

Domanda **1**

Risposta mancante

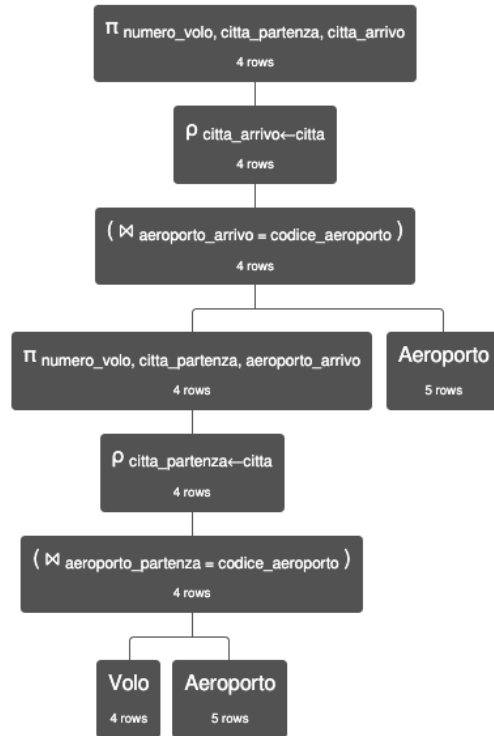
Punteggio max.: 1,00

Con riferimento ad una base di dati sullo schema che includa le relazioni (sono indicati solo gli attributi rilevanti)

VOLO(numero_volo, aeroporto_partenza, aeroporto_arrivo)

AEROPORTO (codice_aeroporto, nome, citta)

scrivere l'interrogazione SQL equivalente alla seguente interrogazione in algebra relazionale (ordinare il risultato per numero_volo)



Answer: (penalty regime: 10, 20, ... %)

1 ||

Verifica risposta

Domanda **2**

Risposta
mancante

Punteggio
max.: 1,00

Con riferimento ad una base di dati sullo schema:

```
CREATE TABLE IF NOT EXISTS frequent_flyer (  
  id_frequent_flyer integer PRIMARY KEY NOT NULL,  
  numero_carta text NOT NULL,  
  livello integer NOT NULL,  
  punti integer NOT NULL)
```

```
CREATE TABLE IF NOT EXISTS utente_registrato (  
  id_utente integer PRIMARY KEY NOT NULL,  
  email text NOT NULL,  
  nome text NOT NULL,  
  cognome text NOT NULL,  
  id_frequent_flyer integer REFERENCES frequent_flyer (id_frequent_flyer))
```

Scrivere l'istruzione SQL per trovare nome, cognome, punti degli utenti registrati con il maggior numero di punti frequent_flyer (potrebbero essere più di uno, con lo stesso punteggio), ordinati per nome, cognome, punti

Answer: (penalty regime: 10, 20, ... %)

1 ||

Verifica risposta

Domanda **3**

Risposta
mancante

Punteggio
max.: 1,00

Con riferimento ad una base di dati sullo schema:

```
CREATE TABLE IF NOT EXISTS aeroporto (  
  codice_aeroporto character(3) PRIMARY KEY NOT NULL,  
  nome_aeroporto text NOT NULL,  
  citta text NOT NULL,  
  codice_continente text ,  
  codice_paese text ,  
  internazionale boolean NOT NULL)
```

```
CREATE TABLE IF NOT EXISTS volo_reale (  
  id_volo_reale integer PRIMARY KEY NOT NULL,  
  data_partenza_programmata date,  
  numero_volo text REFERENCES volo(numero_volo),  
  codice_tipo_aeromobile character(3) REFERENCES tipo_aeromobile(codice_tipo_aeromobile),  
  data_partenza_reale date,  
  data_arrivo_reale date,  
  orario_arrivo_reale time ,  
  orario_partenza_reale time ,  
  UNIQUE (numero_volo, data_partenza_programmata))
```

```
CREATE TABLE IF NOT EXISTS volo (  
  numero_volo text PRIMARY KEY NOT NULL,  
  aeroporto_partenza character(3) REFERENCES aeroporto(codice_aeroporto),  
  aeroporto_arrivo character(3) REFERENCES aeroporto(codice_aeroporto),  
  orario_partenza_previsto time ,  
  orario_arrivo_previsto time)
```

