

1) Si consideri lo schema relazionale $R(A,B,C,D,E,F)$ con le seguenti DF:

$A \rightarrow C, \quad DE \rightarrow F, \quad B \rightarrow D$

- a) Si trovino le chiavi di R.
- b) Dire se lo schema è in 3FN o in FNBC.
- c) Se lo schema non è in FNBC lo si normalizzi e si dica se la decomposizione preserva le dipendenze.
- d) Si aggiunga alle DF su R la dipendenza $A \rightarrow B$. Se A non è chiave si aggiunga qualche altra dipendenza (con un solo attributo a sinistra e a destra) in modo che A diventi chiave.

2) Si considerino i seguenti insiemi di FD sullo schema $R(A,B,C,D)$

$F1 = \{A \rightarrow B, B \rightarrow C\}$

$F2 = \{A \rightarrow B, A \rightarrow C\}$

$F3 = \{A \rightarrow B, AB \rightarrow C\}$

- a) Si esprima (in modo formale) che cosa significa dire che due insiemi di FD sono equivalenti, e si dica come si procede in pratica per verificare tale equivalenza
- b) F1 e F2 sono equivalenti?
- d) F2 e F3 sono equivalenti?
- d) F1 e F3 sono equivalenti?

3) Si consideri lo schema relazionale:

Clienti(CF, Nome, Telefono), con chiavi CF e (Nome, Telefono)

Depositi(Numero, CFCliente, Tipo, Saldo), con chiave Numero; il tipo del deposito può essere 'R' (a risparmio) o 'C' (in conto corrente).

- a) Definire in SQL un'interrogazione che ritorna una relazione vuota se e solo se in Clienti vale la dipendenza $\text{Nome} \rightarrow \text{Telefono}$.
- b) Definire in SQL un'interrogazione per trovare il nome dei clienti che hanno tutti i depositi in conto corrente con saldo negativo (ovvero, non ci interessa lo stato dei loro depositi a risparmio)
- c) (opzionale) Definire in SQL un'interrogazione per trovare il nome dei clienti che hanno tutti i depositi in conto corrente e con saldo negativo.
- d) Si considerino le seguenti interrogazioni per trovare il nome e il telefono dei clienti che hanno un deposito a risparmio o in conto corrente con saldo negativo:

d1) SELECT Nome, Telefono
FROM Clienti, Depositi
WHERE CF = CFCliente AND Saldo < 0 AND (Tipo = 'R' OR Tipo = 'C');

d2) SELECT Nome, Telefono
FROM Clienti
WHERE CF IN (SELECT CFCliente
FROM Depositi
WHERE Saldo < 0 AND (Tipo = 'R' OR Tipo = 'C'));

d3) SELECT Nome, Telefono
FROM Clienti, Depositi
WHERE CF = CFCliente AND Saldo < 0 AND Tipo = 'R'
UNION
SELECT Nome, Telefono
FROM Clienti, Depositi
WHERE CF = CFClienti AND Saldo < 0 AND Tipo = 'C';

Due interrogazioni sono equivalenti se producono lo stesso risultato, ignorando l'ordine. Dire se le tre interrogazioni sono equivalenti, oppure quali di loro lo sono.

- 4) Per l'interrogazione seguente, si fornisca un piano di accesso senza uso di indici ed uno che usa due indici a vostra scelta:

```
SELECT Nome, sum(saldo)
FROM Clienti, Depositi
WHERE CF = CFCliente AND Saldo > 1000
GROUP BY CF, Nome
HAVING count(*) > 1;
```

- 5) (opzionale) Si consideri il seguente schema, che rappresenta informazioni relative alle partenze da un aeroporto; un volo è identificato dalla coppia Compagnia,Codice:

Partenze(Compagnia, Codice, Ora, Destinazione)

Si esprimano le seguenti affermazioni come dipendenze funzionali, *se possibile*:

- a) Non è possibile che due voli della stessa compagnia partano alla stessa ora
- b) Due voli che partono alla stessa ora hanno due destinazioni diverse
- c) Due voli della stessa compagnia possono avere la stessa destinazione

1) Si consideri lo schema relazionale $R(A,B,C,D,E,F)$ con le seguenti DF:

$C \rightarrow E, \quad AF \rightarrow B, \quad D \rightarrow F$

- Si trovino le chiavi di R .
- Dire se lo schema è in 3FN o in FNBC.
- Se lo schema non è in FNBC lo si normalizzi e si dica se la decomposizione preserva le dipendenze.
- Si aggiunga alle DF su R la dipendenza $C \rightarrow D$. Se C non è chiave si aggiunga qualche altra dipendenza (con un solo attributo a sinistra e a destra) in modo che C diventi chiave.

