



Basi di Dati



Query di riepilogo

DB Esami

S (Matr, SNome, Citta, ACorso)

C (CC, CNome, CD)

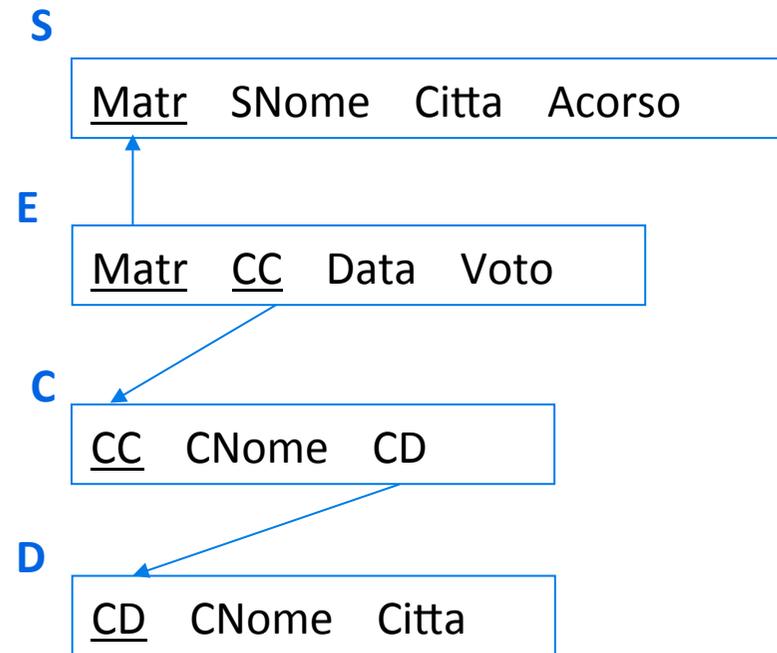
FK: CD REFERENCES D

D (CD, CNome, Citta)

E (Matr, CC, Data, Voto)

FK: Matr REFERENCES S

FK: CC REFERENCES C



DB Esami – Query di riepilogo

- ▶ Matricole degli studenti che hanno sostenuto un esame insieme (stesso corso, stessa data) alla matricola M2

```
SELECT Matr
FROM E
WHERE Matr <> 'M2'
AND (CC,Data) IN (SELECT CC,Data
                  FROM E
                  WHERE Matr = 'M2')
```

DB Esami – Query di riepilogo

- ▶ Per ogni studente, il voto massimo ottenuto, escludendo gli esami con voto pari a 33 e considerando solo gli studenti con più di 2 esami; ordinare il risultato per voti massimi crescenti

```
SELECT Matr, MAX(Voto)
FROM E
WHERE Voto <> 33
GROUP BY Matr
HAVING COUNT(*) > 2
ORDER BY 2
```

DB Esami – Query di riepilogo

- ▶ Selezionare per ogni docente il corso per il quale sono stati sostenuti il maggior numero di esami

```
SELECT C1.CD,C1.CC
FROM C C1,E E1
WHERE C1.CC=E1.CC
GROUP BY C1.CD,C1.CC
HAVING COUNT(*) >= ALL (SELECT COUNT(*)
                        FROM C C2,E E2
                        WHERE C2.CC=E2.CC
                        AND C2.CD=C1.CD
                        GROUP BY C2.CD,C2.CC)
```

DB Esami – Query di riepilogo

- ▶ Studenti che hanno sostenuto tutti gli esami sostenuti dallo studente M7

→ Studenti per i quali non esiste alcun esame sostenuto da M7 che essi non hanno sostenuto

```
SELECT *
FROM S
WHERE Matr <> 'M7'
AND NOT EXISTS
    (SELECT *
     FROM E E1
     WHERE Matr='M7'
     AND NOT EXISTS
        (SELECT *
         FROM E E2
         WHERE E2.Matr=S.Matr
         AND E2.CC=E1.CC))
```

Sistema Informativo per Taxi

TAXI (TARGA, AUTISTA, MODELLO, PARCHEGGIO)

CLIENTE (CF, DATA, POSIZIONE_I, POSIZIONE_F, NOME)

CORSA (TARGA, CF, DATA, COSTO)

FK: TARGA REFERENCES TAXI

FK: CF, DATA REFERENCES CLIENTE

Dove POSIZIONE_I è il luogo di inizio della corsa e POSIZIONE_F il luogo di fine corsa.

Sistema Informativo per Taxi

1. Selezionare i modelli di TAXI che non hanno caricato clienti in parco della Vittoria
2. Selezionare i CLIENTI che hanno utilizzato tutti i taxi utilizzati dal cliente con CF 'xy'
3. Per ogni coppia TARGA, DATA di una CORSA selezionare l'incasso ottenuto dal TAXI sino al giorno precedente a quella data
4. Selezionare la posizione di partenza da cui sono partite il maggior numero di corse, delle quali almeno 3 con costo superiore a 100 EURO.

Sistema Informativo per Taxi

1) Selezionare i modelli di TAXI che non hanno caricato clienti in parco della Vittoria

```
select DISTINCT MODELLO
from TAXI
where MODELLO not in (
  select t1.MODELLO
  from CORSA C, CLIENTE cl, TAXI t1
  where cl.CF = c.CF
  and cl.DATA = c.DATA
  and t1.TARGA = c.TARGA
  and cl.POSIZIONE_I = 'parco della Vittoria')
```

Sistema Informativo per Taxi

2) Selezionare i CLIENTI che hanno utilizzato tutti i taxi utilizzati dal cliente con CF 'xy'

```
select *
from CLIENTE cl
where not exists (
  select *
  from CORSA c1
  where c1.CF = 'xy'
  and not exists (
    select *
    from CORSA c2
    where cl.CF = c2.CF
    and c1.TARGA = c2.TARGA
  )
)
```

Sistema Informativo per Taxi

3) Per ogni coppia TARGA, DATA di una CORSA selezionare l'incasso ottenuto dal TAXI sino al giorno precedente a quella data

```
select C1.TARGA, C1.DATA, SUM(C2.COSTO)
from CORSA c1, CORSA c2
where c1.TARGA = c2.TARGA
and c1.DATA > c2.DATA
group by c1.TARGA, c1. DATA
```

Sistema Informativo per Taxi

- 4) Selezionare la posizione di partenza da cui sono partite il maggior numero di corse, delle quali almeno 3 con costo superiore a 100 EURO.

Sistema Informativo per Taxi

```
select cl.POSIZIONE_I
from CLIENTE cl
where 3 <= (select count(*)
            from CLIENTE cl2, CORSA co2
            where cl2.POSIZIONE_I = cl.POSIZIONE_I
            and cl2.CF = co2.CF
            and cl2.DATA = co2.DATA
            and co2.COSTO > 100)
group by cl.POSIZIONE_I
having count(*) >= ALL (select count(*)
                       from CLIENTE cl1, CORSA co1
                       where cl1.CF = co1.CF
                       and cl1.DATA = co1.DATA
                       and 3 <= (select count(*)
                                from CLIENTE cl3, CORSA co3
                                where cl3.POSIZIONE_I = cl1.POSIZIONE_I
                                and cl3.CF = co3.CF
                                and cl3.DATA = co3.DATA
                                and co3.COSTO > 100)
                       group by cl1.POSIZIONE_I)
```