

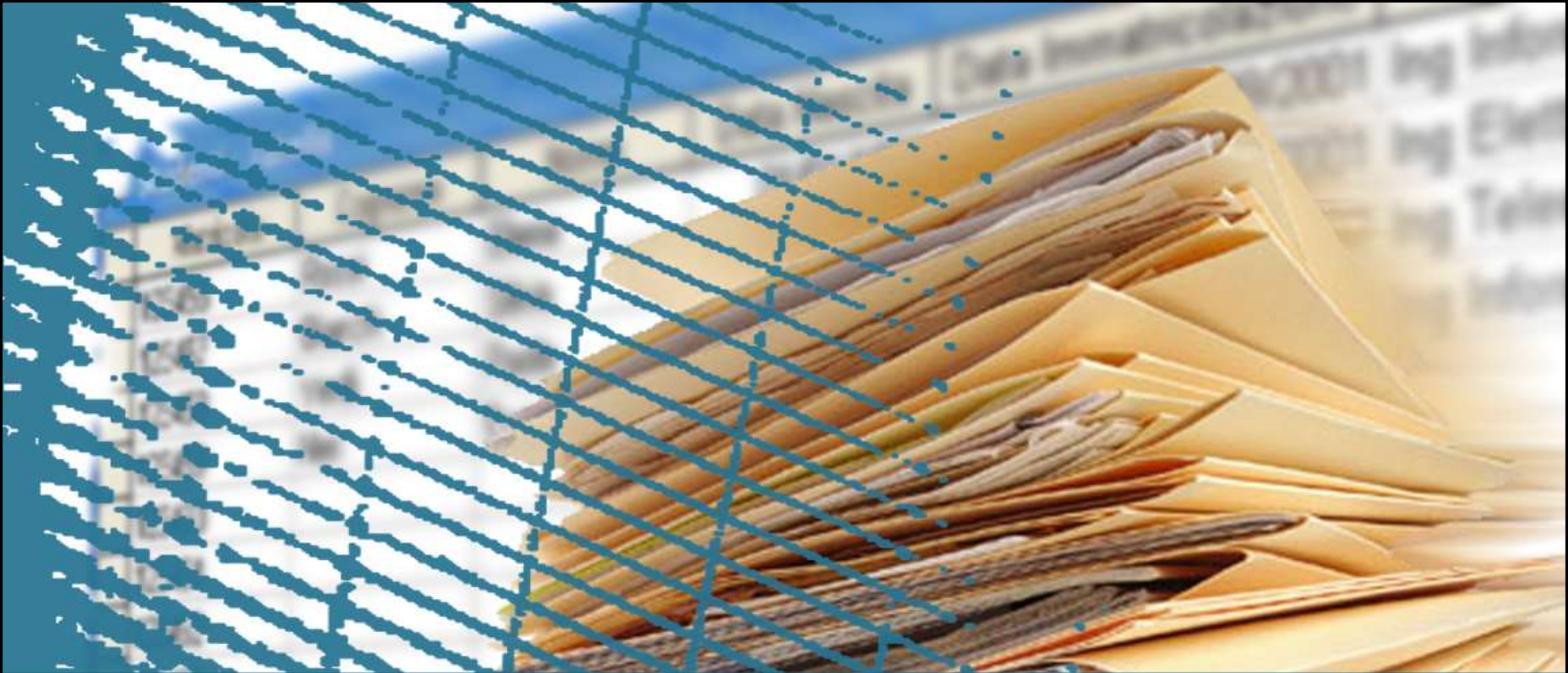


Linguaggio SQL: fondamenti

Istruzioni di aggiornamento

Istruzioni di aggiornamento

- Introduzione
- Istruzione INSERT
- Istruzione DELETE
- Istruzione UPDATE



Istruzioni di aggiornamento

Introduzione

Istruzioni di aggiornamento (1/3)

- Inserimento di tuple
- Cancellazione di tuple
- Modifica di tuple

Istruzioni di aggiornamento (2/3)

