

Basis Data II



Pertemuan Ke-8
(Views & Users Authorisation)

Noor Ifada

noor.ifada@if.trunojoyo.ac.id

Sub Pokok Bahasan

- *Views:*

- *Create*
- *Update*
- *Delete*

- *Users Authorisation:*

- Menambah dan Menghapus *Users*
- Mengubah Nama dan Password *Users*
- Hak Akses: Tabel dan Kolom
- Hak Akses: Basisdata
- meneruskan Hak Akses: *WITH GRANT OPTION*
- Membatalkan Hak Akses

VIEWS

```
CREATE VIEW view_name (column_name) AS  
[SELECT BLOCK]  
[WITH CHECK OPTION]
```

- merupakan '*derived tables*' ⇒ harus didefinisikan dalam bentuk query pada tabel atau view yang lain
- merupakan '*virtual tables*' ⇒ tidak akan dievaluasi, kecuali jika mereka digunakan dalam *query* lain
- *reusable*
- dapat digunakan untuk membuat query yang sulit (atau bahkan tidak mungkin) untuk dilakukan
- dalam suatu kondisi tertentu perintah INSERT, UPDATE, atau DELETE dapat dilakukan terhadap data yang ada di dalam tabel basis melalui view tabel basis tersebut

Create VIEWS

Contoh 1: Buatlah view untuk membuat daftar seluruh nama kota yang ada dalam tabel PLAYERS!

```
CREATE VIEW TOWNS (TOWN) AS
SELECT DISTINCT TOWN
FROM PLAYERS
```

Sintaks Alternatif:

```
CREATE VIEW TOWNS AS
SELECT DISTINCT TOWN
FROM PLAYERS
```

Query SELECT:

```
SELECT *
FROM TOWNS
```

TOWN
Stratford
Inglewood
Eltham
Midhurst
Douglas
Plymouth

Create VIEWS (contd-2)

Contoh 2: Buatlah view untuk membuat daftar nomor pemain dan nomor liga dari seluruh pemain yang memiliki nomor liga!

```
CREATE VIEW CPLAYERS AS
SELECT PLAYERNO, LEAGUENO
FROM PLAYERS
WHERE LEAGUENO IS NOT NULL
```

Query SELECT:

```
SELECT *
FROM CPLAYERS
```

PLAYERNO	LEAGUENO
2	2411
6	8467
8	2983
27	2513
44	1124
57	6409
83	1608
100	6524
104	7060
112	1319

Contoh 3: Dapatkan seluruh informasi mengenai pemain yang memiliki nomor liga yang nomor pemainnya adalah antara 6 dan 44!

```
SELECT *
FROM CPLAYERS
WHERE PLAYERNO BETWEEN 6 AND 44
```

PLAYERNO	LEAGUENO
6	8467
8	2983
27	2513
44	1124

Create VIEWS (contd-3)

Contoh 4: Hapuslah pemain dengan nomor liga 7060!

```
DELETE
FROM   CPLAYERS
WHERE  LEAGUENO = '7060'
```

Ketika query dieksekusi, maka data pemain dengan nomor liga = 7060 yang ada pada **tabel basis** (tabel PLAYERS) akan dihapus

Contoh 5: Buatlah view untuk membuat daftar seluruh pemain yang memiliki nomor liga dengan nomor pemain antara 6 dan 27!

```
CREATE      VIEW SEVERAL AS
SELECT *
FROM       CPLAYERS
WHERE      PLAYERNO BETWEEN 6 AND 27
```

Create VIEWS (contd-4)

Contoh 6: Tentukan (nilai) rata-rata dari keseluruhan penalti yang pernah dilakukan oleh pemain!

Solusi berikut **tidak mungkin** dapat dilakukan:

```
SELECT  PLAYERNO, AVG(SUM(AMOUNT))
FROM    PENALTIES
GROUP BY PLAYERNO;
```

Solusi yang dapat dilakukan dengan menggunakan VIEW:

```
CREATE VIEW SUM_PENALTIES (PLAYERNO, SUM_AMOUNT) AS
SELECT  PLAYERNO, SUM(AMOUNT)
FROM    PENALTIES
GROUP BY PLAYERNO;
```

```
SELECT AVG(SUM_AMOUNT)
FROM SUM_PENALTIES;
```

