

Laporan Hasil Percobaan

Matematika Diskrit



Nama : Taufan Ali
NIM : 2215016135
Kelas : A

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS SAINS DAN TEKNOLOGI TERAPAN
UNIVERSITAS AHMAD DAHLAN
YOGYAKARTA
2023

Tugas : Membuat database dengan data minimal 25 dan atribut minimal 5. Praktekkan kedelapan operator yang ada(**SELECT, PROJECT, JOIN, INTERSECT, UNION, DIFFERENCE, PRODUCT, dan DIVIDE**)

Untuk software, saya menggunakan software mysql workbench. Berikut saya sertakan data yang saya gunakan :

id	nama	usia	alamat	email	pekerjaan
1	Budi Santoso	25	Jalan Merdeka No. 10	budisantoso@example.com	Mahasiswa
2	Rina Setiawati	35	Jalan Sudirman No. 15	rinasetiawati@example.com	Dokter
3	Ahmad Hidayat	42	Jalan Imam Bonjol No. 20	ahmadhidayat@example.com	Pengusaha
4	Siti Rahayu	28	Jalan Diponegoro No. 8	sitirahayu@example.com	Desainer Grafis
5	Rizky Pratama	50	Jalan Gatot Subroto No. 5	rizkypratama@example.com	Akuntan
6	Dewi Puspitasari	31	Jalan Pahlawan No. 12	dewipuspitasari@example.com	Penulis
7	Eko Sutrisno	38	Jalan Veteran No. 18	ekosutrisno@example.com	Guru
8	Anita Setyawati	27	Jalan Ahmad Yani No. 25	anitasetyawati@example.com	Arsitek
9	Yusuf Kurniawan	45	Jalan Hayam Wuruk No. 30	yusufkurniawan@example.com	Insinyur
10	Ratna Dewi	33	Jalan Asia Afrika No. 7	ratnadewi@example.com	Pengacara
11	Feri Susanto	29	Jalan MT Haryono No. 14	ferisusanto@example.com	Peneliti
12	Maya Indah	41	Jalan HR Rasuna Said No. 22	mayaindah@example.com	Manajer
13	Adi Wijaya	36	Jalan Cikini Raya No. 9	adiwijaya@example.com	Programmer
14	Dewi Permata	26	Jalan Ahmad Dahlan No. 16	dewipermata@example.com	Konsultan
15	Arief Santoso	47	Jalan Thamrin No. 28	ariefsantoso@example.com	Pengusaha
16	Sinta Dewi	32	Jalan Kebon Sirih No. 11	sintadewi@example.com	Pengajar
17	Andi Pratama	39	Jalan Tebet Barat No. 19	andipratama@example.com	Dokter
18	Fitriani Putri	30	Jalan Kuningan No. 6	fitrianiputri@example.com	Blogger
19	Rahmat Widodo	49	Jalan Gatot Subroto No. 17	rahmatwidodo@example.com	Pengusaha
20	Dewi Wulandari	34	Jalan Thamrin No. 23	dewiwulandari@example.com	Desainer Interior
21	Budi Setiawan	37	Jalan Asia Afrika No. 13	budisetiawan@example.com	Arsitek
22	Laras Sekar	24	Jalan Sudirman No. 26	larassekar@example.com	Penulis
23	Hendro Gunawan	43	Jalan HR Rasuna Said No. 21	hendrogunawan@example.com	Insinyur
24	Anita Halim	31	Jalan MT Haryono No. 7	anitahalim@example.com	Pengacara
25	Agus Wijaya	40	Jalan Kebon Sirih No. 14	aguswijaya@example.com	Peneliti

id	nama_pekerjaan
1	Manager
2	Pengembang Perangkat Lunak
3	Akuntan
4	Marketing
5	Desainer Grafis
NULL	NULL

Untuk Membuat database, pertama-tama buatlah tabel dengan mengetikkan query

CREATE TABLE *nama_tabel* (

nama_atribut jenis_atribut;

Pada database yang saya buat, saya menggunakan atribut id, nama, usia, alamat, email, pekerjaan.

```
1 • CREATE TABLE data_pengguna (  
2     id INT AUTO_INCREMENT PRIMARY KEY,  
3     nama VARCHAR(50),  
4     usia INT,  
5     alamat VARCHAR(100),  
6     email VARCHAR(50),  
7     pekerjaan VARCHAR(50)  
8 );  
9
```

```
1 • CREATE TABLE data_pekerjaan (  
2     id INT PRIMARY KEY,  
3     nama_pekerjaan VARCHAR(50)  
4 );
```

