



## Interrogazioni nidificate

Operatore EXISTS

# Operatore EXISTS (n.1)

➤ Trovare il nome dei fornitori del prodotto P2

## Operatore EXISTS (n.1)

➤ Trovare il nome dei fornitori del prodotto P2



*Trovare il nome dei fornitori **per cui esiste**  
una fornitura del prodotto P2*

# Operatore EXISTS (n.1)

➤ Trovare il nome dei fornitori del prodotto P2

```
SELECT NomeF  
FROM F  
WHERE EXISTS (...);
```

# Operatore EXISTS (n.1)

➤ Trovare il nome dei fornitori del prodotto P2

```
SELECT NomeF
FROM F
WHERE EXISTS (SELECT *
              FROM FP
              WHERE CodP='P2'
              ...
```

# Operatore EXISTS (n.1)

➤ Trovare il nome dei fornitori del prodotto P2

```
SELECT NomeF
FROM F
WHERE EXISTS (SELECT *
              FROM FP
              WHERE CodP='P2'
              AND FP.CodF=F.CodF );
```

# Condizione di correlazione (n.1)

➤ Trovare il nome dei fornitori del prodotto P2

```
SELECT NomeF
FROM F
WHERE EXISTS (SELECT *
              FROM FP
              WHERE CodP='P2'
              AND FP.CodF=F.CodF );
```

*Condizione di correlazione*

# Funzionamento di EXISTS (n.1)

➤ Trovare il nome dei fornitori del prodotto P2

F

CodF	NomeF	NSoci	Città
F1	Andrea	2	Torino
F2	Luca	1	Milano
F3	Antonio	3	Milano
F4	Gabriele	2	Torino
F5	Matteo	3	Venezia



# Funzionamento di EXISTS (n.1)

➤ Trovare il nome dei fornitori del prodotto P2  
F

CodF	NomeF	NSoci	Città
F1	Andrea	2	Torino
F2	Luca	1	Milano
F3	Antonio	3	Milano
F4	Gabriele	2	Torino
F5	Matteo	3	Venezia

```
SELECT *
```

```
FROM FP
```

```
WHERE CodP='P2'
```

```
AND FP.CodF='F1'
```

↑  
*Valore di CodF nella  
riga corrente di F*

# Funzionamento di EXISTS (n.1)

➤ Trovare il nome dei fornitori del prodotto P2

F

CodF	NomeF	NSoci	Città
F1	Andrea	2	Torino
F2	Luca	1	Milano
F3	Antonio	3	Milano
F4	Gabriele	2	Torino
F5	Matteo	3	Venezia

FP

CodF	CodP	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 *
```

```
FROM FP
```

```
WHERE CodP='P2'
```

```
AND FP.CodF='F1'
```

*Valore di CodF nella  
riga corrente di F*

# Funzionamento di EXISTS (n.1)

➤ Trovare il nome dei fornitori del prodotto P2

F

CodF	NomeF	NSoci	Città
F1	Andrea	2	Torino
F2	Luca	1	Milano
F3	Antonio	3	Milano
F4	Gabriele	2	Torino
F5	Matteo	3	Venezia

FP

CodF	CodP	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

➤ Il predicato **EXISTS** è vero per F1 poiché esiste una fornitura di P2 per F1

- F1 fa parte del risultato dell'interrogazione

# Funzionamento di EXISTS (n.1)

➤ Trovare il nome dei fornitori del prodotto P2

F

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

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

# Funzionamento di EXISTS (n.1)

➤ Trovare il nome dei fornitori del prodotto P2

F

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

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

➤ Il predicato **EXISTS** è falso per F4 poiché non esiste una fornitura di P2 per F4

- F4 non fa parte del risultato dell'interrogazione

# Risultato dell'interrogazione (n.1)

➤ Trovare il nome dei fornitori del prodotto P2

R

NomeF
Andrea
Luca
Antonio

# Predicati con EXISTS

- Il predicato contenente EXISTS è
- vero se l'interrogazione interna restituisce almeno una tupla
  - falso se l'interrogazione interna restituisce l'insieme vuoto

# Predicati con EXISTS

- Il predicato contenente EXISTS è
  - vero se l'interrogazione interna restituisce almeno una tupla
  - falso se l'interrogazione interna restituisce l'insieme vuoto
- Nell'interrogazione interna a EXISTS, la clausola SELECT è obbligatoria, ma irrilevante, perchè gli attributi non sono visualizzati



## Predicati con EXISTS

- Il predicato contenente EXISTS è
  - vero se l'interrogazione interna restituisce almeno una tupla
  - falso se l'interrogazione interna restituisce l'insieme vuoto
- Nell'interrogazione interna a EXISTS, la clausola SELECT è obbligatoria, ma irrilevante, perchè gli attributi non sono visualizzati
- La condizione di correlazione lega l'esecuzione dell'interrogazione interna al valore di attributi della tupla corrente nell'interrogazione esterna

## Visibilità degli attributi

- Un'interrogazione nidificata può far riferimento ad attributi definiti in interrogazioni più esterne
- Un'interrogazione non può far riferimento ad attributi referenziati
  - in un'interrogazione nidificata al suo interno
  - in un'interrogazione allo stesso livello



## Interrogazioni nidificate

Operatore NOT EXISTS

## Operatore NOT EXISTS (n.1)

➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2

# Operatore NOT EXISTS (n.1)

➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2



*Trovare il nome dei fornitori per cui non esiste una fornitura del prodotto P2*

# Operatore NOT EXISTS (n.1)

➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2

