



**SQL language: other definitions**

**Access control**

**DBG**



**Access control**

- Data security
- Resources and privileges
- Management of privileges in SQL
- Management of roles in SQL

**DBG**

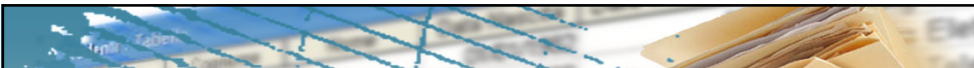
2



**Access control**

**Data security**

**DBG**

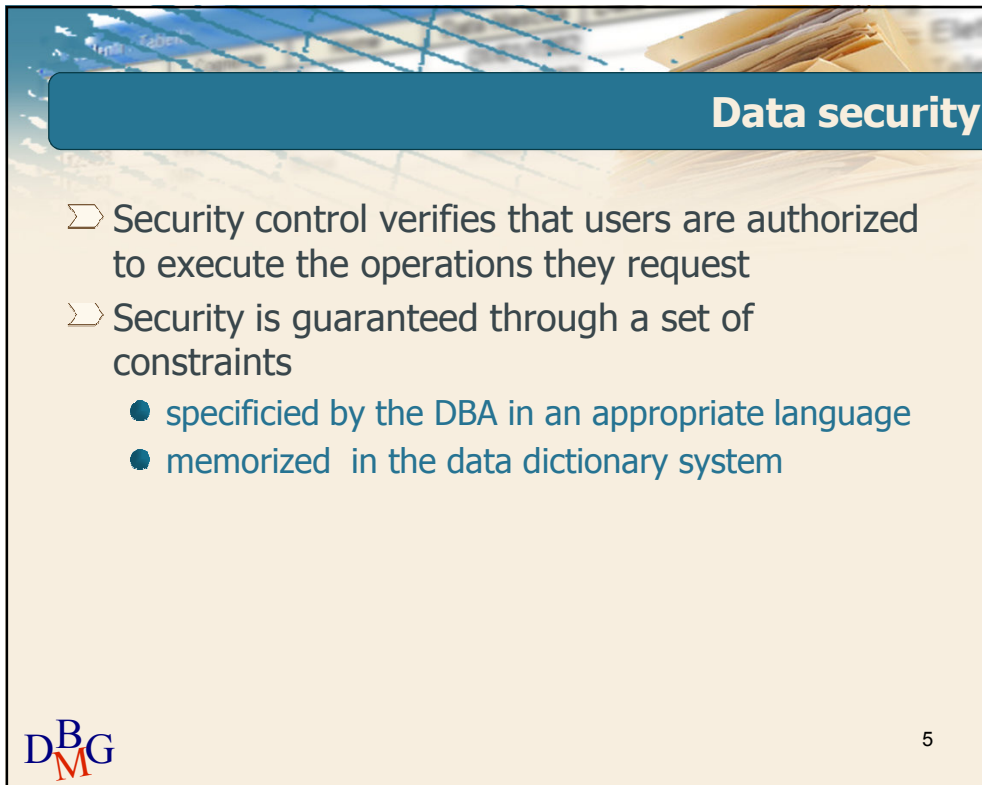


**Data security**

- Protection of data from
  - unauthorized readers
  - alteration or destruction
- The DBMS provides protection tools which are defined by the database administrator (DBA)

**DBG**

4




## Data security

- Security control verifies that users are authorized to execute the operations they request
- Security is guaranteed through a set of constraints
  - specified by the DBA in an appropriate language
  - memorized in the data dictionary system

DBG

5



## Access control

### Resources and privileges

DBG

## Resources

- Any component of the database scheme is a resource
  - table
  - view
  - attribute in a table or view
  - domain
  - procedure
  - ...
- Resources are protected by the definition of *access privileges*

## Access privileges

- Describe access rights to system resources
- SQL provides very flexible access control mechanisms for specifying
  - the resources users can access
  - the resources that have to remain private

## Privileges: characteristics

- Each privilege is characterized by the following information
  - the resource it refers to
  - the type of privilege
    - describes the action allowed on the resource
  - the user granting the privilege
  - the user receiving the privilege
  - the faculty to transmit the privilege to other users

## Types of privilege (1/2)

- INSERT
  - enables the insertion of a new object in the resource
  - valid for tables and views
- UPDATE
  - enables updating the value of an object
  - valid for tables, views and attributes
- DELETE
  - enables removal of objects from the resource
  - valid for tables and views

## Types of privilege (2/2)

### ➤ SELECT

- enables using the resource in a query
- valid for tables and views

### ➤ REFERENCES

- enables referring to a resource in the definition of a table scheme
- can be associated with tables and attributes

### ➤ USAGE

- enables use of the resource (e.g. a new type of data) in the definition of new schemes

