

**NAMA : ARMAN SYAM**  
**NIM : 200250502017**  
**JURUSAN/KELAS : TEKNIK INFORMATIKA A**  
**ANGKATAN : 2020**  
**DOSEN : MUSLIADI KH, S.Kom., M.Kom**  
**MATA KULIAH : SISTEM BASIS DATA**

### 1. SQL Tutorial

Untuk memberikan pelatihan cara atau step-step dalam menajlangkan aplikasi MySql.

---

### 2. SQL Select, SQL Home

Untuk memilih semua data dalam table Database. Dan juga dapat mengembalikan table database hasil output. Contoh :

```
SELECT * FROM Produk ;
```

---

### 3. SQL Select Distinct

Untuk mengembalikan lagi data, dan tidak Bersama dengan entri duplikat pada Database. Contoh :

```
SELECT DISTINCT Kategori FROM Produk ;
```

---

### 4. SQL Where

Untuk memberikan filter pada hasil SELECT yaitu mengekstrak record. Juga untuk memenuhi kondisi secara ketentuan. Contoh :

```
SELECT kolom1, kolom2, ... FROM name_tabel WHERE kondisi;
```

---

### 5. SQL And,Or,Not

Untuk mengkombinasikan dengan SQL Where. And,Or, dan Not digunakan untuk memberikan filter record yang bergantung pada kondisi. Contoh :

```
SELECT kolom1, kolom2, ... FROM name_tabel WHERE kondisi1 AND kondisi2 AND kondisi3;
```

---

#### 6. **SQL Order By**

Untuk mengatur perintah order hasil, dan mengikuti aturan yang standar. Yaitu yang tadinya rendah, menjadi di tingkatkan. Contoh :

```
REVOKE SELECT, UPDATE ON Users FROM admin5 ;
```

---

#### 7. **SQL Insert Into**

Untuk memasukkan sebuah record baru ke dalam tabel database. Contoh :

```
INSERT INTO tabeldata (VALUES  
(1, 'redaksi', 'redaksi@kuliahkomputer.com', 'Kediri')
```

---

#### 8. **SQL Update**

Untuk mengupdate data di table dengn WHERE. Contoh :

```
UPDATE Produks  
SET Kategori = 'Tas', Ukuran= 'Medium'  
WHERE Produk ID = 2;
```

---

#### 9. **SQL Delete**

Untuk menghapus baris table pada Database. Contoh :

```
DELETE FROM Produks  
WHERE NamaProduk= 'K-12 Mechanical';  
DELETE * FROM Produks ;
```

---

#### 10. **SQL Select Top**

Untuk memberikan spesifik pada angka maksimal dalam set hasil pengolahan data. Contoh :

```
SELECT * FROM Produksi  
LIMIT 5 ;  
SELECT TOP 7 PERCENT * FROM Produks ;
```

---

### 11. SQL Min dan Max

Min untuk mendapatkan nilai paling terkecil pada suatu kolom, Sedangkan Max kebalikan dari Min, yaitu untuk mendapatkan Nilai paling terbesar pada suatu kolom. Contoh :

```
SELECT MIN(nama_kolom) FROM nama_tabel WHERE kondisi;  
SELECT MAX(nama_kolom) FROM nama_tabel WHERE kondisi;
```

---

### 12. SQL Count, Avg, dan Sum

Count untuk mendapatkan jumlah hitungan record. Avg untuk mendapatkan rata-rata record. Sedangkan Sum untuk mendapatkan jumlah record, tetapi tentu nilai pada kolom harus numerik. Contoh :

```
SELECT COUNT (nama_kolom) FROM nama_tabel WHERE kondisi;  
SELECT AVG(nama_kolom) FROM nama_tabel WHERE kondisi;  
SELECT SUM(nama_kolom) FROM nama_tabel WHERE kondisi;
```

