

Esercizi SQL del 7 Dicembre 2022

Dati i seguenti schemi di relazione (le chiavi primarie sono sottolineate, le chiavi esterne sono date esplicitamente):

- Dipendenti(Id, Nome, AnnoNascita, Stipendio)
- Manager(Id*, Area)
Id FK(Dipendenti)
- Staff(Id*, Cod*)
Id FK(Dipendenti) Cod FK(Progetti)
- Progetti(Cod, Nome, Budget, Responsabile*)
Responsabile FK(Manager)

(i) Scrivere in SQL le seguenti interrogazioni:

1. trovare il nome e l'anno di nascita dei dipendenti del progetto *Master*;
2. per ogni manager, trovare il nome, l'età e il budget complessivo, ottenuto sommando i budget dei vari progetti che coordina. Ordinare il risultato per il budget complessivo;
3. eliminare i progetti che hanno solo il responsabile come dipendente e il corrispondente record in Staff;
4. aumentare del 20% lo stipendio dei dipendenti che lavorano in almeno due progetti e in almeno uno è il manager;
5. trovare il nome e lo stipendio dei manager dei progetti che hanno più di 30 dipendenti e un budget superiore a 500000 euro;
6. trovare il nome del progetto/progetti avente la più grande differenza fra lo stipendio di un dipendente del progetto e di quello del manager del progetto;
7. per ogni manager, contare il numero di progetti con numero di dipendenti maggiori di 100 e con un budget superiore a 1.000.000 euro. Se un manager non ha alcun progetto con queste caratteristiche restituire 0;
8. per ogni area, trovare il nome e il numero dei progetti del manager (o dei manager) che coordina il numero massimo di dipendenti. Se un dipendente occorre in più progetti deve essere contato una sola volta e il manager non coordina se stesso.

(ii) scrivere in Algebra relazionale la prima interrogazione.

Soluzione:

1. Trovare il nome e l'anno di nascita dei dipendenti del progetto *Master*.

```
SELECT d.Nome, d.AnnoNascita
FROM   Dipendenti d NATURAL JOIN Staff s JOIN Progetti p USING (Cod)
WHERE  p.Nome = 'Master'
```

$$\pi_{\text{Nome}, \text{AnnoNascita}}(\sigma_{\text{NomeP}='Master'}(\text{Dipendenti} \bowtie \text{Staff} \bowtie \rho_{\text{Nome} \leftarrow \text{NomeP}}(\text{Progetti})))$$

2. Per ogni manager, trovare il nome, l'età e il budget complessivo, ottenuto sommando i budget dei vari progetti che coordina. Ordinare il risultato per il budget complessivo.

```
SELECT   d.Nome, 2022 - d.AnnoNascita AS Eta, SUM(p.Budget) AS BudgetTotale
FROM     Progetti p JOIN Dipendenti d ON p.Responsabile = d.Id
GROUP BY p.Responsabile, d.Nome, d.AnnoNascita
ORDER BY BudgetTotale
```

3. Eliminare i progetti che hanno solo il responsabile come dipendente e il corrispondente record in Staff.

```
DELETE FROM Staff s
WHERE NOT EXISTS (SELECT *
                  FROM Staff s1
                  WHERE s1.Cod = s.Cod AND s1.Id <> s.Id)
AND s.id = (SELECT p.Responsabile
            FROM Progetti p
            WHERE p.Cod = s.Cod)
```

```
DELETE FROM Progetti
WHERE Cod NOT IN (SELECT Cod
                  FROM Staff)
```

4. Aumentare del 20% lo stipendio dei dipendenti che lavorano in almeno due progetti e in almeno uno è il manager.

```
UPDATE Dipendenti
SET Stipendio = Stipendio * 1.2
WHERE Id IN (SELECT p.Responsabile
             FROM Progetti p JOIN Staff s ON p.Responsabile = s.Id
             WHERE p.Cod <> s.Cod)
```

5. Trovare il nome e lo stipendio dei manager dei progetti che hanno più di 30 dipendenti e un budget superiore a 500000 euro.

```
SELECT DISTINCT m.Id, m.Nome, m.Stipendio
FROM Progetti p JOIN Dipendenti m ON p.Responsabile = m.Id
      JOIN Staff s USING (Cod)
WHERE p.Budget > 500000
GROUP BY p.Cod, m.Id, m.Nome, m.Stipendio
HAVING COUNT(*) > 30
```

6. Trovare il nome del progetto/progetti avente la più grande differenza fra lo stipendio di un dipendente del progetto e di quello del manager del progetto;

```
SELECT DISTINCT p.Nome
FROM Progetti p JOIN Dipendenti m ON p.Responsabile = m.Id JOIN Staff s USING(Cod)
      JOIN Dipendenti d ON d.Id = s.Id
WHERE m.Stipendio - d.Stipendio = (SELECT MAX(m1.Stipendio - d1.Stipendio)
                                FROM Progetti p1 JOIN Dipendenti m1 ON
                                      p1.Responsabile = m1.Id JOIN Staff s1
                                      USING(Cod) JOIN Dipendenti d1
                                      ON d1.Id = s1.Id)
```

7. per ogni manager, contare il numero di progetti con numero di dipendenti maggiori di 100 e con un budget superiore a 1.000.000 euro. Se un manager non ha alcun progetto con queste caratteristiche restituire 0;

```
WITH ProgettiRilevanti(Cod, Responsabile)
AS (SELECT p.Cod, p.Responsabile
```

```

FROM      Progetti p NATURAL JOIN Staff s
WHERE      p.Budget > 1000000
GROUP BY  p.Cod, p.Responsabile
HAVING     COUNT(*) > 100)

SELECT     m.Id, COUNT(p.Cod) AS NumProgRilevanti
FROM Manager m LEFT JOIN ProgettiRilevanti p ON m.Id = p.Responsabile
GROUP BY m.Id

```

8. Per ogni area, trovare il nome e il numero dei progetti del manager (o dei manager) che coordina il numero massimo di dipendenti. Se un dipendente occorre in più progetti deve essere contato una sola volta e il manager non coordina se stesso.

```

WITH ManagerCoordina(Id, Area, NumProg, NumDip)
AS ( SELECT     m.Id, m.Area, COUNT(DISTINCT p.Cod), COUNT(DISTINCT s.Id)
      FROM       Progetti p JOIN Staff s ON s. Cod = p. Cod AND s.Id <> p.Responsabile
      JOIN Manager m ON p.Responsabile = m.Id
      GROUP BY m.Id, m.Area)

SELECT mc.Area, d.Nome, mc.NumProg
FROM   ManagerCoordina mc NATURAL JOIN Dipendenti d
WHERE mc.NumDip = (SELECT MAX(mc1.NumDip)
                  FROM   ManagerCoordina mc1
                  WHERE mc.Area = mc1.Area)

```