

Prova intermedia di Sistemi Informativi T del 2 aprile 2019 (matr. Pari – C)

Date le relazioni:

PROGETTI

<u>CODICE</u>	NOME	BUDGET	TIPO
PYZ12	IGUANA	250.000	STRATEGY
PXX21	GANIMEDE	180.000	FUTURE
PYY12	ANNAPURNA	845.000	STRATEGY
...

ASSEGNAIMENTI

<u>IMPIEGATO</u>	<u>PROGETTO</u>	SEDE	PERC-TEMPO
CARLO ROSSI	PXX21	ROMA	35
ANNA CORLI	PYK32	BOLOGNA	80
MARTA VERDI	PXY10	FIRENZE	65
...

ES 1. Si scriva una vista SQL **TTP (PTIPO, NPROG, BMED)** che, considerando solamente progetti attivi presso le sedi di Roma e Milano, restituisca per ogni tipo di progetto il numero totale di progetti di quel tipo ed il loro budget medio, se tale budget medio risulta inferiore a 500000 Euro.

```
CREATE VIEW TTP (PTIPO, NPROG, BMED)
AS SELECT TIPO, COUNT(*), AVG (BUDGET)
   FROM PROGETTI
   WHERE CODICE IN (SELECT PROGETTO FROM ASSEGNAIMENTI
                    WHERE SEDE='Roma' OR SEDE='Milano')
   GROUP BY TIPO HAVING AVG (BUDGET) < 500000
```

Oppure, es:

```
CREATE VIEW TTP (PTIPO, NPROG, BMED)
AS SELECT TIPO, COUNT(*), AVG (BUDGET)
   FROM PROGETTI
   WHERE CODICE IN (SELECT PROGETTO FROM ASSEGNAIMENTI
                    WHERE SEDE IN ('Roma', 'Milano'))
   GROUP BY TIPO HAVING AVG (BUDGET) <= 49999
```

ES 2. Utilizzando anche la vista **TTP**, si scriva una query SQL che restituisca l'elenco (con nome, tipo e budget) di tutti i progetti del tipo che ha il più basso valore del budget medio così come calcolato dalla vista.

```
SELECT NOME, TIPO, BUDGET
   FROM PROGETTI WHERE TIPO IN (SELECT PTIPO FROM TTP
                                WHERE BMED <= ALL (SELECT BMED FROM TTP))
```

Oppure, es:

```
SELECT NOME, TIPO, BUDGET
   FROM PROGETTI JOIN TTP T ON TIPO=PTIPO
   WHERE NOT EXISTS (SELECT * FROM TTP WHERE BMED > T.BMED)
```

ES 3 (facoltativo). Si forniscano (dietro al foglio) definizioni e vincoli fondamentali del modello relazionale (chiave primaria, chiave esterna, entity integrity, referential integrity).

Risultati su AlmaEsami

MATRICOLA	
COGNOME E NOME	
FIRMA	

Prova intermedia di Sistemi Informativi T del 2 aprile 2019 (matr. Pari – D)

Date le relazioni: **IMPIEGATI**

<u>MATRICOLA</u>	<u>NOME</u>	<u>STIPENDIO</u>	<u>QUALIFICA</u>
EYZ12	CARLO ROSSI	2.500	PROGRAMMATORE
EXX21	ANNA CORLI	3.800	PROGETTISTA ER
EYY12	MARTA VERDI	3.250	DBA
...

ASSEGNAIMENTI

<u>IMPIEGATO</u>	<u>PROGETTO</u>	<u>SEDE</u>	<u>PERC-TEMPO</u>
EYY12	IGUANA	ROMA	35
EXX21	GANIMEDE	BOLOGNA	80
EYY12	ANNAPURNA	FIRENZE	65
...

ES 1. Si scriva una vista SQL **QQL (QUAL, NIMP, SMED)** che restituisca per ogni tipo di qualifica il numero totale di impiegati aventi tale qualifica ed il loro stipendio medio, se tale stipendio medio risulta di almeno 2000 Euro.

```
CREATE VIEW QQL (QUAL, NIMP, SMED)
AS SELECT QUALIFICA, COUNT(*), AVG(STIPENDIO)
   FROM IMPIEGATI
   GROUP BY QUALIFICA HAVING AVG(STIPENDIO) > 1999
```

ES 2. Utilizzando anche la vista **QQL**, si scriva una query SQL che restituisca l'elenco (con nome, qualifica e stipendio) di tutti gli impiegati che non lavorano presso le sedi di Roma e Milano ed aventi la qualifica che ha il più elevato valore dello stipendio medio così come calcolato dalla vista.

```
SELECT NOME, QUAL, STIPENDIO
   FROM IMPIEGATI JOIN QQL ON QUALIFICA=QUAL
  WHERE MATRICOLA NOT IN (SELECT IMPIEGATO FROM ASSEGNAIMENTI
                          WHERE SEDE IN ('Roma', 'Milano'))
     AND SMED >= ALL (SELECT SMED FROM QQL)
```

Oppure, es:

```
SELECT NOME, QUAL, STIPENDIO
   FROM IMPIEGATI
  WHERE MATRICOLA NOT IN (SELECT IMPIEGATO FROM ASSEGNAIMENTI
                          WHERE SEDE='Roma' OR SEDE='Milano')
     AND QUALIFICA IN (SELECT QUAL FROM QQL
                      WHERE SMED = (SELECT MAX(SMED) FROM QQL))
```

ES 3 (facoltativo). Si forniscano (dietro al foglio) definizioni e vincoli fondamentali del modello relazionale (chiave primaria, chiave esterna, entity integrity, referential integrity).

Risultati su AlmaEsami

MATRICOLA	
COGNOME E NOME	
FIRMA	