BASIS DATA

Dasar MySQL Data Retrieval Language

Lavina 2341760062

Praktikum 1

```
74  SELECT * FROM jadwal;  3ms
jadwal
SELECT * FROM jadwal LIMIT 100
```

Search results Cost: 9ms 1 2 3 4 Total 370

	kode_jad int(10)	kode_kela varchar(10	kode_dos varchar(4)	kode_mk varchar(5)	kode_rua varchar(5)	kode_hari varchar(3)	jp_mulai int(3)	jp_selesai int(3)
1	1	2021020101	D001	02001	0504	001	7	9
2	2	2021010103	D001	02001	0506	002	9	11
3	3	2021010105	D001	02001	0806	003	10	12
4	4	2021010102	D001	02001	0506	004	1	3
5	5	2021010106	D001	02001	0806	004	4	6
6	6	2021010101	D001	02001	0506	004	7	9
7	7	2021010104	D001	02001	0506	005	10	12
8	8	2021010206	D002	02037	0702	001	7	12
9	9	2021020202	D002	02036	0708	003	2	4
10	10	2021010205	D002	02037	0713	004	1	6
11	11	2021020209	D002	02025	0719	004	7	12
12	12	2021020301	D003	02012	0508	002	1	4

```
74  SELECT * FROM prodi;  2ms
prodi
SELECT * FROM prodi LIMIT 100
```

Search results

	kode_pr varchar(3)	nama_prodi varchar(100)
1	001	D3 Manajemen Informat
2	002	D4 Teknik Informatika

```
74  SELECT * FROM kelas;  2ms
```

kelas

```
SELECT * FROM kelas LIMIT 100
```

Search results

	kode_ke varchar(10	kode_pro varchar(3)	nama_kel varchar(5)
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
8	2021010201	001	MI-2A
9	2021010202	001	MI-2B
10	2021010203	001	MI-2C
11	2021010204	001	MI-2D
12	2021010205	001	MI-2E

```
74  SELECT * FROM ruang;  1ms
```

ruang

```
SELECT * FROM ruang LIMIT 100
```

Search results

	kode_ru varchar(5)	nama_rua varchar(20	deskripsi_ruang varchar(100)
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 Inf
10	0617	LSI2	Laboratorium Sistem Inf
11	0618	LSI3	Laboratorium Sistem Inf

```
74  SELECT * FROM dosen;  2ms
```

dosen

```
SELECT * FROM dosen LIMIT 100
```

Search results

	kode_d varchar(4)	nama_dosen varchar(100)
1	D001	Abdul Chalim, SAg., MPd.I
2	D002	Ade Ismail
3	D003	Agung Nugroho Pramudhita ST., MT.
4	D004	Ahmadi Yuli Ananta ST., MM.
5	D005	Ane Fany Novitasari, SH.MKn.
6	D006	Annisa Puspa Kirana MKom.
7	D007	Annisa Taufika Firdausi ST., MT.
8	D008	Anugrah Nur Rahmanto SSn., MDs.
9	D009	Ariadi Retno Ririd SKom., MKom.
10	D010	Arie Rachmad Syulistyo SKom., MKom.
11	D011	Arief Prasetyo SKom., MKom.
12	D012	Arwin Sumari ST., MT., DR

```
74  SELECT * FROM hari;  2ms
```

hari

```
SELECT * FROM hari LIMIT 100
```

Search results

	kode_hi varchar(3)	nama_hai varchar(10
1	001	Senin
2	002	Selasa
3	003	Rabu
4	004	Kamis
5	005	Jumat
6	006	Sabtu
7	007	Minggu

```
74  SELECT * FROM mk;  2ms
```

mk

```
SELECT * FROM mk LIMIT 100
```

Search results

	kode_m varchar(5)	nama_mk varchar(100)
1	02001	Agama
2	02002	Alajabar Linier
3	02003	Algoritma dan Struktur Data
4	02004	Aljabar Linier
5	02005	Analisis Dan Desan Berorientasi Objek
6	02006	Bahasa Indonesia
7	02007	Bahasa Inggris
8	02008	Bahasa Inggris 2
9	02009	Bahasa Inggris Persiapan Kerja
10	02010	Basis Data
11	02011	Desain Pemrograman Web
12	02012	Digital Entrepreneurship

```
74  SELECT * FROM jp;  1ms
```

jp

```
SELECT * FROM jp LIMIT 100
```

Search results

	kode_jp int(3)	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
12	12	17:10:00	18:00:00

