

# **PERTEMUAN 5**

## **MENGGUNAKAN PERINTAH SQL SELECT DAN PARAMETER**



**Microsoft Office**

# **ACCESS**

**STMIK ANTAR BANGSA**

**2022**

## I. PENDAHULUAN

SQL adalah bahasa komputer untuk bekerja dengan sekumpulan fakta dan hubungan di antara keduanya. Program database relasional, seperti Microsoft Office Access, gunakan SQL untuk bekerja dengan data. Tidak seperti banyak bahasa komputer, SQL tidak sulit untuk dibaca dan dipahami, bahkan untuk pemula. Seperti banyak bahasa komputer, SQL adalah standar internasional yang dikenali oleh badan standar seperti ISO dan ANSI.

Anda menggunakan SQL untuk menjelaskan kumpulan data yang bisa membantu Anda menjawab pertanyaan. Saat Anda menggunakan SQL, Anda harus menggunakan sintaks yang benar. Sintaks adalah sekumpulan aturan dimana elemen bahasa digabungkan dengan benar. Sintaks SQL didasarkan pada sintaks bahasa Inggris, dan menggunakan banyak elemen yang sama seperti sintaks Visual Basic for Applications (VBA).

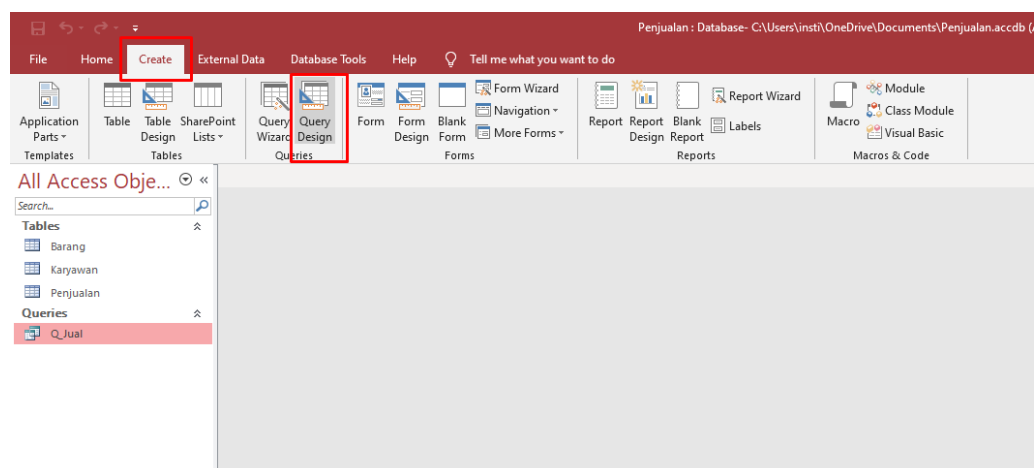
Saat Anda ingin mengambil data dari database, Anda meminta data dengan menggunakan bahasa kueri terstruktur, atau SQL. SQL adalah bahasa komputer yang sangat mirip dengan bahasa Inggris, tetapi program database tersebut memahaminya. Setiap kueri yang Anda jalankan menggunakan SQL di balik layar.

Memahami cara kerja SQL bisa membantu Anda membuat kueri yang lebih baik, dan dapat memudahkan Anda memahami cara memperbaiki kueri yang tidak mengembalikan hasil yang Anda inginkan.

