

Linguaggio SQL: Esercizi (join, group by)

Esercizi join, group by

1. Sia dato il seguente schema relazionale (le chiavi primarie sono sottolineate, gli attributi opzionali sono indicati con “*”)

STUDENTE (MatrS, NomeS, Città)

CORSO (CodC, NomeC, MatrD)

DOCENTE (MatrD, NomeD)

ESAME (CodC, MatrS, Data, Voto)

- a) Per ogni studente, visualizzare la matricola e il voto massimo, minimo e medio conseguito negli esami

```
SELECT MatrS, MAX (Voto), MIN(Voto), AVG (Voto)
FROM ESAME E
GROUP BY MatrS
```

- b) Per ogni studente, visualizzare la matricola, il nome e il voto massimo, minimo e medio conseguito negli esami

```
SELECT S.MatrS, NomeS, MAX (Voto), MIN(Voto), AVG (Voto)
FROM ESAME E, STUDENTE S
WHERE S.MatrS = E.MatrS
GROUP BY S.MatrS, NomeS
```

- c) Per ogni studente che ha una media voti superiore al 28, visualizzare la matricola, il nome e il voto massimo, minimo e medio conseguito negli esami

```
SELECT S.MatrS, NomeS, MAX (Voto), MIN(Voto), AVG (Voto)
FROM ESAME E, STUDENTE S
WHERE S.MatrS = E.MatrS
GROUP BY S.MatrS, NomeS
HAVING AVG(Voto) > 28
```

- d) Per ogni studente della città di Torino che ha una media voti superiore al 28 e ha sostenuto esami in almeno 10 date diverse, visualizzare la matricola, il nome e il voto massimo, minimo e medio conseguito negli esami

```
SELECT S.MatrS, NomeS, MAX (Voto), MIN(Voto), AVG (Voto)
FROM ESAME E, STUDENTE S
WHERE S.MatrS = E.MatrS and Città = ‘Torino’
GROUP BY S.MatrS, NomeS
HAVING AVG(Voto) > 28 AND COUNT(DISTINCT Data)>10
```

2. Sia dato il seguente schema relazionale (le chiavi primarie sono sottolineate, gli attributi opzionali sono indicati con “*”)

PERSONA (CodFisc, NomeP, DataNascita)

ISTRUTTORE (CodI, NomeI)

LEZIONE_PRIVATA (CodFisc, Data, Ora, CodI)

a) Per ogni persona visualizzare il codice fiscale e il numero di lezioni frequentate

```
SELECT CodFisc, COUNT(*), COUNT(DISTINCT CodI)
FROM LEZIONE_PRIVATA LP
GROUP BY CodFisc
```

b) Per ogni persona visualizzare il codice fiscale, il nome e il numero di lezioni frequentate

```
SELECT P.CodFisc, NomeP, COUNT(*)
FROM LEZIONE_PRIVATA LP, PERSONA P
WHERE P.CodFisc = LP.CodFisc
GROUP BY P.CodFisc, NomeP
```

c) Per ogni persona visualizzare il codice fiscale, il nome, il numero di lezioni frequentate e il numero di istruttori (diversi) con cui ha fatto lezione

```
SELECT P.CodFisc, NomeP, COUNT(*), COUNT(DISTINCT CodI)
FROM LEZIONE_PRIVATA LP, PERSONA P
WHERE P.CodFisc = LP.CodFisc
GROUP BY P.CodFisc, Nome
```

d) Per ogni persona nata dopo il 1970 che ha frequentato almeno 5 lezioni, visualizzare il codice fiscale, il nome, il numero di lezioni frequentate e il numero di istruttori (diversi) con cui ha fatto lezione

```
SELECT P.CodFisc, NomeP, COUNT(*), COUNT(DISTINCT CodI)
FROM LEZIONE_PRIVATA LP, PERSONA P
WHERE P.CodFisc = LP.CodFisc AND DataNascita > 1970
GROUP BY P.CodFisc, NomeP
HAVING COUNT(*)>=5
```

3. Sia dato il seguente schema relazionale (le chiavi primarie sono sottolineate, gli attributi opzionali sono indicati con “*”)

CORSO (CodCorso, NomeC, Anno, Semestre)

ORARIO-LEZIONI (CodCorso, GiornoSettimana, OraInizio, OraFine, Aula)