Untuk menginputkan data kedalam tabel gunakan query INSERT INTO

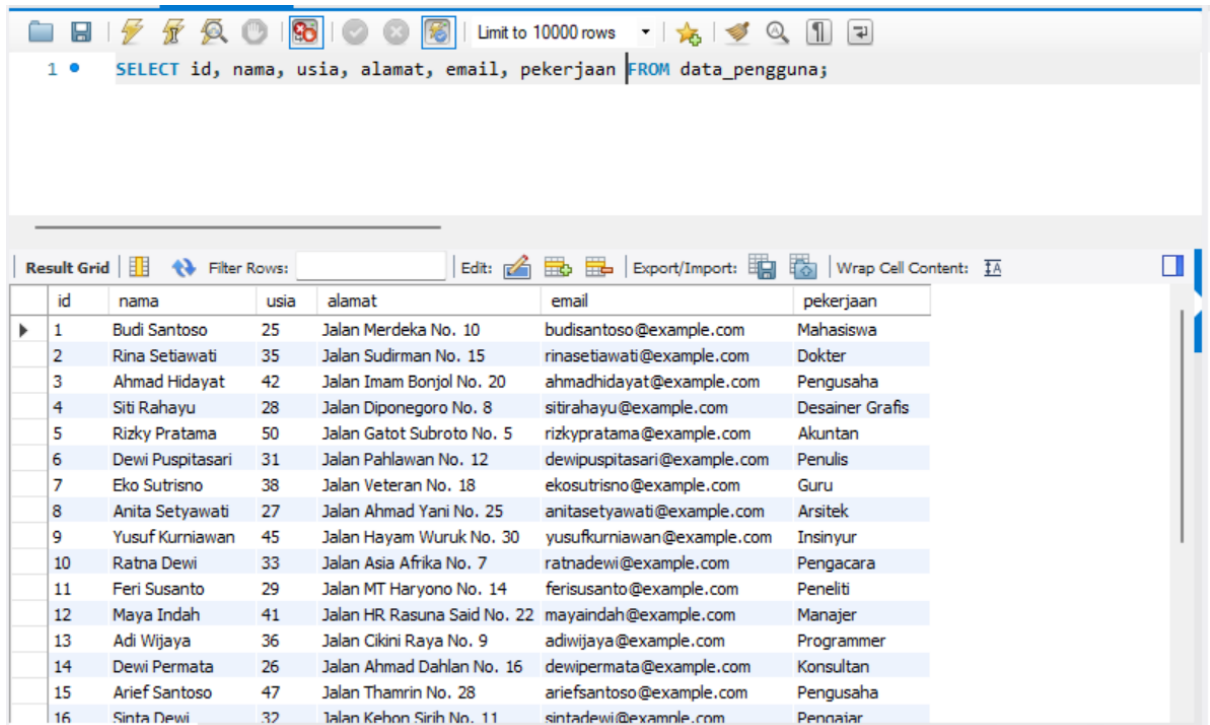
nama_tabel(nama_atribut)

