

Basi di dati - Tutoraggi

Laurea in Ingegneria Informatica

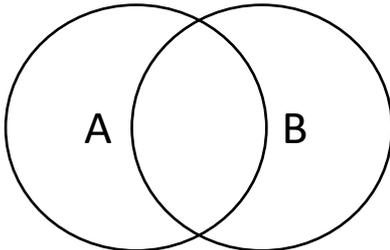
Università Tor Vergata

Tutor: Romolo Marotta

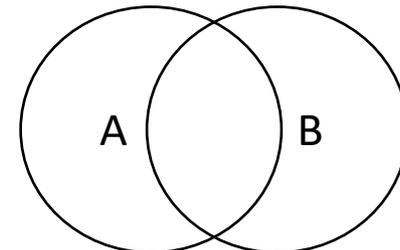
Structured Query Language

1. Richiami
2. Esercizi

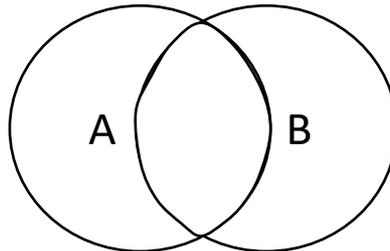
SQL JOINS



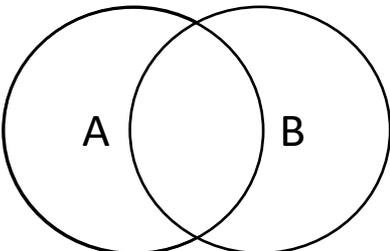
```
SELECT <select_list>  
FROM TableA A  
LEFT JOIN TableB B  
ON A.Key = B.Key
```



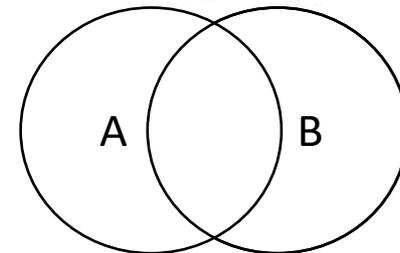
```
SELECT <select_list>  
FROM TableA A  
RIGHT JOIN TableB B  
ON A.Key = B.Key
```



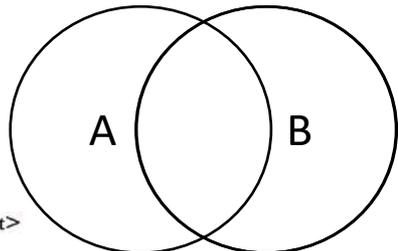
```
SELECT <select_list>  
FROM TableA A  
INNER JOIN TableB B  
ON A.Key = B.Key
```



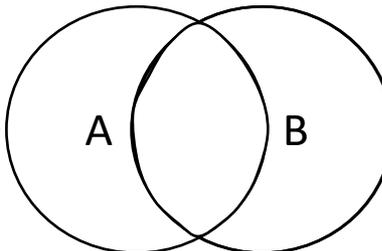
```
SELECT <select_list>  
FROM TableA A  
LEFT JOIN TableB B  
ON A.Key = B.Key  
WHERE B.Key IS NULL
```



```
SELECT <select_list>  
FROM TableA A  
RIGHT JOIN TableB B  
ON A.Key = B.Key  
WHERE A.Key IS NULL
```



```
SELECT <select_list>  
FROM TableA A  
FULL OUTER JOIN TableB B  
ON A.Key = B.Key
```



```
SELECT <select_list>  
FROM TableA A  
FULL OUTER JOIN TableB B  
ON A.Key = B.Key  
WHERE A.Key IS NULL  
OR B.Key IS NULL
```

© C.L. Moffatt, 2008

Esercizi

Esercizio 4.14

- Dato il seguente modello relazionale:
 - 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.**
- **Le nazioni da cui parte e arriva il volo AZ274.**
- **I tipi di aereo usati nei voli che partono da Torino.**
- **I tipi di aereo e il corrispondente numero di passeggeri per i tipi di aereo usati nei voli che partono da Torino. Se la descrizione dell'aereo non è disponibile, visualizzare solamente il tipo.**

Esercizio 4.14

- Dato il seguente modello relazionale:
 - 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 4.14

- Dato il seguente modello relazionale:
 - 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 AZ274.**

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

Esercizio 4.14

- Dato il seguente modello relazionale:
 - 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 4.14

- Dato il seguente modello relazionale:
 - 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. Se la descrizione dell'aereo non è disponibile, visualizzare solamente il tipo.**

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

Esercizio 4.14

- Dato il seguente modello relazionale:
 - 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**
- **Le città da cui partono voli diretti a Bologna, ordinate alfabeticamente**
- **Il numero di voli internazionali che partono il giovedì da Napoli**
- **Il numero di voli internazionali che partono ogni settimana da città italiane**

Esercizio 4.14

- Dato il seguente modello relazionale:
 - 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 AEROPORTO as A1 join VOLO on CittaPart=A1.Citta
join AEROPORTO as A2 on CittaArr=A2.Citta
where A1.Nazione <> A2.Nazione
```

Esercizio 4.14

- Dato il seguente modello relazionale:
 - 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, ordinate alfabeticamente**

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

Esercizio 4.14

- Dato il seguente modello relazionale:
 - 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 giovedì da Napoli**

