

Basi di Dati

Esercitazione

Algebra Relazionale e SQL

Tanti autori diversi ☺

11 novembre 2021

Fornitori (CodiceFornitore, Nome, Indirizzo, Città)

Prodotti (CodiceProdotto, Nome, Marca, Modello)

Catalogo (Fornitore, Prodotto, Costo)

con vincoli di integrità referenziale

fra Prodotto e la chiave di Prodotti

fra Fornitore e la chiave di Fornitori

1. Trovare Nome, Marca e Modello dei prodotti acquistabili con meno di 2000 €.
2. Trovare i nomi dei fornitori che distribuiscono prodotti IBM (IBM è una marca di prodotto).
3. Trovare i codici dei prodotti che sono forniti da almeno due fornitori.
4. Trovare i codici dei fornitori che distribuiscono tutti i prodotti.
5. Trovare i nomi dei fornitori che forniscono tutti i prodotti IBM presenti nel catalogo.

Fornitori (CodiceFornitore, Nome, Indirizzo, Città)

Prodotti (CodiceProdotto, Nome, Marca, Modello)

Catalogo (Fornitore, Prodotto, Costo)

con vincoli di integrità referenziale

fra Prodotto e la chiave di Prodotti

fra Fornitore e la chiave di Fornitori

- [Dati di esempio per RelaX](#)
- [Script SQL](#)

Nome	CodiceFornitore	Indirizzo	Città
Ladroni	001	Via Ostiense	Roma
Risparmietti	002	Viale Marconi	Roma
Teloporto	010	Via Roma	Milano

Fornitori

Fornitore	Prodotto	Costo
001	0002	€ 3.200
001	0003	€ 2.200
002	0001	€ 1.900
002	0002	€ 2.500
002	0003	€ 1.800
010	0001	€ 2.200
010	0003	€ 2.000

Catalogo

CodiceProdotto	Nome	Marca	Modello
0001	Notebook	IBM	390 x
0002	Desktop	IBM	510
0003	Desktop	ACER	730

Prodotti

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

1. Trovare Nome, Marca e Modello dei prodotti acquistabili con meno di 2000 €.

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

1. Trovare Nome, Marca e Modello dei prodotti acquistabili con meno di 2000 €.

proj_{Nome, Marca, Modello}

(**sel**_{Costo < 2000} (**Prodotti**) $\bowtie_{CP=P}$
Catalogo)

(Prodotti $\bowtie_{CP=P}$ Catalogo)

Modello	Marca	Nome	CodiceProdotto
390 x	IBM	Notebook	0001
510	IBM	Desktop	0002
730	ACER	Desktop	0003

Modello	Marca	Nome	CP	Fornitore	Prodotto	Costo
510	IBM	Desktop	0002	001	0002	€ 3.200
730	ACER	Desktop	0003	001	0003	€ 2.200
390 x	IBM	Notebook	0001	002	0001	€ 1.900
510	IBM	Desktop	0002	002	0002	€ 2.500
730	ACER	Desktop	0003	002	0003	€ 1.800
390 x	IBM	Notebook	0001	010	0001	€ 2.200
730	ACER	Desktop	0003	010	0003	€ 2.000

$\text{sel}_{\text{Costo} < 2000} (\text{Prodotti} \bowtie_{\text{CP}=\text{P}} \text{Catalogo})$

Modello	Marca	Nome	CP	Fornitore	Prodotto	Costo
510	IBM	Desktop	0002	001	0002	€ 3.200
730	ACER	Desktop	0003	001	0003	€ 2.200
390 x	IBM	Notebook	0001	002	0001	€ 1.900
510	IBM	Desktop	0002	002	0002	€ 2.500
730	ACER	Desktop	0003	002	0003	€ 1.800
390 x	IBM	Notebook	0001	010	0001	€ 2.200
730	ACER	Desktop	0003	010	0003	€ 2.000

$\text{proj}_{\text{Nome, Marca, Modello}}(\text{sel}_{\text{Costo} < 2000}(\text{Prodotti} \bowtie_{\text{CP}=\text{P}} \text{Catalogo}))$

Modello	Marca	Nome	CP	Fornitore	Prodotto	Costo
510	IBM	Desktop	0002	001	0002	€ 3.200
730	ACER	Desktop	0003	001	0003	€ 2.200
390 x	IBM	Notebook	0001	002	0001	€ 1.900
510	IBM	Desktop	0002	002	0002	€ 2.500
730	ACER	Desktop	0003	002	0003	€ 1.800
390 x	IBM	Notebook	0001	010	0001	€ 2.200
730	ACER	Desktop	0003	010	0003	€ 2.000

Esercizio 1

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

1. Trovare Nome, Marca e Modello dei prodotti acquistabili con meno di 2000 €.

proj_{Nome, Marca, Modello}

(Prodotti $\bowtie_{CP=P}$ **sel**_{Costo < 2000} **(Catalogo)**)

Nome	Marca	Modello
Notebook	IBM	390 x
Desktop	ACER	730

Esercizio 1

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

1. Trovare Nome, Marca e Modello dei prodotti acquistabili con meno di 2000 €.

proj_{Nome, Marca, Modello}

(sel_{Costo < 2000} (**Prodotti** ⋈_{CP=P} **Catalogo**))

Ora in SQL

Esercizio 1

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

1. Trovare Nome, Marca e Modello dei prodotti acquistabili con meno di 2000 €.

proj_{Nome, Marca, Modello}

(sel_{Costo < 2000} (**Prodotti** ⋈_{CP=P} **Catalogo**))

```
SELECT nome, marca, modello
```

```
FROM catalogo JOIN prodotti ON (prodotto=codiceprodotto)
```

```
Where costo < 2000
```