```
SELECT NomeF  
FROM F  
WHERE NOT EXISTS (...);
```

# Operatore NOT EXISTS (n.1)

➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2

```
SELECT NomeF
FROM F
WHERE NOT EXISTS (SELECT *
                  FROM FP
                  WHERE CodP='P2'
                  ...
```

# Operatore NOT EXISTS (n.1)

➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2

```
SELECT NomeF
FROM F
WHERE NOT EXISTS (SELECT *
                  FROM FP
                  WHERE CodP='P2'
                  AND FP.CodF=F.CodF );
```



# Operatore NOT EXISTS (n.1)

➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2

```
SELECT NomeF
FROM F
WHERE NOT EXISTS (SELECT *
                  FROM FP
                  WHERE CodP='P2'
                  AND FP.CodF=F.CodF );
```

*Condizione di correlazione*

# Funzionamento di NOT EXISTS (n.1)

➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2

F



CodF	NomeF	NSoci	Città
F1	Andrea	2	Torino
F2	Luca	1	Milano
F3	Antonio	3	Milano
F4	Gabriele	2	Torino
F5	Matteo	3	Venezia

# Funzionamento di NOT EXISTS (n.1)

➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2

F



CodF	NomeF	NSoci	Città
F1	Andrea	2	Torino
F2	Luca	1	Milano
F3	Antonio	3	Milano
F4	Gabriele	2	Torino
F5	Matteo	3	Venezia

```
SELECT *
```

```
FROM FP
```

```
WHERE CodP='P2' AND
```

```
FP.CodF='F1'
```

*Valore di CodF nella  
riga corrente di F*

# Funzionamento di NOT EXISTS (n.1)


➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2

F



CodF	NomeF	NSoci	Città
F1	Andrea	2	Torino
F2	Luca	1	Milano
F3	Antonio	3	Milano
F4	Gabriele	2	Torino
F5	Matteo	3	Venezia

FP



CodF	CodP	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 *  
FROM FP  
WHERE CodP='P2' AND  
FP.CodF='F1'
```

# Funzionamento di NOT EXISTS (n.1)


➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2

F



CodF	NomeF	NSoci	Città
F1	Andrea	2	Torino
F2	Luca	1	Milano
F3	Antonio	3	Milano
F4	Gabriele	2	Torino
F5	Matteo	3	Venezia

FP



CodF	CodP	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

➤ Il predicato **NOT EXISTS** è falso per F1 perché esiste una fornitura di P2 per F1

- F1 *non* fa parte del risultato dell'interrogazione

# Funzionamento di NOT EXISTS (n.1)

➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2

F

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

# Funzionamento di NOT EXISTS (n.1)


➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2

F



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

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

# Funzionamento di NOT EXISTS (n.1)

➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2

F

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





# Funzionamento di NOT EXISTS (n.1)

➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2

F

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

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

# Funzionamento di NOT EXISTS (n.1)

➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2

F

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

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

➤ Il predicato **NOT EXISTS** è vero per F4 perché non esiste una fornitura di P2 per F4

- F4 fa parte del risultato dell'interrogazione

# Funzionamento di NOT EXISTS (n.1)

➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2

F

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



# Funzionamento di NOT EXISTS (n.1)

➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2

F

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

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

# Risultato dell'interrogazione (n.1)

➤ Trovare il nome dei fornitori che *non* forniscono il prodotto P2

R

NomeF
Gabriele
Matteo

# Predicato con NOT EXISTS

- Il predicato contenente NOT EXISTS è
  - vero se l'interrogazione interna restituisce l'insieme vuoto
  - falso se l'interrogazione interna restituisce almeno una tupla
- La condizione di correlazione lega l'esecuzione dell'interrogazione interna al valore di attributi della tupla corrente nell'interrogazione esterna



# Interrogazioni nidificate

Correlazione tra interrogazioni

# Correlazione tra interrogazioni

- Può essere necessario legare la computazione di un'interrogazione nidificata al valore di uno o più attributi in un'interrogazione più esterna
  - il legame è espresso da una o più condizioni di correlazione



# Condizione di correlazione

- Una condizione di correlazione
- è indicata nella clausola **WHERE** dell'interrogazione nidificata che la richiede
  - è un predicato che lega attributi di tabelle nella **FROM** dell'interrogazione nidificata con attributi di tabelle nella **FROM** di interrogazioni più esterne

# Condizione di correlazione

- Una condizione di correlazione
  - è indicata nella clausola WHERE dell'interrogazione nidificata che la richiede
  - è un predicato che lega attributi di tabelle nella FROM dell'interrogazione nidificata con attributi di tabelle nella FROM di interrogazioni più esterne
- Non si possono esprimere condizioni di correlazione
  - in interrogazioni allo stesso livello di nidificazione
  - contenenti riferimenti ad attributi di una tabella nella FROM di un'interrogazione nidificata