VALUES

(data)

```
10 INSERT INTO data_pengguna (nama, usia, alamat, email, pekerjaan)  
11 VALUES  
12 ('Budi Santoso', 25, 'Jalan Merdeka No. 10', 'budisantoso@example.com', 'Mahasiswa'),  
13 ('Rina Setiawati', 35, 'Jalan Sudirman No. 15', 'rinasetiawati@example.com', 'Dokter'),  
14 ('Ahmad Hidayat', 42, 'Jalan Imam Bonjol No. 20', 'ahmadhidayat@example.com', 'Pengusaha'),  
15 ('Siti Rahayu', 28, 'Jalan Diponegoro No. 8', 'sitirahayu@example.com', 'Desainer Grafis'),  
16 ('Rizky Pratama', 50, 'Jalan Gatot Subroto No. 5', 'rizkypratama@example.com', 'Akuntan'),  
17 ('Dewi Puspitasari', 31, 'Jalan Pahlawan No. 12', 'dewipuspitasari@example.com', 'Penulis'),  
18 ('Eko Sutrisno', 38, 'Jalan Veteran No. 18', 'ekosutrisno@example.com', 'Guru'),  
19 ('Anita Setyawati', 27, 'Jalan Ahmad Yani No. 25', 'anitasetyawati@example.com', 'Arsitek'),  
20 ('Yusuf Kurniawan', 45, 'Jalan Hayam Wuruk No. 30', 'yusufkurniawan@example.com', 'Insinyur'),  
21 ('Ratna Dewi', 33, 'Jalan Asia Afrika No. 7', 'ratnadewi@example.com', 'Pengacara'),  
22 ('Feri Susanto', 29, 'Jalan MT Haryono No. 14', 'ferisusanto@example.com', 'Peneliti'),  
23 ('Maya Indah', 41, 'Jalan HR Rasuna Said No. 22', 'mayaindah@example.com', 'Manajer'),  
24 ('Adi Wijaya', 36, 'Jalan Cikini Raya No. 9', 'adiwijaya@example.com', 'Programmer'),  
25 ('Dewi Permata', 26, 'Jalan Ahmad Dahlan No. 16', 'dewipermata@example.com', 'Konsultan'),  
26 ('Arief Santoso', 47, 'Jalan Thamrin No. 28', 'ariefsantoso@example.com', 'Pengusaha'),  
27 ('Sinta Dewi', 32, 'Jalan Kebon Sirih No. 11', 'sintadewi@example.com', 'Pengajar'),  
28 ('Andi Pratama', 39, 'Jalan Tebet Barat No. 19', 'andipratama@example.com', 'Dokter'),  
29 ('Fitriani Putri', 30, 'Jalan Kuningan No. 6', 'fitrianiputri@example.com', 'Blogger'),  
30 ('Rahmat Widodo', 49, 'Jalan Gatot Subroto No. 17', 'rahmatwido@example.com', 'Pengusaha'),
```

1. SELECT. Digunakan untuk mengambil data dari tabel berdasarkan kriteria tertentu. Berikut adalah contoh query SELECT untuk mengambil data dari tabel "data_pengguna" dengan atribut "id", "nama", "usia", "alamat", "email", dan "pekerjaan" di MySQL:



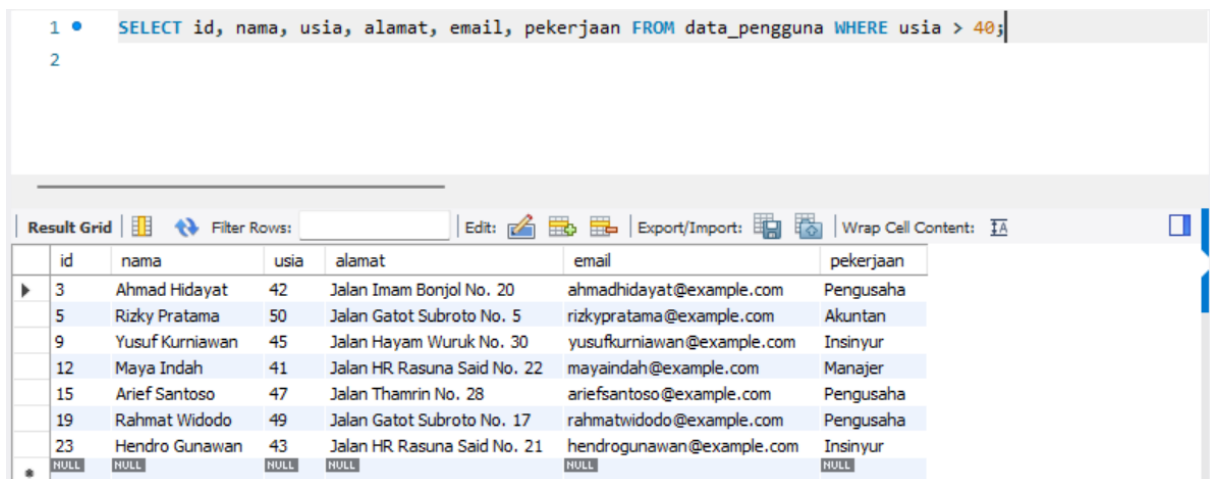
The screenshot shows a MySQL query editor with the following query entered:

```
1 • SELECT id, nama, usia, alamat, email, pekerjaan FROM data_pengguna;
```