Scrivere l'istruzione SQL che trova per ogni aeroporto quanti sono i voli con data di partenza programmata il '2023-07-04'. Mostrare codice aeroporto di partenza, città dell'aeroporto di partenza, e numero dei voli in partenza, ordinati per codice aeroporto di partenza

Answer: (penalty regime: 10, 20, ... %)

1 ||

Verifica risposta

Domanda **4**

Risposta
mancante

Punteggio
max.: 1,00

Con riferimento ad una base di dati sullo schema:

```
CREATE TABLE IF NOT EXISTS volo_reale (  
  id_volo_reale integer PRIMARY KEY NOT NULL,  
  data_partenza_programmata date,  
  numero_volo text REFERENCES volo(numero_volo),  
  codice_tipo_aeromobile character(3) REFERENCES tipo_aeromobile(codice_tipo_aeromobile),  
  data_partenza_reale date,  
  data_arrivo_reale date,  
  orario_arrivo_reale time ,  
  orario_partenza_reale time ,  
  UNIQUE (numero_volo, data_partenza_programmata))
```

```
CREATE TABLE IF NOT EXISTS volo (  
  numero_volo text PRIMARY KEY NOT NULL,  
  aeroporto_partenza character(3) REFERENCES aeroporto(codice_aeroporto),  
  aeroporto_arrivo character(3) REFERENCES aeroporto(codice_aeroporto),  
  orario_partenza_previsto time ,  
  orario_arrivo_previsto time)
```

Scrivere l'istruzione SQL per trovare il codice degli aeroporti con più di un volo con data partenza programmata il giorno '2023-07-06'. Mostrare il codice dell'aeroporto e il numero di voli programmati in partenza il '2023-07-06', ordinati per codice dell'aeroporto

Answer: (penalty regime: 10, 20, ... %)

1 ||

Verifica risposta

Domanda **5**

Risposta
mancante

Punteggio
max.: 1,00

Con riferimento ad una base di dati sullo schema:

```
CREATE TABLE IF NOT EXISTS volo_reale (  
  id_volo_reale integer PRIMARY KEY NOT NULL,  
  data_partenza_programmata date,  
  numero_volo text REFERENCES volo(numero_volo),  
  codice_tipo_aeromobile character(3) REFERENCES tipo_aeromobile(codice_tipo_aeromobile),  
  data_partenza_reale date,  
  data_arrivo_reale date,  
  orario_arrivo_reale time ,  
  orario_partenza_reale time ,  
  UNIQUE (numero_volo, data_partenza_programmata))
```

```
CREATE TABLE IF NOT EXISTS volo (  
  numero_volo text PRIMARY KEY NOT NULL,  
  aeroporto_partenza character(3) REFERENCES aeroporto(codice_aeroporto),  
  aeroporto_arrivo character(3) REFERENCES aeroporto(codice_aeroporto),  
  orario_partenza_previsto time ,  
  orario_arrivo_previsto time)
```

Scrivere l'istruzione SQL che trova per ogni aeroporto i voli più lunghi (cioè con la massima durata prevista) in partenza il '2023-07-04'. La durata prevista di un volo è calcolata come differenza tra l'orario di arrivo previsto e l'orario di partenza previsto (per semplicità supponiamo che tutti i voli partano e arrivino nello stesso giorno). Mostrare codice aeroporto di partenza, codice aeroporto di arrivo, numero volo, durata prevista ordinati per codice aeroporto di partenza e numero volo

Answer: (penalty regime: 10, 20, ... %)