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

2. Trovare i nomi dei fornitori che distribuiscono prodotti IBM (IBM è la marca di un prodotto).

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

2. Trovare i nomi dei fornitori che distribuiscono prodotti IBM (IBM è la marca di un prodotto).

```
projNome (
  selMarca = 'IBM' ( (Fornitori ⋈CF=F Catalogo)
    ⋈P=CP (projCP,Marca (Prodotti)) )
```

(Fornitori $\bowtie_{CF=F}$ Catalogo)

Nome	CodiceFornitore	Indirizzo	Città
Ladroni	001	Via Ostense	Roma
Risparmietti	002	Viale Marconi	Roma
Teloporto	010	Via Roma	Milano

Nome	CF	Indirizzo	Città	Fornitore	Prodotto	Costo
Ladroni	001	Via Ostiense	Roma	001	0002	€ 3.200
Ladroni	001	Via Ostiense	Roma	001	0003	€ 2.200
Risparmiet.	002	Viale Marconi	Roma	002	0001	€ 1.900
Risparmiet.	002	Viale Marconi	Roma	002	0002	€ 2.500
Risparmiet.	002	Viale Marconi	Roma	002	0003	€ 1.800
Teloporto	010	Via Roma	Milano	010	0001	€ 2.200
Teloporto	010	Via Roma	Milano	010	0003	€ 2.000

Le Relazioni

$((\text{Fornitori} \bowtie_{CF=F} \text{Catalogo}) \bowtie_{P=CP} (\text{proj}_{CP, \text{Marca}} (\text{Prodotti})))$

Modello	Marca	Nome	CodiceProdotto
390 x	IBM	Notebook	0001
510	IBM	Desktop	0002
730	ACER	Desktop	0003

Nome	CF	..	Città	Fornitore	Prodotto	Costo	Modello	Marca	..	CP
Ladroni	001	..	Roma	001	0002	€ 3.200	510	IBM	..	0002
Ladroni	001	..	Roma	001	0003	€ 2.200	730	ACER	..	0003
Risparmiet.	002	..	Roma	002	0001	€ 1.900	390 x	IBM	..	0001
Risparmiet.	002	..	Roma	002	0002	€ 2.500	510	IBM	..	0002
Risparmiet.	002	..	Roma	002	0003	€ 1.800	730	ACER	..	0003
Teloporto	010	..	Milano	010	0001	€ 2.200	390 x	IBM	..	0001
Teloporto	010	..	Milano	010	0003	€ 2.000	730	ACER	..	0003

```

sel  $\text{Marca} = \text{'IBM'}$ 
  ( (Fornitori  $\bowtie_{CF=F}$  Catalogo)  $\bowtie_{P=CP}$  (proj $_{CP, \text{Marca}}$  (Prodotti)) )
  
```

Nome	CF	..	Città	Fornitore	Prodotto	Costo	Modello	Marca	..	CP
Ladroni	001	..	Roma	001	0002	€ 3.200	510	IBM	..	0002
Ladroni	001	..	Roma	001	0003	€ 2.200	730	ACER	..	0003
Risparmiet.	002	..	Roma	002	0001	€ 1.900	390 x	IBM	..	0001
Risparmiet.	002	..	Roma	002	0002	€ 2.500	510	IBM	..	0002
Risparmiet.	002	..	Roma	002	0003	€ 1.800	730	ACER	..	0003
Teloporto	010	..	Milano	010	0001	€ 2.200	390 x	IBM	..	0001
Teloporto	010	..	Milano	010	0003	€ 2.000	730	ACER	..	0003

$\text{proj}_{\text{Nome}} (\sigma_{\text{Marca} = \text{'IBM'}}$
 $((\text{Fornitori} \bowtie_{\text{CF}=\text{F}} \text{Catalogo}) \bowtie_{\text{P}=\text{CP}} (\text{proj}_{\text{CP,Marca}} (\text{Prodotti})))$

Nome	CF	Città	Fornitore	Prodotto	Costo	Modello	Marca	CP
Ladroni	001	Roma	001	0002	€ 3.200	510	IBM	0002
Ladroni	001	Roma	001	0003	€ 2.200	730	ACER	0003
Risparmiet.	002	Roma	002	0001	€ 1.900	390 x	IBM	0001
Risparmiet.	002	Roma	002	0002	€ 2.500	510	IBM	0002
Risparmiet.	002	Roma	002	0003	€ 1.800	730	ACER	0003
Teloporto	010	Milano	010	0001	€ 2.200	390 x	IBM	0001
Teloporto	010	Milano	010	0003	€ 2.000	730	ACER	0003

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

2. Trovare i nomi dei fornitori che distribuiscono prodotti IBM (IBM è la marca di un prodotto).

```
projNome (
  selMarca = 'IBM' ( (Fornitori ⋈CF=F Catalogo)
    ⋈P=CP (projCP,Marca (Prodotti)) )
```

Esercizio 2

```
select *  
from prodotti P, catalogo C, fornitori F  
where P.CP = C.P and F.CF = C.F;
```

Output pane

	CodiceProdotto oid	Nome text	Marca text	Modello text	Fornitore oid	Prodotto oid	Costo real	Nome text	CodiceFornitore oid	Indirizzo text	Città text
1	2	Desktop	IBM	510	1	2	3200	Ladro	1	Via Ost	Roma
2	3	Desktop	ACER	790	1	3	2200	Ladro	1	Via Ost	Roma
3	1	Notebook	IBM	390 X	2	1	1900	Rispa	2	Viale M	Roma
4	2	Desktop	IBM	510	2	2	2500	Rispa	2	Viale M	Roma
5	3	Desktop	ACER	790	2	3	1800	Rispa	2	Viale M	Roma
6	1	Notebook	IBM	390 X	10	1	2200	Telop	10	Via Rom	Mila
7	3	Desktop	ACER	790	10	3	2000	Telop	10	Via Rom	Mila

