

**TUGAS 8**  
**RUMUS PERHITUNGAN DAN VIEW**



**Adam Arthur Faizal**

**M3119001**

**TI A**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**SEKOLAH VOKASI**

**UNIVERSITAS SEBELAS MARET**

**SURAKARTA**

**2020**

```
mysql> select nama_merk, harga * id_jenis from barang join merk_barang;
```

nama_merk	harga * id_jenis
Gudang Garam	10000
Gudang Garam	10000
Gudang Garam	10000
Gudang Gula	10000
Gudang Gula	10000
Gudang Gula	10000
Gudang Beras	10000
Gudang Beras	10000
Gudang Beras	10000

```
9 rows in set (0,00 sec)
```

Jika dilihat lebih teliti, terdapat operasi aritmetika pada query diatas, dimana data-data yang terdapat pada atribut harga dikalikan dengan data-data yang ada pada atribut id\_jenis.

```
mysql> select nama_merk, harga * id_jenis as modal_per_item_barang from barang join merk_barang;
```

nama_merk	modal_per_item_barang
Gudang Garam	10000
Gudang Garam	10000
Gudang Garam	10000
Gudang Gula	10000
Gudang Gula	10000
Gudang Gula	10000
Gudang Beras	10000
Gudang Beras	10000
Gudang Beras	10000

```
9 rows in set (0,00 sec)
```

Jika dilihat lebih teliti, juga terdapat operasi aritmetika pada query diatas, dimana data-data yang terdapat pada atribut harga dikalikan dengan data-data yang ada pada atribut id\_jenis dan diberi alias dengan nama modal\_per\_item\_barang

```
mysql> select sum(harga * id_jenis) as total_modal from barang;
```

total_modal
30000

```
1 row in set (0,00 sec)
```

Jika dilihat lebih teliti, terdapat operasi aritmetika yang dibungkus dengan fungsi agregasi serta diberi alias dengan nama total\_modal pada tabel barang.

```
mysql> select id_jenis, sum(harga * id_jenis) as total_jenis from barang group by id_jenis;
+-----+-----+
| id_jenis | total_jenis |
+-----+-----+
|      1 |      30000 |
+-----+-----+
1 row in set (0,00 sec)
```

```
mysql> select id_jenis, avg(harga * id_jenis) as rata_modal_per_jenis from barang group by id_jenis;
+-----+-----+
| id_jenis | rata_modal_per_jenis |
+-----+-----+
|      1 |      10000.0000 |
+-----+-----+
1 row in set (0,00 sec)
```

```
mysql> select id_jenis, count(id_barang) from barang group by id_jenis having count(id_jenis) > 0;
+-----+-----+
| id_jenis | count(id_barang) |
+-----+-----+
|      1 |      3 |
+-----+-----+
1 row in set (0,01 sec)
```

Query diatas juga memiliki operasi aritmetika yang dibungkus dengan fungsi agregasi

```
mysql> select nama_merk, harga, case when harga > '10000' then 'mahal'
-> else 'murah' end as status_harga from barang join merk_barang;
```

nama_merk	harga	status_harga
Gudang Garam	10000	murah
Gudang Garam	10000	murah
Gudang Garam	10000	murah
Gudang Gula	10000	murah
Gudang Gula	10000	murah
Gudang Gula	10000	murah
Gudang Beras	10000	murah
Gudang Beras	10000	murah
Gudang Beras	10000	murah

```
9 rows in set (0,00 sec)
```

Query diatas berfungsi untuk membuat sebuah view yang bisa kita manipulasi layaknya tabel biasa

```
mysql> create view barang2 as select nama_merk, harga, case
-> when harga > '10000' then 'mahal' else 'murah' end
-> as status_harga from barang join merk_barang;
```

Query OK, 0 rows affected (0,26 sec)

```
mysql> select * from barang2;
```

nama_merk	harga	status_harga
Gudang Garam	10000	murah
Gudang Garam	10000	murah
Gudang Garam	10000	murah
Gudang Gula	10000	murah
Gudang Gula	10000	murah
Gudang Gula	10000	murah
Gudang Beras	10000	murah
Gudang Beras	10000	murah
Gudang Beras	10000	murah

```
9 rows in set (0,01 sec)
```