

Matricola	Nome e Cognome	Valutazione

a) Si consideri il seguente schema relazionale relativo alla gestione delle riunioni di un dipartimento:

**R** Riunione(id, iddipartimento data, ora)

**O** Odg(idriunione, idodg, descrizione, docente)

**D** Docente(id, iddipartimento nome, cognome)

**PR** presentiRiunione(idriunione, iddocente)

1. identificare le chiavi primarie ed esterne dello schema [0 punti corretta, -1 sbagliata]
2. Rispondere alle seguenti query in algebra relazionale
  - a. Trovare i docenti, mostrando nome e cognome, che hanno partecipato a tutte le riunioni [4 punti]
  - b. Trovare le riunioni che non hanno esattamente due record in odg [4 punti]
3. Rispondere alle seguenti query in SQL
  - a. Per ogni docente trovare il numero di riunioni a cui non ha partecipato [4 punti]
  - b. Trovare gli anni in cui il numero complessivo di partecipanti alle riunioni è stato maggiore di quello medio [4 punti]
4. Implementare una asserzione che eviti di far inserire nella tabella "presentiRiunione" docenti che appartengono a dipartimenti differenti rispetto a quello dove la riunione è stata indetta [5 punti]

Trovare i docenti, mostrando nome e cognome, che hanno partecipato a tutte le riunioni

$$R_1 = \pi_{D.id, D.nome, D.cognome, PR.idriunione} (D \bowtie PR)$$

$D.id = PR.iddocente$

$$R_2 = \pi_{id} (\sigma_{idriunione \neq id} (R))$$

$$\pi_{nome, cognome} (R_1 \div R_2)$$

Trovare le riunioni che non hanno esattamente due record in odg

$$R1 = (R \bowtie O)$$

$R.id = O.id_{riunione}$

$$R2 = R1$$

$$R3 = (R1 \times R2)$$

$R1.id_{riunione} = R2.id_{riunione}$   
 $R1.odg \overset{AND}{=} R2.odg$

$$\pi_{R2.id} (R2 \setminus R3)$$

SOL

Per ogni docente trovare il numero di riunioni a cui non ha partecipato

SELECT D.id, COUNT(PR.id\_riunione) AS conteggio

FROM Docente D, presenza Riunione PR

WHERE D.id NOT IN (SELECT D1.id

FROM Docente D1, presenza Riunione PR1

WHERE D1.id = PR1.id\_docente)

GROUP BY D.id

Trovare gli anni in cui il numero complessivo di partecipanti alle riunioni è stato maggiore di quello medio [4 punti]

Numero di partecipanti alle riunioni

```
Create VIEW numeroPresenti AS
```

```
Select R.data, count(PR.idlocaute) AS Presenti
```

```
From presentiRiunione PR, Riunione R
```

```
Where R.id = PR.idriunione
```

```
Group by R.data
```

```
Select NP.data
```

```
From NumeroPresenti NP
```

```
Where NP.Presenti > ( Select AVG (NP2.presenti)
```

```
From NumeroPresenti NP2
```

```
Where NP.data = NP2.data )
```

Implementare una asserzione che eviti di far inserire nella tabella "presentiRiunione" docenti che appartengono a dipartimenti differenti rispetto a quello dove la riunione è stata indetta [5 punti]

Create assertion stesso dipartimento

check NOT EXIST ( Select PR. id docente

From presentiRiunione PR, Docente D, Riunione R

Where PR. id docente = D. id

AND PR. id riunione = R. id

AND D. dipartimento <> R. dipartimento )