Esercizio 2

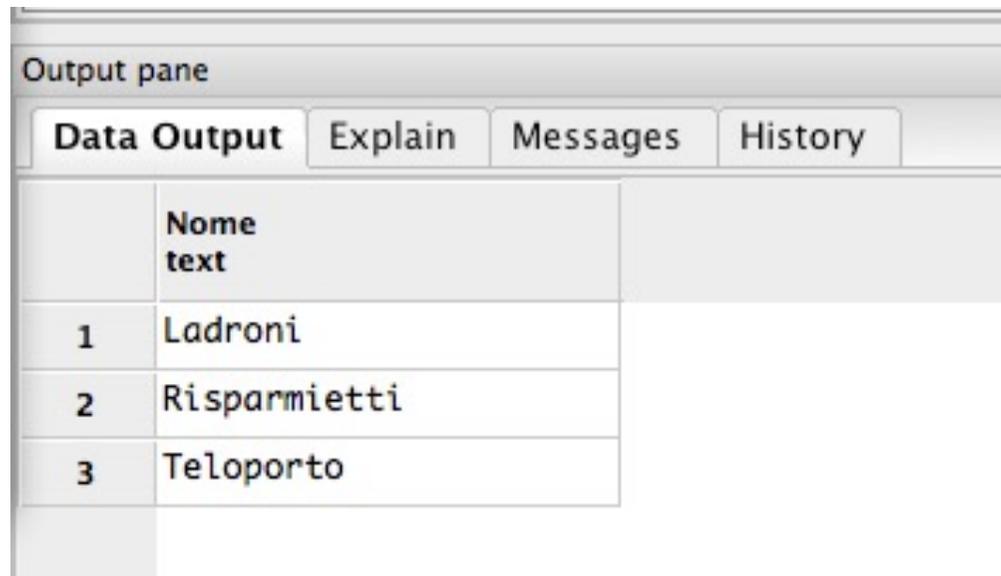
```
select *  
from prodotti P, catalogo C, fornitori F  
where P.CP = C.P and C.F = F.CF  
and P.marca = 'IBM';
```

Output pane

	CodiceProdotto oid	Nome text	Marca text	Modello text	Fornitore oid	Prodotto oid	Costo real	Nome text	CodiceFornitore oid	Indirizzo text	Città text
1	2	Desktop	IBM	510	1	2	3200	Ladro	1	Via Ost	Roma
2	1	Notebook	IBM	390 X	2	1	1900	Rispa	2	Viale M	Roma
3	2	Desktop	IBM	510	2	2	2500	Rispa	2	Viale M	Roma
4	1	Notebook	IBM	390 X	10	1	2200	Telop	10	Via Rom	Mila

Esercizio 2

```
select distinct fornitori.nome  
from prodotti, catalogo, fornitori  
where CP = P and F = CF and marca = 'IBM';
```

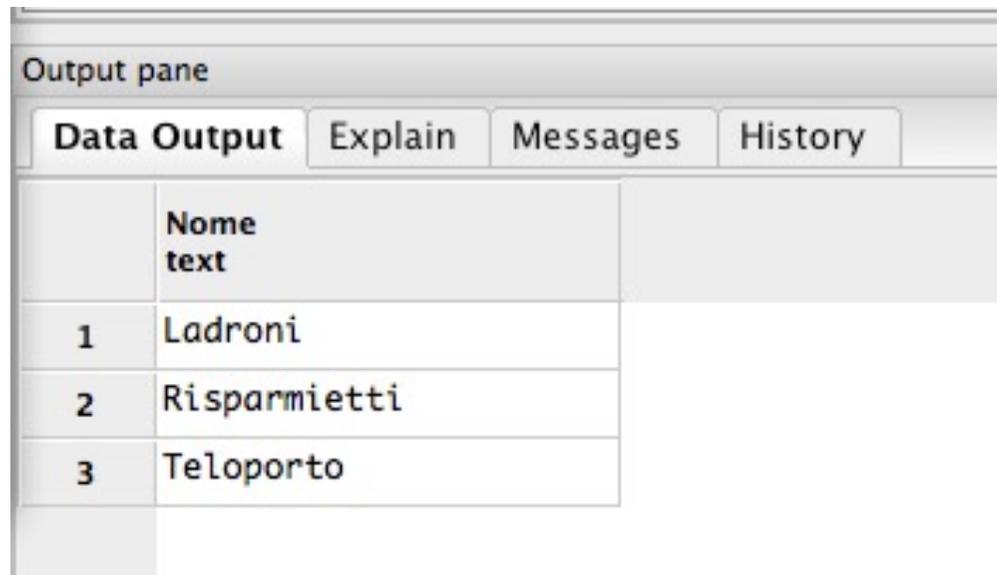


The screenshot shows a database interface with an "Output pane" containing four tabs: "Data Output", "Explain", "Messages", and "History". The "Data Output" tab is active, displaying a table with three rows of results. The table has two columns: "Nome" (text) and an unlabeled column. The rows are numbered 1, 2, and 3, corresponding to the supplier names "Ladroni", "Risparmietti", and "Teloporto".

	Nome text	
1	Ladroni	
2	Risparmietti	
3	Teloporto	

Esercizio 2

```
SELECT DISTINCT fornitori.nome
FROM (fornitori JOIN catalogo ON codicefornitore = fornitore)
      JOIN prodotti ON prodotto = codiceprodotto
WHERE marca = 'IBM'
```



The screenshot shows a database interface with an "Output pane" containing four tabs: "Data Output", "Explain", "Messages", and "History". The "Data Output" tab is active, displaying a table with three rows of results. The table has two columns: "Nome" (text) and an unlabeled column. The rows are numbered 1, 2, and 3, corresponding to the values "Ladroni", "Risparmietti", and "Teloporto" respectively.