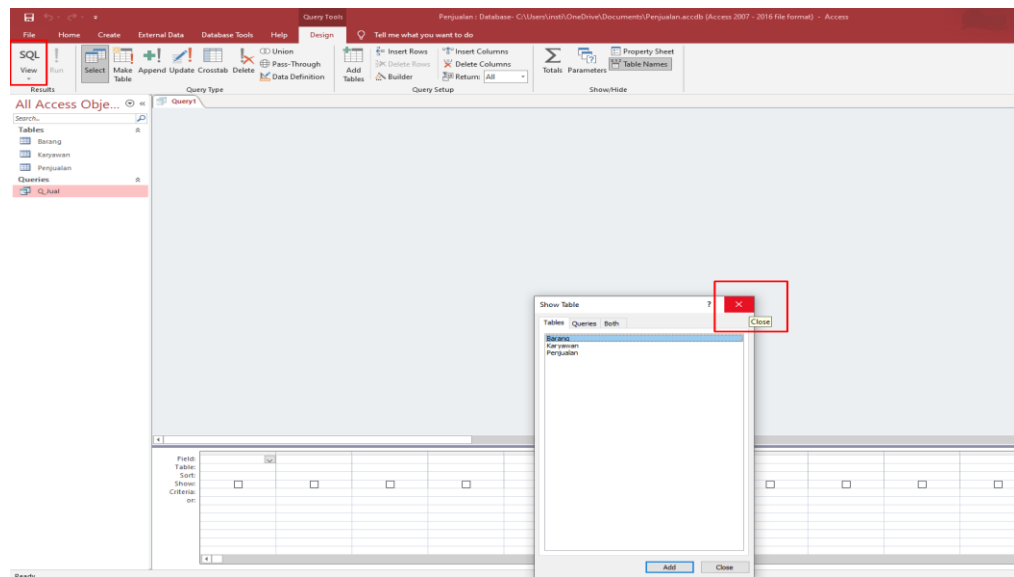
## II. PERINTAH SELECT

Perintah SELECT adalah perintah untuk menampilkan data yang ada di dalam suatu tabel, yang mana dalam pengembangannya perintah ini akan menjadi sebuah perintah yang sangat penting dan berpengaruh hingga saat pemrograman di stored procedures dan triggers. Sebelum anda melakukan perintah SQL, terlebih dahulu anda lakukan hal berikut ini untuk membuka perintah SQL tersebut.

### 1. Pilih Create lalu klik pada Query Design

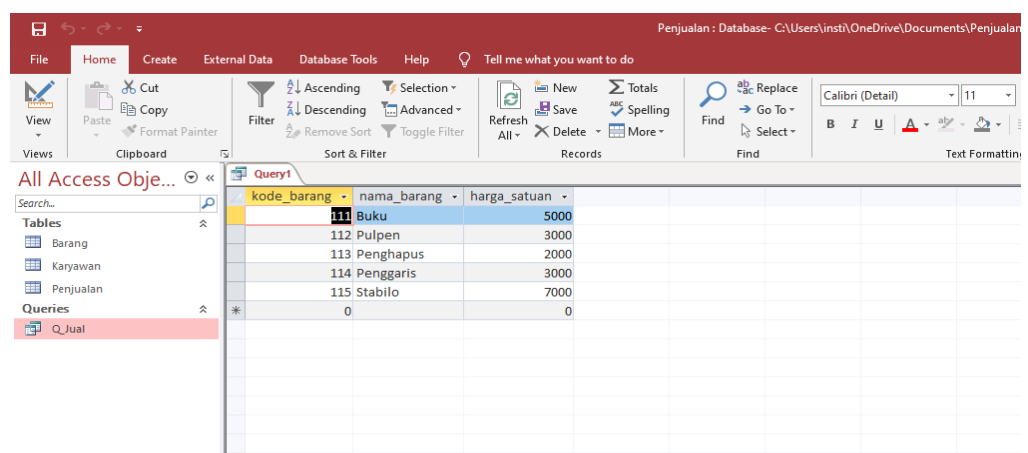
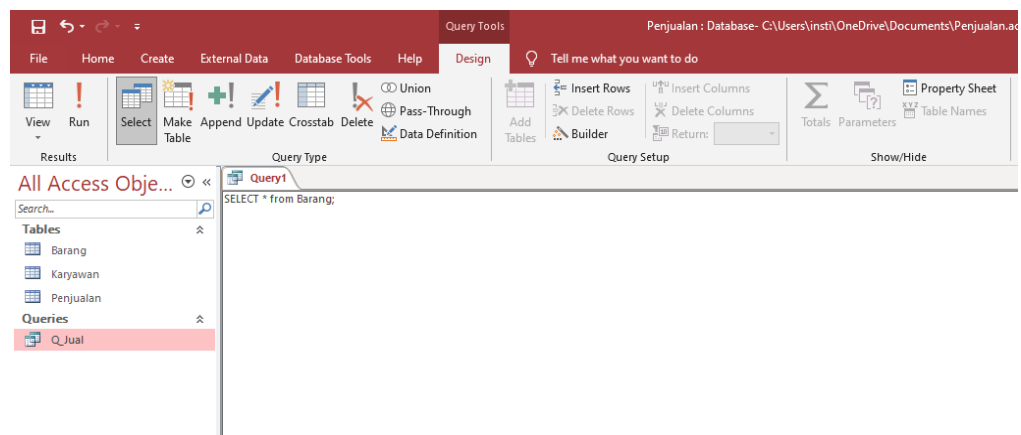


