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



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



kode_film	jenis	judul	jml_keping	jml_film		
A01	action	Spiderman 1	2	3		
DØ1	drama	Love Story	1 2	3		
		Suster Ngesot	1 2	3		

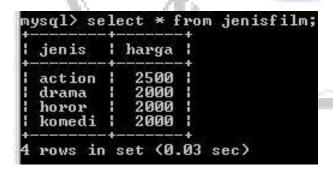
1. Menampilkan se<mark>luruh d</mark>ata dalam tabel

Langkah-langkah dapat dilihat pada pertemuan 4 pada video 1

Untuk menampilkan semua isi dan kolom tabel, kita gunakan perintah

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

SELECT * FROM jenisfilm;



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 |
| horor |
| komedi |
+ rows in set (0.00 sec)
```

3. <u>Menampilkan data dengan kondisi tertentu</u>

Langkah-langkah dapat dili<mark>hat p</mark>ada 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 | 2000 |
| 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;
 nysql> select * from jenisfilm where harga>=1000 and harga<=3000;
  jenis
         | harga
  action
  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</pre>

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

SELECT * FROM jenisfilm WHERE harga BETWEEN 1000 AND 2000;



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;



7. <u>Menampilkan data dengan operator IN</u>

Langkah-langkah dapat dilihat pada pertemuan 4 pada video 7

Untuk menampilkan data – data yang memiliki beberapa niai 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);

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:

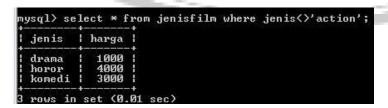
SELECT * FROM jenisfilm where NOT jenis='action'

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';



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%';

kode_film	jenis	¦ judul	jml_keping	jml_film		
AØ1	action	Spiderman 1	2	3		
		Suster Ngesot	1 2	1 3		

11. Menampilkan data dengan memberikan nama lain

Langkah-langkah dapat dilihat p<mark>ada pe</mark>rtemuan 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 |
| komedi |
| trows 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;</pre>
```

Berikut contoh penggunaan penamaan alias tabel:

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

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;

jenis	1	harga	1	kode_film	1	jenis	1	judu1	1	jml_keping	1	jml_film	
action	î	2500		AØ1	î	action		Spiderman 1		2	î	3	
action	्र	2500		DØ1	3	drama	1	Love Story	1	2		3	
action		2500		HØ1	9	horor	1	Suster Ngesot	ì	2		3	
drama		2000		AØ1	3	action	į	Spiderman 1	3	2	्	3	
drama	<u> </u>	2000	i	DØ1		drama	i	Love Story		2	i	3	
drama	3	2000		HØ1		horor	3	Suster Ngesot		2	ं	3	
horor	3	2000	-	A01	3	action	्	Spiderman 1	3	2	ं	3	
horor	3	2000		DØ1	3	drama	्	Love Story	्	2	्	3	
horor	्	2000	i	HØ1	3	horor	1	Suster Ngesot	3	2		3	
komedi	ं	2000		AØ1	3	action	į	Spiderman 1	3	2	ं	3	
komedi	्र	2000	-	DØ1	्	drama	्	Love Story	्र	2	ं	3	
komedi	្ន	2000	1	HØ1		horor	i	Suster Ngesot	1	2	1	3	