The result grid displays the following data:

	id	nama	usia	alamat	email	pekerjaan
▶	1	Budi Santoso	25	Jalan Merdeka No. 10	budisantoso@example.com	Mahasiswa
	2	Rina Setiawati	35	Jalan Sudirman No. 15	rinasetiawati@example.com	Dokter
	3	Ahmad Hidayat	42	Jalan Imam Bonjol No. 20	ahmadhidayat@example.com	Pengusaha
	4	Siti Rahayu	28	Jalan Diponegoro No. 8	sitirahayu@example.com	Desainer Grafis
	5	Rizky Pratama	50	Jalan Gatot Subroto No. 5	rizkypratama@example.com	Akuntan
	6	Dewi Puspitasari	31	Jalan Pahlawan No. 12	dewipuspitasari@example.com	Penulis
	7	Eko Sutrisno	38	Jalan Veteran No. 18	ekosutrisno@example.com	Guru
	8	Anita Setyawati	27	Jalan Ahmad Yani No. 25	anitasetyawati@example.com	Arsitek
	9	Yusuf Kurniawan	45	Jalan Hayam Wuruk No. 30	yusufkurniawan@example.com	Insinyur
	10	Ratna Dewi	33	Jalan Asia Afrika No. 7	ratnadewi@example.com	Pengacara
	11	Feri Susanto	29	Jalan MT Haryono No. 14	ferisusanto@example.com	Peneliti
	12	Maya Indah	41	Jalan HR Rasuna Said No. 22	mayaindah@example.com	Manajer
	13	Adi Wijaya	36	Jalan Cikini Raya No. 9	adiwijaya@example.com	Programmer
	14	Dewi Permata	26	Jalan Ahmad Dahlan No. 16	dewipermata@example.com	Konsultan
	15	Arief Santoso	47	Jalan Thamrin No. 28	ariefsantoso@example.com	Pengusaha
	16	Sinta Dewi	32	Jalan Kehon Sirih No. 11	sintadewi@example.com	Pennair

Query di atas akan mengambil semua data yang ada dalam tabel "data_pengguna" dan mengembalikan kolom "id", "nama", "usia", "alamat", "email", dan "pekerjaan" dari setiap baris. Gunakan klausa WHERE jika ingin menambahkan kriteria atau kondisi tertentu. Berikut adalah contoh query SELECT dengan kondisi:



The screenshot shows a MySQL query editor with the following query entered:

```
1 • SELECT id, nama, usia, alamat, email, pekerjaan FROM data_pengguna WHERE usia > 40;
2
```

The result grid displays the following data:

	id	nama	usia	alamat	email	pekerjaan
▶	3	Ahmad Hidayat	42	Jalan Imam Bonjol No. 20	ahmadhidayat@example.com	Pengusaha
	5	Rizky Pratama	50	Jalan Gatot Subroto No. 5	rizkypratama@example.com	Akuntan
	9	Yusuf Kurniawan	45	Jalan Hayam Wuruk No. 30	yusufkurniawan@example.com	Insinyur
	12	Maya Indah	41	Jalan HR Rasuna Said No. 22	mayaindah@example.com	Manajer
	15	Arief Santoso	47	Jalan Thamrin No. 28	ariefsantoso@example.com	Pengusaha
	19	Rahmat Widodo	49	Jalan Gatot Subroto No. 17	rahmatwido@example.com	Pengusaha
	23	Hendro Gunawan	43	Jalan HR Rasuna Said No. 21	hendrogunawan@example.com	Insinyur
*	NULL	NULL	NULL	NULL	NULL	NULL

Query di atas akan mengambil data pengguna yang memiliki usia lebih dari 40 tahun.

