

4. MySQL SELECT

Objektif :

Setelah menyelesaikan pelajaran ini, anda seharusnya mampu :

- Mengetahui konsep Select
- Mengetahui cara untuk menampilkan data dengan perintah select
- Dapat mengoperasikan tabel dengan perintah select dengan data dan kondisi tertentu



Sebelum melakukan Activity ini, pastikan database dan kedua tabel saat Activity MySQL DDL belum dihapus, jika sudah dihapus maka buat database dan table seperti pada Activity MySQL DDL, lalu isi tabel dengan

| jenis | harga |
|--------|-------|
| action | 2500 |
| drama | 2000 |
| horor | 2000 |
| komedi | 2000 |

| kode_film | jenis | judul | jml_keping | jml_film |
|-----------|--------|---------------|------------|----------|
| A01 | action | Spiderman 1 | 2 | 3 |
| D01 | drama | Love Story | 2 | 3 |
| H01 | horor | Suster Ngesot | 2 | 3 |

1. Menampilkan seluruh data dalam tabel

Langkah-langkah dapat dilihat pada pertemuan 4 pada video 1

Untuk menampilkan semua isi dan kolom tabel, kita gunakan perintah

```
SELECT * FROM <nama tabel>;
```

Berikut ini contoh perintah untuk menampilkan semua data pada tabel jenisfilm :

```
SELECT * FROM jenisfilm;
```

```
mysql> select * from jenisfilm;
+----+-----+
| jenis | harga |
+----+-----+
| action | 2500 |
| drama | 2000 |
| horor | 2000 |
| komedi | 2000 |
+----+-----+
4 rows in set (0.03 sec)
```

2. Menampilkan field – field tertentu

Langkah-langkah dapat dilihat pada pertemuan 4 pada video 2

Untuk menampilkan semua record pada *field – field* tertentu, kita gunakan perintah

```
SELECT <field 1>, <field 2>, <field 3> FROM <nama tabel>;
```

Berikut ini perintah untuk menampilkan data pada tabel jenisfilm dengan memilih kolom– kolom tertentu, pada kali ini, kolom yang ditampilkan adalah kolom jenis :

```
SELECT jenis FROM jenisfilm;
```

```
mysql> select jenis from jenisfilm;
+-----+
| jenis |
+-----+
| action |
| drama  |
| horror |
| komedi |
+-----+
4 rows in set (0.00 sec)
```

3. Menampilkan data dengan kondisi tertentu

Langkah-langkah dapat dilihat pada pertemuan 4 pada video 3

Data pada tabel, tidak hanya dapat dipilih berdasarkan nama kolomnya saja, akan tetapi dipilih berdasarkan nilai suatu field dengan menggunakan operator kondisi. Perintah yang digunakan adalah

```
SELECT * FROM <nama tabel> WHERE <kondisi>;
```

Berikut ini perintah untuk menampilkan data pada tabel jenisfilm dimana nilai pada kolom jenis adalah drama :

```
SELECT * FROM jenisfilm WHERE jenis='drama' ;
```

```
mysql> select * from jenisfilm where jenis='drama';
+-----+-----+
| jenis | harga |
+-----+-----+
| drama | 20000 |
+-----+-----+
1 row in set (0.03 sec)
```

Macam – macam operator perbandingan yang dikenal MySQL antara lain :

- Lebih besar (>)
- Lebih kecil (<)
- Tidak kurang dari / lebih besar dan sama dengan (>=)
- Tidak lebih dari / lebih kecil dan sama dengan (<=)
- Sama dengan (=)
- Tidak sama dengan (<>)

4. Menampilkan data dengan operator AND

Langkah-langkah dapat dilihat pada pertemuan 4 pada video 4

Untuk menggunakan menggabungkan dua kondisi sekaligus dan menginginkan agar kedua kondisi tersebut harus tercapai untuk ditampilkan, maka gunakan perintah AND di antara kedua syarat tersebut, bentuk umum perintahnya adalah

```
SELECT * FROM <nama tabel> WHERE <kondisi 1> AND <kondisi 2>;
```

Contohnya seperti perintah berikut :

```
SELECT * FROM jenisfilm WHERE harga>=1000 AND harga<=3000;
```

```
mysql> select * from jenisfilm where harga>=1000 and harga<=3000;
+----+-----+
| jenis | harga |
+----+-----+
| action | 2500 |
| drama | 1000 |
| komedi | 3000 |
+----+-----+
3 rows in set (0.00 sec)
```

