



Linguaggio SQL: fondamentali

Operatori insiemistici

Operatori insiemistici

- Operatore UNION
- Operatore INTERSECT
- Operatore EXCEPT

IMPIEGATO (Matricola, Cognome, Nome, CodDip, Stipendio)
DIPARTIMENTO (Codice, NomeDip, Città)

```
SELECT CodDip
FROM IMPIEGATO
WHERE Stipendio > 45.000
EXCEPT
SELECT Codice
FROM DIPARTIMENTO
WHERE Città = 'Torino';
```

```
SELECT Codice
FROM DIPARTIMENTO
WHERE Città = 'Torino';
EXCEPT
SELECT CodDip
FROM IMPIEGATO
WHERE Stipendio > 45.000;
```

```
SELECT Codice
FROM DIPARTIMENTO
WHERE Città = 'Torino';
INTERSECT
SELECT CodDip
FROM IMPIEGATO
WHERE Stipendio > 45.000;
```

```
SELECT CodDip
FROM IMPIEGATO
WHERE Stipendio > 45.000
AND CodDip NOT IN (SELECT Codice
FROM DIPARTIMENTO
WHERE Città = 'Torino');
```

```
SELECT CodDip
FROM IMPIEGATO
WHERE Stipendio > 45.000
AND CodDip = (SELECT CodDip
FROM DIPARTIMENTO
WHERE Città <> 'Torino');
```



Operatori insiemistici

Operatore UNION

Operatore UNION

➤ Operatore insiemistico di unione

A UNION B

➤ Esegue l'unione delle due espressioni relazionali A e B

- le espressioni relazionali A e B possono essere generate da istruzioni SELECT
- richiede la compatibilità di schema tra A e B
- rimozione dei duplicati
 - UNION rimuove i duplicati
 - UNION ALL non rimuove i duplicati

UNION: esempio

- Trovare il codice dei prodotti di colore rosso o forniti dal fornitore F2 (o entrambe le cose)

UNION: esempio

➤ Trovare il codice dei prodotti di colore rosso o forniti dal fornitore F2 (o entrambe le cose)

P

<u>CodP</u>	<u>NomeP</u>	<u>Colore</u>	<u>Taglia</u>	<u>Magazzino</u>
P1	Maglia	Rosso	40	Torino
P2	Jeans	Verde	48	Milano
P3	Camicia	Blu	48	Roma
P4	Camicia	Blu	44	Torino
P5	Gonna	Blu	40	Milano
P6	Bermuda	Rosso	42	Torino

FP

<u>CodF</u>	<u>CodP</u>	<u>Qta</u>
F1	P1	300
F1	P2	200
F1	P3	400
F1	P4	200
F1	P5	100
F1	P6	100
F2	P1	300
F2	P2	400
F3	P2	200
F4	P3	200
F4	P4	300
F4	P5	400

UNION: esempio

➤ Trovare *il codice dei prodotti di colore rosso* o forniti dal fornitore F2 (o entrambe le cose)

```
SELECT CodP  
FROM P  
WHERE Colore='Rosso'
```

UNION: esempio

➤ Trovare *il codice dei prodotti di colore rosso* o forniti dal fornitore F2 (o entrambe le cose)

```
SELECT CodP
FROM P
WHERE Colore='Rosso'
```

P

<u>CodP</u>	NomeP	Colore	Taglia	Magazzino
P1	Maglia	Rosso	40	Torino
P2	Jeans	Verde	48	Milano
P3	Camicia	Blu	48	Roma
P4	Camicia	Blu	44	Torino
P5	Gonna	Blu	40	Milano
P6	Bermuda	Rosso	42	Torino



CodP
P1
P6

UNION: esempio

➤ Trovare il *codice dei prodotti* di colore rosso o *forniti dal fornitore F2* (o entrambe le cose)

```
SELECT CodP
FROM FP
WHERE CodF='F2'
```

UNION: esempio

➤ Trovare il *codice dei prodotti* di colore rosso o *forniti dal fornitore F2* (o entrambe le cose)