	Nome text	
1	Ladroni	
2	Risparmietti	
3	Teloporto	

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

3. Trovare i codici dei prodotti che sono forniti da almeno due fornitori.

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

3. Trovare i codici dei prodotti che sono forniti da almeno due fornitori.

Nota bene: a lezione è stata mostrata solo in SQL e non con questa tecnica

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

3. Trovare i codici dei prodotti che sono forniti da almeno due fornitori.

$\text{proj}_P (\text{sel}_{F \langle \rangle F'} ($
 $\text{Catalogo} \bowtie_{P=P'} \rho_{X' \leftarrow X} (\text{Catalogo}))$

$\rho_{X' \leftarrow X}$ indica una ridenominazione in cui ciascun attributo A viene cambiato in A'

Le Relazioni

Catalogo $\bowtie_{P=P'}$
 $\rho_{X' \leftarrow X}(\text{Catalogo})$

Fornitore	Prodotto	Costo	Fornitore'	Prodotto'	Costo'
001	0002	€ 3.200	001	0002	€ 3.200
002	0002	€ 2.500	001	0002	€ 3.200
001	0003	€ 2.200	001	0003	€ 2.200
002	0003	€ 1.800	001	0003	€ 2.200
010	0003	€ 2.000	001	0003	€ 2.200
002	0001	€ 1.900	002	0001	€ 1.900
010	0001	€ 2.200	002	0001	€ 1.900
002	0002	€ 2.500	002	0002	€ 2.500
001	0002	€ 3.200	002	0002	€ 2.500
002	0003	€ 1.800	002	0003	€ 1.800
001	0003	€ 2.200	002	0003	€ 1.800
010	0003	€ 2.000	002	0003	€ 1.800
010	0001	€ 2.200	010	0001	€ 2.200
002	0001	€ 1.900	010	0001	€ 2.200
010	0003	€ 2.000	010	0003	€ 2.000
001	0003	€ 2.200	010	0003	€ 2.000
002	0003	€ 1.800	010	0003	€ 2.000

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

3. Trovare i codici dei prodotti che sono forniti da almeno due fornitori.

$\text{proj}_P (\text{sel}_F \langle \rangle F' ($
 $\text{Catalogo} \bowtie_{P=P'} \rho_{X' \leftarrow X} (\text{Catalogo}))$

CodiceProdotto
0001
0002
0003

$\rho_{X' \leftarrow X}$ indica una ridenominazione in cui ciascun attributo A viene cambiato in A'

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

3. Trovare i codici dei prodotti che sono forniti da almeno due fornitori.

Variante con aggregazione:

```
proj Catalogo.P ( $\sigma$  NUM_F  $\geq$  2 (agg P;  
Count(F)  $\rightarrow$  NUM_F (Catalogo)))
```

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

3. Trovare i codici dei prodotti che sono forniti da almeno due fornitori.

$$\text{proj}_P \left(\text{sel}_F \langle \rangle F' \left(\text{Catalogo} \bowtie_{P=P'} \rho_{X' \leftarrow X} (\text{Catalogo}) \right) \right)$$

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

3. Trovare i codici dei prodotti che sono forniti da almeno due fornitori.

Nota bene: questa soluzione (basata sul join della relazione con se stessa) non è stata discussa a lezione

Esercizio 3

```
select *  
from catalogo c1, catalogo c2  
where c1.P = c2.P;
```

Output pane

	Fornitore oid	Prodotto oid	Costo real	Fornitore oid	Prodotto oid	Costo real
1	1	2	3200	2	2	2500
2	1	2	3200	1	2	3200
3	1	3	2200	10	3	2000
4	1	3	2200	2	3	1800
5	1	3	2200	1	3	2200
6	2	1	1900	10	1	2200
7	2	1	1900	2	1	1900
8	2	2	2500	2	2	2500
9	2	2	2500	1	2	3200
10	2	3	1800	10	3	2000
11	2	3	1800	2	3	1800
12	2	3	1800	1	3	2200
13	10	1	2200	10	1	2200
14	10	1	2200	2	1	1900
15	10	3	2000	10	3	2000
16	10	3	2000	2	3	1800
17	10	3	2000	1	3	2200

Esercizio 3

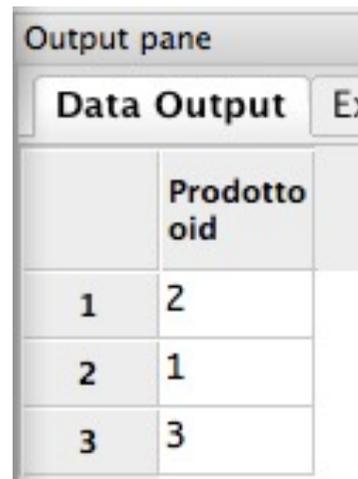
```
select *  
from catalogo c1, catalogo c2  
where c1.p = c2.p  
      c1.f <> c2.f;
```

Output pane

	Fornitore oid	Prodotto oid	Costo real	Fornitore oid	Prodotto oid	Costo real
1	1	2	3200	2	2	2500
2	1	3	2200	10	3	2000
3	1	3	2200	2	3	1800
4	2	1	1900	10	1	2200
5	2	2	2500	1	2	3200
6	2	3	1800	10	3	2000
7	2	3	1800	1	3	2200
8	10	1	2200	2	1	1900
9	10	3	2000	2	3	1800
10	10	3	2000	1	3	2200