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.

```
78  SELECT deskripsi_ruang FROM ruang WHERE nama_ruang='LKJ1';
```

ruang

Search results Cost: 2ms

1 Total 1

	deskripsi_ruang varchar(100)
1	Laboratorium Keamanan Jaringan 1

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.

```
80  SELECT DISTINCT kode_hari FROM jadwal;
```

jadwal

Search results

1 Total 5

	kode_hari varchar(3)
1	001
2	002
3	003
4	004
5	005

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.

```
82  SELECT * FROM ruang WHERE nama_ruang IN ('RT01', 'RT10');
```

ruang

Search results Cost: 4ms

1 Total 2

	kode_ru varchar(5)	nama_rua varchar(20	deskripsi_ruang varchar(100)
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.

```
84  SELECT * FROM ruang WHERE kode_ruang BETWEEN '0501' AND '0508';
```

ruang

Search results Cost: 8ms

1 Total 8

	kode_ruang varchar(5)	nama_ruang varchar(20)	deskripsi_ruang varchar(100)
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

5. 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'.

```
86  SELECT * FROM dosen WHERE nama_dosen LIKE 'E%';
```

dosen

Search results

1 Total 4

	kode_dosen varchar(4)	nama_dosen varchar(100)
1	D028	Eka Larasati Amalia, SST., MT.
2	D029	Ekojono, ST., M.Kom.
3	D030	Elok Nur Hamdana, ST., MT
4	D031	Erfan Rohadi, ST., MEng., PhD.

6. 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.

```
88  SELECT kode_dosen, kode_mk, kode_ruang, kode_hari
89  FROM jadwal GROUP BY kode_hari;  2ms
```

jadwal

Search results Cost: 4ms

1 Total 5

	kode_dosen varchar(4)	kode_mk varchar(5)	kode_ruang varchar(5)	kode_hari varchar(3)
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.

```
91  SELECT kode_jp, jp_mulai
92  FROM jp ORDER BY jp_mulai;
```

jp

Search results

1 Total 12

	kode_jp int(3)	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	8	13:40:00

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.

```
94  SELECT * FROM jadwal
95  WHERE kode_hari = '001' AND jp_mulai = 1;  4ms
```

jadwal

Search results Cost: 6ms 1 Total 29

	kode_jad int(10)	kode_kela varchar(10	kode_dos varchar(4)	kode_mk varchar(5)	kode_rua varchar(5)	kode_hari varchar(3)	jp_mulai int(3)	jp_selesai int(3)
1	17	2021010106	D004	02028	0704	001	1	3
2	21	2021020104	D005	02016	0717	001	1	3
3	57	2021020204	D012	02018	0702	001	1	5
4	75	2021020103	D016	02038	0506	001	1	3
5	84	2021020302	D018	02039	0701	001	1	6
6	91	2021020207	D021	02005	0615	001	1	6
7	98	2021010205	D023	02023	0508	001	1	4
8	103	2021010201	D024	02037	0713	001	1	6
9	107	2021020305	D025	02024	0615	001	1	6

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.

```
97  SELECT kode_hari FROM hari
98  UNION
99  SELECT kode_hari FROM jadwal;
```

Result(RO)

Search results

1 Total 7

	kode_hari varchar
1	001
2	002
3	003
4	004
5	005
6	006
7	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 yang ada di tabel **penugasan** atau **departemen**.

```
101  SELECT kode_hari FROM hari
102  UNION ALL
103  SELECT kode_hari FROM jadwal;
```

Result(RO)

Search results

1 2 3 4 Total 377

	kode_hari varchar
1	001
2	002
3	003
4	004
5	005
6	006
7	007
8	001
9	001

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.

```
105 SELECT kode_dosen, kode_mk, kode_hari, jp_mulai, jp_selesai
106 FROM jadwal WHERE jp_selesai
107 IN (SELECT MAX(jp_selesai) FROM jadwal); 7ms
```

Result(RO)

Search results Cost: 7ms

1 Total 99

	kode_dos varchar	kode_mk varchar	kode_hari varchar	jp_mulai int	jp_selesai int
1	D001	02001	003	10	12
2	D001	02001	005	10	12
3	D002	02037	001	7	12
4	D002	02025	004	7	12
5	D004	02032	001	10	12
6	D004	02034	002	7	12
7	D005	02016	001	10	12
8	D006	02037	004	7	12
9	D006	02037	005	7	12
10	D007	02011	004	8	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.

```
109 SELECT kode_dosen, kode_mk, kode_hari, jp_mulai, jp_selesai
110 FROM jadwal WHERE jp_selesai < ALL
111 (SELECT jp_selesai FROM jadwal WHERE jp_selesai = 6);  3ms
```

Result(RO)