FP

<u>CodF</u>	<u>CodP</u>	Qta
F1	P1	300
F1	P2	200
F1	P3	400
F1	P4	200
F1	P5	100
F1	P6	100
F2	P1	300
F2	P2	400
F3	P2	200
F4	P3	200
F4	P4	300
F4	P5	400

```
SELECT CodP
FROM FP
WHERE CodF='F2'
```



CodP
P1
P2

UNION: esempio

- Trovare il codice dei prodotti di colore rosso o forniti dal fornitore F2 (o entrambe le cose)

```
SELECT CodP
FROM P
WHERE Colore='Rosso'
UNION
SELECT CodP
FROM FP
WHERE CodF='F2';
```

UNION: esempio

- Trovare il codice dei prodotti di colore rosso o forniti dal fornitore F2 (o entrambe le cose)

```
SELECT CodP
FROM P
WHERE Colore='Rosso'
UNION
SELECT CodP
FROM FP
WHERE CodF='F2';
```

CodP
P1
P6

CodP
P1
P2



R

CodP
P1
P2
P6

UNION: esempio

➤ Trovare il codice dei prodotti di colore rosso o forniti dal fornitore F2 (o entrambe le cose)

```
SELECT CodP
FROM P
WHERE Colore='Rosso'
UNION
SELECT CodP
FROM FP
WHERE CodF='F2';
```

R

CodP
P1
P2
P6

Rimozione del duplicato

UNION: esempio

➤ Trovare il codice dei prodotti di colore rosso o forniti dal fornitore F2 (o entrambe le cose)

*Compatibilità
di schema*

```
SELECT CodP
FROM P
WHERE Colore='Rosso'
UNION
SELECT CodP
FROM FP
WHERE CodF='F2';
```

UNION ALL: esempio

- Trovare il codice dei prodotti di colore rosso o forniti dal fornitore F2 (o entrambe le cose)

```
SELECT CodP
FROM P
WHERE Colore='Rosso'
UNION ALL
SELECT CodP
FROM FP
WHERE CodF='F2';
```

UNION ALL: esempio

➤ Trovare il codice dei prodotti di colore rosso o forniti dal fornitore F2 (o entrambe le cose)

```
SELECT CodP
FROM P
WHERE Colore='Rosso'
UNION ALL
SELECT CodP
FROM FP
WHERE CodF='F2';
```

CodP
P1
P6

CodP
P1
P2



R

CodP
P1
P1
P2
P6



Operatori insiemistici

Operatore INTERSECT

Operatore INTERSECT

➤ Operatore insiemistico di intersezione

$A \text{ INTERSECT } B$

➤ Esegue l'intersezione delle due espressioni relazionali A e B

- le espressioni relazionali A e B possono essere generate da istruzioni SELECT
- richiede la compatibilità di schema tra A e B

INTERSECT: esempio

➤ Trovare le città che sono sia sede di fornitori, sia magazzino di prodotti

P

<u>CodP</u>	NomeP	Colore	Taglia	Magazzino
P1	Maglia	Rosso	40	Torino
P2	Jeans	Verde	48	Milano
P3	Camicia	Blu	48	Roma
P4	Camicia	Blu	44	Torino
P5	Gonna	Blu	40	Milano
P6	Bermuda	Rosso	42	Torino

F

<u>CodF</u>	NomeF	NSoci	Sede
F1	Andrea	2	Torino
F2	Luca	1	Milano
F3	Antonio	3	Milano
F4	Gabriele	2	Torino
F5	Matteo	3	Venezia

INTERSECT: esempio

➤ Trovare le città che sono sia sede di fornitori, sia magazzino di prodotti

```
SELECT Sede  
FROM F
```

F

<u>CodF</u>	NomeF	NSoci	Sede
F1	Andrea	2	Torino
F2	Luca	1	Milano
F3	Antonio	3	Milano
F4	Gabriele	2	Torino
F5	Matteo	3	Venezia



Sede
Torino
Milano
Milano
Torino
Venezia

INTERSECT: esempio

➤ Trovare *le città che sono* sia sede di fornitori, sia *magazzino di prodotti*

```
SELECT Magazzino  
FROM P
```