2) Si considerino i seguenti insiemi di FD sullo schema $R(A,B,C,D)$

$F1 = \{B \rightarrow A, A \rightarrow C\}$

$F2 = \{B \rightarrow A, B \rightarrow C\}$

$F3 = \{B \rightarrow A, AB \rightarrow C\}$

- Si esprima (in modo formale) che cosa significa dire che due insiemi di FD sono equivalenti, e si dica come si procede in pratica per verificare tale equivalenza
- $F1$ e $F2$ sono equivalenti?
- $F2$ e $F3$ sono equivalenti?
- $F1$ e $F3$ sono equivalenti?

3) Si consideri lo schema relazionale:

$Clienti(CF, Nome, Telefono)$, con chiavi CF e $(Nome, Telefono)$

$Depositi(Numero, CF_{Cliente}, Tipo, Saldo)$, con chiave $Numero$; il tipo del deposito può essere 'R' (a risparmio) o 'C' (in conto corrente).

- Definire in SQL un'interrogazione che ritorna una relazione vuota se e solo se in $Clienti$ vale la dipendenza $Nome \rightarrow Telefono$.
- Definire in SQL un'interrogazione per trovare il nome dei clienti che hanno tutti i depositi in conto corrente con saldo negativo (ovvero, non ci interessa lo stato dei loro depositi a risparmio)
- (opzionale) Definire in SQL un'interrogazione per trovare il nome dei clienti che hanno tutti i depositi in conto corrente e con saldo negativo.
- Si considerino le seguenti interrogazioni per trovare il nome e il telefono dei clienti che hanno un deposito a risparmio con saldo non compreso tra -100 e 100:

d1) SELECT Nome, Telefono
FROM Clienti, Depositi
WHERE CF = CFCliente AND Tipo = 'R' AND (Saldo < -100 OR Saldo > 100);

d2) SELECT Nome, Telefono
FROM Clienti
WHERE CF IN (SELECT CFCliente
FROM Depositi
WHERE Tipo = 'R' AND (Saldo < -100 OR Saldo > 100));

d3) SELECT Nome, Telefono
FROM Clienti, Depositi
WHERE CF = CFCliente AND Tipo = 'R' AND Saldo < -100
UNION
SELECT Nome, Telefono
FROM Clienti, Depositi
WHERE CF = CFClienti AND Tipo = 'R' AND Saldo > 100;

Due interrogazioni sono equivalenti se producono lo stesso risultato, ignorando l'ordine. Dire se le tre interrogazioni sono equivalenti, oppure quali di loro lo sono.

- 4) Per l'interrogazione seguente, si fornisca un piano di accesso senza uso di indici ed uno che usa due indici a vostra scelta:

```
SELECT Nome, sum(saldo)
FROM Clienti, Depositi
WHERE CF = CFCliente AND Saldo > 1000
GROUP BY CF, Nome
HAVING count(*) > 1;
```

- 5) (opzionale) Si consideri il seguente schema, che rappresenta informazioni relative alle partenze da un aeroporto; un volo è identificato dalla coppia Compagnia,Codice:

Partenze(Compagnia, Codice, Ora, Destinazione)

Si esprimano le seguenti affermazioni come dipendenze funzionali, *se possibile*:

- a) Non è possibile che due voli che partono alla stessa ora abbiano la stessa destinazione
- b) Due voli con la stessa destinazione possono essere della stessa compagnia
- c) Se due voli sono della stessa compagnia, allora partono ad ore diverse

Basi di Dati, Dicembre 2002, secondo compitino A, soluzioni, Versione 0.2

1) Si consideri lo schema relazionale $R(A,B,C,D,E,F)$ con le seguenti DF:

$$A \rightarrow C, \quad DE \rightarrow F, \quad B \rightarrow D$$

- a) Si trovino le chiavi di R .
- b) Dire se lo schema è in 3FN o in FNBC.
- c) Se lo schema non è in FNBC lo si normalizzi e si dica se la decomposizione preserva le dipendenze.
- d) Si aggiunga alle DF su R la dipendenza $A \rightarrow B$. Se A non è chiave si aggiunga qualche altra dipendenza (con un solo attributo a sinistra e a destra) in modo che A diventi chiave.