Sintaks Alternatif:

```
CREATE VIEW SUM_PENALTIES AS
SELECT  PLAYERNO, SUM(AMOUNT) AS SUM_AMOUNT
FROM    PENALTIES
GROUP BY PLAYERNO;
```

Create VIEWS (contd-5)

Contoh 7: Buatlah view untuk membuat daftar urutan digit dari 0 sampai 9!

```
CREATE VIEW DIGITS AS
SELECT 0 DIGIT UNION
SELECT 1 UNION
SELECT 2 UNION
SELECT 3 UNION
SELECT 4 UNION
SELECT 5 UNION
SELECT 6 UNION
SELECT 7 UNION
SELECT 8 UNION
SELECT 9
```

```
SELECT * FROM DIGITS
```

DIGIT
0
1
2
3
4
5
6
7
8
9

Update VIEWS: WITH CHECK OPTION

Contoh 8: Buatlah view untuk membuat daftar pemain yang lahir sebelum tahun 1960!

```
CREATE VIEW VETERANS AS
SELECT *
FROM PLAYERS
WHERE BIRTH_DATE < '1960-01-01';
SELECT * FROM VETERANS;
```

PLAYERNO	NAME	INITIALS	BIRTH_DATE	SEX	JOINED	STREET	HOUSENO	POSTCODE	TOWN	PH
2	Everett R		1948-09-01	M	1975	Stoney Road	43	3575NH	Stratford	07
39	Bishop D		1956-10-29	M	1980	Eaton Square	78	9629CD	Stratford	07
83	Hope PK		1956-11-11	M	1982	Magdalene Road	16A	1812UP	Stratford	07

Kemudian lakukan perubahan tanggal lahir dari pemain nomor 2 (dari 1 September 1948 menjadi 1 September 1970)!

```
UPDATE VETERANS
SET BIRTH_DATE = '1970-09-01'
WHERE PLAYERNO = 2;
SELECT * FROM VETERANS;
```

PLAYERNO	NAME	INITIALS	BIRTH_DATE	SEX	JOINED	STREET	HOUSENO	POSTCODE	TOWN	PH
39	Bishop D		1956-10-29	M	1980	Eaton Square	78	9629CD	Stratford	07
83	Hope PK		1956-11-11	M	1982	Magdalene Road	16A	1812UP	Stratford	07

Update VIEWS: WITH CHECK OPTION (contd-2)

Penggunaan WITH CHECK OPTION pada VIEW adalah melakukan pengecekan validitas perintah UPDATE, INSERT dan DELETE:

- ◆ perintah UPDATE adalah benar jika seluruh baris yang di-UPDATE tetap menjadi isi ('virtual') tabel VIEW
- ◆ perintah INSERT adalah benar jika baris baru yang di-INSERT merupakan isi ('virtual') tabel VIEW
- ◆ perintah DELETE adalah benar jika baris baru yang di-DELETE merupakan isi ('virtual') tabel VIEW

Maka, view pada **Contoh 8** diubah menjadi seperti berikut:

```
CREATE VIEW VETERANS AS
SELECT *
FROM PLAYERS
WHERE BIRTH_DATE < '1960-01-01'
WITH CHECK OPTION
```



Delete VIEWS

Contoh 9: Hapuslah view CPLAYERS!

DROP VIEW CPLAYERS

Contoh 10: Hapuslah view VETERANS!

DROP VIEW VETERANS

USERS AUTHORISATION

- Untuk proteksi data: USER, PASSWORD, HAK AKSES
- Hak Akses meliputi:
 - ➔ Hak akses terhadap suatu **Kolom** tertentu dari suatu tabel
 - ➔ Hak akses terhadap suatu **Tabel**
 - ➔ Hak akses terhadap tabel-tabel yang ada pada suatu **Basisdata** tertentu
 - ➔ Hak akses suatu **User** terhadap seluruh basisdata yang ada di dalam server



Menambah & Menghapus USERS

```
CREATE USER 'user_name'@'host_name' IDENTIFIED  
BY password_user
```

Contoh 1: Buatlah dua user baru, yaitu EVE dengan password EVE_PASS, dan JUN dengan password JUN_PASS!

```
CREATE USER  
  'EVE'@'localhost' IDENTIFIED BY 'EVE_PASS',  
  'JUN'@'localhost' IDENTIFIED BY 'JUN_PASS'
```

```
DROP USER 'user_name'@'host_name'
```