5. Menampilkan data dengan operator BETWEEN ... AND ...

Langkah-langkah dapat dilihat pada pertemuan 4 pada video 5

Untuk menampilkan data berdasarkan *range* nilai angka kita gunakan perintah BETWEEN ... AND... Bentuk umum perintahnya adalah

```
SELECT * FROM <nama tabel> WHERE <field acuan>
BETWEEN <nilai rendah> AND <nilai tinggi>;
```

Perintah ini sama halnya dengan perintah berikut

```
SELECT * FROM <nama tabel> WHERE <field acuan> >= <nilai rendah
```

Contoh penggunaan perintah BETWEEN... AND... adalah sebagai berikut :

```
SELECT * FROM jenisfilm WHERE harga BETWEEN 1000 AND 2000;
```

```
mysql> select * from jenisfilm where harga between 1000 and 3000;
+----+-----+
| jenis | harga |
+----+-----+
| action | 2500 |
| drama | 1000 |
| komedi | 3000 |
+----+-----+
3 rows in set (0.00 sec)
```

6. Menampilkan data dengan operator OR

Langkah-langkah dapat dilihat pada pertemuan 4 pada video 6

Untuk menggunakan menggabungkan dua kondisi sekaligus dan menginginkan agar minimal salah satu kondisi tersebut harus tercapai untuk ditampilkan, maka gunakan perintah OR di antara kedua syarat tersebut, bentuk umum perintahnya adalah

```
SELECT * FROM <nama tabel> WHERE <kondisi 1> OR <kondisi 2>;
```

Contohnya seperti perintah berikut :

```
SELECT * FROM jenisfilm WHERE harga>=1000 OR harga<=3000;
```

```
mysql> select * from jenisfilm where harga=1000 or harga=3000;
+----+-----+
| jenis | harga |
+----+-----+
| drama | 1000 |
| komedi | 3000 |
+----+-----+
2 rows in set (0.00 sec)
```

7. Menampilkan data dengan operator IN

Langkah-langkah dapat dilihat pada pertemuan 4 pada video 7

Untuk menampilkan data – data yang memiliki beberapa nilai tertentu, kita dapat gunakan perintah IN, bentuk umum perintahnya adalah

```
SELECT * FROM <nama tabel> IN (<nilai 1>, <nilai 2>, <nilai n>);
```

Contohnya seperti perintah berikut :

```
SELECT * FROM jenisfilm where harga IN (1000, 2000);
```

```
mysql> select * from jenisfilm where harga in (1000,3000);
+-----+-----+
| jenis | harga |
+-----+-----+
| drama | 1000  |
| komedi | 3000  |
+-----+-----+
2 rows in set (0.05 sec)
```

8. Menampilkan data dengan operator NOT

Langkah-langkah dapat dilihat pada pertemuan 4 pada video 8

Untuk membalikkan hasil kondisi, kita dapat gunakan perintah NOT sebelum kondisi yang akan dibalikkan hasilnya. Berikut adalah contoh penggunaan perintah NOT :

```
SELECT * FROM jenisfilm where NOT jenis='action' ;
```

```
mysql> select * from jenisfilm where not jenis='action';
+-----+-----+
| jenis | harga |
+-----+-----+
| drama | 1000  |
| horror | 4000  |
| komedi | 3000  |
+-----+-----+
3 rows in set (0.00 sec)
```

9. Menampilkan data dengan operator <>

Langkah-langkah dapat dilihat pada pertemuan 4 pada video 9

Untuk menampilkan semua data kecuali suatu data dengan nilai tertentu, kita dapat gunakan operator tidak sama dengan (<>). Berikut adalah contoh penggunaan operator tidak sama dengan :

```
SELECT * FROM jenisfilm WHERE jenis <> 'action' ;
```

```
mysql> select * from jenisfilm where jenis<>'action';
+-----+-----+
| jenis | harga |
+-----+-----+
| drama | 1000  |
| horror | 4000  |
| komedi | 3000  |
+-----+-----+
3 rows in set (0.01 sec)
```

