

# Basi di Dati



## Esercitazione 8 **SQL - partel**

# Esercizio 1

Dare le definizioni SQL delle tre tabelle:

FONDISTA(Nome, Nazione, Età)

GAREGGIA(NomeFondista, NomeGara, Piazzamento)

GARA(Nome, Luogo, Nazione, Lunghezza)

# Soluzione 1

```
Create Table FONDISTA(  
    Nome character(20) primary key,  
    Nazione character(20),  
    Età smallint  
)
```

```
Create Table GARA(  
    Nome character(20) primary key,  
    Luogo character(20),  
    Nazione character(20),  
    Lunghezza integer  
)
```

Dare le definizioni SQL delle tre tabelle:

FONDISTA(Nome, Nazione, Età)

GAREGGIA(NomeFondista, NomeGara, Piazzamento)

GARA(Nome, Luogo, Nazione, Lunghezza)

# Soluzione 1

Dare le definizioni SQL delle tre tabelle:

FONDISTA(Nome, Nazione, Età)

GAREGGIA(NomeFondista, NomeGara, Piazzamento)

GARA(Nome, Luogo, Nazione, Lunghezza)

```
Create Table GAREGGIA(  
  NomeFondista character(20) references FONDISTA(Nome),  
  NomeGara character(20) references GARA(Nome),  
  Piazzamento smallint,  
  primary key (NomeFondista, NomeGara)  
)
```



# Esercizio 2

Considerare la base di dati relazionale definita per mezzo delle seguenti istruzioni:

```
create table TARIFFA(  
    tipoAuto numeric not null primary key,  
    costoAlKm numeric not null  
);  
create table AUTOMOBILE(  
    targa numeric not null primary key,  
    tipologia numeric not null references TARIFFA(tipoAuto),  
    lunghezza char(20) not null  
);
```

# Esercizio 2

Considerare la base di dati relazionale definita per mezzo delle seguenti istruzioni:

```
create table TRANSITO(  
  codice numeric not null primary key,  
  auto numeric not null references AUTOMIBILE(targa),  
  orarioIngresso numeric not null,  
  orarioUscita numeric not null,  
  KmPercorsi numeric not null  
);
```

Formulare in SQL l'interrogazione che restituisce, per ogni transito, I dati del veicolo, del transito e il costo del pedaggio (ottenuto moltiplicando il costo al Km per i Km percorsi)

# Soluzione 2

TARIFFA(TipoAuto, CostoAlKm)

AUTOMOBILE(Targa, tipologia, lunghezza)

TRANSITO(Codice, Auto, OrarioIngresso, OrarioUscita, KmPercorsi)

```
select A.Targa, A.tipologia, A.lunghezza, T.Codice, T.OrarioIngresso, T.OrarioUscita,  
       TA.CostoAlKm * T.KmPercorsi  
from TRANSITO as T join AUTOMOBILE as A on T.Auto = A.Targa  
     join TARIFFA as TA on A.tipologia = TA.TipoAuto
```

# Esercizio 3

Dato il seguente schema:

DISCO(NroSerie, TitoloAlbum, Anno, Prezzo)

CONTIENE(NroSerieDisco, CodiceReg, NroProg)

ESECUZIONE(CodiceReg, TitoloCanz, Anno)

AUTORE(Nome, TitoloCanzone)

CANTANTE(NomeCantante, CodiceReg)

scrivere le interrogazioni SQL che permettono di determinare:

- I cantautori (persone che hanno cantato e scritto la stessa canzone) il cui nome inizia per 'D'



# Soluzione 3

```
select C.NomeCantante
from CANTANTE as C join ESECUZIONE as E on C.CodiceReg = E.CodiceReg
      join AUTORE as A on E.TitoloCanz = A.TitoloCanz
Where A.Nome = C.NomeCantante and A.Nome like 'D%'
```

# Esercizio 3

Dato il seguente schema:

DISCO(NroSerie, TitoloAlbum, Anno, Prezzo)  
CONTIENE(NroSerieDisco, CodiceReg, NroProg)  
ESECUZIONE(CodiceReg, TitoloCanz, Anno\*)  
AUTORE(Nome, TitoloCanzone)  
CANTANTE(NomeCantante, CodiceReg)

scrivere le interrogazioni SQL che permettono di determinare:

- I titoli dei dischi che contengono canzoni di cui non si conosce l'anno di registrazione

# Soluzione 3

```
select D.TitoloAlbum  
from DISCO as D join CONTIENE as C on D.NroSerie = C.NroSerieDisco  
      join ESECUZIONE as E on C.CodiceReg = E.CodiceReg  
where E.Anno is null
```

# Esercizio 4

Si consideri una base di dati che gestisce dati relativi ai voli in partenza da un dato aeroporto, con le seguenti relazioni:

AEROPORTI(Codice, Città, Nome)

AEREI(Codice, Nome, NumeroPosti)

VOLI(Compagnia, Numero, Destinazione, OraPart, OraArr, Aereo) con vincoli di integrità referenziale fra Destinazione e la relazione AEROPORTI e tra Aereo e la relazione AEREI  
Formulare in SQL:

1. l'interrogazione che trova le città raggiungibili con un volo diretto che utilizzi un aereo con almeno 200 posti;
2. l'interrogazione che trova le città raggiungibili con voli diretti e, per ciascuna, mostra il numero di tali voli.

# Soluzione 4

AEROPORTI(Codice, Città, Nome)

AEREI(Codice, Nome, NumeroPosti)

VOLI(Compagnia, Numero, Destinazione, OraPart, OraArr, Aereo) con vincoli di integrità  
Formulare in SQL:

1. l'interrogazione che trova le città raggiungibili con un volo diretto che utilizzi un aereo con almeno 200 posti;

```
select Città
from AEROPORTI as A join VOLI as V on A.Codice = V.Destinazione join AEREI
on V.Aereo = AEREI.Codice
where AEREI.NumPosti >= 200
```



# Soluzione 4

AEROPORTI(Codice, Città, Nome)

AEREI(Codice, Nome, NumeroPosti)

VOLI(Compagnia, Numero, Destinazione, OraPart, OraArr, Aereo) con vincoli di integrità  
Formulare in SQL:

1. l'interrogazione che trova le città raggiungibili con voli diretti e, per ciascuna, mostra il numero di tali voli.

```
select Città, Count(*)  
from AEROPORTI as A join VOLI as V on A.Codice = V.Destinazione  
group by Città
```

# Esercizio 5

Dato il seguente schema:

AEROPORTO(Citta, Nazione, NumPiste\*)

VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)

AEREO(TipoAereo, NumPasseggeri, QtaMerci)

scrivere le interrogazioni SQL che permettono di determinare:

- Le città con un aeroporto di cui non è noto il numero di piste

# Soluzione 5

AEROPORTO(Citta, Nazione, NumPiste)

VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)

AEREO(TipoAereo, NumPasseggeri, QtaMerci)

scrivere le interrogazioni SQL che permettono di determinare:

- Le città con un aeroporto di cui non è noto il numero di piste

```
select Citta  
from AEROPORTO  
where NumPiste is null
```

# Esercizio 5

Dato il seguente schema:

AEROPORTO(Citta, Nazione, NumPiste)

VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)

AEREO(TipoAereo, NumPasseggeri, QtaMerci)

scrivere le interrogazioni SQL che permettono di determinare:

- Le nazioni da cui parte e arriva il volo con codice AZ274

# Soluzione 5

AEROPORTO(Citta, Nazione, NumPiste)

VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)

AEREO(TipoAereo, NumPasseggeri, QtaMerci)

scrivere le interrogazioni SQL che permettono di determinare:

- Le nazioni da cui parte e arriva il volo con codice AZ274

```
select A1.Nazione, A2.Nazione
from AEROPORTO as A1 join VOLO as V on A1.Citta = V.CittaPart join
AEROPORTO as A2 on V.CittaArr = A2.Citta
where IdVolo = 'AZ274'
```



# Esercizio 5

Dato il seguente schema:

AEROPORTO(Citta, Nazione, NumPiste)

VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)

AEREO(TipoAereo, NumPasseggeri, QtaMerci)

scrivere le interrogazioni SQL che permettono di determinare:

- I tipi di aereo usati nei voli che partono da Torino

# Soluzione 5

AEROPORTO(Citta, Nazione, NumPiste)

VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)

AEREO(TipoAereo, NumPasseggeri, QtaMerci)

scrivere le interrogazioni SQL che permettono di determinare:

