

Matricola	Nome e Cognome	Valutazione

Si consideri il seguente schema relazionale:

ImpresaEdile(id, ragionesociale, idtitolare)

Lavoratore(id, nome, cognome, impresa)

Cliente(id, nome, cognome, immobile)

Cantiere(id, idAzienda, idImmobile, data_inizio, data_fine)

LavoratoriCantiere(idCantiere, idLavoratore, ruolo, oreTotali)

1. Identificare le chiavi primarie ed esterne dello schema [0 corrette, -1 errate]

2. Algebra

- a. Identificare le imprese che non hanno nessun cantiere [3 punti].

PROJ_id(Impresa) – RID_{idAzienda->id}(PROJ_idAzienda(Cantiere))

- b. Per ogni cantiere identificare i lavoratori che hanno lavorato per più ore [4 punti].

$c_1 := S_{1, \text{controle}} \rightarrow \text{id}_C$, (Lavoratore control)
 $\text{id}_{\text{Lavoratore}} \rightarrow \text{id}_L$
 $\text{fond } \rightarrow r_1$,
 $\text{gretatore} \rightarrow \text{gret}$

$$C_2 := \overline{\prod_{\substack{\text{ideale} \\ \text{arbeits} \\ \text{laborator}}} \left(\mathcal{G} \left(\text{Lavoratori, Contiere} \times C_1 \right) \middle| \begin{array}{l} \text{ideale Contiere} = \text{ideale} \wedge \\ \text{arbeitslabor} < \text{delt} \end{array} \right)}$$

$$\pi_{\text{LavoroTo-Contro} - C_2}^{\text{Cust}} \left(\text{LavoroTo-Contro} - C_2 \right)$$

Matricola	Nome e Cognome	Valutazione

3. SQL

- a. Identificare i clienti che hanno avuto il maggior numero di cantieri, indicare anche il numero complessivo dei lavoratori per tutti i cantieri di quel cliente [5 punti].

```
SELECT count(DISTINCT ca.id) x, count(lc.idLavoratore), c.id
FROM Cantiere ca, Cliente c, LavoratoriCantiere lc
WHERE c.immobile=ca.idImmobile AND lc.idCantiere = ca.id
GROUP BY c.id
HAVING x >= ALL (SELECT count(DISTINCT ca.id)
FROM Cantiere ca, Cliente c, WHERE c.immobile=ca.idImmobile
GROUP BY c.id)
```

- b. Trovare i cantieri che sono stati aperti per più tempo, per questi indicare anche il numero di ore complessive di tutti i lavoratori impiegati in tale cantiere [6 punti].

```
SELECT cantiere.id, sum(oretotali)
FROM cantiere, lavoratoriCantiere
WHERE data_fine IS NOT NULL AND data_fine-data_inizio >= SELECT
MAX(data_fine-data_inizio) FROM cantiere WHERE data_fine IS NOT
NULL) ANDcantiere.id=lavoratoriCantiere.idCantiere
GROUP BY cantiere.id
```

4. Trigger

Implementare un trigger che, quando la data di fine lavori di un cantiere è valorizzata, calcoli per ogni lavoratore in tale cantiere il numero di ore totali lavorate. Si consideri una giornata lavorativa standard di **8 ore**. Ignorare la possibilità di avere domeniche e festivi [5 punti].

```
CREATE TRIGGER T1
AFTER UPDATE OF data_fine on CANTIERE
FOR EACH ROW
UPDATE lavoratoriCantiere
SET oretotali=(new.data_fine-new.data_inizio)*8
WHERE idCantiere=new.id
```

- a. Identificare i clienti che hanno avuto il maggior numero di cantieri, indicare anche il numero complessivo dei lavoratori per tutti i cantieri di quel cliente [5 punti].

Select cl.id, count (distinct ca.id) conteggio, count (distinct lc.id) lavoratori
From Cliente cl, Cantieri ca, LavoratoriCantieri lc

Where cl.immobile = ca. immobile

AND ca.id = lc.idcantiere

Group by cl.id

HAVING conteggio >= ALL (Select count (distinct ca.id))

From cantieri ca1, Cliente cl1

Where cl1.immobile = ca1. immobile

Group by cl1.id

)

Trovare i cantieri che sono stati aperti per più tempo, per questi indicare anche il numero di ore complessive di tutti i lavoratori impiegati in tale cantiere [6 punti].

Select ca.id, sum (lc.oretotali)

From Cantiere ca, LavoratoriCantiere lc

Where lc.idcantiere = ca.id

AND (ca.data_fine - ca.data_inizio) = (Select max (ca.data_fine - ca.data_inizio))

From Cantiere ca)

Group by ca.id

ImpresaEdile(id, ragionesociale, idtitolare)

Lavoratore(id, nome, cognome, impresa)

Cliente(id, nome, cognome, immobile)

Cantiere(id, idAzienda, idImmobile, data_inizio, data_fine)

LavoratoriCantiere(idCantiere, idLavoratore, ruolo, oretotali)