

1. SELECT

SELECT berfungsi untuk menampilkan data pada table secara keseluruhan.

Syntax Dasar :

```
/** Syntax dasar SELECT **/  
SELECT * FROM nama_tabel  
/** ATAU **/  
/** Syntax dasar SELECT **/  
SELECT nama_field, nama_field, nama_field FROM nama_tabel
```

Contoh Penggunaan :

```
SELECT * FROM admlink;  
/** penggunaan tanda * (bintang) artinya digunakan memilih semua field **/  
  
SELECT nama, kota FROM admlink;  
/** nama dan kota merupakan sebuah nama_field **/  
  
Distinct – Digunakan untuk menampilkan daftar isi yang ada pada field yang diinginkan  
SUM – Digunakan untuk menjumlahkan isi yang ada pada field yang diinginkan  
Max – Digunakan untuk menampilkan nilai maximum pada field yang diinginkan  
Min – Digunakan untuk menampilkan nilai minimum pada field yang diinginkan
```

Contoh Penggunaan :

```
SELECT Distinct (nama) FROM admlink  
SELECT Sum (Harga) FROM admlink  
SELECT Max (Usia) FROM admlink  
SELECT Min (nomer) FROM admlink
```

2. WHERE

where memiliki fungsi untuk memfilter data/record. penggunaan **select** dapat menampilkan keseluruhan data, sedangkan dengan menambah perintah **where** setelah **select** maka data yang ditampilkan dapat difilter/disaring. Dengan menggunakan **where** teman-teman juga dapat menggunakan beberapa **operator** layaknya di bahasa pemrograman seperti :

- Operator Sama Dengan =
- Operator Tidak Sama Dengan <>
- Operator Lebih Besar Dari >
- Operator Lebih Kecil Dari <
- Operator Lebih Besar Sama Dengan >=
- Operator Lebih Kecil Sama Dengan <=
- Berdasarkan jangkauan nilai **BETWEEN**
- Dan untuk memfilter berdasarkan pola huruf yang sama **LIKE**

Syntax Dasar :

```
SELECT * FROM nama_tabel WHERE nama_field = 'nilai'
```

Contoh Penggunaan :

```
Menampilkan customers yang memiliki id 1  
SELECT * FROM customers WHERE id = '1';  
/** query diatas untuk menampilkan customers yang memiliki id 1 **/  
Menampilkan customers yang tidak memiliki nama 'bagus mantonafi'  
SELECT * FROM customers WHERE nama <> 'Bagus Mantonafi';  
/** query diatas untuk menampilkan customers yang memiliki nama selain 'bagus mantonafi' **/
```

3. INSERT

Untuk memasukan atau menambah record/data baru ke table

Syntax Dasar :

```
Tanpa mendefinisikan nama field  
INSERT INTO nama_tabel VALUES('nilai_masukan', 'nilai_masukan', 'dst');
```

Dengan mendefinisikan nama field

```
INSERT INTO nama_tabel('nama_field', 'nama_field', 'dst' )
VALUES('nilai_masukan', 'nilai_masukan', 'dst');
```

Contoh Penggunaan :

Memasukan data berikut ini ke tabel :

nama :Sholahudin
kota :Jakarta
negara:Indonesia

Tanpa mendefinisikan nama field

```
INSERT INTO customers VALUES(' ', 'Sholahudin', 'Jakarta', 'Indonesia');
```

Dengan mendefinisikan nama field

```
INSERT INTO customers('nama', 'kota', 'negara')
VALUES('Sholahudin', 'Jakarta', 'Indonesia');
```

4. UPDATE

Memperbarui atau memperbaiki record/data yang sudah ada pada tabel.

```
UPDATE customers
SET nama='Hendry'
WHERE id = '4';
```

Setiap mengubah data dengan klausa **UPDATE** diharapkan menambah klausa **WHERE** di akhir. Tujuannya agar tidak mengubah seluruh record. Silahkan mencoba tanpa klausa **WHERE** agar tahu dampak yang ditimbulkan

5. DELETE

Penggunaan **DELETE** digunakan untuk menghapus record pada tabel. baik itu satu data/record atau lebih.

```
DELETE FROM customers
WHERE negara = 'USA';
```

Sama halnya klausa **UPDATE**, klausa **DELETE** juga harus diakhiri dengan **WHERE** penggunaan klausa **DELETE** tanpa **WHERE** dapat menyebabkan seluruh record yang telah dimasukan ke tabel terhapus

SOAL!!!!