2. PROJECT. Dalam MySQL, project dapat dilakukan dengan menggunakan klausa SELECT untuk memilih kolom tertentu dari tabel. Dalam MySQL, project dapat dilakukan dengan menggunakan klausa SELECT untuk memilih kolom tertentu dari tabel.

```
1 • SELECT id, nama, email FROM data_pengguna;
```

	id	nama	email
▶	1	Budi Santoso	budisantoso@example.com
	2	Rina Setiawati	rinasetiawati@example.com
	3	Ahmad Hidayat	ahmadhidayat@example.com
	4	Siti Rahayu	sitirahayu@example.com
	5	Rizky Pratama	rizkypratama@example.com
	6	Dewi Puspitasari	dewipuspitasari@example.com
	7	Eko Sutrisno	ekosutrisno@example.com
	8	Anita Setyawati	anitasetyawati@example.com
	9	Yusuf Kurniawan	yusufkurniawan@example.com
	10	Ratna Dewi	ratnadewi@example.com
	11	Feri Susanto	ferisusanto@example.com
	12	Maya Indah	mayaindah@example.com
	13	Adi Wijaya	adiwijaya@example.com
	14	Dewi Permata	dewipermata@example.com
	15	Arief Santoso	ariefsantoso@example.com
	16	Sinta Dewi	sintadewi@example.com

Query di atas akan memilih kolom "id", "nama", dan "email" dari tabel "data_pengguna" dan mengembalikan hasilnya. Dapat menambahkan kolom lain yang ingin dipilih dengan memasukkannya dalam daftar kolom yang dipisahkan oleh koma(,).

3. JOIN. Operasi JOIN digunakan untuk menggabungkan baris dari dua atau lebih tabel berdasarkan kunci yang terkait. Berikut ini adalah contoh query JOIN menggunakan klausa JOIN dalam MySQL untuk menggabungkan tabel "data_pengguna" dengan tabel lain (misalnya, "pekerjaan") berdasarkan kunci yang sesuai:

```
1 SELECT data_pengguna.nama, data_pengguna.usia, data_pekerjaan.nama_pekerjaan
2 FROM data_pengguna
3 JOIN data_pekerjaan ON data_pengguna.pekerjaan = data_pekerjaan.nama_pekerjaan;
```

	nama	usia	nama_pekerjaan
▶	Siti Rahayu	28	Desainer Grafis
	Rizky Pratama	50	Akuntan

Hasil tersebut akan menggabungkan informasi nama, usia, dan nama pekerjaan dari kedua tabel berdasarkan nilai pekerjaan yang cocok antara keduanya.