2. Tutup tampilan Show Table, lalu klik pada SQL View



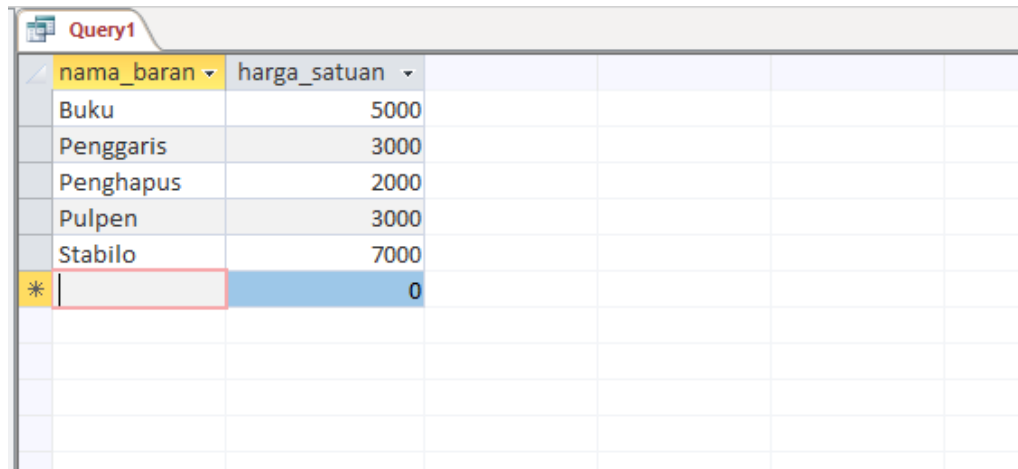
3. Disinilah anda bisa membuat perintah SQL, penulisan huruf besar kecil pada nama table dan fiel harus sesuai.

Masukan kode SQL **Select \* from Barang** untuk menampilkan seluruh data yang ada pada tabel Barang, kemudian klik **Run !** maka akan tampil hasilnya.



4. Menggunakan Order By Ascending atau Descending untuk mengurutkan nama atau angka berdasarkan dari yang tertinggi ke rendah atau sebaliknya. begitu juga dengan huruf, mengurutkan dari A -> Z (ASC) atau dari Z -> A (DESC).

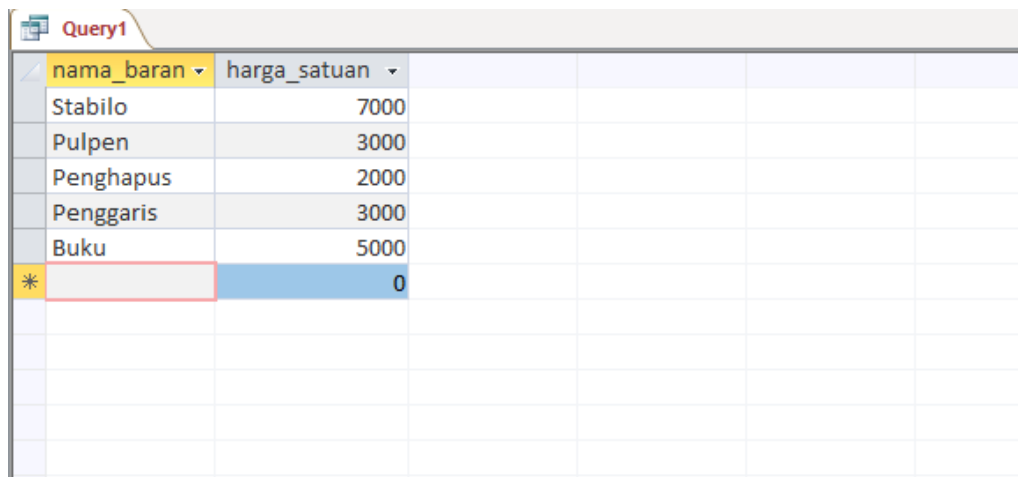
```
SELECT Barang.nama_barang, Barang.harga_satuan FROM Barang  
ORDER BY Barang.nama_barang ASC;
```



The screenshot shows a database query result in a table with two columns: 'nama\_barang' and 'harga\_satuan'. The data is sorted by 'nama\_barang' in ascending order. The items listed are Buku (5000), Penggaris (3000), Penghapus (2000), Pulpen (3000), and Stabilo (7000). A new row with an asterisk and a value of 0 is at the bottom.

nama_barang	harga_satuan
Buku	5000
Penggaris	3000
Penghapus	2000
Pulpen	3000
Stabilo	7000
*	0

```
SELECT Barang.nama_barang, Barang.harga_satuan FROM Barang  
ORDER BY Barang.nama_barang ASC;
```



The screenshot shows a database query result in a table with two columns: 'nama\_barang' and 'harga\_satuan'. The data is sorted by 'nama\_barang' in ascending order. The items listed are Stabilo (7000), Pulpen (3000), Penghapus (2000), Penggaris (3000), and Buku (5000). A new row with an asterisk and a value of 0 is at the bottom.

nama_barang	harga_satuan
Stabilo	7000
Pulpen	3000
Penghapus	2000
Penggaris	3000
Buku	5000
*	0

### III. PERINTAH WHERE

Setelah membahas fungsi SELECT pada materi sebelumnya, kali ini kita akan bahas fungsi WHERE yang digunakan untuk menyatakan kondisi atau perintah yang diinginkan berdasarkan Field atau Record.