P

<u>CodP</u>	NomeP	Colore	Taglia	Magazzino
P1	Maglia	Rosso	40	Torino
P2	Jeans	Verde	48	Milano
P3	Camicia	Blu	48	Roma
P4	Camicia	Blu	44	Torino
P5	Gonna	Blu	40	Milano
P6	Bermuda	Rosso	42	Torino



Magazzino
Torino
Milano
Roma
Torino
Milano
Torino

INTERSECT: esempio

➤ Trovare le città che sono sia sede di fornitori, sia magazzino di prodotti

```
SELECT Sede  
FROM F  
INTERSECT  
SELECT Magazzino  
FROM P;
```

INTERSECT: esempio

➤ Trovare le città che sono sia sede di fornitori, sia magazzino di prodotti

```
SELECT Sede
FROM F
INTERSECT
SELECT Magazzino
FROM P;
```

Sede
Torino
Milano
Milano
Torino
Venezia

Magazzino
Torino
Milano
Roma
Torino
Milano
Torino



R

Torino
Milano

Equivalenza con altri operatori

- L'operazione di intersezione può essere eseguita anche mediante
- il join
 - l'operatore IN

Equivalenza con il join

- La clausola **FROM** contiene le relazioni interessate dall'intersezione
- La clausola **WHERE** contiene condizioni di join tra gli attributi presenti nella clausola **SELECT** delle espressioni relazionali A e B

Equivalenza con il join: esempio

- Trovare le città che sono sia sede di fornitori, sia magazzino di prodotti

Equivalenza con il join: esempio

- Trovare le città che sono sia sede di fornitori, sia magazzino di prodotti

```
SELECT Sede  
FROM F, P  
...
```

Equivalenza con il join: esempio

- Trovare le città che sono sia sede di fornitori, sia magazzino di prodotti

```
SELECT Sede  
FROM F, P  
WHERE F.Sede=P.Magazzino;
```

Equivalenza con l'operatore IN

- Una delle due espressioni relazionali diviene un'interrogazione nidificata mediante l'operatore **IN**
- Gli attributi nella clausola **SELECT** esterna, uniti eventualmente da un costruttore di tupla, costituiscono la parte sinistra dell'operatore **IN**

Equivalenza con IN: esempio

- Trovare le città che sono sia sede di fornitori, sia magazzino di prodotti

Equivalenza con IN: esempio

➤ Trovare le città che sono sia sede di fornitori, sia magazzino di prodotti

```
SELECT Magazzino
FROM P
WHERE Magazzino IN (SELECT Sede
                    FROM F);
```



Operatori insiemistici

Operatore EXCEPT

Operatore EXCEPT

➤ Operatore insiemistico di differenza

$A \text{ EXCEPT } B$

➤ Sottrae l'espressione relazionale B all'espressione relazionale A

- richiede la compatibilità di schema tra A e B

EXCEPT: esempio

➤ Trovare le città che sono sede di fornitori, ma non magazzino di prodotti

P

<u>CodP</u>	NomeP	Colore	Taglia	Magazzino
P1	Maglia	Rosso	40	Torino
P2	Jeans	Verde	48	Milano
P3	Camicia	Blu	48	Roma
P4	Camicia	Blu	44	Torino
P5	Gonna	Blu	40	Milano
P6	Bermuda	Rosso	42	Torino

F

<u>CodF</u>	NomeF	NSoci	Sede
F1	Andrea	2	Torino
F2	Luca	1	Milano
F3	Antonio	3	Milano
F4	Gabriele	2	Torino
F5	Matteo	3	Venezia

EXCEPT: esempio

➤ Trovare *le città che sono sede di fornitori*, ma non magazzino di prodotti

```
SELECT Sede  
FROM F
```

F

<u>CodF</u>	NomeF	NSoci	Sede
F1	Andrea	2	Torino
F2	Luca	1	Milano
F3	Antonio	3	Milano
F4	Gabriele	2	Torino
F5	Matteo	3	Venezia



Sede
Torino
Milano
Milano
Torino
Venezia

EXCEPT: esempio

➤ Trovare *le città che sono* sede di fornitori, ma non *magazzino di prodotti*

```
SELECT Magazzino  
FROM P
```

