



## Linguaggio SQL: costrutti avanzati

Controllo dell'accesso

# Controllo dell'accesso

- Sicurezza dei dati
- Risorse e privilegi
- Gestione dei privilegi in SQL
- Gestione dei ruoli in SQL

- Protezione dei dati da
  - letture non autorizzate
  - alterazione o distruzione
- Il DBMS fornisce strumenti per realizzare le protezioni, che sono definite dall'amministratore della base dati (DBA)

# Sicurezza dei dati

- Il controllo della sicurezza verifica che gli utenti siano autorizzati a eseguire le operazioni che richiedono di eseguire
- La sicurezza è garantita attraverso un insieme di vincoli
  - specificati dal DBA in un opportuno linguaggio
  - memorizzati nel dizionario dei dati del sistema

- Qualsiasi componente dello schema di una base di dati è una risorsa
  - tabella
  - vista
  - attributo all'interno di una tabella o di una vista
  - dominio
  - procedura
  - ...
- Le risorse sono protette mediante la definizione di *privilegi di accesso*

## Privilegi di accesso

- Descrivono i diritti di accesso alle risorse del sistema
- SQL offre meccanismi di controllo dell'accesso molto flessibili mediante i quali è possibile specificare
  - le risorse a cui possono accedere gli utenti
  - le risorse che devono essere mantenute private

# Privilegi: caratteristiche

➤ Ogni privilegio è caratterizzato dalle seguenti informazioni

- la risorsa a cui si riferisce
- il tipo di privilegio
  - descrive l'azione permessa sulla risorsa
- l'utente che concede il privilegio
- l'utente che riceve il privilegio
- la facoltà di trasmettere il privilegio ad altri utenti

## Tipi di privilegi (1/2)

### ➤ INSERT

- permette di inserire un nuovo oggetto nella risorsa
- vale per le tabelle e le viste

### ➤ UPDATE

- permette di aggiornare il valore di un oggetto
- vale per le tabelle, le viste e gli attributi

### ➤ DELETE

- permette di rimuovere oggetti dalla risorsa
- vale per le tabelle e le viste

## Tipi di privilegi (2/2)

### ➤ SELECT

- permette di utilizzare la risorsa all'interno di un'interrogazione
- vale per le tabelle e le viste

### ➤ REFERENCES

- permette di far riferimento a una risorsa nella definizione dello schema di una tabella
- può essere associato a tabelle e attributi

### ➤ USAGE

- permette di utilizzare la risorsa (per esempio, un nuovo tipo di dato) nella definizione di nuovi schemi

## Privilegi del creatore della risorsa

- Alla creazione di una risorsa, il sistema concede tutti i privilegi su tale risorsa all'utente che ha creato la risorsa
- Solo il creatore della risorsa ha il privilegio di eliminare una risorsa (**DROP**) e modificarne lo schema (**ALTER**)
  - il privilegio di eliminare e modificare una risorsa non può essere concesso a nessun altro utente

## Privilegi dell'amministratore del sistema

- L'amministratore del sistema (utente `system`) possiede tutti i privilegi su tutte le risorse

# Gestione dei privilegi in SQL

- I privilegi sono concessi o revocati mediante le istruzioni SQL
- GRANT
    - concede privilegi su una risorsa a uno o più utenti
  - REVOKE
    - toglie a uno o più utenti i privilegi che erano stati loro concessi

GRANT *ElencoPrivilegi* ON *NomeRisorsa* TO *ElencoUtenti*  
[WITH GRANT OPTION]

## ➤ *ElencoPrivilegi*

- specifica l'elenco dei privilegi
- ALL PRIVILEGES
  - parola chiave per identificare tutti i privilegi

## ➤ *NomeRisorsa*

- specifica la risorsa sulla quale si vuole concedere il privilegio

## ➤ *ElencoUtenti*

- specifica gli utenti a cui viene concesso il privilegio

## Esempio n. 1

GRANT ALL PRIVILEGES ON P  
TO Neri, Bianchi

- Agli utenti Neri e Bianchi sono concessi tutti i privilegi sulla tabella P

GRANT *ElencoPrivilegi* ON *NomeRisorsa* TO *ElencoUtenti*  
[WITH GRANT OPTION]

## ➤ WITH GRANT OPTION

- facoltà di trasferire il privilegio ad altri utenti

## Esempio n. 2

GRANT SELECT ON F TO Rossi  
WITH GRANT OPTION

- All'utente Rossi è concesso il privilegio di **SELECT** sulla tabella F
- L'utente Rossi ha facoltà di trasferire il privilegio ad altri utenti

# REVOKE

```
REVOKE ElencoPrivilegi ON NomeRisorsa FROM ElencoUtenti  
[RESTRICT|CASCADE]
```

- Il comando REVOKE può togliere
- tutti i privilegi che erano stati concessi
  - un sottoinsieme dei privilegi concessi

## Esempio n. 1

REVOKE UPDATE ON P FROM Bianchi

- All'utente Bianchi è revocato il privilegio di UPDATE sulla tabella P

```
REVOKE ElencoPrivilegi ON NomeRisorsa FROM ElencoUtenti  
[RESTRICT|CASCADE]
```

## ➤ RESTRICT

- il comando non deve essere eseguito qualora la revoca dei privilegi all'utente comporti qualche altra revoca di privilegi
  - Esempio: l'utente ha ricevuto i privilegi con **GRANT OPTION** e ha propagato i privilegi ad altri utenti
- valore di default

REVOKE UPDATE ON P FROM Bianchi

- All'utente Bianchi è revocato il privilegio di UPDATE sulla tabella P
  - il comando non è eseguito se comporta la revoca del privilegio ad altri utenti

REVOKE *ElencoPrivilegi* ON *NomeRisorsa* FROM *ElencoUtenti*  
[RESTRICT|CASCADE]

## ➤ CASCADE

- revoca anche tutti i privilegi che erano stati propagati
  - genera una reazione a catena
- per ogni privilegio revocato sono
  - revocati in cascata tutti i privilegi concessi
  - rimossi tutti gli elementi della base di dati che erano stati creati sfruttando questi privilegi

REVOKE SELECT ON F FROM Rossi CASCADE

- All'utente Rossi è revocato il privilegio di **SELECT** sulla tabella F
- L'utente Rossi aveva ricevuto il privilegio con **GRANT OPTION**
  - se Rossi ha propagato il privilegio ad altri utenti, il privilegio è revocato in cascata
  - se Rossi ha creato una vista utilizzando il privilegio di **SELECT**, la vista è rimossa

## Concetto di ruolo (1/2)

- Il ruolo è un profilo di accesso
  - definito dall'insieme di privilegi che lo caratterizzano
- Ogni utente ricopre un ruolo predefinito
  - gode dei privilegi associati al ruolo

## Concetto di ruolo (2/2)

### ➤ Vantaggi

- controllo dell'accesso più flessibile
  - possibilità che un utente ricopra ruoli diversi in momenti diversi
- semplificazione dell'attività di amministrazione
  - possibilità di definire un profilo di accesso in un momento diverso dalla sua attivazione
  - facilità nella definizione del profilo di nuovi utenti

### ➤ Definizione di un ruolo

`CREATE ROLE NomeRuolo`

### ➤ Definizione dei privilegi di un ruolo e del ruolo di un utente

- istruzione `GRANT`

### ➤ Un utente in momenti diversi può ricoprire ruoli diversi

- associazione dinamica di un ruolo a un utente

`SET ROLE NomeRuolo`