---

### 13. SQL Like

Untuk menemukan pola data yang spesifik dalam kolom database. Contoh :

```
SELECT * FROM Produks WHERE Kategori LIKE '%Kursi%';
```

---

### 14. SQL Between

Untuk memberi filter hasil pengolahan database, dan juga mengembalikan hasil yang sesuai dengan spesifikasi data. Contoh :

```
SELECT * FROM Produks  
WHERE ProdukID BETWEEN 5 and 25 ;
```

---

### 15. SQL Aliases

Untuk memberi nama terbaru pada table tanpa mengubah nama awalnya. SQL ini sangat sering digunakan karena dapat membantu mengenali table, dan mempermudah dalam menulis query. Contoh :

```
SELECT ID as produkID, ProductName AS Produk  
FROM Produks ;  
SELECT p.ID FROM Produks AS p WHERE p.ID=2;
```

---

#### 16. SQL Inner Join

Untuk menggabungkan dua baris dari table yang berbeda. Contoh :

```
SELECT Orders. ID, Developers.Name  
FROM Orders  
INNER JOIN Developers ON Orders. ID = Developers. ID;
```

---

#### 17. SQL Left Join

Untuk menerima data dari sebelah kiri, lalu mencocokkan entri di table sebelah kanan.

Contoh :

```
SELECT Siswas.NameSiswa, Pelajarans.MataPelajaran FROM Siswas  
LEFT JOIN Pelajarans ON Siswas.Siswa ID = Pelajaran.Siswa_id ;
```

---

#### 18. SQL Right Join

Untuk menerima data dari sebelah kanan, lalu mencocokkan entri di table sebelah kiri.

Contoh :

```
SELECT Produks.ProdukID, Orders.OrderID  
FROM Produks  
RIGHT JOIN Orders ON Produks,ProdukID = Orders.Produk_id ;
```

---

#### 19. SQL Full Join

Untuk merekam semua data yang sesuai antara table pada sisi kanan, dan juga kiri dalam Database. Contoh :

```
SELECT Siswas.NameSiswa, Pelajarans.ID  
FROM Siswas  
FUL OUTER JOIN Gurus ON Siswas, ID=Pelajarans.siswa_id  
ORDER BY Siswas>NamaSiswa ;
```

---

#### 20. SQL Union

Untuk menggabungkan hasil secara ganda dengan dua atau juga lebih statement Select.

Contoh :

```
SELECT NamaProduk FROM Produks  
UNION  
SELECT NamaProduk FROM Produks  
ORDER BY NamaProduk ;
```

---

## 21. SQL Union All

Untuk menggabungkan yang lebih dari satu set hasil, juga untuk menjaga data Duplicate.

Contoh :

```
SELECT Provinsi FROM Siswas  
UNION ALL  
SELECT Provinsi FROM Gurus  
ORDER BY Provinsi ;
```

---

## 22. SQL Group By

Untuk mengatur data identis ke dalam grup yang sama, misalnya baris data dengan nilai yang sama. Contoh :

```
SELECT COUNT (ProdukID), Kategori  
FROM Produks  
GROUP BY Kategori ;
```

---

## 23. SQL Having

Untuk memberikan filter hasil pengolahan data ke dalam baris. Having digunakan untuk mengeksekusi fungsi agregat dari data. Contoh :

```
SELECT COUNT (ProdukID), NamaProduk  
FROM Produks  
GROUP BY NamaProduk  
HAVING COUNT (ProdukID) > 5 ;
```

---

## 24. SQL Exists

Untuk memeriksa kembali tersedianya data yang hanya dengan menulis Subquery. Jika bias d temukan, maka akan ditampilkan pada statmen yang di jalangkan. Dan fitur ini bias di gabungkan dengan INSERT, SELECT, DELETE, dan UPDATE. Contoh :

```
SELECT ProdukID, NamaProduk  
FROM Produks  
WHERE EXISTS (SELECT Produk_id FROM Orders WHERE Orders.Produk_id =  
ProdukID AND Produk.Kategori = "Kursi") ;
```