4. INTERSECT. Dalam MySQL, perintah INTERSECT tidak secara langsung didukung. Namun, Anda dapat mencapai hasil yang serupa dengan menggunakan perintah JOIN dan subquery. Berikut adalah contoh query yang dapat digunakan untuk mencari irisan (intersection) antara dua set data_pengguna dengan usia di bawah 30 tahun:

```
1 • SELECT id, nama, usia, alamat, email, pekerjaan FROM data_pengguna WHERE usia < 30 AND id IN (  
2     SELECT id  
3     FROM data_pengguna  
4     WHERE usia < 30  
5 );
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |

id	nama	usia	alamat	email	pekerjaan
1	Budi Santoso	25	Jalan Merdeka No. 10	budisantoso@example.com	Mahasiswa
4	Siti Rahayu	28	Jalan Diponegoro No. 8	sitirahayu@example.com	Desainer Grafis
8	Anita Setyawati	27	Jalan Ahmad Yani No. 25	anitasetyawati@example.com	Arsitek
11	Feri Susanto	29	Jalan MT Haryono No. 14	ferisusanto@example.com	Peneliti
14	Dewi Permata	26	Jalan Ahmad Dahlan No. 16	dewipermata@example.com	Konsultan
22	Laras Sekar	24	Jalan Sudirman No. 26	larassekar@example.com	Penulis
NULL	NULL	NULL	NULL	NULL	NULL

Dalam query di atas, subquery yang ada di dalam klausa IN mengembalikan semua id pengguna yang memiliki usia di bawah 30 tahun. Kemudian, query utama menggunakan kondisi WHERE untuk memfilter baris pengguna dengan usia di bawah 30 tahun dan juga id yang ada dalam hasil subquery tersebut.

5. UNION. Untuk menggabungkan (union) hasil dari dua set data_pengguna yang berbeda dalam MySQL, Anda dapat menggunakan perintah UNION. Berikut adalah contoh query yang menggabungkan data pengguna dengan usia di bawah 30 tahun dan data pengguna dengan pekerjaan "Mahasiswa":

Limit to 10000 rows

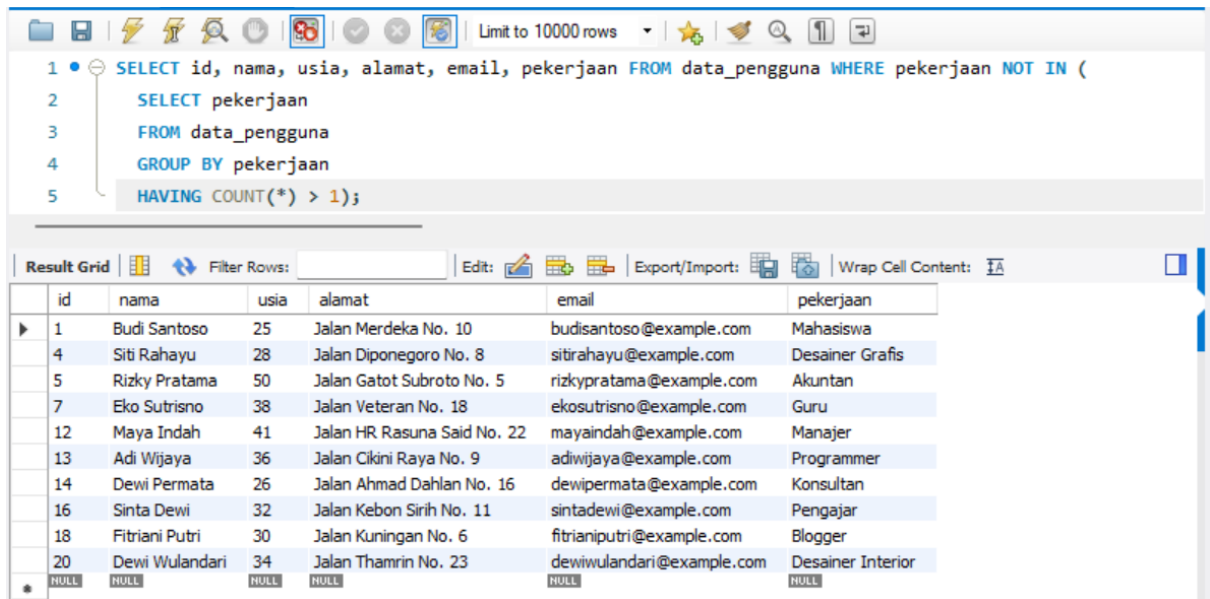
```
1 • SELECT id, nama, usia, alamat, email, pekerjaan FROM data_pengguna WHERE usia < 30  
2 UNION  
3 SELECT id, nama, usia, alamat, email, pekerjaan FROM data_pengguna WHERE pekerjaan = 'Mahasiswa';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

id	nama	usia	alamat	email	pekerjaan
1	Budi Santoso	25	Jalan Merdeka No. 10	budisantoso@example.com	Mahasiswa
4	Siti Rahayu	28	Jalan Diponegoro No. 8	sitirahayu@example.com	Desainer Grafis
8	Anita Setyawati	27	Jalan Ahmad Yani No. 25	anitasetyawati@example.com	Arsitek
11	Feri Susanto	29	Jalan MT Haryono No. 14	ferisusanto@example.com	Peneliti
14	Dewi Permata	26	Jalan Ahmad Dahlan No. 16	dewipermata@example.com	Konsultan
22	Laras Sekar	24	Jalan Sudirman No. 26	larassekar@example.com	Penulis

Dalam query di atas, dua bagian dari UNION terpisah oleh kata kunci UNION. Bagian pertama SELECT mengambil baris-baris dari tabel "data_pengguna" dengan kondisi usia < 30 tahun. Bagian kedua SELECT mengambil baris-baris dari tabel "data_pengguna" dengan kondisi pekerjaan = 'Mahasiswa'. Hasil UNION dari kedua SELECT akan menggabungkan dan menghapus duplikat baris dari kedua set data tersebut.