Contoh 2: Hapuslah user JUN!

```
DROP USER 'JUN'@'localhost'
```

Mengubah Nama & Password USERS

```
RENAME USER 'user_name'@'host_name' TO  
'user_name'@'host_name'
```

Contoh 3: Ubahlah nama user EVE dan JUN menjadi EVE1 dan JUN1

```
RENAME USER  
'EVE'@'localhost' TO 'EVE1'@'localhost',  
'JUN'@'localhost' TO 'JUN1'@'localhost'
```

```
SET PASSWORD FOR 'user_name'@'host_name' =  
PASSWORD('new_password')
```

Contoh 4: Ubahlah password user JUN1 menjadi JUN1_PASS!

```
SET PASSWORD FOR 'JUN1'@'localhost' = PASSWORD('JUN1_PASS')
```

Hak Akses: Tabel & Kolom

- **SELECT** — user berhak untuk mengakses tabel tertentu dengan menggunakan pernyataan SELECT
- **INSERT** — user berhak untuk menambah baris pada tabel tertentu dengan menggunakan pernyataan INSERT
- **DELETE** — user berhak untuk menghapus baris dari tabel tertentu dengan menggunakan pernyataan DELETE
- **UPDATE** — user berhak untuk mengubah nilai dari suatu tabel tertentu dengan menggunakan pernyataan UPDATE
- **REFERENCES** — user berhak untuk membuat Foreign Key (FK) yang mengacu pada suatu tabel
- **CREATE** — user berhak untuk membuat tabel dengan suatu nama tertentu
- **ALTER** — user berhak untuk mengubah tabel dengan menggunakan pernyataan ALTER TABLE
- **INDEX** — user berhak untuk menentukan indeks pada satu tabel
- **DROP** — user berhak untuk menghapus tabel
- **ALL** atau **ALL PRIVILEGES** — merupakan 'singkatan' atas seluruh hak akses yang telah disebutkan di atas



Hak Akses: Tabel & Kolom

(contd-2)

Contoh 5: Berikanlah hak SELECT untuk JUN1 pada tabel PLAYERS!

```
GRANT SELECT
ON PLAYERS
TO JUN1
```

Contoh 6: Berikanlah hak SELECT untuk user baru BOB pada tabel PLAYERS!

```
GRANT SELECT
ON PLAYERS
TO 'BOB'@'localhost' IDENTIFIED BY 'BOB_PASS'
```

Contoh 7: Berikanlah hak INSERT dan UPDATE untuk EVE1 dan JUN1 pada seluruh kolom dari tabel TEAMS!

```
GRANT INSERT, UPDATE
ON TEAMS
TO EVE1, JUN1
```



Hak Akses: Basisdata

- **SELECT** — user berhak untuk mengakses seluruh tabel dan view dari suatu basisdata dengan menggunakan pernyataan SELECT
- **INSERT** — user berhak untuk menambah baris pada seluruh tabel yang ada dalam suatu basisdata dengan menggunakan pernyataan INSERT
- **DELETE** — user berhak untuk menghapus baris dari seluruh tabel yang ada dalam suatu basisdata dengan menggunakan pernyataan DELETE
- **UPDATE** — user berhak untuk mengubah nilai dari seluruh tabel yang ada dalam suatu basisdata dengan menggunakan pernyataan UPDATE
- **REFERENCES** — user berhak untuk membuat Foreign Key (FK) yang mengacu pada tabel-tabel yang ada dalam suatu basisdata
- **CREATE** — **CREATE** — user berhak untuk membuat tabel-tabel baru dalam suatu basisdata dengan menggunakan pernyataan CREATE TABLE
- **ALTER** — user berhak untuk mengubah tabel-tabel pada suatu basisdata dengan menggunakan pernyataan ALTER TABLE



Hak Akses: Basisdata (contd-2)

- **DROP** — user berhak untuk menghapus seluruh tabel dan view yang ada dalam suatu basisdata
- **INDEX** — user berhak untuk menentukan dan menghapus indeks dari tabel-tabel yang ada dalam suatu basisdata
- **CREATE TEMPORARY TABLES** — user berhak untuk membuat tabel-tabel sementara di dalam suatu basisdata
- **CREATE VIEW** — user berhak untuk membuat view baru dalam suatu basisdata dengan menggunakan pernyataan CREATE VIEW
- **SHOW VIEW** — user berhak untuk melihat definisi view dari seluruh view yang ada dalam suatu basisdata dengan menggunakan pernyataan SHOW VIEW
- **CREATE ROUTINE** — user berhak untuk membuat prosedur (*stored procedure*) dan fungsi (*stored function*) baru untuk suatu basisdata
- **ALTER ROUTINE** — user berhak untuk mengubah dan menghapus prosedur (*stored procedure*) dan fungsi (*stored function*) dari suatu basisdata



