

Laboratorio di Basi di Dati 2

Terzo progetto

Data consegna: entro l'esame orale, su AulaWeb.

Cosa consegnare: Un documento contenente: (i) descrizione del software utilizzato, come richiesto al punto 1; (ii) per ogni domanda contenuta nei punti successivi quanto richiesto dal testo ed eventuali script SQL o altri programmi che sono stati sviluppati relativamente a tali punti.

Software da utilizzare. Un DBMS a scelta tra PostgreSQL, Oracle e Microsoft SQL Server. La documentazione prodotta deve essere commisurata alla quantità di documentazione e informazioni disponibili relativamente a meccanismi di controllo dell'accesso nel DBMS scelto. In laboratorio è presente solo PostgreSQL.

NOTA: Per effettuare alcune operazioni richieste dall'esercitazione sono necessari i privilegi di amministrazione. Vi suggerisco quindi, se potete, di utilizzare una macchina personale per questa esercitazione. Nel caso non sia possibile, contattatemi e provvederò a concedervi i privilegi strettamente necessari alla creazione dei ruoli.

Punto 1: Comprensione dello strumento. Basandosi sulla documentazione disponibile e su prove pratiche, acquisire familiarità con le funzionalità di controllo dell'accesso supportate dal DBMS. Produrre una breve relazione in cui si descrivano tali elementi evidenziando le principali differenze rispetto a quanto visto a lezione. Riportare le fonti delle informazioni inserite nella documentazione.

Punto 2: Schema. Consideriamo una seguente semplice base di dati relativa ad un dipartimento universitario e costituita dalle seguenti relazioni (definite dall'amministratore del DB, cioè voi):

Personale(id, nome, stipendio)

Studente(matricola, nome, media)

Corso(codice, titolo, prof^{Personale})

Esame(corso^{Corso}, studente^{Studente}, data, voto)

Punto 3: Ruoli e utenti. Definire i ruoli `capoufficio`, `direttore`, `prof`, `impiegato`, `studente`, tali che:

- Il ruolo `prof` è ruolo senior di `impiegato` ma non può amministrare tale ruolo.
- Il ruolo `capoufficio` è ruolo senior di `impiegato` e può amministrare tale ruolo.
- Il ruolo `direttore` è ruolo senior di `prof` e può amministrare tale ruolo.
- `direttore` e `capoufficio` sono abilitati a creare relazioni.

Creare ora gli utenti:

- **alice**, **bianca** e **carlo** con ruolo **studente**;
- **marta** con ruolo **impiegato**, **luca** con ruolo **capoufficio**, **nino** con ruolo **direttore** e la possibilità di amministrare tale ruolo;
- **donatella**, **elena**, **fabio** e **olga**.

nino ora delega (se possibile) il ruolo **prof** a **donatella**, **elena**, **fabio**. Quali ruoli potrebbe delegare **nino** ad **olga**? Delegare tutti tali ruoli.

Punto 4: Assegnazione privilegi ai ruoli. Assegnare a tali ruoli i seguenti privilegi (la prima indicazione specifica quale utente deve effettuare i relativi comandi):

admin Il capoufficio può assumere e licenziare aggiungendo/cancellando tuple in/da **Personale**. Può inoltre effettuare modifiche e interrogazioni, nonché amministrare i privilegi su tale tabella.

admin Il direttore gestisce i corsi e le relative titolarità (tabella **Corso**) e può amministrare i privilegi su tale tabella.

admin Gli studenti possono consultare le relazioni **Studente**, **Corso**, **Esame** gli impiegati la relazione **Impiegato**, i professori tutte e quattro le relazioni.

admin Gli impiegati possono richiedere l'aggiornamento della media in **Studente** che viene calcolata con apposita funzione.

admin Tutti possono fare interrogazioni aggregate su **Esame** per recuperare il numero di studenti iscritti ad un corso e il voto medio.

nino Gli studenti possono aggiungere ed eliminare tuple in **Esame** relativamente alle iscrizioni ai corsi, ma non specificare nè modificare il voto (che sarà quindi sempre nullo al momento dell'inserimento della tupla).

nino I professori possono inserire o modificare i voti in **Esame**.

Riassumere in una tabella i privilegi effettivamente in possesso dei vari utenti e ruoli dopo l'esecuzione di tali comandi.

Punto 4: Privilegi basati sul contenuto. Modificare i privilegi precedenti in modo tale che gli impiegati possano consultare la relazione **Impiegato** ma visualizzare gli stipendi solo relativamente agli impiegati che guadagnano meno di 2000 e gli studenti possano consultare le tuple di **Esame** solo per esami il cui voto non è specificato o è superiore a 17. (Non sono cioè visualizzabili i voti insufficienti). Sarebbe possibile e come fare sì che sia studenti che impiegati possano consultare solo le loro informazioni e i professori solo le informazioni relativamente ai loro corsi?

Punto 5: Revoca dei privilegi. Provare ad eseguire almeno un comando di revoca di ruolo ed uno di revoca di privilegio che si propagherebbero ad altri utenti/ruoli analizzando le differenze di comportamento nel caso **CASCADE** rispetto a quello **RESTRICT**.