10. Menampilkan data dengan menggunakan operator LIKE

Langkah-langkah dapat dilihat pada pertemuan 4 pada video 10

Bila kita ingin menyeleksi data string yang mengandung karakter tertentu pada posisi tertentu, kita gunakan operator LIKE. Kita dapat menggunakan karakter persen (%) sebagai karakter yang menjadi nilai bebas pencarian. Berikut adalah contoh penggunaan operator LIKE :

```
SELECT * FROM film WHERE judul LIKE 'S%';
```

```
mysql> select * from film where judul like 'S%';
```

| kode_film | jenis | judul | jml_keping | jml_film |
|-----------|--------|---------------|------------|----------|
| A01 | action | Spiderman 1 | 2 | 3 |
| H01 | horor | Suster Ngesot | 2 | 3 |

```
2 rows in set (0.39 sec)
```

11. Menampilkan data dengan memberikan nama lain

Langkah-langkah dapat dilihat pada pertemuan 4 pada video 11

Kita dapat menampilkan field dengan merubah penamaan *field* tersebut dengan perintah AS setelah nama *field* yang akan diubah namanya saat pemanggilan. Perubahan nama ini tidak akan mempengaruhi nama *field* aslinya pada tabel. Bentuk umum perintahnya adalah :

```
SELECT <nama field 1> AS <nama baru field 1>, <nama field 2> AS <nama baru field 2>,  
<nama field n> AS <nama baru field n> FROM <nama tabel>;
```

Berikut contoh penggunaan perintah AS :

```
SELECT jenis AS jenis_film FROM jenisfilm;
```

```
mysql> select jenis as jenis_film from jenisfilm;
```

| jenis_film |
|------------|
| action |
| drama |
| horor |
| komed |

```
4 rows in set (0.00 sec)
```

12. Menampilkan data dengan menggunakan nama lain table

Langkah-langkah dapat dilihat pada pertemuan 4 pada video 12

Kita dapat menampilkan data dengan memberi nama alias pada sebuah tabel dengan menambahkan nama alias tabel setelah nama asli tabel, kemudian setiap nama *field* pada tabel tersebut yang dipanggil harus ditambahkan nama alias tabel tersebut diikuti dengan tanda titik sebelum nama *field* tersebut. Perubahan nama ini tidak akan mempengaruhi nama asli tabel. Bentuk umum perintahnya adalah :

```
SELECT <alias tabel>.<nama field 1>,  
<alias tabel>.<nama field 2>, <alias tabel>.<nama field n>  
FROM <nama tabel> <alias tabel>;
```

Berikut contoh penggunaan penamaan alias tabel :

```
SELECT j.jenis, j.harga FROM jenisfilm j;
```

```
mysql> select j.jenis, j.harga from jenisfilm j;  
+-----+-----+  
| jenis | harga |  
+-----+-----+  
| action | 2500 |  
| drama | 2000 |  
| horror | 2000 |  
| komedi | 2000 |  
+-----+-----+  
4 rows in set (0.00 sec)
```


13. Menampilkan data dari dua table sekaligus

Langkah-langkah dapat dilihat pada pertemuan 4 pada video 13

Kita dapat menampilkan beberapa tabel sekaligus dan menggabungkannya dalam satu tabel dengan mengetik beberapa nama tabel yang diikuti dengan tanda koma. Berikut adalah contohnya :

```
SELECT * FROM jenisfilm, film;
```

```
mysql> select * from jenisfilm,film;
+-----+-----+-----+-----+-----+-----+-----+
| jenis | harga | kode_film | jenis | judul          | jml_keping | jml_film |
+-----+-----+-----+-----+-----+-----+-----+
| action | 2500 | A01       | action | Spiderman 1    | 2          | 3        |
| action | 2500 | D01       | drama  | Love Story     | 2          | 3        |
| action | 2500 | H01       | horror | Suster Ngesot  | 2          | 3        |
| drama  | 2000 | A01       | action | Spiderman 1    | 2          | 3        |
| drama  | 2000 | D01       | drama  | Love Story     | 2          | 3        |
| drama  | 2000 | H01       | horror | Suster Ngesot  | 2          | 3        |
| horror | 2000 | A01       | action | Spiderman 1    | 2          | 3        |
| horror | 2000 | D01       | drama  | Love Story     | 2          | 3        |
| horror | 2000 | H01       | horror | Suster Ngesot  | 2          | 3        |
| komedi | 2000 | A01       | action | Spiderman 1    | 2          | 3        |
| komedi | 2000 | D01       | drama  | Love Story     | 2          | 3        |
| komedi | 2000 | H01       | horror | Suster Ngesot  | 2          | 3        |
+-----+-----+-----+-----+-----+-----+-----+
12 rows in set (0.00 sec)
```