#### 1. WHERE >

```
SELECT * from Barang where harga_satuan>2000;
```

Query1		
kode_barang	nama_barang	harga_satuan
111	Buku	5000
112	Pulpen	3000
114	Penggaris	3000
115	Stabilo	7000
*	0	0

#### 2. WHERE >=

```
SELECT * from Barang where harga_satuan>=5000;
```

Query1		
kode_barang	nama_barang	harga_satuan
111	Buku	5000
115	Stabilo	7000
*	0	0

### 3. WHERE =

```
SELECT * from Barang where harga_satuan=7000;
```

Query1			
kode_barang	nama_barang	harga_satuan	
115	Stabilo	7000	
*	0	0	

### 4. WHERE <

```
SELECT * from Penjualan where jumlah_beli<2;
```

Query1					
kode_karyawan	nama_karyawan	kode_barang	nama_barang	harga_satuan	jumlah_beli
221	Budi	111	Buku	2000	1
223	Sahrul	113	Penghapus	2000	1
225	Santi	115	Stabilo	7000	1
*	0	0		0	0

### 5. WHERE <=

```
SELECT * from Penjualan where jumlah_beli<=2;
```

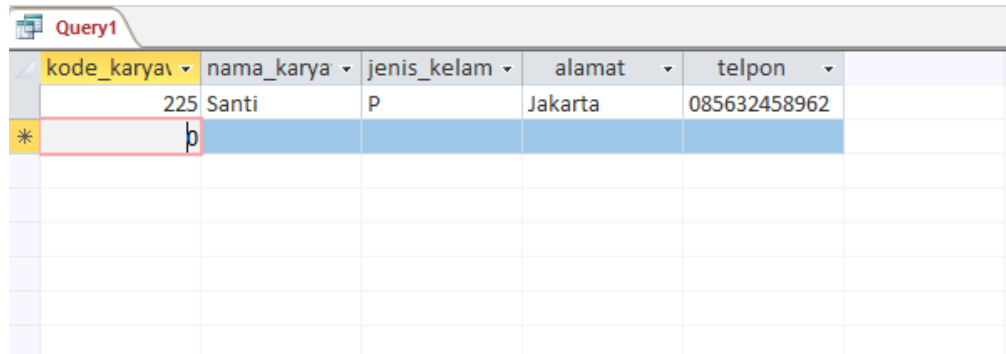
Query1					
kode_karyawan	nama_karyawan	kode_barang	nama_barang	harga_satuan	jumlah_beli
221	Budi	111	Buku	2000	1
222	Putri	112	Pulpen	3000	2
223	Sahrul	113	Penghapus	2000	1
224	Joko	114	Penggaris	3000	2
225	Santi	115	Stabilo	7000	1
*	0	0		0	0

#### IV. FUNGSI SQL PARAMETER AND, OR, & NOT

##### 1. FUNGSI AND

Fungsi AND menampilkan data yang kedua kondisinya harus benar-benar ada pada data yang telah dibuat.

```
SELECT * from Karyawan WHERE jenis_kelamin='P' AND  
alamat='Jakarta';
```



kode_karyav	nama_karya	jenis_kelam	alamat	telpon
225	Santi	P	Jakarta	085632458962
*	p			

##### 2. FUNGSI OR

Penambahan OR akan menampilkan data yang salah satu kondisinya benar maka dia akan tetap bisa menampilkan data melalui salah satu fungsi yang benar tersebut.

```
SELECT * from Karyawan WHERE jenis_kelamin='L' OR  
alamat='Bekasi';
```



kode_karyav	nama_karya	jenis_kelam	alamat	telpon
221	Budi	L	Jakarta	089694415421
223	Sahrul	L	Bekasi	085654545650
224	Joko	L	Tangerang	088963547500
*	0			

### 3. FUNGSI NOT

Fungsi NOT ini untuk menampilkan data selain yang kita tulis pada kode SQL. Disini ditulis NOT harga\_satuan=3000. Berarti tidak akan menampilkan data pada Field harga\_satuan yang bernilai 3000.

```
SELECT * from Barang WHERE NOT harga_satuan=3000;
```

Query1				
	kode_barang	nama_barang	harga_satuan	
	111	Buku	5000	
	113	Penghapus	2000	
	115	Stabilo	7000	
*	0		0	