➤ INSERT

- inserimento di nuove tuple in una tabella

➤ DELETE

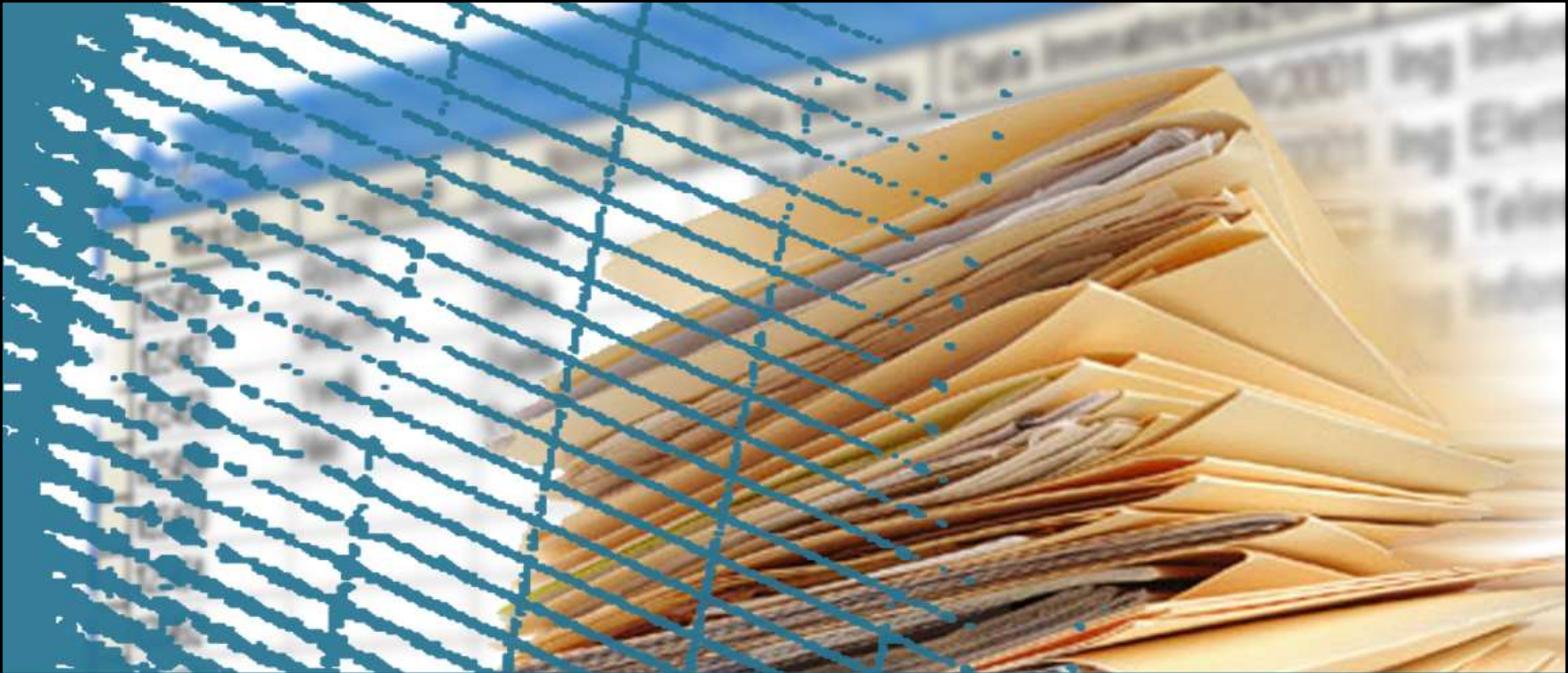
- cancellazione di tuple da una tabella

➤ UPDATE

- modifica del contenuto di tuple in una tabella

Istruzioni di aggiornamento (3/3)

- Le operazioni di aggiornamento modificano lo stato della base di dati
 - è necessario verificare che siano rispettati i vincoli di integrità
- Ogni istruzione può aggiornare il contenuto di una sola tabella



Istruzioni di aggiornamento

Istruzione INSERT

Istruzione INSERT

- Inserimento di una sola tupla
 - assegnazione di un valore costante ad ogni attributo
- Inserimento di più tuple
 - lette da altre tabelle mediante un'istruzione SELECT

Inserimento di una tupla

```
INSERT INTO NomeTabella  
           [(ElencoColonne)]  
VALUES (ElencoCostanti);
```

Inserimento di una tupla: esempio (n.1)

- Inserire un prodotto con
CodP = P7, NomeP = Maglia, Colore = Viola
Taglia = 40, Città = Venezia

```
INSERT INTO P (CodP, NomeP, Colore, Taglia, Città)  
VALUES ('P7', 'Maglia', 'Viola', 40, 'Venezia');
```

- È inserita nella tabella P una nuova tupla con i valori specificati

Inserimento di una tupla: esempio (n.1)

- Inserire un prodotto con
CodP = P7, NomeP = Maglia, Colore = Viola
Taglia = 40, Città = Venezia

```
INSERT INTO P (CodP, NomeP, Colore, Taglia, Città)  
VALUES ('P7', 'Maglia', 'Viola', 40, 'Venezia');
```