Table Admlink

ID	Voucher	Remarks	ADMNO	AMOUNT	A_PolicyNo	BILLDATE
D10BC00001	000001/DN/10/06/16	BANK CENTRAL ASIA	2	4525000	11002011600001	2016-06-10
T10NK00001	000003/DN/10/06/16	PT NIPPON KAIJI KENTEI KYOKAI IND	1	13961644	11001011600001	2016-06-17
D10BC00001	000004/DN/10/06/16	BANK CENTRAL ASIA	5	4525000	11002011600002	2016-06-16
O10AT00001	000006/DN/10/06/16	BANK CENTRAL ASIA	4	4000000	11002011600002	2016-06-22
T10PA00001	000007/DN/10/06/16	PT SATRIA DHARMA PUSAKA	6	6025000	11001101600001	2016-06-16
D10CT00001	000010/DN/10/06/16	PT CARE TECHNOLOGIES	7	11250000	11015031600001	2016-06-20
D10CT00001	000011/DN/10/06/16	PT CARE TECHNOLOGIES	10	11250000	11015031600001	2016-06-20
D10BC00001	000014/DN/10/06/16	BANK CENTRAL ASIA	9	10000000	11002011600001	2016-06-21
D10BC00001	000016/DN/10/06/16	BANK CENTRAL ASIA	8	9500000	11002011600001	2016-06-21
D10CT00001	000038/DN/11/06/16	CARE TECHNOLOGIES	3	6603000	11002011600003	2016-06-29

Table Voucher

Voucher	Remarks	Branch
000001/CN/10/06/16	BANK CENTRAL ASIA	10
000006/CN/10/11/16	PT BAHTERA ARTHAGUNA PARAMA	10
000204/DN/10/08/16	PT NIPPON KAIJI KENTEI KYOKAI IND	10
000040/DN/10/09/16	PT ASURANSI TRI PAKARTA QQ PT CARE TECHNOLOGIES	10
000025/CN/11/09/16	BANK CENTRAL ASIA	11
000056/DN/10/09/16	BANK CENTRAL ASIA	10
000031/CN/10/09/16	PT PRIMA ADJUSTERINDO MANDIRI	10
000072/DN/10/09/16	PT BAHTERA ARTHAGUNA PARAMA QQ PT JAPENANSI	10
000171/DN/10/08/16	BANK CENTRAL ASIA	10
000200/CN/10/08/16	BANK CENTRAL ASIA	10

1. Tampilkan data dengan kondisi "admno = 1" dari table Admlink
2. Tampilkan data dengan kondisi "admno = 1, 2, 3" dari table Admlink
3. Tampilkan hanya field voucher, docno, dan amount pada table admlink
4. Tampilkan data dengan kondisi "ID = D10BC00001" dari table Admlink
5. Tampilkan data yang memiliki jumlah amount diatas 5.000.000 dari table Admlink
6. Tampilkan data yang memiliki jumlah amount dibawah 5.000.000 dari table Admlink
7. Tampilkan data yang memiliki jumlah amount diantara 5.000.000 hingga 10.000.000 dari table Admlink
8. Tampilkan data dengan kondisi pada field "voucher" yang memiliki karakter "CN" saja dari table Admlink
9. Tampilkan data dengan kondisi tanggal "Billdate" paling lama dari table Admlink
10. Tampilkan hanya field ID, Voucher, Docno, Amount, dan Remarks dengan kondisi "ID = T10NK00001 " yang pada field "voucher" yang memiliki karakter "DN" saja dan jumlah amount diatas 5.000.000 dari table Admlink
11. Tampilkan data dengan field Voucher, Branch, ID, Amount, Billdate dari table Admlink dan table Voucher