Esercizio 3

```
select distinct c1.P  
from catalogo c1, catalogo c2  
where c1.P = c2.P  
        c1.F <> c2.F;
```



	Prodotto oid
1	2
2	1
3	3

Soluzione con aggregazione e having

```
select distinct C.P  
from Catalogo C  
group by C.P  
having count(*) > 1;
```

Soluzione con vista

```
create view numfornitoriperprodotto
as select prodotto, count(*) as numerofornitori
   from catalogo
   group by prodotto
```

```
select *
from numfornitoriperprodotto
where numerofornitori >= 2
```

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

4. Trovare i codici dei fornitori che distribuiscono tutti i prodotti presenti nel catalogo.

Nota bene, assumiamo che il catalogo contenga tutti i fornitori e prodotti di interesse

Esercizio 4

Nome	CodiceFornitore	Indirizzo	Città
Ladroni	001	Via Ostense	Roma
Risparmietti	002	Viale Marconi	Roma
Teloporto	010	Via Roma	Milano

Fornitore	Prodotto	Costo
001	0002	€ 3.200
001	0003	€ 2.200
002	0001	€ 1.900
002	0002	€ 2.500
002	0003	€ 1.800
010	0001	€ 2.200
010	0003	€ 2.000

CodiceProdotto	Nome	Marca	Modello
0001	Notebook	IBM	390 x
0002	Desktop	IBM	510
0003	Desktop	ACER	730

4. Trovare i codici dei fornitori che distribuiscono tutti i prodotti presenti nel catalogo.

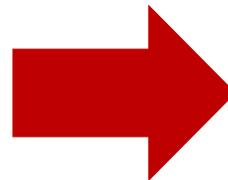
$\text{proj}_F(\text{Catalogo}) -$

$\text{proj}_F((\text{proj}_F(\text{Catalogo}) \bowtie \text{proj}_P(\text{Catalogo}))$

$- \text{proj}_{F,P}(\text{Catalogo}))$

$\text{proj}_F(\text{Catalogo})$ – Fornitori che non forniscono tutti i prodotti

Fornitore	Prodotto	Costo
001	0002	€ 3.200
001	0003	€ 2.200
002	0001	€ 1.900
002	0002	€ 2.500
002	0003	€ 1.800
010	0001	€ 2.200
010	0003	€ 2.000



Fornitore
001
002
010

– Fornitori che non forniscono tutti i prodotti

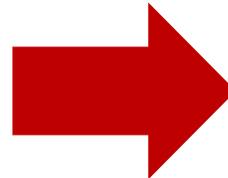
$\text{proj}_F(\text{Catalogo}) \bowtie \text{proj}_P(\text{Catalogo})$

Tutte le coppie Fornitore/Prodotto

Prodotto
0001
0002
0003

\bowtie

Fornitore
001
002
010



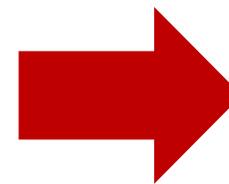
Prodotto	Fornitore
0001	001
0002	002
0003	010
0001	002
0002	010
0003	001
0001	010
0002	001
0003	002

$$\text{proj}_F(\text{Catalogo}) \bowtie \text{proj}_P(\text{Catalogo}) - \text{proj}_{F,P}(\text{Catalogo})$$

Tutte le coppie Fornitore/Prodotto – Fornitore/Prodotto

Prodotto	Fornitore
0001	001
0002	002
0003	010
0001	002
0002	010
0003	001
0001	010
0002	001
0003	002

Prodotto	Fornitore
0002	001
0003	001
0001	002
0002	002
0003	002
0001	010
0003	010

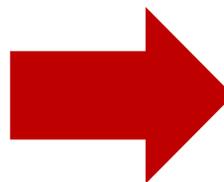


Le forniture mancate

$$\text{proj}_F ((\text{proj}_F (\text{Catalogo}) \bowtie \text{proj}_P (\text{Catalogo})) \\ - \text{proj}_{F,P} (\text{Catalogo}))$$

Fornitori che non forniscono tutti i prodotti

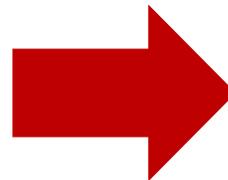
Prodotto	Fornitore
0001	001
0002	002
0003	010
0001	002
0002	010
0003	001
0001	010
0002	001
0003	002



Fornitore
001
010

$\text{proj}_F(\text{Catalogo})$ - Fornitori che non forniscono tutti i prodotti

Fornitore	Prodotto	Costo
001	0002	€ 3.200
001	0003	€ 2.200
002	0001	€ 1.900
002	0002	€ 2.500
002	0003	€ 1.800
010	0001	€ 2.200
010	0003	€ 2.000