- Omettere la lista dei campi equivale a specificare tutti i campi secondo l'ordine di creazione delle colonne nella tabella
- Se la tabella cambia schema, l'istruzione non è più applicabile

Inserimento di una tupla: esempio (n.2)

- Inserire un prodotto con
CodP = P8, Città = Genova, Taglia = 42

```
INSERT INTO P (CodP, Città, Taglia)  
VALUES ('P8', 'Genova', 42);
```

- È inserita nella tabella P una nuova tupla con i valori specificati
- a NomeP e Colore è assegnato il valore NULL
- Per tutti gli attributi il cui valore non è specificato, il dominio dell'attributo deve consentire il valore NULL

Inserimento: integrità referenziale

➤ Inserire una nuova fornitura con
CodF = F20, CodP = P20, Qta = 1000

```
INSERT INTO FP (CodF, CodP, Qta)
VALUES ('F20', 'P20', 1000);
```

➤ *Se esistono vincoli di integrità referenziale*
(dichiarati a livello di definizione delle tabelle → **DDL**)
ad es. **FP.CodF → P.CodF** e/o **FP.CodP → P.CodP**

- è necessario che P20 e F20 siano già presenti rispettivamente in P e F
- se il vincolo non è soddisfatto, l'inserimento non viene eseguito

Inserimento di più tuple

```
INSERT INTO NomeTabella  
          [(ElencoColonne)]  
          Interrogazione;
```

- Sono inserite in *NomeTabella* tutte le tuple selezionate dall'interrogazione *Interrogazione*
- *Interrogazione* è un'istruzione **SELECT** arbitraria
 - non può contenere la clausola **ORDER BY**

Inserimento di più record: esempio

FORNITURE-TOTALI (CodP, TotQta)

- Per ogni prodotto, inserire nella tabella FORNITURE-TOTALI la quantità totale fornita
- dati aggregati estratti dalla tabella FP

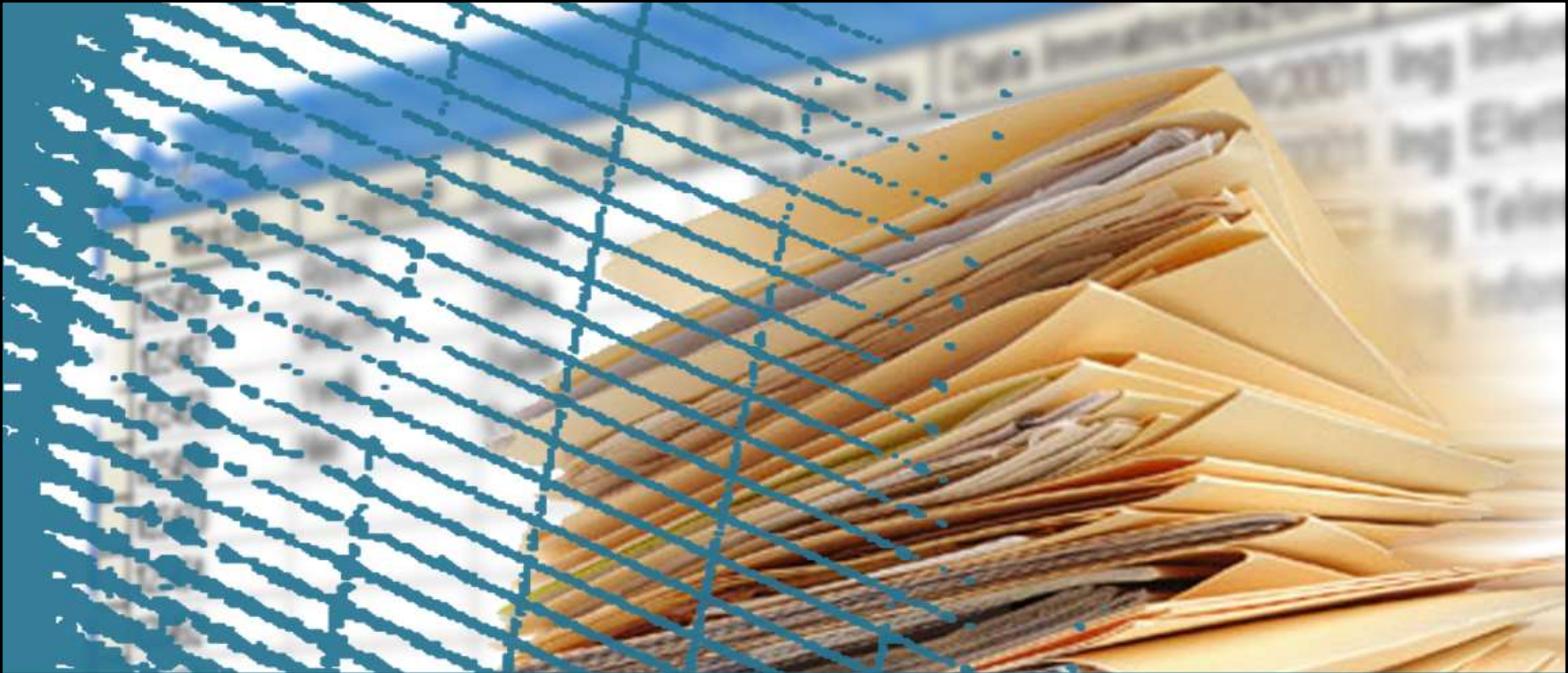
```
SELECT CodP, SUM(Qta)
FROM FP
GROUP BY CodP
```