6. DIFFERENCE. Dalam MySQL, tidak ada perintah khusus yang disebut "DIFFERENT" untuk menggabungkan hasil yang berbeda antara dua set data. Namun, agar dapat mencapai hasil yang serupa dengan menggunakan perintah UNION dan subquery. Berikut adalah contoh query yang dapat digunakan untuk mendapatkan data pengguna yang memiliki pekerjaan yang berbeda dengan data pengguna lainnya:



```
1 SELECT id, nama, usia, alamat, email, pekerjaan FROM data_pengguna WHERE pekerjaan NOT IN (  
2     SELECT pekerjaan  
3     FROM data_pengguna  
4     GROUP BY pekerjaan  
5     HAVING COUNT(*) > 1);
```

Result Grid

	id	nama	usia	alamat	email	pekerjaan
▶	1	Budi Santoso	25	Jalan Merdeka No. 10	budisantoso@example.com	Mahasiswa
	4	Siti Rahayu	28	Jalan Diponegoro No. 8	sitirahayu@example.com	Desainer Grafis
	5	Rizky Pratama	50	Jalan Gatot Subroto No. 5	rizkypratama@example.com	Akuntan
	7	Eko Sutrisno	38	Jalan Veteran No. 18	ekosutrisno@example.com	Guru
	12	Maya Indah	41	Jalan HR Rasuna Said No. 22	mayaindah@example.com	Manajer
	13	Adi Wijaya	36	Jalan Cikini Raya No. 9	adiwijaya@example.com	Programmer
	14	Dewi Permata	26	Jalan Ahmad Dahlan No. 16	dewipermata@example.com	Konsultan
	16	Sinta Dewi	32	Jalan Kebon Sirih No. 11	sintadewi@example.com	Pengajar
	18	Fitriani Putri	30	Jalan Kuningan No. 6	fitrianiputri@example.com	Blogger
	20	Dewi Wulandari	34	Jalan Thamrin No. 23	dewiwulandari@example.com	Desainer Interior
*	NULL	NULL	NULL	NULL	NULL	NULL

Dalam query di atas, subquery yang ada di dalam klausa WHERE mengembalikan daftar pekerjaan yang memiliki lebih dari satu pengguna. Kemudian, query utama menggunakan klausa NOT IN untuk memfilter baris pengguna yang pekerjaannya tidak termasuk dalam hasil subquery tersebut. Hasil dari query di atas akan mengembalikan data pengguna yang memiliki pekerjaan yang berbeda dengan pengguna lainnya. Pastikan untuk menyesuaikan kolom dan kondisi yang sesuai dengan struktur tabel dan kebutuhan.

7. PRODUCT. Operasi PRODUCT menghasilkan setiap kemungkinan gabungan antara baris-baris dari kedua tabel tersebut. Berikut ini adalah contoh query yang melakukan operasi PRODUCT antara tabel "data_pengguna" dengan dirinya sendiri:

1 • `SELECT * FROM data_pengguna AS u1, data_pengguna AS u2;`

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	id	nama	usia	alamat	email	pekerjaan	id	nama	usia
▶	25	Agus Wijaya	40	Jalan Kebon Sirih No. 14	aguswijaya@example.com	Peneliti	1	Budi Santoso	25
	24	Anita Halim	31	Jalan MT Haryono No. 7	anitahalim@example.com	Pengacara	1	Budi Santoso	25
	23	Hendro Gunawan	43	Jalan HR Rasuna Said No. 21	hendrogunawan@example.com	Insinyur	1	Budi Santoso	25
	22	Laras Sekar	24	Jalan Sudirman No. 26	larassekar@example.com	Penulis	1	Budi Santoso	25
	21	Budi Setiawan	37	Jalan Asia Afrika No. 13	budisetiawan@example.com	Arsitek	1	Budi Santoso	25
	20	Dewi Wulandari	34	Jalan Thamrin No. 23	dewiwulandari@example.com	Desainer Interior	1	Budi Santoso	25
	19	Rahmat Widodo	49	Jalan Gatot Subroto No. 17	rahmatwido@example.com	Pengusaha	1	Budi Santoso	25
	18	Fitriani Putri	30	Jalan Kuningan No. 6	fitrianiputri@example.com	Blogger	1	Budi Santoso	25
	17	Andi Pratama	39	Jalan Tebet Barat No. 19	andipratama@example.com	Dokter	1	Budi Santoso	25
	16	Sinta Dewi	32	Jalan Kebon Sirih No. 11	sintadewi@example.com	Pengajar	1	Budi Santoso	25
	15	Arief Santoso	47	Jalan Thamrin No. 28	ariesantoso@example.com	Pengusaha	1	Budi Santoso	25
	14	Dewi Permata	26	Jalan Ahmad Dahlan No. 16	dewipermata@example.com	Konsultan	1	Budi Santoso	25
	13	Adi Wijaya	36	Jalan Cikini Raya No. 9	adiwijaya@example.com	Programmer	1	Budi Santoso	25
	12	Maya Indah	41	Jalan HR Rasuna Said No. 22	mayaindah@example.com	Manajer	1	Budi Santoso	25
	11	Feri Susanto	29	Jalan MT Haryono No. 14	ferisusanto@example.com	Peneliti	1	Budi Santoso	25