Fornitore
001
002
010

-

Fornitore
001
010

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

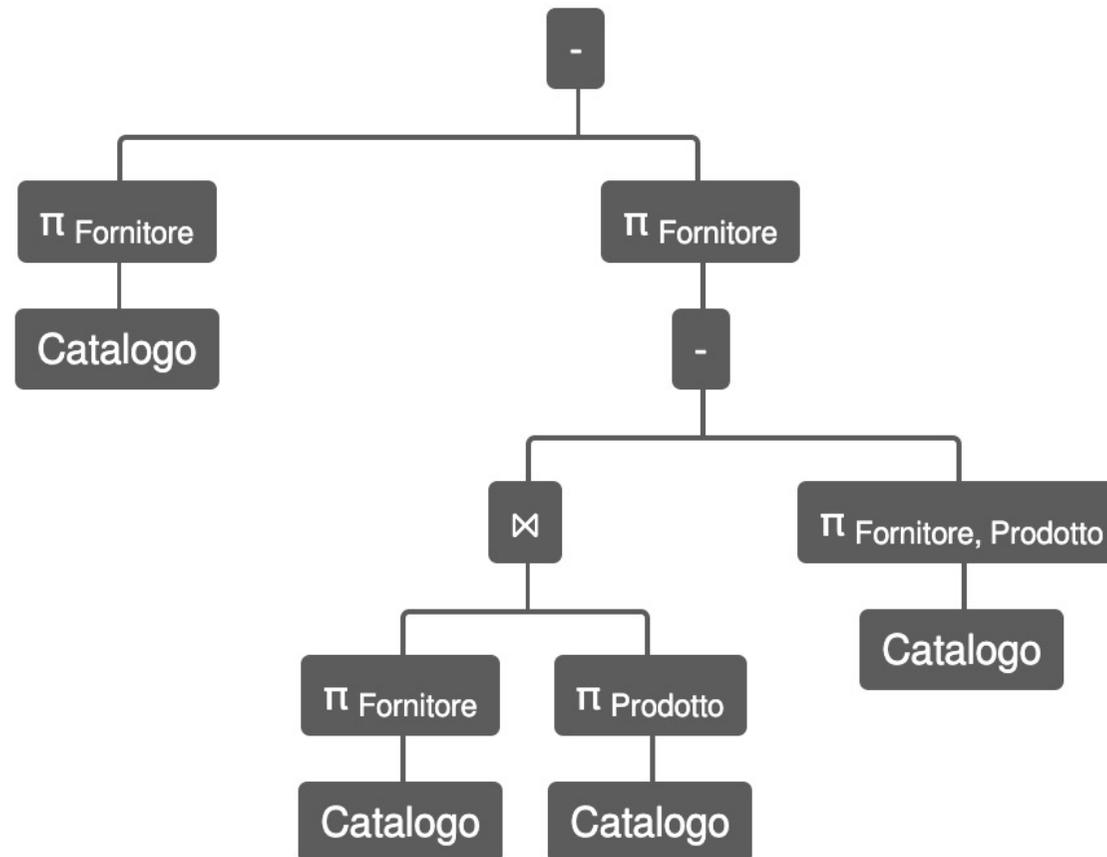
4. Trovare i codici dei fornitori che distribuiscono tutti i prodotti presenti nel catalogo.

$\text{proj}_F(\text{Catalogo}) -$

$\text{proj}_F((\text{proj}_F(\text{Catalogo}) \bowtie \text{proj}_P(\text{Catalogo}))$
 $- \text{proj}_{F,P}(\text{Catalogo}))$

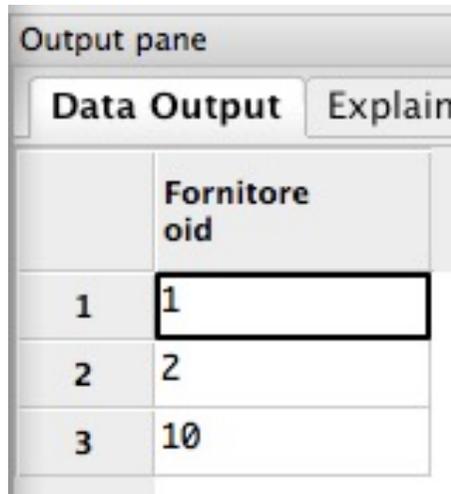
$\text{proj}_F(\text{Catalogo}) -$

$\text{proj}_F((\text{proj}_F(\text{Catalogo}) \bowtie \text{proj}_P(\text{Catalogo})) - \text{proj}_{F,P}(\text{Catalogo}))$



$\text{proj}_F(\text{Catalogo})$ - mancate forniture

```
select distinct F  
from Catalogo
```



	Fornitore oid
1	1
2	2
3	10

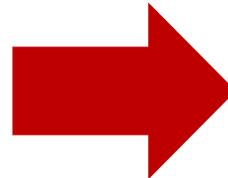
$$\text{proj}_F(\text{Catalogo}) \bowtie \text{proj}_P(\text{Catalogo})$$

Tutte le coppie Fornitore/Prodotto

Prodotto
0001
0002
0003

 \bowtie

Fornitore
001
002
010



Prodotto	Fornitore
0001	001
0002	002
0003	010
0001	002
0002	010
0003	001
0001	010
0002	001
0003	002

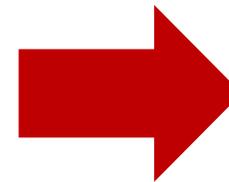
Esercizio 4

$\text{proj}_F(\text{Catalogo}) \bowtie \text{proj}_P(\text{Catalogo}) - \text{proj}_{F,P}(\text{Catalogo})$

Tutte le coppie Fornitore/Prodotto – Fornitore/Prodotto

Prodotto	Fornitore
0001	001
0002	002
0003	010
0001	002
0002	010
0003	001
0001	010
0002	001
0003	002

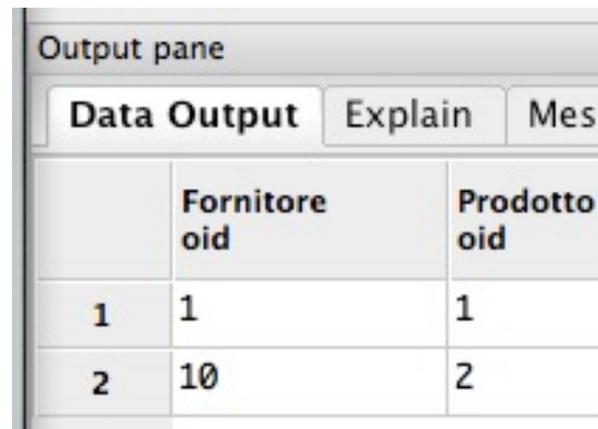
Prodotto	Fornitore
0002	001
0003	001
0001	002
0002	002
0003	002
0001	010
0003	010



