

Keamanan Basis Data

Sasaran Prosedur Keamanan

- ⦿ Pengaksesan oleh pihak-pihak yang tidak berhak
- ⦿ Celah kelemahan sistem:
 - Kondisi basis data yang semakin kompleks
 - basis data terdistribusi
 - Akses basis data terbuka lewat internet/intranet

Perencanaan keamanan data

- ⦿ Ancaman-ancaman:
 1. Kehilangan yang tidak disengaja
 2. Pencurian dan penggelapan
 3. Kehilangan Privasi
 4. Kehilangan Integritas Data
 5. Kehilangan Ketersediaan Data

Fitur-Fitur Keamanan DBMS

1. View atau Subskema
2. Aturan Otorisasi
3. Prosedur yang membatasi penggunaan basis data
4. Prosedur Enkripsi
5. Skema Pengesahan
6. Pencadangan

View

- ⦿ **Pengertian:**

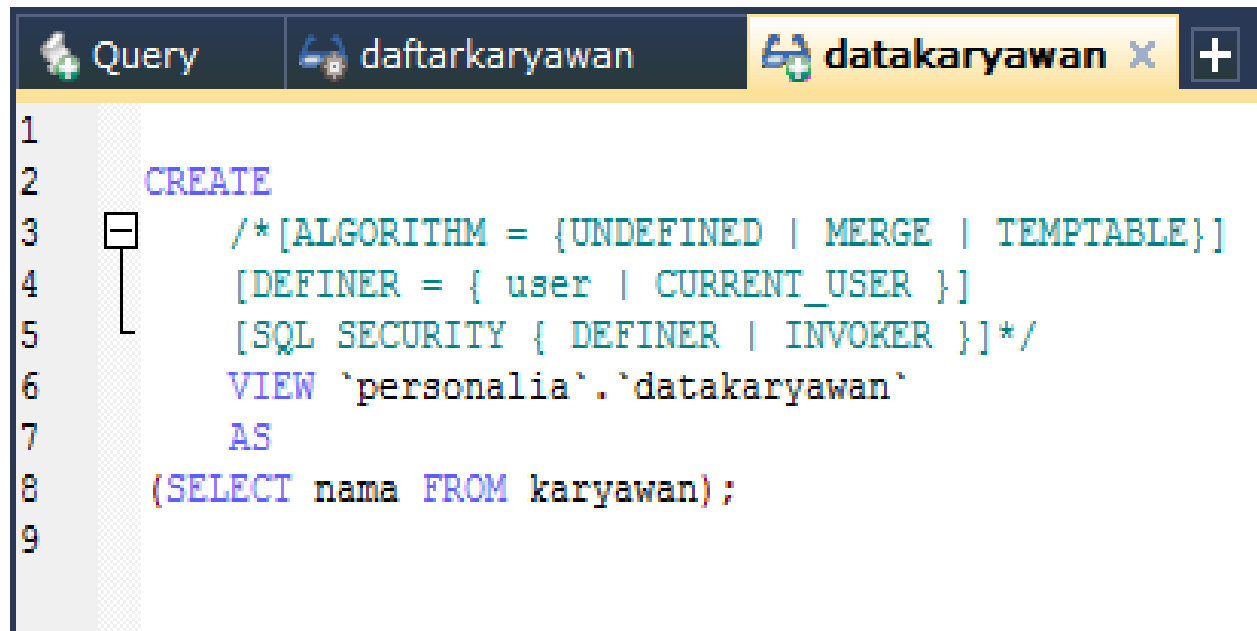
Himpunan bagian basis data yang dapat ditampilkan ke satu atau lebih pengguna.
- ⦿ **Diciptakan dengan melakukan permohonan pada satu atau lebih tabel dasar, menghasilkan tabel hasil yang dinamis untuk pengguna.**

☉ Keunggulan:

1. dapat dibentuk hanya untuk menampilkan data yang dapat diakses oleh pengguna,
2. Mencegah pengguna lain melihat data-data yang bersifat pribadi