P

<u>CodP</u>	NomeP	Colore	Taglia	Magazzino
P1	Maglia	Rosso	40	Torino
P2	Jeans	Verde	48	Milano
P3	Camicia	Blu	48	Roma
P4	Camicia	Blu	44	Torino
P5	Gonna	Blu	40	Milano
P6	Bermuda	Rosso	42	Torino



Magazzino
Torino
Milano
Roma
Torino
Milano
Torino

EXCEPT: esempio

➤ Trovare le città che sono sede di fornitori, ma non magazzino di prodotti

```
SELECT Sede  
FROM F  
EXCEPT  
SELECT Magazzino  
FROM P;
```

EXCEPT: esempio

➤ Trovare le città che sono sede di fornitori, ma non magazzino di prodotti

```
SELECT Sede  
FROM F  
EXCEPT  
SELECT Magazzino  
FROM P;
```

Sede
Torino
Milano
Milano
Torino
Venezia

Magazzino
Torino
Milano
Roma
Torino
Milano
Torino



R

Venezia

Equivalenza con l'operatore NOT IN

- L'operazione di differenza può essere eseguita anche mediante l'operatore NOT IN
- l'espressione relazionale B è nidificata all'interno dell'operatore NOT IN
 - gli attributi nella clausola SELECT dell'espressione relazionale A, uniti eventualmente da un costruttore di tupla, costituiscono la parte sinistra dell'operatore NOT IN

Equivalenza con l'operatore NOT IN: esempio

- Trovare le città che sono sede di fornitori, ma non magazzino di prodotti

Equivalenza con l'operatore NOT IN: esempio

➤ Trovare le città che sono sede di fornitori, ma non magazzino di prodotti

```
SELECT Sede
FROM F
WHERE Sede NOT IN (SELECT Magazzino
                   FROM P);
```



Linguaggio SQL: fondamenti

Istruzioni di aggiornamento

Istruzioni di aggiornamento

- Introduzione
- Istruzione INSERT
- Istruzione DELETE
- Istruzione UPDATE

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

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

Istruzione INSERT

- Inserimento di una sola tupla
 - assegnazione di un valore costante ad ogni attributo

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 il prodotto P7 con nome: maglia, colore: viola, taglia: 40, magazzino: Venezia

Inserimento di una tupla: esempio (n.1)

➤ Inserire il prodotto P7 con nome: maglia, colore: viola, taglia: 40, magazzino: Venezia

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

Inserimento di una tupla: esempio (n.1)

- Inserire il prodotto P7 con nome: maglia, colore: viola, taglia: 40, magazzino: Venezia

```
INSERT INTO P (CodP, NomeP, Colore, Taglia, Magazzino)  
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 il prodotto P7 con nome: maglia, colore: viola, taglia: 40, magazzino: Venezia

```
INSERT INTO P (CodP, NomeP, Colore, Taglia, Magazzino)  
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

Inserimento di una tupla: esempio (n.1)

- Inserire il prodotto P7 con nome: maglia, colore: viola, taglia: 40, magazzino: Venezia

```
INSERT INTO P (CodP, NomeP, Colore, Taglia, Magazzino)  
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 il prodotto P8 con magazzino: Genova, taglia: 42

Inserimento di una tupla: esempio (n.2)

➤ Inserire il prodotto P8 con magazzino: Genova, taglia: 42

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

Inserimento di una tupla: esempio (n.2)

- Inserire il prodotto P8 con magazzino: Genova, taglia: 42

```
INSERT INTO P (CodP, Magazzino, 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

Inserimento di una tupla: esempio (n.2)

- Inserire il prodotto P8 con magazzino : Genova, taglia: 42

```
INSERT INTO P (CodP, Magazzino, 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

Inserimento: integrità referenziale

- Inserire una nuova fornitura relativa al fornitore F20, prodotto P20 e quantità 1000

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

Inserimento: integrità referenziale

- Inserire una nuova fornitura relativa al fornitore F20, prodotto P20 e quantità 1000

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

- Vincolo di integrità referenziale
 - è necessario che P20 e F20 siano già presenti rispettivamente in P e F

Inserimento: integrità referenziale

- Inserire una nuova fornitura relativa al fornitore F20, prodotto P20 e quantità 1000

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

- Vincolo di integrità referenziale
 - è necessario che P20 e F20 siano già presenti rispettivamente in P e F
 - se il vincolo non è soddisfatto, l'inserimento non deve essere eseguito

Inserimento di più record