Inserimento di più record: esempio

FORNITURE-TOTALI (CodP, TotQta)

- Per ogni prodotto, inserire nella tabella FORNITURE-TOTALI la quantità totale fornita

```
INSERT INTO FORNITURE-TOTALI (CodP, TotQta)
      (SELECT CodP, SUM(Qta)
       FROM FP
       GROUP BY CodP);
```



Istruzioni di aggiornamento

Istruzione DELETE

Istruzione DELETE

```
DELETE FROM NomeTabella  
[WHERE predicato];
```

- Cancellazione dalla tabella *NomeTabella* di tutte le tuple che soddisfano il predicato
- Qualora la semplice cancellazione causi la violazione di vincoli di *integrità referenziale* (dichiarati a livello di definizione delle tabelle → **DDL**) il DB mette in atto le previste *azioni compensative* per mantenere l'integrità

Istruzione DELETE: esempio (n.1)

➤ Cancellare tutte le forniture

```
DELETE FROM FP;
```

➤ Senza clausola **WHERE** tutte le tuple soddisfano il predicato di selezione

- elimina il contenuto della tabella FP
- la tabella *non* è eliminata

Istruzione DELETE: esempio (n.2)

- Cancellare la tupla corrispondente al fornitore con codice **F1**

```
DELETE FROM F  
WHERE CodF = 'F1';
```

- Qualora la semplice cancellazione causi la violazione di vincoli di *integrità referenziale* (dichiarati a livello di definizione delle tabelle → **DDL**) ad es. **FP.CodF** → **P.CodF** e/o **FP.CodP** → **P.CodP** il DB mette in atto le previste *azioni compensative* per mantenere l'integrità

Istruzione DELETE: esempio (n.2)

➤ Ad esempio, se l'azione compensativa impostata nel DB prevede l'**eliminazione in cascata** delle *tuple referenzianti* in FP, l'istruzione

```
DELETE FROM F  
WHERE CodF = 'F1';
```