```
select count(*)  
from VOLO  
join AEROPORTO on CittaArr=Citta  
where Nazione <> 'Italia' and CittaPart='Napoli'  
and GiornoSett = 4
```

Esercizio 4.14

- Dato il seguente modello relazionale:
 - 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 ogni settimana da città italiane**

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

Esercizio 4.14

- Dato il seguente modello relazionale:
 - 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**
- **Gli aeroporti italiani da cui partono solo voli interni, usando operatori insiemistici**
- **Gli aeroporti italiani da cui partono solo voli interni, usando un'interrogazione nidificata con l'operatore `not in`**
- **Gli aeroporti italiani da cui partono solo voli interni, usando un'interrogazione nidificata con l'operatore `not exist`**
- **Gli aeroporti italiani da cui partono solo voli interni, usando una join ed un operatore di conteggio**

Esercizio 4.14

- Dato il seguente modello relazionale:
 - 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 A1.Citta, count(A1.Citta)
from AEROPORTO as A1
join VOLO on A1.Citta=CittaPart
join AEROPORTO as A2 on A2.Citta=CittaArr
where A1.Nazione = 'Francia' and A2.Nazione = 'Italia'
group by A1.Citta
having count(A1.Citta) > 20
```

Esercizio 4.14

- Dato il seguente modello relazionale:
 - AEROPORTO(Citta, Nazione, NumPiste*)
 - VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)
 - AEREO(TipoAereo, NumPasseggeri, QtaMerci)
- Scrivere le interrogazioni SQL che permettono di determinare:
- **Gli aeroporti italiani da cui partono solo voli interni, usando operatori insiemistici**

```
select distinct CittaPart
from VOLO, AEROPORTO where CittaPart=Citta and Nazione='Italia'
except
select CittaPart
from AEROPORTO as A1 join VOLO on A1.Citta=CittaPart
join AEROPORTO as A2 on A2.Citta=CittaArr
where A1.Nazione = 'Italia' and A2.Nazione <> 'Italia'
```

Esercizio 4.14

- Dato il seguente modello relazionale:
 - AEROPORTO(Citta, Nazione, NumPiste*)
 - VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)
 - AEREO(TipoAereo, NumPasseggeri, QtaMerci)
- Scrivere le interrogazioni SQL che permettono di determinare:
- **Gli aeroporti italiani da cui partono solo voli interni, usando un'interrogazione nidificata con l'operatore not in**

```
select distinct CittaPart
from VOLO, AEROPORTO where CittaPart=Citta
and Nazione='Italia' and CittaPart not in
(select CittaPart
 from AEROPORTO as A1 join VOLO on A1.Citta=CittaPart
 join AEROPORTO as A2 on A2.Citta=CittaArr
 where A1.Nazione = 'Italia' and A2.Nazione <> 'Italia')
```

Esercizio 4.14

- Dato il seguente modello relazionale:
 - AEROPORTO(Citta, Nazione, NumPiste*)
 - VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)
 - AEREO(TipoAereo, NumPasseggeri, QtaMerci)
- Scrivere le interrogazioni SQL che permettono di determinare:
- **Gli aeroporti italiani da cui partono solo voli interni, usando un'interrogazione nidificata con l'operatore `not exist`**

```
select distinct CittaPart
from VOLO
join AEROPORTO as A1 on CittaPart=A1.Citta
where Nazione='Italia' and not exists
  (select * from VOLO join AEROPORTO as A2 on
   A2.Citta=CittaArr
   where A1.Citta=CittaPart and A2.Nazione <> 'Italia' )
```

Esercizio 4.14

- Dato il seguente modello relazionale:
 - AEROPORTO(Citta, Nazione, NumPiste*)
 - VOLO(IdVolo, GiornoSett, CittaPart, OraPart, CittaArr, OraArr, TipoAereo)
 - AEREO(TipoAereo, NumPasseggeri, QtaMerci)
- Scrivere le interrogazioni SQL che permettono di determinare:
- **Gli aeroporti italiani da cui partono solo voli interni, usando una join ed un operatore di conteggio**

```
select CittaPart
from AEROPORTO as A1
join VOLO on A1.Citta=CittaPart
left join AEROPORTO as A2
on (CittaArr = A2.Citta and A2.Nazione <> 'Italia')
where A1.Nazione = 'Italia'
group by CittaPart
having count(A2.Nazione) = 0
```

Esercizi per casa

Esercizi per casa (4.15)

- Dato il seguente schema:
 - DISCO(NroSerie, TitoloAlbum, Anno, Prezzo)
 - CONTIENE(NroSerieDisco, CodiceReg, NroProg)
 - ESECUZIONE(CodiceReg, TitoloCanz, Anno)
 - AUTORE(Nome, TitoloCanzone)
 - CANTANTE(NomeCantante, CodiceReg)

formulare le interrogazioni SQL che permettono di determinare:

1. I cantautori (persone che hanno cantato e scritto la stessa canzone) il cui nome inizia per 'D';
2. I titoli dei dischi che contengono canzoni di cui non si conosce l'anno di registrazione;
3. I pezzi del disco con numero di serie 78574, ordinati per numero progressivo, con indicazione degli interpreti per i pezzi che hanno associato un cantante;
4. Gli autori e i cantanti puri, ovvero autori che non hanno mai registrato una canzone e cantanti che non hanno mai scritto una canzone;
5. I cantanti del disco che contiene il maggior numero di canzoni;
6. Gli autori solisti di “collezioni di successi” (dischi in cui tutte le canzoni sono di un solo cantante e in cui almeno tre registrazioni sono di anni precedenti la pubblicazione del disco);
7. I cantanti che non hanno mai registrato una canzone come solisti;
8. I cantanti che non hanno mai inciso un disco in cui comparissero come unici cantanti;
9. I cantanti che hanno sempre registrato canzoni come solisti.