## Correlazione tra interrogazioni (n.1)

- Per ogni prodotto, trovare il codice del fornitore che ne fornisce la quantità massima

# Correlazione tra interrogazioni (n.1)

➤ Per ogni prodotto, trovare il codice del fornitore che ne fornisce la quantità massima

```
SELECT CodP, CodF  
FROM FP AS FPX  
WHERE Qta = (...
```

```
)
```

*Quantità massima  
per il prodotto  
corrente*

# Correlazione tra interrogazioni (n.1)

➤ Per ogni prodotto, trovare il codice del fornitore che ne fornisce la quantità massima

```
SELECT CodP, CodF
FROM FP AS FPX
WHERE Qta = (SELECT MAX(Qta)
             FROM FP AS FPY
             ... )
```

*Quantità  
massima*

# Correlazione tra interrogazioni (n.1)

➤ Per ogni prodotto, trovare il codice del fornitore che ne fornisce la quantità massima

```
SELECT CodP, CodF
FROM FP AS FPX
WHERE Qta = (SELECT MAX(Qta)
             FROM FP AS FPY
             WHERE FPY.CodP=FPX.CodP);
```

*Quantità  
massima  
per il  
prodotto  
corrente*

# Correlazione tra interrogazioni (n.1)

- Per ogni prodotto, trovare il codice del fornitore che ne fornisce la quantità massima

```
SELECT CodP, CodF
FROM FP AS FPX
WHERE Qta = (SELECT MAX(Qta)
             FROM FP AS FPY
             WHERE FPY.CodP=FPX.CodP);
```

*Condizione di correlazione*

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, CodF
FROM FP AS FPX
WHERE Qta = (SELECT MAX(Qta)
             FROM FP AS FPY
             WHERE FPY. CodP=FPX.CodP);
```



## Schema di esempio (n.2)

VIAGGIO (CodV, LuogoPartenza, LuogoArrivo,  
OraPartenza, OraArrivo)

## Correlazione tra interrogazioni (n.2)

VIAGGIO (CodV, LuogoPartenza, LuogoArrivo,  
OraPartenza, OraArrivo)

- Trovare il codice dei viaggi che hanno una durata inferiore alla durata media dei viaggi sullo stesso percorso (caratterizzato dallo stesso luogo di partenza e di arrivo)

## Correlazione tra interrogazioni (n.2)

VIAGGIO (CodV, LuogoPartenza, LuogoArrivo,  
OraPartenza, OraArrivo)

- Trovare il codice dei viaggi che hanno una durata inferiore alla durata media dei viaggi sullo stesso percorso (caratterizzato dallo stesso luogo di partenza e di arrivo)

```
SELECT CodV  
FROM VIAGGIO AS VA  
WHERE OraArrivo-OraPartenza < (...
```

} *Durata  
media  
dei viaggi  
sul percorso  
corrente*

## Correlazione tra interrogazioni (n.2)

VIAGGIO (CodV, LuogoPartenza, LuogoArrivo,  
OraPartenza, OraArrivo)

- Trovare il codice dei viaggi che hanno una durata inferiore alla durata media dei viaggi sullo stesso percorso (caratterizzato dallo stesso luogo di partenza e di arrivo)

```
SELECT CodV
```

```
FROM VIAGGIO AS VA
```

```
WHERE OraArrivo-OraPartenza <
```

```
(SELECT AVG(OraArrivo-OraPartenza)
```

```
FROM VIAGGIO AS VB
```

```
... )
```

*Durata  
media  
dei viaggi*

## Correlazione tra interrogazioni (n.2)

VIAGGIO (CodV, LuogoPartenza, LuogoArrivo,  
OraPartenza, OraArrivo)

- Trovare il codice dei viaggi che hanno una durata inferiore alla durata media dei viaggi sullo stesso percorso (caratterizzato dallo stesso luogo di partenza e di arrivo)

```
SELECT CodV
FROM VIAGGIO AS VA
WHERE OraArrivo-OraPartenza <
      (SELECT AVG(OraArrivo-OraPartenza)
       FROM VIAGGIO AS VB
        WHERE VB.LuogoPartenza=VA.LuogoPartenza
              AND VB.LuogoArrivo=VA.LuogoArrivo);
```

## Correlazione tra interrogazioni (n.2)

VIAGGIO (CodV, LuogoPartenza, LuogoArrivo,  
OraPartenza, OraArrivo)

- Trovare il codice dei viaggi che hanno una durata inferiore alla durata media dei viaggi sullo stesso percorso (caratterizzato dallo stesso luogo di partenza e di arrivo)

```
SELECT CodV
FROM VIAGGIO AS VA
WHERE OraArrivo-OraPartenza <
      (SELECT AVG(OraArrivo-OraPartenza)
FROM VIAGGIO AS VB Condizioni di correlazione
WHERE VB.LuogoPartenza=VA.LuogoPartenza
AND VB.LuogoArrivo=VA.LuogoArrivo);
```