```
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

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);
```

Istruzione DELETE

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

- Cancellazione dalla tabella *NomeTabella* di tutte le tuple che soddisfano il predicato

Istruzione DELETE

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

- Cancellazione dalla tabella *NomeTabella* di tutte le tuple che soddisfano il predicato
- Occorre sempre verificare che la cancellazione non causi la violazione di vincoli di integrità referenziale

Istruzione DELETE: esempio (n.1)

➤ Cancellare tutte le forniture

```
DELETE FROM FP;
```

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

Istruzione DELETE: esempio (n.2)

- Cancellare la tupla corrispondente al fornitore con codice F1

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

Istruzione DELETE: esempio (n.2)

- Cancellare la tupla corrispondente al fornitore con codice F1

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

- Se in FP esistono forniture che fanno riferimento ai fornitori cancellati, la base di dati perde la propria integrità
- avviene una violazione del vincolo di integrità referenziale tra FP e F
 - occorre propagare la cancellazione

Istruzione DELETE: esempio (n.2)

- Cancellare la tupla corrispondente al fornitore con codice F1

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

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

Istruzione DELETE: esempio (n.2)

- Cancellare la tupla corrispondente al fornitore con codice F1

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

- Per mantenere la coerenza è necessario completare le operazioni su entrambe le tabelle

Istruzione DELETE: esempio complesso

➤ Cancellare i fornitori di Milano

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

- Se in FP esistono forniture che fanno riferimento ai fornitori cancellati, è violato il vincolo di integrità referenziale tra FP e F
- occorre cancellare anche tali forniture in FP

Istruzione DELETE: esempio complesso

➤ Cancellare i fornitori di Milano

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

```
DELETE FROM FP  
WHERE CodF IN (SELECT CodF  
                FROM F  
                WHERE Sede='Milano');
```

Istruzione DELETE: esempio complesso

➤ Cancellare i fornitori di Milano

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

```
DELETE FROM FP  
WHERE CodF IN (SELECT CodF  
                FROM F  
                WHERE Sede='Milano');
```

➤ In quale ordine devono essere eseguite le due operazioni di cancellazione?

Istruzione DELETE: esempio complesso

➤ Ordine corretto di esecuzione

```
DELETE FROM FP
WHERE CodF IN (SELECT CodF
                FROM F
                WHERE Sede='Milano');
```

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

Istruzione UPDATE

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

- Tutti i record 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 e assegnare NULL a magazzino

Aggiornamento di una tupla

- Aggiornare le caratteristiche del prodotto P1:
assegnare giallo al colore, incrementare la taglia
di 2 e assegnare NULL a magazzino

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

Aggiornamento di una tupla

- Aggiornare le caratteristiche del prodotto P1: assegnare giallo al colore, incrementare la taglia di 2 e assegnare NULL a magazzino

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

- È aggiornata la tupla individuata dal codice P1

Aggiornamento multiplo

- Aggiornare il numero dei soci al doppio del valore per tutti i fornitori di Milano

Aggiornamento multiplo

- Aggiornare il numero dei soci al doppio del valore per tutti i fornitori di Milano

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

Aggiornamento multiplo

- Aggiornare il numero dei soci al doppio del valore per tutti i fornitori di Milano

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

- Sono aggiornate tutte le tuple individuate dal predicato nella clausola **WHERE**

Aggiornamento con sottointerrogazione

- Aggiornare a 10 la quantità fornita per tutti i fornitori di 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');
```

Aggiornamento di più tabelle

- Modificare con il valore F9 il codice del fornitore F2

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

- Se in FP esistono forniture che fanno riferimento ai codici dei fornitori aggiornati, è violato il vincolo di integrità referenziale
 - occorre aggiornare anche tali forniture in FP

Aggiornamento di più tabelle

- Modificare con il valore F9 il codice del fornitore F2

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

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

Aggiornamento di più tabelle

- Modificare con il valore F9 il codice del fornitore F2

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

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

- Per mantenere la coerenza è necessario completare l'aggiornamento di entrambe le tabelle