1 ||

Verifica risposta

Domanda **6**

Risposta non
ancora data

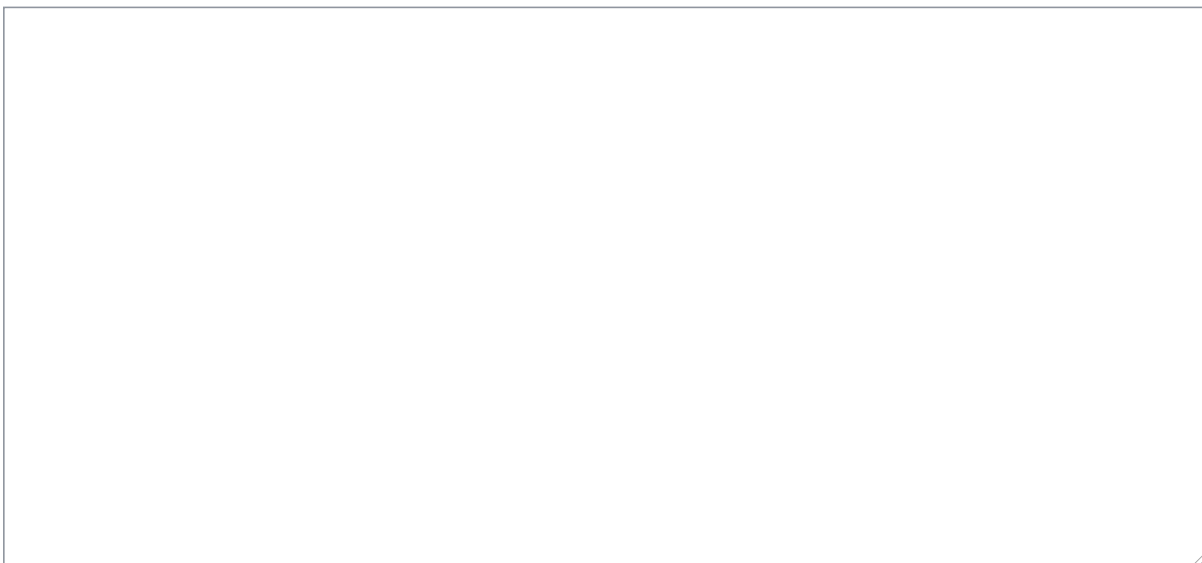
Punteggio
max.: 1,00

Con riferimento ad una base di dati sullo schema:

```
CREATE TABLE IF NOT EXISTS aeroporto (  
  codice_aeroporto character(3) PRIMARY KEY NOT NULL,  
  nome_aeroporto text NOT NULL,  
  citta text NOT NULL,  
  codice_continente text ,  
  codice_paese text ,  
  internazionale boolean NOT NULL)
```

```
CREATE TABLE IF NOT EXISTS volo (  
  numero_volo text PRIMARY KEY NOT NULL,  
  aeroporto_partenza character(3) REFERENCES aeroporto(codice_aeroporto),  
  aeroporto_arrivo character(3) REFERENCES aeroporto(codice_aeroporto),  
  orario_partenza_previsto time ,  
  orario_arrivo_previsto time)
```

Scrivere l'espressione dell'algebra relazionale che trova il codice_aeroporto, codice_paese, codice_continente degli aeroporti internazionali che non hanno voli diretti all'aeroporto con codice 'JFK'



Domanda **7**

Risposta non ancora data

Punteggio max.: 1,00

Considerare le seguenti relazioni (tutte senza valori nulli)

- $R_1(A, B)$, con vincolo di integrità referenziale fra B e la chiave D di R_2 e con cardinalità $N_1 = 1000$
- $R_2(D, E, F, G)$, con vincolo di integrità referenziale fra F, G e la chiave H, P di R_3 e con cardinalità $N_2 = 400$
- $R_3(H, P, Q)$, con cardinalità $N_3 = 500$

Indicare la cardinalità del risultato di ciascuna delle seguenti espressioni (in cui il simbolo \wedge indica l'AND), specificando l'intervallo nel quale essa può variare; indicare simboli e numeri.

	Min (simboli)	Max (simboli)	Min (valore)	Max (valore)
$\pi_{EF}(R_2)$				
$\pi_{HP}(R_3)$				
$R_3 \bowtie_{(Q=A)} R_1$				
$R_1 \bowtie_{(B=D)} R_2$				
$(R_1 \bowtie_{(B=D)} R_2) \bowtie_{(F=H) \wedge (G=P)} R_3$				

Rispondere nella casella sottostante, su cinque righe e quattro colonne, come avreste fatto nel modulo qui sopra

Paragrafo

Percorso: p

[◀ Esercizi autovalutazione \(copia\) \(nascosto\)](#)

Vai a...

[Turno 1 \(copia\) \(nascosto\) ▶](#)