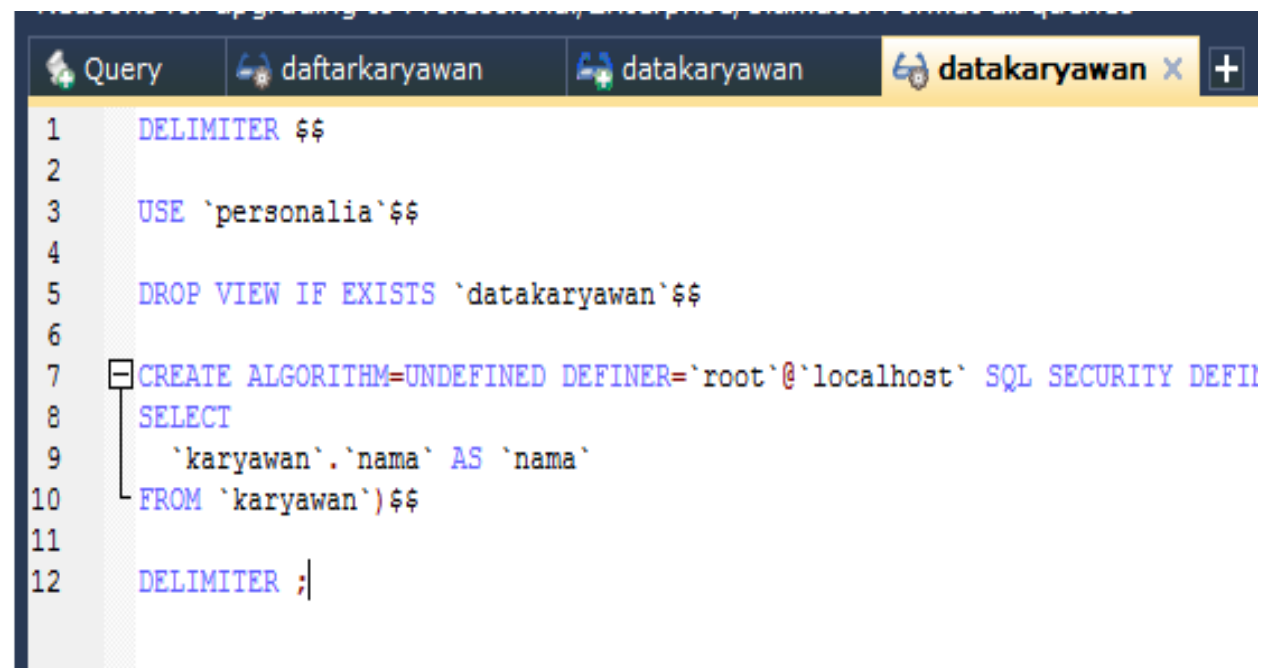
☉ Sintak:

Create View namaview AS <ekspresi query>



The screenshot shows a MySQL query editor window with a dark theme. The title bar at the top contains tabs for 'Query', 'daftarkaryawan', and 'datakaryawan'. The 'datakaryawan' tab is active, showing a SQL script to create a view. The script is as follows:

```
1
2 CREATE
3     /*[ALGORITHM = {UNDEFINED | MERGE | TEMPTABLE}]
4     [DEFINER = { user | CURRENT_USER }]
5     [SQL SECURITY { DEFINER | INVOKER }]*/
6     VIEW `personalia`.`datakaryawan`
7     AS
8     (SELECT nama FROM karyawan);
9
```



The screenshot shows a MySQL query editor window with a dark theme. The title bar at the top contains tabs for 'Query', 'daftarkaryawan', 'datakaryawan', and 'datakaryawan'. The 'datakaryawan' tab is active, showing a SQL script to drop an existing view and create a new one. The script is as follows:

```
1 DELIMITER $$
2
3 USE `personalia`$$
4
5 DROP VIEW IF EXISTS `datakaryawan`$$
6
7 CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY DEFINER
8 SELECT
9     `karyawan`.`nama` AS `nama`
10 FROM `karyawan` ) $$
11
12 DELIMITER ;
```

Aturan Otorisasi

- ⦿ Administrator basis data (DBA)
- ⦿ Administrator Jaringan
- ⦿ Pengembang Aplikasi

Tabel Otorisasi pada Basis Data Akuntansi dari Suatu Organisasi

Subjek	Objek	Aksi	Batasan
Dept. Pemasaran	Rekaman Pelanggan	Tambah	Batas Kredit<5.000.000
Transaksi Pesanan	Rekaman Pelanggan	Baca	-
Terminal 19	Rekaman Pelanggan	Modifikasi	Hanya Saldo
Dept. Akuntansi	Rekaman Pesanan	Hapus	-
Adisti Prananingrum	Rekaman Pesanan	Tambah	Pesanan<2.000.000
Program AN-4	Rekaman Pesanan	Modifikasi	-

Prosedur yang Didefinisikan Pengguna

- ⦿ Prosedur pengguna dapat dirancang untuk menyediakan identifikasi pengguna lebih lanjut

Enkripsi

- ⦿ Enkripsi/Penyandian: proses menyandikan data sehingga manusia tidak bisa membaca data tersebut
- ⦿ Difasilitasi oleh : SQL Server, Oracle

Skema Pengesahan

- ⦿ Untuk mengidentifikasi pengguna yang akan masuk ke Sistem
- ⦿ Pendekatan:
 - Biometrik
 - Kerberos pada MIT (Massachusetts Institute of Technology)
 - DES Authentication pada Sun Microsystem

Prosedur Pencadangan Basis Data

- ⦿ Prosedur Back-Up Basis Data
- ⦿ Fasilitas pencatatan (transaction log)
- ⦿ Fasilitas check point: DBMS menyinkronkan berkas-berkas secara periodik
- ⦿ Manajer pemulihan: DBMS melakukan recovery ke kondisi yang benar dan memulai kembali pemrosesan transaksi.