- a) Trovare codice corso, nome corso e numero totale di ore di lezione settimanali per i corsi del terzo anno per cui il numero complessivo di ore di lezione settimanali è superiore a 10 e le lezioni sono in più di tre giorni diversi della settimana.

```
SELECT C.CodCorso, NomeC, SUM(OraFine-OraInizio)
FROM CORSO C, ORARIO-LEZIONI OL
WHERE C.CodCorso = OL.CodCorso AND Anno = 3
GROUP BY C.CodCorso, NomeC
HAVING SUM(OraFine-OraInizio) >10 AND COUNT(DISTINCT GiornoSettimana)>3
```

4. Sia dato il seguente schema relazionale (le chiavi primarie sono sottolineate, gli attributi opzionali sono indicati con “*”)

ALLOGGIO (CodA, Indirizzo, Città, Superficie, CostoAffittoMensile)

CONTRATTO-AFFITTO (CodC, DataInizio, DataFine*, NomePersona, CodA)

N.B. Superficie espressa in metri quadri. Per i contratti in corso, DataFine è NULL.

- a) Trovare il nome delle persone che hanno stipulato più di due contratti di affitto per lo stesso appartamento (in tempi diversi).

```
SELECT NomePersona
FROM CONTRATTO-AFFITTO
GROUP BY CodA, NomePersona
HAVING COUNT(*)>2
```

- b) Trovare, per le città in cui sono stati stipulati almeno 100 contratti, la città, il costo mensile massimo degli affitti, il costo mensile medio degli affitti, la durata massima dei contratti, la durata media dei contratti e il numero totale di contratti stipulati.

```
SELECT Città, MAX(CostoAffittoMensile), AVG(CostoAffittoMensile),
MAX(DataFine-DataInizio), AVG(DataFine-DataInizio), COUNT(*)
FROM ALLOGGIO A, CONTRATTO-AFFITTO CA
WHERE A.CodA=CA.CodA
GROUP BY Città
HAVING COUNT(*) >=100
```

Esercizi query annidate (in, not in, exists, not exists)

1. Sia dato il seguente schema relazionale (le chiavi primarie sono sottolineate, gli attributi opzionali sono indicati con “*”)

ORCHESTRA (CodO, NomeO, NomeDirettore, NumElementi)

SALA (CodS, NomeS, Città, Capienza)

CONCERTI (CodC, Data, CodO, CodS, PrezzoBiglietto)

- a) Trovare il codice e il nome delle orchestre con più di 30 elementi che hanno tenuto concerti sia a Torino, sia a Milano e non hanno mai tenuto concerti a Bologna.

Soluzione con operatori IN e NOT IN

```
SELECT O.CodO, NomeO
FROM ORCHESTRA O
WHERE NumElementi > 30
AND O.CodO IN ( SELECT CodO
                FROM SALA S, CONCERTI C
                WHERE Città = 'Torino' AND S.CodS=C.CodS)
AND O.CodO IN ( SELECT CodO
                FROM SALA S, CONCERTO C
                WHERE Città = 'Milano' AND S.CodS=C.CodS)
AND O.CodO NOT IN (SELECT CodO
                  FROM SALA S, CONCERTO C
                  WHERE Città = 'Bologna' AND S.CodS=C.CodS)
```

Soluzione con operatori EXISTS e NOT EXISTS

```
SELECT O.CodO, NomeO
FROM ORCHESTRA O
WHERE NumElementi > 30
AND EXISTS (SELECT *
            FROM SALA SA, CONCERTI CA
            WHERE Città = 'Torino' AND SA.CodS=CA.CodS
            AND CA.CodO = O.CodC) CONDIZIONE DI CORRELAZIONE
AND EXISTS (SELECT *
            FROM SALA SB, CONCERTO CB
            WHERE Città = 'Milano' AND SB.CodS=CB.CodS
            AND CB.CodO = O.CodC) CONDIZIONE DI CORRELAZIONE
AND NOT EXISTS (SELECT *
                FROM SALA SC, CONCERTO CC
                WHERE Città = 'Bologna' AND SC.CodS=CC.CodS
                AND CC.CodO = O.CodC) CONDIZIONE DI CORRELAZIONE
```

2. Sia dato il seguente schema relazionale (le chiavi primarie sono sottolineate, gli attributi opzionali sono indicati con “*”)

CORSO (CodCorso, NomeC, Anno, Semestre)

ORARIO-LEZIONI (CodCorso, GiornoSettimana, OraInizio, OraFine, Aula)

a) Trovare le aule in cui non si tengono mai lezioni di corsi del primo anno.