Hak Akses: Basisdata (contd-3)

- **EXECUTE ROUTINE** — user berhak untuk membatalkan prosedur (*stored procedure*) dan fungsi (*stored function*) yang ada pada suatu basisdata
- **LOCK TABLES** — user berhak untuk memblok tabel-tabel yang ada dalam suatu basisdata
- **ALL** atau **ALL PRIVILEGES** — merupakan 'singkatan' atas seluruh hak akses yang telah disebutkan sebelumnya

Hak Akses: Basisdata (contd-4)

Contoh 8: Berikanlah hak SELECT untuk JUN1 pada seluruh tabel yang ada di dalam basisdata TENNIS!

```
GRANT SELECT
ON TENNIS.*
TO JUN1
```

Contoh 9: Berikanlah hak CREATE, UPDATE dan REMOVE tabel-tabel dan view baru untuk JUN1 di dalam basisdata TENNIS!

```
GRANT CREATE, ALTER, DROP, CREATE VIEW
ON TENNIS.*
TO JUN1
```



Hak Akses: User

Contoh 10: Berikanlah hak CREATE, ALTER, dan DROP untuk EVE1 pada seluruh tabel dari seluruh basisdata!

```
GRANT CREATE, ALTER, DROP  
ON *.*  
TO EVE1
```

Contoh 11: Berikanlah hak kepada JUN1 untuk membuat user baru!

```
GRANT CREATE USER  
ON *.*  
TO JUN1
```



Meneruskan Hak Akses: WITH GRANT OPTION

Contoh 12: Berikanlah hak REFERENCES untuk JUN1 pada tabel TEAMS dan berikanlah ijin padanya untuk meneruskan hak tersebut kepada user lain!

```
GRANT REFERENCES  
ON TEAMS  
TO JUN1  
WITH GRANT OPTION
```

Dengan adanya klausa WITH GRANT OPTION, maka JUN1 dapat meneruskan hak akses REFERENCES tersebut kepada EVE1

```
GRANT REFERENCES  
ON TEAMS  
TO EVE1
```



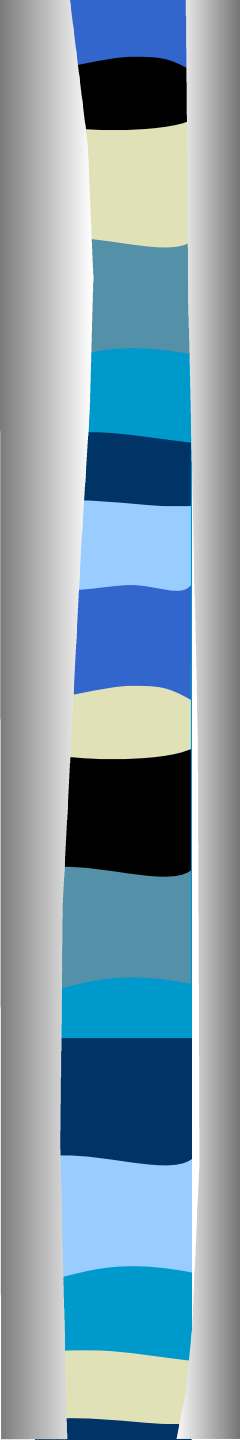
Membatalkan Hak Akses

Contoh 13: Batalkan hak SELECT untuk JUN1 pada tabel PLAYERS!

```
REVOKE SELECT  
ON PLAYERS  
FROM JUN1
```

Contoh 14: Batalkan hak INSERT dan SELECT untuk EVE1 pada tabel COMMITTEE_MEMBERS!

```
REVOKE INSERT, SELECT  
ON COMMITTEE_MEMBERS  
FROM EVE1
```



Keterangan lebih detail mengenai:

- User Authorisation

Cek MySQL 5.0 Reference Manual
(Chapter 5.4. The MySQL Access Privilege System)