

SQL -Caratteristiche Evolute-

Atzeni, Ceri, Paraboschi, Torlone
Basi Di Dati:
Modelli e Linguaggi di Interrogazione,
McGraw-Hill Italia Capitolo 5

Check

- La clausola check permette di specificare vincoli generici:
check(Condizione)
- Si ipotizzi che un impiegato debba avere un capo del proprio dipartimento, a meno che la sua matricola non inizi per 1.
check (Matricola like '1%' or
Dipart = (select Dipart
from Impiegato I
where I.matricola = Superiore))

SQL

1

Check (continua)

- Con tale costrutto e' possibile specificare anche i vincoli già visti (di tupla e di integrità referenziale)

```
create table Impiegato (  
  Matr char(6)  
  check(Matr is not null and 2 > (select count(*) from Impiegato  
    where Matr=I.Matr)),  
  Cognome char(20) check (Cognome is not null),  
  Nome char(20) check (Nome is not null),  
  Dipart char(10) check(Dipart in(select NomDip  
    from Dipartimento))  
)
```

SQL

2

Asserzioni

- Le asserzioni sono dei vincoli che non appartengono ad attributi o relazioni particolari, ma fanno riferimento direttamente allo schema.

```
create assertion Nome check (Condizione)
```

```
create assertion AlmenoUnImpiegato  
check (1 <= (select count(*) from Impiegato))
```

- set constraints immediate
- set constraints deferred

SQL

3

Viste

- In SQL le viste sono definite
`create view Nome [(ListaAttributi)]
as SqlSelect
[with [local | cascaded] check option]`
- In molti sistemi, le viste possono essere
 - Materializzate: la query è effettivamente eseguita ed il risultato è memorizzato in forma di tabella
 - Hanno il vantaggio di essere già pronte ma devono essere tenute allineate alla tabelle sottostanti.
 - Virtuali: si memorizza la sola query.
- E' possibile utilizzare Viste nella definizione di Viste. Non sono però consentito dipendenze ricorsive (né immediate, né transitive) introdotte da SQL:1999.

SQL

4

Viste (continua)

```
create view ImpAmm (Matr, Nome, Cogn, Stip) as  
select Matr, Nome, Cogn, Stip from Impiegati  
where Dipart='Amministrazione'
```

```
create view ImpAmmPov (Matr, Nome, Cogn, Stip) as  
select * from ImpAmm  
where Stip < 50  
with local check option
```

SQL

5

Viste (continua)

- Le viste sono utilizzabili per scrivere interrogazioni.
- Tale caratteristica è particolarmente utile quando è richiesto l' utilizzo combinato di operatori aggregati.
- Impiegato (Matr, Nome, Stip, Dip)
- Trovare il dipartimento che spende il massimo in stipendi.

```
select Dip  
from Impiegato  
group by Dip  
having sum(Stip) >=  
all (select sum(Stip)  
from Impiegato  
groupby (Dip))
```

SQL

6

Viste (continua)

```
create view BudgStip (Dip;TotStip)  
select Dip, sum(Stip) from Impiegato group by Dip
```

```
select Dip from BudgStip  
where TotStip = (select max(TotStip)  
from BudgStip))
```

SQL

7

Viste Aggiornabili

- Sintatticamente, posso pensare di modificare una vista
- I sistemi impongono una serie di restrizioni
 - Come regola, bisogna immaginare che una vista è modificabile sse le modifiche possono essere applicate **univocamente** alle tabelle sottostanti.
- Per le viste aggiornabili, la parola chiave *check* controlla che la riga modificata non esca dalla vista.

SQL

8

Funzioni Scalari

- Gli interpreti SQL mettono a disposizione una serie di funzioni di utilità:
 - `Current_date`, `current_time`, ..
 - Conversioni di data (`extract(year,current-date())`)
 - Cast di vario tipo
 - Esempio: `date-> stringa`
 - Funzioni per formattazione
 - Funzioni matematiche (`sqrt`, `abs`, `pow`)
 - Chiamate al sistema operativo

SQL

9

Funzioni Condizionali

- Coalesce: accetta una sequenza di espressioni e restituisce il primo diverso da null
`Select cognome, coalesce (Dip,'Ignoto') from Impiegato`
- Nullif: accetta una espressione e un valore costante; se l'espressione è pari al valore costante restituisce NULL
 - `Select cognome, nullif (Dip,'Ignoto') from Impiegato`

SQL

10

Funzioni Condizionali (continua)

- Case
 - Estrarre l'ammontare delle tasse per un veicolo

```
Select targa
  case tipo
    when 'auto' then 2.58*Kwatt
    when 'camion' then 1.00*NAssi
    else null
  end
From veicoli
Where anno > 2000
```

SQL

11

Funzioni Condizionali (continua)

- Case può essere usato anche negli update

```
Update Impiegato
Set stipendio =
case
  when Dipart = 'Amministrazione' then Stipendio * 1,10
  when Dipart = 'Produzione' then Stipendio * 1,15
  else Stipendio
end
```

SQL

12

Risorse e Privilegi

- SQL prevede che ogni utente sia identificato. Ciò può avvenire sia appoggiandosi al S.O. che con meccanismi propri.
- Le risorse tipiche (ma non le sole) che sono protette sono le tabelle e le viste.
- Di norma, il creatore di una risorsa ne diventa il proprietario e può eseguire su di essa ogni operazione.
- Il proprietario può concedere ad alcuni utenti privilegi sulle sue risorse.
- In SQL esiste il concetto di "ruolo" (già previsto in molti sistemi) che semplifica drasticamente la gestione degli utenti

SQL

13

Privilegi

- insert (tabelle e viste)
- update (tabelle e viste)
- select (tabelle e viste)
- references (tabelle ed attributi)
- usage (domini)
- All privileges
- grant Privilegi on Risorsa to Utenti [with grant option]
- revoke Privilegi on Risorsa from Utenti [restrct | cascade]
- grant insert on Impiegati to stefano

SQL

14