Le forniture mancate

Esercizio 4

```
create view forniture_mancate as  
select c1.F, c2.P  
from Catalogo c1, Catalogo c2  
except  
select F, P  
from Catalogo;
```



	Fornitore oid	Prodotto oid
1	1	1
2	10	2

$\text{proj}_F(\text{Catalogo}) - \text{Forniture Mancate}$

```
select distinct F
from catalogo
```

except

```
select F
from forniture_mancate
```

Output pane

Data Output Explain

	Fornitore oid
1	1
2	2
3	10

Output pane

Data Output Explain Mes:

	Fornitore oid	Prodotto oid
1	1	1
2	10	2

Soluzione con la doppia nidificazione con not exist

```
select codicefornitore
from fornitori f
where not exists (
    select *
    from prodotti p
    where not exists (
        select *
        from catalogo
        where prodotto=p.codiceprodotto
        and fornitore=f.codicefornitore
    )
)
```

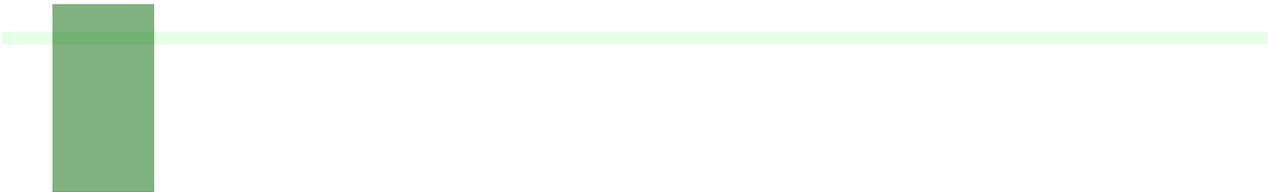


Soluzione con due viste (i fornitori che forniscono un numero di prodotti pari al numero complessivo di prodotti stessi)

```
create view contaprodotto as
SELECT fornitore, count(prodotto) as numeroprodotto
FROM catalogo
GROUP BY fornitore;
```

```
create view contaprodotto totale as
SELECT count(*) as numerototaleprodotti
FROM prodotti;
```

```
SELECT *
FROM contaprodotto, contaprodotto totale
where numeroprodotto=numerototaleprodotti
```



Stesso principio (i fornitori che forniscono un numero di prodotti pari al numero complessivo di prodotti stessi) ma con interrogazione nidificata

```
create view contaprodotto as
SELECT fornitore,
       count(prodotto) as numeroprodotto
FROM catalogo
GROUP BY fornitore;
```

```
SELECT *
FROM contaprodotto
where numeroprodotto =
       (select count(*) from prodotti)
```



Stesso principio (i fornitori che forniscono un numero di prodotti pari al numero complessivo di prodotti stessi) ma con la nidificazione nella HAVING

```
SELECT fornitore,  
       count(prodotto) as numeroprodotto  
FROM catalogo  
GROUP BY fornitore  
HAVING count(prodotto) =  
       (select count(*) from prodotti)
```



Soluzione con doppia differenza (tutti i fornitori esclusi quelli che hanno un prodotto non fornito)

```
create view mancateforniture as
SELECT distinct c1.fornitore, c2.prodotto
from catalogo c1, catalogo c2
EXCEPT
SELECT fornitore, prodotto
FROM catalogo;
```

```
select distinct fornitore from catalogo
Except
select fornitore from mancateforniture
```



Esercizi aggiuntivi, da svolgere autonomamente

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

5. Trovare i nomi dei fornitori che forniscono tutti i prodotti IBM presenti nel catalogo.

La soluzione è identica a quella della interrogazione 4, con, al posto della relazione Catalogo la vista catalogoIBM :

```
CatalogoIBM := projP, F (
  selMarca='IBM' (Catalogo ⋈P=CP
  Prodotti) )
```

5. Trovare i nomi dei fornitori che forniscono tutti i prodotti IBM presenti nel catalogo.

La soluzione è identica a quella della interrogazione 4, con, al posto della relazione Catalogo la vista catalogoIBM :

CatalogoIBM := proj_{P,F} (
sel_{Marca='IBM'} (Catalogo ⋈_{P=CP} Prodotti))

Soluzione con viste ed EXCEPT (1)

```
create view forniture_mancate_IBM as  
  select C1.P, C2.F  
  from  
    Catalogo C1,  
    Catalogo C2,  
    Prodotti P  
  where  
    C1.P = P.CP  
    and P.Marca = 'IBM'  
  except  
  select C.P, C.F  
  from Catalogo C;
```

Soluzione con viste ed EXCEPT (2)

```
select F.Nome
from
  Catalogo C,
  Fornitori F
where C.F = F.CF prodotti p
except
select F.Nome
from
  forniture_mancate_IBM FM,
  Fornitori F
where FM.F = F.CF;
```

Soluzione con la doppia nidificazione con not exist

```
select codicefornitore
from fornitori f
where not exists (
    select *
    from prodotti p
    where marca = 'IBM'
        and not exists (
            select *
            from catalogo
            where prodotto=p.codiceprodotto
                and fornitore=f.codicefornitore
        )
    )
```

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

6. Trovare tutte le coppie di prodotti dell' IBM per cui esiste un solo fornitore da cui i due prodotti possono essere acquistati insieme. Trovare per ogni coppia anche il codice del relativo fornitore.