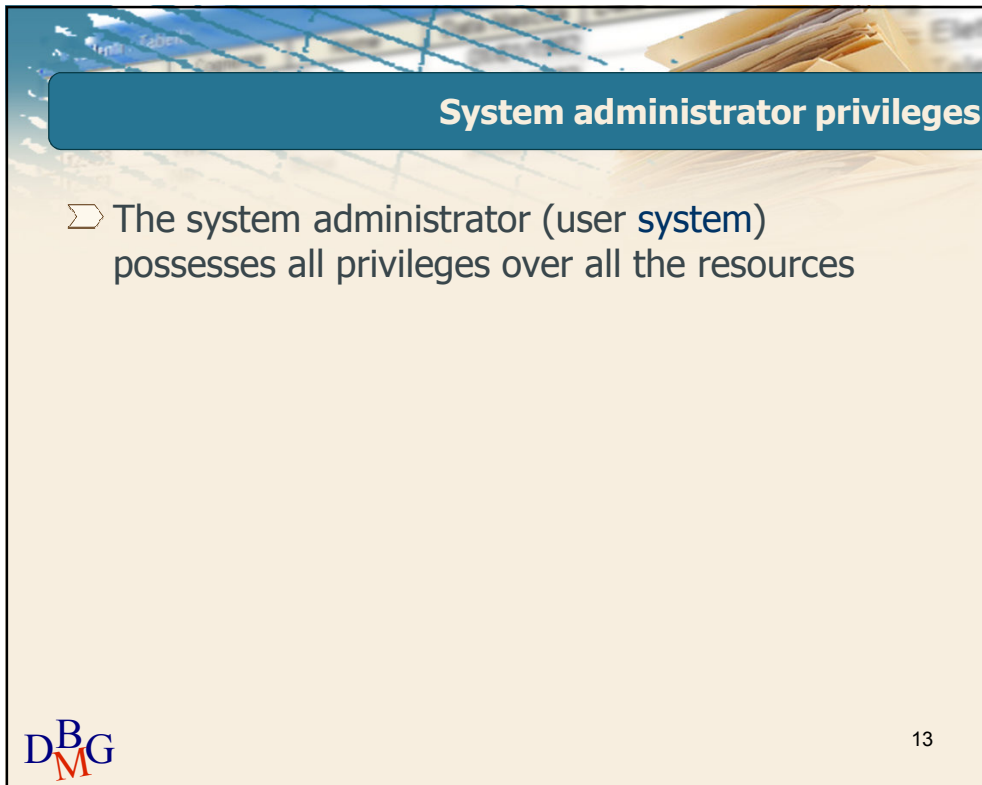
## Resource creator privileges

- When a resource is created, the system grants all privileges over that resource to the user that created it
- Only the resource creator has the privilege to eliminate a resource (**DROP**) and modify a scheme (**ALTER**)
  - the privilege to eliminate and modify a resource cannot be granted to any other user

## Diapositiva 12

---

**sp1** Would you prefer me to use "eliminate a resource", "modify a scheme" here, or would you prefer "drop a resource", "alter a scheme"?  
sp; 10/02/2013



### System administrator privileges

- The system administrator (user system) possesses all privileges over all the resources

DBG

13



### Access control

## Management of privileges in SQL

DBG



## Management of privileges in SQL

- Privileges are granted or revoked using SQL instructions
  - GRANT
    - grants privileges over a resource to one or more users
  - REVOKE
    - revokes privileges granted to one or more users

## GRANT

```
GRANT PrivilegeList ON ResourceName TO UserList
      [WITH GRANT OPTION]
```

- *PrivilegeList*
  - specifies the list of privileges
  - ALL PRIVILEGES
    - Keyword for identifying all privileges
- *ResourceName*
  - specifies the resource for which the privilege is granted
- *UserList*
  - Specifies the users who are granted the privilege

## Example n. 1

```
GRANT ALL PRIVILEGES ON P  
TO Black, Whitei
```

- ⇒ Users Black and White are granted all privileges for table P



17

## GRANT

```
GRANT PrivilegeList ON ResourceName TO UserList  
[WITH GRANT OPTION]
```

- ⇒ WITH GRANT OPTION
  - faculty to transfer the privilege to other users



18

## Example n. 2

```
GRANT SELECT ON S TO Red
WITH GRANT OPTION
```

- ⇒ User Red is granted the privilege to **SELECT** in table S
- ⇒ User Red has the faculty to grant the privilege to other users

## REVOKE

```
REVOKE PrivilegeList ON ResourceName FROM UserList
[RESTRICT|CASCADE]
```

- ⇒ The command **REVOKE** can remove
  - all the privileges that have been granted
  - a subset of privileges granted

## Example n. 1

```
REVOKE UPDATE ON P FROM White
```

- ⇒ User White's privilege to UPDATE table P is revoked

## REVOKE

```
REVOKE PrivilegeList ON ResourceName FROM UserList  
[RESTRICT|CASCADE]
```

### ⇒ RESTRICT

- the command must not be executed if revoking the user's privileges entails revoking other privileges
  - Example: the user has received the privileges with the GRANT OPTION and has propagated the privileges to other users
- default value

## Example n. 1

```
REVOKE UPDATE ON P FROM White
```

- ⇒ User White's privilege to UPDATE table P is revoked
  - the command is not executed if it entails revoking the privilege of other users

## REVOKE

```
REVOKE PrivilegeList ON ResourceName FROM UserList  
[RESTRICT|CASCADE]
```

- ⇒ CASCADE
  - revokes also all the privileges which have been propagated
    - generates a chain reaction
  - for each privilege revoked
    - all granted privileges are revoked in a cascade
    - all database elements which have been created exploiting these privileges are removed

## Example n. 2

REVOKE SELECT ON S FROM Red CASCADE

- ⇒ User Red's privilege to SELECT table S is revoked
- ⇒ User Red had received the privilege through GRANT OPTION
  - if Rossi has propagated the privilege to other users, the privilege is revoked in cascade
  - if Rossi has created a view using the SELECT privilege, the view is removed

## Access control

### Management of roles in SQL

## Concept of role (1/2)

- The role is an access profile
  - Defined by **its** set of privileges
- Each user has a defined role
  - it enjoys the privileges associated with that role

## Concept of role (2/2)

- Advantages
  - access control is more flexible
    - a user can have different roles at different times
  - it simplifies administration
    - an access profile need not be defined at the moment of its activation
    - it is easy to define new user profiles

## Roles in SQL-3

➤ Definition of a role

```
CREATE ROLE RoleName
```

➤ Definition of role privileges and user roles

- instruction GRANT

➤ A user can have different roles at different times

- dynamic association of a role with a user

```
SET ROLE RoleName
```