---

## 25. SQL Select Into

Untuk memberi pilihan spesifikasi pada data dari table, lalu menyalin kepada table yang lainnya. Contoh :

```
SELECT * INTO ProduksBackup2021  
FROM Produks ;  
SELECT ProdukID, NamaProduk, Kategori, Ukuran INTO ProduksBackup2021  
FROM Produks ;
```

---

## 26. SQL Insert Into Select

Untuk memasukkan baris data baru pada table Database. Contoh :

```
INSERT INTO SELECT Produks (NamaProduk, Kategori, Ukuran)  
VALUES ('K-12 Mechanical', 'Keyboard', 'Medium') ;
```

---

## 27. SQL Operators

Untuk menjangkau perintah yang Bersama dengan fitur SQL Count, Avg, dan juga Sum.

---

## 28. SQL Database, SQL Creat Database

Untuk membuat sebuah Database terbaru, dan harus mempunyai akses sebagai Admin agar bisa digunakan. Contoh :

```
CREATE DATABASE name_database
```

---

## 29. SQL Drop DB

Untuk menghapus seluruh isi pada Database, termasuk semua data dan paramaternya.  
Contoh :

```
DROP DATABASE belajar ;
```

---

### 30. SQL Create Table

Untuk membuat sebuah table terbaru di dalam database. Contoh :

```
CREATE TABLE Produk (
  ProdukID int,
  NamaProduk varchar(255),
  Kategori varchar(255),
  Ukuran varchar(255)
);
```

---

### 31. SQL Drop Table

Untuk menghapus table data, dengan tipe data dan parameter yang sudah diberikan. Contoh :

```
DROP TABLE Buku ;
```

---

### 32. SQL Alter Table

Untuk menambah kolom dari sebuah Tabel. Contoh :

```
ALTER TABLE Produk
ADD TgProduksi date ;
ALTER TABLE Produk
DROP COLUMN Size ;
```

---

### 33. SQL Drop Index, Ms Acces functions, MySQL functions, SQL Server functions

Untuk menghapus Index yang sudah dipilih. Contoh :

```
DROP INDEX idx_produk On Produk
DROP INDEX Products.idx_Product
```

---

### 34. SQL Drop Table

Untuk menghapus table data, dengan tipe data dan parameter yang sudah diberikan. Contoh :

```
DROP TABLE Buku ;
```

---

### 35. SQL Create Index

Untuk membuat Index terbaru yang ada pada table, dan juga dapat mudah untuk mendapat data dalam database dengan cepat. Contoh :

```
CREATE INDEX idx_produk  
ON Produks (NamaProduk) ;
```

---

### 36. SQL Views

Untuk membuat sebuah versi yang lebih muda dari berbagai table dalam database. Views dapat memperlihatkan aneka kolom dan juga baris data. Contoh :

```
VIEWS [Present List Produks] AS  
SELECT ProdukID, NamaProduk  
FROM Produks  
WHERE Discountinued = No ;
```

---

### 37. SQL Dates

Untuk mengembalikan tugas ke titik progress pengelolaan data yang lain. Contoh :

```
DATES bliss_buku ;
```

---

### 38. SQL Data Types

Untuk membatalkan tugas ke titik progress pengolahan data. Contoh :

```
DELETE FROM Produks  
WHERE ProduksID = 9 ;  
DATA TYPES ;
```

---

### 39. SQL Primary Key

Untuk memberikan user lain mengakses database. Contoh :

```
PRIMARY KEY SELECT, UPDATE ON Users TO Admin5 ;
```

---

### 40. SQL Foreign Key

Untuk mencabut akses database dari user lain. Contoh :

```
FOREIGN KEY SELECT, UPDATE ON Users FROM admin5 ;
```



#### **41. SQL Comments**

Untuk memberikan komentar atau koreksi pada pengelola dan pembuatan sebuah table database.