Search results Cost: 3ms

1 2 Total 112

	kode_dos varchar	kode_mk varchar	kode_hari varchar	jp_mulai int	jp_selesai int
1	D001	02001	004	1	3
2	D002	02036	003	2	4
3	D003	02012	002	1	4
4	D003	02012	002	1	4
5	D003	02017	003	2	5
6	D003	02017	003	2	5
7	D003	02017	005	2	5

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.

```
113 SELECT AVG(jp_selesai - jp_mulai) FROM jadwal;
```

jadwal

Search results

1 Total 1

	AVG(jp_selesai - jp_r newdecimal
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.

```
115 SELECT MAX(jp_mulai) FROM jadwal;
```

jadwal

Search results

1 Total 1

	MAX(jp_mula int
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.

```
117 SELECT MIN(jp_selesai) FROM jadwal;
```

jadwal

Search results

1 Total 1

	MIN(jp_selesai) int
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.

```
119 SELECT SUM(jp_selesai - jp_mulai) FROM jadwal
120 WHERE kode_dosen = 'D001';  2ms
```

jadwal

Search results

1 Total 1

	SUM(jp_selesai - jp_mulai) newdecimal
1	14

```
122  SELECT * FROM jadwal WHERE kode_dosen = 'D001';  1ms
```

jadwal

Search results Cost: 3ms 1 Total 7

	kode_jad int(10)	kode_kela varchar(10	kode_dos varchar(4)	kode_mk varchar(5)	kode_ruai varchar(5)	kode_hari varchar(3)	jp_mulai int(3)	jp_selesai int(3)
1	1	2021020101	D001	02001	0504	001	7	9
2	2	2021010103	D001	02001	0506	002	9	11
3	3	2021010105	D001	02001	0806	003	10	12
4	4	2021010102	D001	02001	0506	004	1	3
5	5	2021010106	D001	02001	0806	004	4	6
6	6	2021010101	D001	02001	0506	004	7	9
7	7	2021010104	D001	02001	0506	005	10	12

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.

```
124  SELECT count(kode_ruang) FROM ruang;
```

ruang

Search results 1 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!
2. Bagaimana sintaksis untuk menampilkan semua kolom jadwal pada hari kamis yang dimulai pada pukul 08:40 ? Tunjukkan outputnya!

```
126  SELECT * FROM jadwal WHERE kode_hari = '004' AND jp_mulai = 3;  1ms
```

jadwal

Search results Cost: 3ms 1 Total 3

	kode_jad int(10)	kode_kela varchar(10	kode_dos varchar(4)	kode_mk varchar(5)	kode_ruai varchar(5)	kode_hari varchar(3)	jp_mulai int(3)	jp_selesai int(3)
1	80	2021020106	D017	02004	0805	004	3	6
2	194	2021020206	D044	02020	0707	004	3	5
3	195	2021020207	D044	02020	0707	004	3	5

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!

```
128  SELECT count(kode_dosen) FROM jadwal WHERE kode_hari = '002';
```

jadwal

Search results Cost: 2ms

1 Total 1

	count(kode_dosen) bigint
1	79

```
128  SELECT kode_dosen FROM jadwal WHERE kode_hari = '002';
```

jadwal

Search results Cos

1 Total 79

	kode_dos varchar(4)
70	D080
71	D081
72	D081
73	D082
74	D082
75	D082
76	D082
77	D083
78	D085
79	D085

4. 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.

```sql
(SELECT DISTINCT kode_dosen FROM jadwal WHERE kode_hari = '004')
INTERSECT
(SELECT DISTINCT kode_dosen FROM jadwal
WHERE kode_dosen IN (
  SELECT kode_dosen FROM dosen WHERE nama_dosen LIKE 'A%'
));  3ms
```

Result(RO)

Search results Cost: 3ms 1

	kode_dosen varchar
2	D002
3	D004
4	D005
5	D006
6	D007
7	D008
8	D011
9	D013

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

```sql
SELECT kode_ruang, kode_hari, jp_mulai FROM jadwal WHERE kode_hari = '001' ORDER BY jp_mulai;
```

jadwal

```sql
SELECT kode_ruang, kode_hari, jp_mulai FROM jadwal WHERE kode_hari = '001' ORDER BY jp_mulai LIMIT 1
```

Search results Cost: 3ms 1 Total 89

	kode_rua varchar(5)	kode_hari varchar(3)	jp_mulai int(3)
28	0708	001	1
29	0504	001	1
30	0619	001	2
31	0705	001	2
32	0703	001	2
33	0703	001	2
34	0806	001	3
35	0706	001	3
36	0504	001	4
37	0716	001	4
38	0704	001	4
39	0507	001	4
40	0507	001	4