
SQL
7 lezione
esercizi
Francesca Gasparini
gasparini@disco.unimib.it

esercizio 1



Specificare in SQL **vincoli (assertion)** che garantiscano le seguenti restrizioni:

- a) Ogni cliente non può acquistare più di 100 CD.
- b) Ogni cliente non può acquistare più di 10 CD allo stesso giorno.
- c) Non possono esistere clienti che hanno acquistato sia CD dei Subsonica che dei Metallica.

esercizio 1



- a) Ogni cliente non può acquistare più di 100 CD.

(a) create assertion UpperBound
check (100 > = all (select count(*)
from ACQUISTO
group by cliente)

esercizio 1



- b) Ogni cliente non può acquistare più di 10 CD nello stesso giorno.

(b) create assertion DailyUpperBound
check(10 > = all (select count(*)
from ACQUISTO
group by cliente, data)

esercizio 1



- c) Non possono esistere clienti che hanno acquistato sia CD dei Subsonica che dei Metallica.

(c) create assertion NoMetSub
check(not exists (select *
from ACQUISTO join CD on (cd = codice-cd)
where artista = 'Metallica'
intersect
select *
from ACQUISTO join CD on (cd = codice-cd)
where artista = 'Subsonica'))

esercizio 2



3. Specificare in SQL vincoli di campo che garantiscano le seguenti restrizioni:

- a) Il codice-cd nella tabella CD deve essere una stringa di 6 caratteri che inizia o finisce con '01'.

(a) codice-cd char(6) check(codice-cd like '01____' or
codice-cd like '____01')

esercizio 2



b) Il prezzo di ogni CD non può superare di più di 10 Euro la media dei prezzi.

(b) prezzo integer check(prezzo -10 <= (select avg(prezzo) from CD))

esercizio 3

Dare una sequenza di comandi di aggiornamento che modifichi l'attributo salario, aumentando del 10% gli stipendi sotto 30000, e diminuendo del 5% gli stipendi sopra 30000

Impiegato(id, cognome, nome, salario, dipartimento)

```
update Employee set Salary=Salary/2
where Salary <= 30000
```

```
update Employee set Salary=Salary*0.95
where Salary > 30000
```

```
update Employee set Salary=Salary*2.2
where Salary<= 15000
```

esercizio 4

Creare una vista con gli attributi di Impiegato, escludendo l'attributo salario

Impiegato(id, cognome, nome, salario, dipartimento)

permettere a Carlo di accedere al contenuto di questa nuova vista

```
create view
Impiegato_senza_sal(id, cognome, nome, dipartimento)
AS
select id, cognome, nome, dipartimento
from impiegato
```

```
grant select on Impiegato_senza_sal to Carlo
```

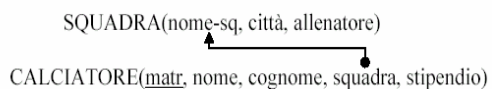
esercizio 4

Definire una vista che mostra per ogni dipartimento il valore medio degli stipendi superiore alla media

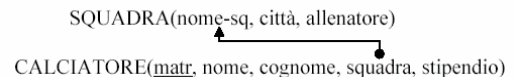
```
create view Stipenti_superiore (dipartimento, salario) as
select dipartimento, avg(Salario)
from impiegato
where Salario > (select avg(Salario)
from Impiegato as E1
where dipartimento=E1.dipartimento)
group by dipartimento
```

esercizio 5

Si assuma il seguente schema di base di dati per l'archiviazione d'informazioni relative alla militanza di calciatori in quadre:



esercizio 5



a) Classifica delle squadre per numero di giocatori.

```
select nome-sq, città, count(*) as N-giocatori
from SQUADRA join CALCIATORE on (nome-sq=squadra)
group by nome-sq, città
order by N-giocatori
```

esercizio 5

SQUADRA(nome-sq, città, allenatore)
CALCIATORE(matr, nome, cognome, squadra, stipendio)

b) Calciatori dell'Inter dallo stipendio più alto.

```
select matricola, nome, cognome
from CALCIATORE
where nome-sq='Inter' and stipendio >= all (select stipendio
                                           from CALCIATORE
                                           where nome-sq='Inter')
```

esercizio 5

SQUADRA(nome-sq, città, allenatore)
CALCIATORE(matr, nome, cognome, squadra, stipendio)

c) Calciatori con lo stipendio più alto, squadra per squadra.

```
select matricola, nome, cognome, squadra
from CALCIATORE C
where stipendio >= all (select stipendio
                       from CALCIATORE
                       where nome-sq=C.nome-sq)
```

query correlate

esercizio 5

SQUADRA(nome-sq, città, allenatore)
CALCIATORE(matr, nome, cognome, squadra, stipendio)

d) Squadre in cui militano i calciatori dallo stipendio più alto.

```
select nome-sq, città
from SQUADRA join CALCIATORE on (nome-sq=squadra)
where stipendio = (select max(stipendio)
                  from CALCIATORE)
```

esercizio 5

SQUADRA(nome-sq, città, allenatore)
CALCIATORE(matr, nome, cognome, squadra, stipendio)

e) Calciatori il cui stipendio supera la media degli stipendi di squadra.

```
select matricola, nome, cognome
from CALCIATORE C
where stipendio > (select avg(stipendio)
                  from CALCIATORE
                  where nome-sq=C.nome-sq)
```

esercizio 5

SQUADRA(nome-sq, città, allenatore)
CALCIATORE(matr, nome, cognome, squadra, stipendio)

f) Calciatori che guadagnano più dei propri omonimi che militano in altre squadre.

```
select matricola, nome, cognome
from CALCIATORE C
where stipendio >= all (select stipendio
                       from CALCIATORE
                       where nome=C.nome and cognome=C.cognome and
                             nome-sq<>C.nome-sq)
```

esercizio 5

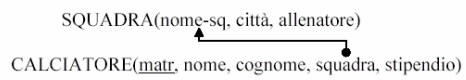
SQUADRA(nome-sq, città, allenatore)
CALCIATORE(matr, nome, cognome, squadra, stipendio)

g) Totale degli stipendi pagati, squadra per squadra.

```
select nome-sq, città, sum(stipendio) as Tot-stipendi
from SQUADRA join CALCIATORE on (nome-sq=squadra)
group by nome-sq, città
```

esercizio 5

SQUADRA(nome-sq, città, allenatore)
CALCIATORE(matr, nome, cognome, squadra, stipendio)



h) Calciatori che militano nelle squadre della città col maggior numero di squadre.

```
select matricola, nome, cognome
from SQUADRA join CALCIATORE on (nome-sq=squadra)
where città in (select città
               from SQUADRA
               group by città
               having count(*) >= all (select count(*)
                                       from SQUADRA
                                       group by città))
```