Soluzione con operatore NOT IN

```
SELECT DISTINCT Aula
FROM ORARIO-LEZIONI
WHERE Aula NOT IN (SELECT Aula
                   FROM CORSO C, ORARIO-LEZIONI OL
                   WHERE Anno = 1 AND OL.CodCorso = C.CodCorso)
```

Soluzione con operatore NOT EXISTS

```
SELECT DISTINCT Aula
FROM ORARIO-LEZIONE OL1
WHERE NOT EXISTS
  (SELECT *
   FROM CORSO C, ORARIO-LEZIONI OL
   WHERE Anno =1 AND OL.CodCorso = C.CodCorso
   AND OL.Aula = OL1.Aula ) CONDIZIONE DI CORRELAZIONE
```

3. Sia dato il seguente schema relazionale (le chiavi primarie sono sottolineate, gli attributi opzionali sono indicati con “*”)

ALLOGGIO (CodA, Indirizzo, Città, Superficie)

CONTRATTO-AFFITTO (CodC, DataInizio, DataFine*, NomePersona, CodA, RettaMensile)

- a) Trovare il nome delle persone che non hanno mai affittato alloggi con superficie superiore a 80 metri quadri.

Soluzione con operatore NOT IN

```
SELECT DISTINCT NomePersona
FROM CONTRATTO-AFFITTO
WHERE NomePersona NOT IN
      (SELECT CA.NomePersona
       FROM CONTRATTO-AFFITTO CA, ALLOGGIO A
       WHERE A.Superficie > 80 AND CA.CodA = A.CodA)
```

Soluzione con operatore NOT EXISTS

```
SELECT DISTINCT NomePersona
FROM CONTRATTO-AFFITTO CA1
WHERE NOT EXISTS (SELECT *
                  FROM CONTRATTO-AFFITTO CA, ALLOGGIO A
                  WHERE A.Superficie > 80
                  AND CA.CodA = A.CodA
                  AND CA.NomePersona= CA1.NomePersona)
      CONDIZIONE DI CORRELAZIONE
```

- b) Trovare il codice e l'indirizzo degli appartamenti di Torino in cui la retta mensile è sempre stata superiore a 500 euro e per cui sono stati stipulati al più 5 contratti di affitto.

```
SELECT A.CodA, Indirizzo
FROM APPARTAMENTO A, CONTRATTO-AFFITTO CA
WHERE A.CodA=CA.CodA AND Citta='Torino'
AND A.CodA NOT IN
      (SELECT CodA
       FROM CONTRATTO-AFFITTO
       WHERE RettaMensile<=500)
GROUP BY A.CodA, Indirizzo
HAVING COUNT(*)<=5
```

Esercizi con condizione di correlazione

1. Sia dato il seguente schema relazionale (le chiavi primarie sono sottolineate, gli attributi opzionali sono indicati con “*”)

ALLOGGIO (CodA, Indirizzo, Città, Superficie)
 CONTRATTO-AFFITTO (CodC, DataInizio, DataFine, NomePersona, CodA, CostoAffittoMensile)

- a) Trovare il codice, l'indirizzo e la città degli alloggi che hanno una superficie superiore alla superficie media degli alloggi delle città in cui si trovano.

```
SELECT CodA, Indirizzo, Città
FROM ALLOGGIO A1
WHERE Superficie > (SELECT AVG(Superficie)
                    FROM ALLOGGIO A2
                    WHERE A2.Città=A1.Città) CONDIZIONE DI CORRELAZIONE
```

2. Sia dato il seguente schema relazionale (le chiavi primarie sono sottolineate, gli attributi opzionali sono indicati con “*”)

AEREI (Matr, Modello, NumPosti)
 ORARIO (Sigla, CittàPartenza, CittàArrivo, OraPart, OraArr)
 VOLI (Siglia, Matr, Data, PostiPren)

- a) Trovare le tratte (città di partenza, città di arrivo) che non sono state mai effettuate con un aereo modello Boing-747.

Soluzione con costruttore di tupla

```
SELECT DISTINCT CittàPartenza, CittàArrivo
FROM ORARIO
WHERE (CittàPartenza, CittàArrivo) NOT IN
      (SELECT CittàPartenza, CittàArrivo
       FROM AEREI A, VOLI V, ORARIO O
       WHERE Modello=747
       AND V.Matr = A.Matr AND V.Sigla = O.Sigla)
```

Soluzione con operatore NOT EXISTS

```
SELECT DISTINCT CittàPartenza, CittàArrivo
FROM ORARIO O1
WHERE NOT EXISTS
  (SELECT *
   FROM AEREI A, VOLI V, ORARIO O
   WHERE Modello=747
   AND V.Matr = A.Matr AND V.Sigla = O.Sigla
   AND O.CittàPartenza=O1.CittàPartenza AND O.CittàArrivo=O1.CittàArrivo)
CONDIZIONI DI CORRELAZIONE
```