

Esercitazioni Basi di Dati

Marco Abbadini - 2023

marco.abbadini@unibg.it

<https://cs.unibg.it/abbadini>



Introduzione

Risolveremo esercizi mostrando:

- Interrogazioni nidificate
- Interrogazioni con aggregazione
- Interrogazioni con ricorsione
- Esercizi da temi d'esame

Interrogazioni SQL

- Linguaggio inizialmente dichiarativo, successivamente introdotti costrutti procedurali, controllo di flusso, etc..
- Specifica definita da standard:
 - SQL-86 SQL-89 SQL-92 SQL:1999 SQL:2003 SQL:2006 SQL:2008 SQL:2011 SQL:2016 -- consideriamo come riferimento la specifica 2008
- Diverse soluzioni sono presenti in prodotti reali che si differenziano per dettagli implementativi → consultare la documentazione ufficiale in fase di progetto
- Attenzione: si tratta di un prodotto industriale, aspetti come la performance sono fondamentali → non si lavora più con relazioni insiemistiche poiché eliminare i duplicati è oneroso

Struttura valida di una interrogazione

In quale ordine devono essere scritte le clausole in una interrogazione valida?

1. SELECT
2. FROM
3. WHERE
4. GROUP BY
5. HAVING
6. ORDER BY
7. LIMIT

SELECT e FROM sono i costrutti che fanno parte di ciascuna interrogazione

Ordine generale di esecuzione

Qual è l'ordine di esecuzione?

1. FROM
2. WHERE
3. GROUP BY
4. HAVING
5. SELECT
6. ORDER BY
7. LIMIT

Attenzione: alcune variazioni in caso di query nidificata

Esercizio 1

UTENTE (Codice, Nome, Cognome, Indirizzo, Telefono)

PRESTITO (Collocazione, CodUtente, DataPrestito, DataRest)

COPIA (Collocazione, ISBN, DataAcq)

DATILIBRO (ISBN, Titolo, AnnoPub, CasaEd, PrimoAut, Genere)

- A. Trovare i titoli dei libri di informatica prestati nel giugno '02 in ordine alfabetico (ogni titolo compaia una volta)
- B. Estrarre nome, cognome e codice degli utenti che non hanno mai preso in prestito libri di informatica
- C. Trovare il genere dei libri con il maggior numero di prestiti
- D. Trovare gli utenti che hanno preso in prestito libri di tutti i generi

Esercizio 2 - Tema d'esame 3 Febbraio 2003

CLIENTE(Id, Cognome, Nome, NumNoleggi, Tipologia)

FILM(Titolo, Regista, Genere, Durata)

CASSETTA(Id, Titolo)

DVD(Id, Titolo)

NOLEGGIO(IdCassODvd, CodCliente, DataPrestito, DataRestituzione)

- A. Estrarre in SQL tutti i film di cui sono stati noleggiati più volte i DVD che le videocassette
- B. Estrarre in SQL tutti i film le cui cassette sono state noleggiate più di venti volte nel mese di Gennaio ma non presenti in formato DVD
- C. Determinare il cognome e il nome dei clienti che hanno noleggiato tutti i film (in formato cassetta o DVD)
- D. Determinare il nome e cognome dei clienti che hanno noleggiato DVD solo di genere giallo

Interrogazioni ricorsive - Premessa

Ci sono alcune restrizioni sull'uso di interrogazioni ricorsive in una vista ricorsiva; nello specifico la query deve essere monotona, cioè il risultato sull'istanza I1 di una vista di una relazione deve essere un superset del suo risultato sull'istanza I2 se I1 è un superset di I2.

Intuitivamente, se alcune tuple vengono aggiunte ad una vista allora la query ricorsiva che viene eseguita sulla medesima vista deve almeno produrre lo stesso sottoinsieme di tuple prodotte in una precedente invocazione, e possibilmente alcune tuple in aggiunta.

Interrogazioni ricorsive - Restrizioni

Le query ricorsive non devono usare nessuno dei seguenti costrutti, in quanto la loro presenza potrebbe rendere la query non-monotona:

- Aggregazioni sulla vista ricorsiva
- **not exists** su una sotto-query che usa la vista ricorsiva
- Operatore **except** con operando destro (sottraendo) una vista ricorsiva

Esercizio 3 - Interrogazioni ricorsive

Dato lo schema

`prereq(id_corso, id_prereq)`

Trovare tutti i pre-requisiti del corso "ID0018"

Esercizio 4 - Ranking

Come risolvere in maniera semplice il problema del ranking

Dato lo schema

SQUADRA(nome, media_punti)

costruire una tabella che mostri il ranking delle squadre

Viene concesso l'utilizzo della sola funzione count()

Esercizio 5 - Tema d'esame 8 Settembre 2010

AEROPORTO(Id, Città, Nazione, NumPiste)

VOLO(NumeroVolo, GiornoSett, IdAeropPartenza, IdAeropArrivo, Compagnia, OraPartenza, OraArrivo, CodAereo)

AEREO(Codice, Tipo, NumPasseggeri)

- A. Esprimere in SQL la query che estrae la coppia di città tra cui viene offerta la maggiore capacità complessiva di trasporto passeggeri (ipotizzando che la capacità sia simmetrica, come si potrebbe rendere la query più efficiente?)
- B. Estrarre in SQL le compagnie che servono la città di Bergamo con più di settanta voli alla settimana, usando aerei di un solo tipo per tutti i loro voli