

Escursioni (id, titolo, descrizione, durata, difficoltà, costo)

Date Escursione (id, data, id\_escursione, id\_guida)

Partecipante (id\_Partecipante, id\_Escursione)

Persone (id, nome, cognome)

Trovare le escursioni che hanno un costo massimo

ALGEBRA

Per confrontare elementi delle stesse tabelle devo fare sempre una self-join

$R_1 = \text{escursione}$

$R_2 = \text{escursione}$

$R_3 = \sigma_{R_1.\text{costo} < R_2.\text{costo}}(R_1 \times R_2)$

$R_4 = \pi_{id, titolo, diff}(escursione) - \pi_{id, titolo, diff}(R_3)$

id	Cost
1	100
2	200
3	300

 $\times$ 

id	Cost
1	100
2	200
3	300

SQL

Select titolo, descrizione, difficoltà

From Escursione

Where costo = (Select max(cost) escursioni)

trovare i partecipanti che hanno partecipato a tutte le escursioni

## Algebra

Partecipante

idPort	idEsc
1	1
1	3
2	1
1	2
1	1
3	1
3	2

Dobbiamo usare il quoziente perché quello che fa è: prodotto cartesiano di un idPort con tutte le escursioni se il risultato è un sottoinsieme delle nostre tabelle allora quel idPort va in output

$$R_1 = \pi_{idEsc \rightarrow id} (Partecipante \div \pi_{id} (Escursione))$$

$$R_2 = \pi_{nome, cognome} (R_1 \bowtie_{idPort = id} Persone)$$

## SQL

Select Nome, Cognome

From Partecipante  $p_1$ , Persone  $p_2$  ON  $p_1.idPort = p_2.id$

Where Not Exist (

Select \* from escursione  $e$  where not exist (

Select \* from Partecipante  $p_2$

where  $p_2.idPort = p_1.idPort$  AND  $p_2.idEscursione = e.id$

)

)

trovare le guide che non hanno mai partecipato ad escursioni di difficoltà massima

### Algebra

$R_1 = \text{Escursioni}$

$R_2 = \text{Escursioni}$

$DE = \text{Date Escursione}$

$R_3 = \pi_{id} (R_1) - \pi_{R_1.id} (\sigma_{R_1.difficulta < R_2.difficulta} (R_1 \times R_2))$

$R_4 = \pi_{idguide} (DE \bowtie_{DE.idescursione = R_3.id})$

$\pi_{idguide} (DE) - R_4$

### SQL

SELECT idguide

FROM Dateescursione de1

WHERE NOT EXISTS (SELECT \* FROM escursione e

WHERE e.difficulta = (SELECT max(diff)  
FROM escursione))

AND exist (SELECT \* FROM Dateescursione de2

WHERE de2.idese = de1.idese

AND de2.idguide = de2.idguide))

Trovare le coppie di persone che hanno partecipato sempre alle stesse escursioni

Algebra:

$$R_1 = \rho_{idPort \Rightarrow idP_1}(\text{partecipante})$$

$$R_2 = \rho_{idPort \Rightarrow idP_2}(\text{partecipante})$$

$$R_3 = \pi_{idP_1, idP_2} (R_1 \bowtie_{R_1.idP_1 \neq R_2.idP_2 \wedge R_1.idEscursione = R_2.idEscursione} R_2)$$

SQL

Guarda le slide e tempo lungo