadwal
WHERE kode_hari = '003'
ORDER BY jp_mulai ASC;

wal

Search results



kode_ruang
varchar

> 1	0620
> 2	0707
> 3	0703
> 4	0703
> 5	0502
> 6	0713
> 7	0615
> 8	0715
> 9	0805
> 10	0720
> 11	0507
> 12	0507
> 13	0506
> 14	0705