- I tipi di aereo usati nei voli che partono da Torino

```
select TipoAereo  
from VOLO  
where CittaPart = 'Torino'
```

# Esercizio 5

Dato il seguente schema:

AEROPORTO(Citta, Nazione, NumPiste)

VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)

AEREO(TipoAereo, NumPasseggeri, QtaMerci)

scrivere le interrogazioni SQL che permettono di determinare:

- I tipi di aereo e il corrispondente numero di passeggeri per i tipi di aereo usati nei voli che partono da Torino

# Soluzione 5

AEROPORTO(Citta, Nazione, NumPiste)

VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)

AEREO(TipoAereo, NumPasseggeri, QtaMerci)

scrivere le interrogazioni SQL che permettono di determinare:

- I tipi di aereo e il corrispondente numero di passeggeri per i tipi di aereo usati nei voli che partono da Torino

```
select A.TipoAereo, A.NumPasseggeri
from AEREO as A join VOLO as V on A.TipoAereo = V.TipoAereo
where V.CittaPart = 'Torino'
```

# Esercizio 5

Dato il seguente schema:

AEROPORTO(Citta, Nazione, NumPiste)

VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)

AEREO(TipoAereo, NumPasseggeri, QtaMerci)

scrivere le interrogazioni SQL che permettono di determinare:

- Le città da cui partono voli internazionali



# Soluzione 5

AEROPORTO(Citta, Nazione, NumPiste)

VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)

AEREO(TipoAereo, NumPasseggeri, QtaMerci)

scrivere le interrogazioni SQL che permettono di determinare:

- Le città da cui partono voli internazionali

```
select distinct CittaPart
from AEROPORTI as A1 join VOLO on A1.Citta = CittaPart join AEROPORTO as
A2 on CittaArr = A2.Citta
where A1.Nazione != A2.Nazione
```

# Esercizio 5

Dato il seguente schema:

AEROPORTO(Citta, Nazione, NumPiste)

VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)

AEREO(TipoAereo, NumPasseggeri, QtaMerci)

scrivere le interrogazioni SQL che permettono di determinare:

- Le città da cui partono voli diretti a Bologna, ordinati

# Soluzione 5

AEROPORTO(Citta, Nazione, NumPiste)

VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)

AEREO(TipoAereo, NumPasseggeri, QtaMerci)

scrivere le interrogazioni SQL che permettono di determinare:

- Le città da cui partono voli diretti a Bologna, ordinati

```
select distinct CittaPart
from VOLO
where CittaArr = 'Bologna'
order by CittaPart
```

# Esercizio 5

Dato il seguente schema:

AEROPORTO(Citta, Nazione, NumPiste)

VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)

AEREO(TipoAereo, NumPasseggeri, QtaMerci)

scrivere le interrogazioni SQL che permettono di determinare:

- Il numero di voli internazionali che partono il mercoledì da Roma

# Soluzione 5

AEROPORTO(Citta, Nazione, NumPiste)

VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)

AEREO(TipoAereo, NumPasseggeri, QtaMerci)

scrivere le interrogazioni SQL che permettono di determinare:  
- Il numero di voli internazionali che partono il mercoledì da Roma

```
select count(*)  
from VOLO join AEROPORTO on CittaArr = Citta  
where CittaPart = 'Roma' and GiornoSett = 'Mercoledì' and Nazione != 'Italia'
```



# Esercizio 5

Dato il seguente schema:

AEROPORTO(Citta, Nazione, NumPiste)

VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)

AEREO(TipoAereo, NumPasseggeri, QtaMerci)

scrivere le interrogazioni SQL che permettono di determinare:

- Le città francesi da cui partono più di venti voli alla settimana diretti in Italia

# Soluzione 5

AEROPORTO(Citta, Nazione, NumPiste)

VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)

AEREO(TipoAereo, NumPasseggeri, QtaMerci)

scrivere le interrogazioni SQL che permettono di determinare:

- Le città francesi da cui partono più di venti voli alla settimana diretti in Italia

```
select CittaPart
from VOLO join AEROPORTO as A1 on CittaPart = A1.Citta join AEROPORTO
as A2 on CittaArr = A2.Citta
where A1.Nazione = 'Francia' and A2.Nazione = 'Italia'
group by CittaPart
having count(*) > 20
```