

Basi di Dati

More SQL

Michele Beretta

michele.beretta@unibg.it



Prova in itinere 20/04/2017

```
Studente(Matr, Nome, DataNascita, Città)
Corso(Codice, Titolo, CFU, Docente)
PianoStudi(MatrStud, CodCorso, Anno)
Esame(MatrStud, CodCorso, Data, Voto, NroProgr)
```

1. Estrarre gli studenti ordinati per la media dei voti pesata per CFU, solo per chi ha acquisito almeno 60 CFU in totale.
2. Assegnare in SQL a ciascun esame il numero progressivo nella carriera dello studente in base alla data di superamento.

Prova in itinere 20/04/2017

Estrarre gli studenti ordinati per la media dei voti pesata per CFU, solo per chi ha acquisito almeno 60 CFU in totale.

```
SELECT MatrStud, SUM(CFU * Voto) / SUM(CFU) AS MediaPesata
FROM Esame JOIN Corso ON CodCorso = Codice
GROUP BY MatrStud
HAVING SUM(CFU) >= 60
ORDER BY MediaPesata
```

Prova in itinere 20/04/2017

Assegnare in SQL a ciascun esame il numero progressivo nella carriera dello studente in base alla data di superamento.

```
UPDATE Esame E1
SET NroProgr = 1 + (
    SELECT COUNT(*)
    FROM Esame E2
    WHERE E1.MatrStud = E2.MatrStud AND E2.Data < E1.Data
)
```

Prova in itinere 19/04/2016

```
Studente(Matr, Nome, DataNascita, Città)
Corso(Codice, Titolo, CFU, Docente)
PianoStudi(MatrStud, CodCorso, Anno)
Esame(MatrStud, CodCorso, Data, Voto, NroProgr)
```

Per semplicità, suppongo di avere una funzione year che ritorna l'anno.

```
SELECT Matr, 110 / 30 * SUM(CFU * Voto) / SUM(CFU) as Media110
FROM Esame
  JOIN Corso ON CodCorso = Codice
  JOIN Studente ON MatrStud = Matr
WHERE year(DataNascita) = 1995
GROUP BY Matr
HAVING SUM(CFU) >= 100
```

Tema d'esame settembre 2010

Aeroporto(Id, Città, Nazione, NumPiste)

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

Aereo(Codice, Tipo, NumPasseggeri)

Esprimere in SQL la query che estrae la coppia di città tra cui viene offerta la maggior capacità complessiva di trasporto passeggeri.

Semplificazione: esiste un aereo bidirezionale tra ogni coppia di città.

Tema d'esame settembre 2010

```
CREATE VIEW Trasporto(Da, A, NumPasseggeri) AS
SELECT P.Città, A.Città, SUM(NumPasseggeri)
FROM Volo
    JOIN Aereo ON CodAereo = Codice
    JOIN Aeroporto P ON IdAeropPartenza = P.Id
    JOIN Aeroporto A ON IdAeropArrivo = A.Id
GROUP BY P.Città, A.Città

SELECT Da, A
FROM Trasporto
WHERE NumPasseggeri = (
    SELECT MAX(NumPasseggeri) FROM Trasporto
)
```

Tema d'esame febbraio 2003

```
Cliente(Id, Cognome, Nome, NumNoleggi, Tipologia)
Film(Titolo, Regista, Genere, Durata)
Cassetta(Id, Titolo)
Dvd(Id, Titolo)
Noleggio(IdCass0Dvd, CodCliente, DataPrestito, DataRestituzione)
```

1. Estrarre in SQL tutti i film di cui sono stati noleggiati più volte i DVD che le videocassette.
2. Estrarre in SQL tutti i film le cui cassette sono state noleggiate più di 20 volte nel mese di gennaio, ma che non sono presenti in formato DVD. Fare in modo che vengano stampati i titoli in ordine opposto a quello alfabetico e che vengano visualizzati solo quelli in posizione 2 e 3 della lista.
3. Determinare cognome e nome dei clienti che hanno noleggiato tutti i film.
4. Determinare cognome e nome dei clienti che hanno noleggiato DVD solo di genere giallo.

Tema d'esame febbraio 2003

Estrarre in SQL tutti i film di cui sono stati noleggiati più volte i DVD che le videocassette.

```
CREATE VIEW NolDvd(Titolo, Numero) AS
SELECT Titolo, COUNT(*)
FROM Dvd JOIN Noleggio ON Id = IdCassODvd
GROUP BY Titolo

CREATE VIEW NolCassetta(Titolo, Numero) AS
SELECT Titolo, COUNT(*)
FROM Cassetta JOIN Noleggio ON Id = IdCassODvd
GROUP BY Titolo

SELECT Titolo
FROM NolDvd D JOIN NolCassetta C ON D.Titolo = C.Titolo
WHERE D.Numero > C.Numero
```

Tema d'esame febbraio 2003

Estrarre in SQL tutti i film le cui cassette sono state noleggiate più di 20 volte nel mese di gennaio 2003, ma che non sono presenti in formato DVD. Fare in modo che vengano stampati i titoli in ordine opposto a quello alfabetico e che vengano visualizzati solo quelli in posizione 2 e 3 della lista.

```
SELECT Titolo
FROM Cassetta JOIN Noleggio ON Id = IdCass0Dvd
WHERE DataPrestito BETWEEN 1/1/2003 AND 31/1/2003
  AND Titolo NOT IN (SELECT Titolo FROM Dvd)
GROUP BY Titolo
HAVING COUNT(*) > 20
ORDER BY Titolo DESC
LIMIT 2
OFFSET 1
```

Tema d'esame febbraio 2003

Determinare cognome e nome dei clienti che hanno noleggiato tutti i film.

Nota: Noleggio usa IdCass0Dvd, quindi gli ID delle cassette e gli ID dei DVD sono *diversi e riconoscibili* e si può fare l'unione tra i due senza avere conflitti.

```
CREATE VIEW AllCassDvd(Id, Titolo) AS
SELECT Id, Titolo
FROM Cassetta
UNION
SELECT Id, Titolo
FROM Dvd
```

```
SELECT Cognome, Nome
FROM Cliente C
WHERE NOT EXISTS (
    SELECT *
    FROM Film F
    WHERE NOT EXISTS (
        SELECT *
        FROM Noleggio
        JOIN AllCassDvd ON IdCass0Dvd = Id
        WHERE Titolo = F.Titolo
        AND CodCliente = C.Id
    )
)
```

Tema d'esame febbraio 2003

Determinare cognome e nome dei clienti che hanno noleggiato DVD solo di genere giallo.

```
SELECT Cognome, Nome
FROM Cliente C
  JOIN Noleggio N ON C.Id = N.CodCliente
  JOIN Dvd D ON N.IdCassODvd = D.Id
WHERE NOT EXISTS (
  SELECT *
  FROM Noleggio N2
    JOIN Dvd D2 ON N2.IdCassODvd = D2.Id
    JOIN Film F ON D2.Titolo = F.Titolo
  WHERE N2.CodCliente = C.Id
    AND F.Genere <> 'Giallo'
)
```