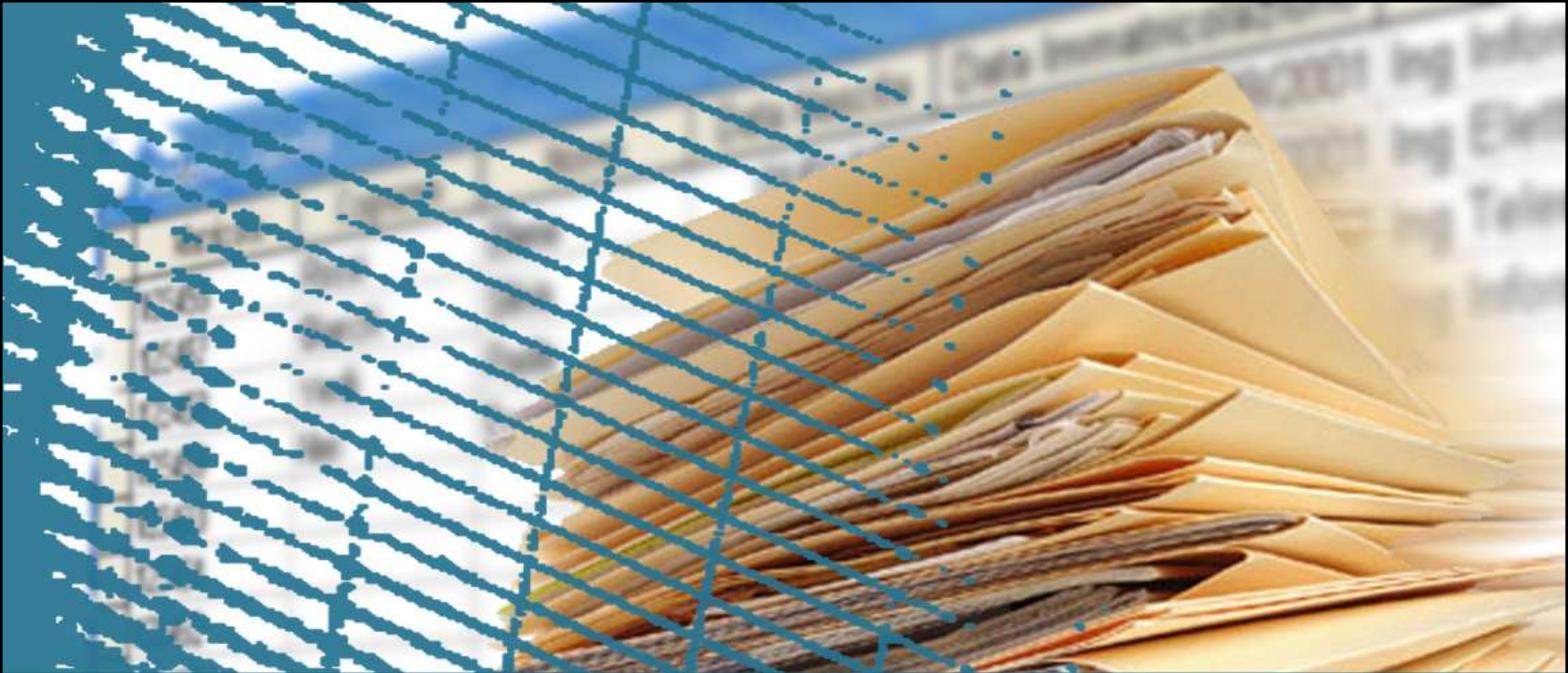
comporta l'eliminazione (automatica) di tutte le tuple in FP dove **CodF = 'F1'**

Istruzione DELETE: esempio (n.3)

➤ Cancellare i fornitori con sede a Milano

```
DELETE FROM F  
WHERE Sede = 'Milano';
```

➤ Ad esempio, se l'azione compensativa impostata nel DB prevede l'**annullamento dell'azione** nel caso che esistano *tuple referenzianti* in FP, l'istruzione **non viene eseguita**



Istruzioni di aggiornamento

Istruzione UPDATE

Istruzione UPDATE

```
UPDATE NomeTabella  
SET colonna = espressione  
    {, colonna = espressione}  
[WHERE predicato];
```

- Tutte le tuple della tabella *NomeTabella* che soddisfano il predicato sono modificati in base alle assegnazioni *colonna = espressione* nella clausola **SET**

Aggiornamento di una tupla

- Aggiornare le caratteristiche del prodotto P1:
- assegnare Giallo al Colore
 - incrementare la Taglia di 2
 - assegnare NULL a Città

```
UPDATE P
SET Colore = 'Giallo',
    Taglia=Taglia+2,
    Città = NULL
WHERE CodP = 'P1';
```

L'istruzione aggiorna *l'unica tupla* relativa al prodotto di codice P1

Aggiornamento multiplo

- Aggiornare il numero dei soci al doppio del valore per tutti i fornitori con sede a Milano

```
UPDATE F  
SET Nsoci = 2 * NSoci  
WHERE Città = 'Milano';
```

L'istruzione aggiorna *tutte le tuple* relative a fornitori con sede a Milano

Aggiornamento con sottointerrogazione

- Aggiornare a 10 la quantità fornita per tutti i fornitori di Milano

```
UPDATE FP
SET Qta = 10
WHERE CodF IN (SELECT CodF
                FROM F
                WHERE Sede = 'Milano');
```

L'istruzione aggiorna *tutte le tuple* relative a fornitori con sede a Milano

Aggiornamento in tabelle referenziate

- Assegnare il valore CodF = F9
al fornitore avente CodF = F2

```
UPDATE F  
SET CodF = 'F9'  
WHERE CodF = 'F2';
```

se l'azione compensativa impostata nel DB prevede la **modifica in cascata** delle *tuple referenzianti* in FP, l'istruzione comporta la **modifica** (automatica) in FP di tutti i valori **F2** in **F9**

Aggiornamento in tabelle referenziate

- Assegnare il valore CodF = F9
al fornitore avente CodF = F2

```
UPDATE F  
SET CodF = 'F9'  
WHERE CodF = 'F2';
```

Se l'azione compensativa impostata nel DB prevede l'**annullamento dell'azione** nel caso che esistano *tuple referenzianti* in FP, l'istruzione **non viene eseguita**