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

CatalogoIBM :=

$\text{proj}_{P,F}(\text{sel}_{\text{Marca}='IBM'}(\text{Catalogo} \bowtie_{P=CP} \text{Prodotti}))$

CatalogoIBM' := $\rho_{X' \leftarrow X}(\text{CatalogoIBM})$

PC := $\text{proj}_{P,F}(\text{CatalogoIBM}) -$

$\text{proj}_{C.P,C.F}(\text{sel}_{C.F \neq C'.F}$
 $(\text{CatalogoIBM} \bowtie_{C.P=C'.P} \text{CatalogoIBM'}))$

PC' := $\rho_{X' \leftarrow X}(\text{PC})$

$\text{proj}_{PC.P,PC'.P,PC.F}(\text{sel}_{PC.P > PC'.P}(\text{PC} \bowtie_{PC.F=PC'.F} \text{PC'}))$

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

```
create view furniture_ibm as
select c1.fornitore, c1.prodotto as prodotto1,
c2.prodotto as prodotto2
from catalogo c1, catalogo c2, prodotti p1, prodotti p2
where c1.fornitore = c2.fornitore and
c1.prodotto > c2.prodotto
and c1.prodotto = p1.codiceprodotto
and c2.prodotto = p2.codiceprodotto
and p1.marca = 'IBM' and p2.marca = 'IBM'

select prodotto1, prodotto2
from furniture_ibm group by prodotto1, prodotto2
having count(*) = 1
```

7. Trovare il codice e il nome del prodotto che è venduto al prezzo massimo (mostrando anche il costo).

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

7. Trovare il codice e il nome del prodotto che è venduto al prezzo massimo (mostrando anche il costo).

```
projCP, Nome, CostoMax  
  (proj P, CostoMax  
    (sel Costo >= CostoMax  
      (Catalogo × agg max(Costo) → CostoMax  
      (Catalogo))) ⋈P = CP (Prodotti))
```

8. Trovare per ogni prodotto il costo minimo a cui viene venduto (mostrando codice del prodotto e costo).

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

8. Trovare per ogni prodotto il costo minimo a cui viene venduto (mostrando codice del prodotto e costo).

```
projCatalogo.P, minCosto  
(agg P; min(Costo) → minCosto (Catalogo))
```

8. Trovare per ogni prodotto il costo minimo a cui viene venduto (mostrando codice del prodotto e costo).

```
select C.P, min(C.Costo) as "Costo Minimo"  
from Catalogo C  
group by C.P;
```

9. Trovare per ogni prodotto, il fornitore che lo vende a costo più alto, mostrando il nome del prodotto, il nome del fornitore e il costo stesso.

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

9. Trovare per ogni prodotto, il fornitore che lo vende a costo più alto, mostrando il nome del prodotto, il nome del fornitore e il costo stesso.

```
create view prodotti_costi_massimi as  
select  
    C.P, max(C.Costo) as "Costo Massimo"  
from Catalogo C  
group by C.P;
```

```
select P.Nome as "Nome prodotto",  
        F.Nome as "Nome fornitore",  
        C.Costo  
from  
        Catalogo C,  
        prodotti_costi_massimi CM,  
        Fornitori F,  
        Prodotti P  
where  
        C.P = CM.P  
        and F.CF = C.F  
        and P.CP = C.P  
        and C.Costo = CM."Costo Massimo";
```

10. Trovare quali prodotti del fornitore "Ladroni" sono venduti anche dal fornitore "Risparmietti", ma a un prezzo più basso. Riportare il codice dei prodotti.

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

10. Trovare quali prodotti del fornitore "Ladroni" sono venduti anche dal fornitore "Risparmietti", ma a un prezzo più basso. Riportare il codice dei prodotti.

```
LADRONI = ren LADRONI (proj Catalogo.P, Catalogo.Costo (sel
  Fornitori.Nome = 'Ladroni' (Catalogo ⋈ Catalogo.F =
  Fornitori.CF (Fornitori))))
```

```
RISPARMIETTI = ren RISPARMIETTI (proj Catalogo.P,
  Catalogo.Costo (sel Fornitori.Nome = 'Risparmietti'
  (Catalogo ⋈ Catalogo.F = Fornitori.CF (Fornitori))))
```

```
proj LADRONI.P (sel RISPARMIETTI.Costo < LADRONI.Costo
  (LADRONI ⋈ LADRONI.P = RISPARMIETTI.P (RISPARMIETTI)))
```

10. Trovare quali prodotti del fornitore "Ladroni" sono venduti anche dal fornitore "Risparmietti", ma a un prezzo più basso. Riportare il codice dei prodotti.

```
select C1.P
from
    Catalogo C1, Catalogo C2,
    Fornitori F1, Fornitori F2
where C1.F = F1.CF
    and C2.F = F2.CF
    and F1.Nome = 'Ladroni'
    and F2.Nome = 'Risparmietti'
    and C1.P = C2.P
    and C1.Costo > C2.Costo;
```

11. Trovare il codice e il nome dei fornitori che vendono Notebook oppure Desktop.

Fornitori (CF, Nome, Indirizzo, Città)

Prodotti (CP, Nome, Marca, Modello)

Catalogo (F, P, Costo)

11. Trovare il codice e il nome dei fornitori che vendono Notebook oppure Desktop.

```
proj Fornitori.Nome, Catalogo.F ((sel Nome='Notebook' ∨  
Nome='Desktop' (Catalogo ⋈ Catalogo.P = Prodotti.CP  
(Prodotti))) ⋈ Catalogo.F = Fornitori.CF (Fornitori))
```

11. Trovare il codice e il nome dei fornitori che vendono Notebook oppure Desktop.

```
select distinct F.CF, F.Nome
from
    Fornitori F,
    Catalogo C,
    Prodotti P
where C.F = F.CF
    and C.P = P.CP
    and
    (P.Nome = 'Notebook' or
     P.Nome = 'Desktop'));
```