Dalam query di atas, kita menggunakan alias "u1" dan "u2" untuk mengidentifikasi kedua referensi tabel "data_pengguna". Query tersebut akan menghasilkan hasil gabungan (cartesian product) dari setiap baris pada "data_pengguna" dengan setiap baris lainnya dalam tabel yang sama.

8. DIVIDE. Dalam MySQL, operasi DIVIDE (atau juga dikenal sebagai division) tidak tersedia secara langsung. Operasi DIVIDE pada basis data relasional menghasilkan hasil yang berbeda dari operasi DIVIDE dalam matematika. Namun, agar dapat mencapai hasil yang mirip dengan operasi DIVIDE menggunakan kombinasi operasi JOIN, subquery, dan kondisi COUNT. Berikut adalah contoh query yang mengilustrasikan cara mendapatkan data_pengguna yang memiliki semua pekerjaan yang ada di tabel lain (misalnya, tabel "data_pekerjaan"):

```
1 • SELECT dp.id, dp.nama, dp.usia, dp.alamat, dp.email, dp.pekerjaan FROM data_pengguna dp
2 WHERE NOT EXISTS (SELECT * FROM data_pekerjaan
3 WHERE NOT EXISTS (SELECT * FROM data_pengguna WHERE pekerjaan = dp.pekerjaan));
```

Result Grid						
Filter Rows:						
Edit:						
Export/Import:						
Wrap Cell Content:						
id	nama	usia	alamat	email	pekerjaan	
1	Budi Santoso	25	Jalan Merdeka No. 10	budisantoso@example.com	Mahasiswa	
2	Rina Setiawati	35	Jalan Sudirman No. 15	rinasetiawati@example.com	Dokter	
3	Ahmad Hidayat	42	Jalan Imam Bonjol No. 20	ahmadhidayat@example.com	Pengusaha	
4	Siti Rahayu	28	Jalan Diponegoro No. 8	sitirahayu@example.com	Desainer Grafis	
5	Rizky Pratama	50	Jalan Gatot Subroto No. 5	rizkypratama@example.com	Akuntan	
6	Dewi Puspitasari	31	Jalan Pahlawan No. 12	dewipuspitasari@example.com	Penulis	
7	Eko Sutrisno	38	Jalan Veteran No. 18	ekosutrisno@example.com	Guru	
8	Anita Setyawati	27	Jalan Ahmad Yani No. 25	anitasetyawati@example.com	Arsitek	
9	Yusuf Kurniawan	45	Jalan Hayam Wuruk No. 30	yusufkurniawan@example.com	Insinyur	
10	Ratna Dewi	33	Jalan Asia Afrika No. 7	ratnadewi@example.com	Pengacara	
11	Feri Susanto	29	Jalan MT Haryono No. 14	ferisusanto@example.com	Peneliti	
12	Maya Indah	41	Jalan HR Rasuna Said No. 22	mayaindah@example.com	Manajer	
13	Adi Wijaya	36	Jalan Cikini Raya No. 9	adiwijaya@example.com	Programmer	
14	Dewi Permata	26	Jalan Ahmad Dahlan No. 16	dewipermata@example.com	Konsultan	
15	Arief Santoso	47	Jalan Thamrin No. 28	ariefsantoso@example.com	Pengusaha	
16	Sinta Dewi	32	Jalan Kehon Sirih No. 11	sintadewi@example.com	Pennaiar	

Dalam query di atas, subquery yang ada di dalam klausa WHERE mencari semua pekerjaan yang ada di tabel "data_pekerjaan". Kemudian, query utama menggunakan subquery lain untuk memeriksa apakah setiap pekerjaan tersebut ada dalam tabel "data_pengguna". Jika tidak ada data_pengguna dengan pekerjaan yang tidak cocok, baris tersebut akan dikeluarkan dari hasil.