a) Gli attributi A,B e E non appaiono a destra delle DF, quindi devono far parte di ogni chiave. ABE è chiave ed è l'unica

b) Non è in 3FN (per $A \rightarrow C$) e quindi nemmeno in FNBC.

c) $R(ABCDEF) \{A \rightarrow C, DE \rightarrow F, B \rightarrow D\} \rightarrow$

$$R1(AC) \{A \rightarrow C\}, R2(ABDEF) \{DE \rightarrow F, B \rightarrow D\} \rightarrow$$

$$R1(AC) \{A \rightarrow C\}, R3(DEF) \{DE \rightarrow F\}, R4(DEAB) \{B \rightarrow D\} \rightarrow$$

$$R1(AC) \{A \rightarrow C\}, R3(DEF) \{DE \rightarrow F\}, R5(BD) \{B \rightarrow D\}, R6(BAE) \{ \}$$

Preserva le dipendenze

Oppure:

$$R(ABCDEF) \{A \rightarrow C, DE \rightarrow F, B \rightarrow D\} \rightarrow$$

$$R1(AC) \{A \rightarrow C\}, R2(ABDEF) \{DE \rightarrow F, B \rightarrow D\} \rightarrow$$

$$R1(AC) \{A \rightarrow C\}, R3(BD) \{B \rightarrow D\}, R4(ABEF) \{BE \rightarrow F\} \rightarrow$$

$$R1(AC) \{A \rightarrow C\}, R3(BD) \{B \rightarrow D\}, R4(BEF) \{BE \rightarrow F\}, R5(ABE)$$

Non preserva le dipendenze ($DE \rightarrow F$ è andata perduta)

d) Con $A \rightarrow B$, A non è chiave e per diventarlo occorre almeno una delle seguenti dipendenze:
 $A \rightarrow E, BC \rightarrow E, CD \rightarrow E, B \rightarrow E, C \rightarrow E, D \rightarrow E$.

2) Si considerino i seguenti insiemi di FD sullo schema $R(A,B,C,D)$

$$F1 = \{A \rightarrow B, B \rightarrow C\}$$

$$F2 = \{A \rightarrow B, A \rightarrow C\}$$

$$F3 = \{A \rightarrow B, AB \rightarrow C\}$$

a) Si esprima (in modo formale) che cosa significa dire che due insiemi di FD sono equivalenti, e si dica come si procede in pratica per verificare tale equivalenza

b) $F1$ e $F2$ sono equivalenti?

d) $F2$ e $F3$ sono equivalenti?

d) $F1$ e $F3$ sono equivalenti?

a) $F \sim G$ se: $F \subseteq G^+$ e $G \subseteq F^+$, ovvero se: $F^+ = G^+$, ovvero se: $f \in F \Rightarrow f \in G^+$ e $g \in G \Rightarrow g \in F^+$. In pratica, si verifica se $\forall X \rightarrow Y \in F. Y \subseteq X^+_G$, e $\forall X \rightarrow Y \in G. Y \subseteq X^+_F$,

- b) $F1$ e $F2$ non sono equivalenti: $B \rightarrow C$ non è in $F2^+$
- c) $F2$ e $F3$ sono equivalenti perché, in particolare, $F2 \models AB \rightarrow C$ ed $F3 \models A \rightarrow C$
- d) $F1$ e $F3$ non sono equivalenti, $B \rightarrow C$ non è in $F3^+$

3) Si consideri lo schema relazionale:

Clienti(CF, Nome, Telefono), con chiavi CF e (Nome, Telefono)

Depositi(Numero, CFCliente, Tipo, Saldo), con chiave Numero; il tipo del deposito può essere 'R' (a risparmio) o 'C' (in conto corrente).

- a) Definire in SQL un'interrogazione che ritorna una relazione vuota se e solo se in *Clienti* vale la dipendenza $\text{Nome} \rightarrow \text{Telefono}$.
- b) Definire in SQL un'interrogazione per trovare il nome dei clienti che hanno tutti i depositi in conto corrente con saldo negativo (ovvero, non ci interessa lo stato dei loro depositi a risparmio)
- c) (opzionale) Definire in SQL un'interrogazione per trovare il nome dei clienti che hanno tutti i depositi in conto corrente e con saldo negativo.
- d) Si considerino le seguenti interrogazioni per trovare il nome e il telefono dei clienti che hanno un deposito a risparmio o in conto corrente con saldo negativo:

d1) *SELECT* Nome, Telefono
 FROM Clienti, Depositi
 WHERE CF = CFCliente *AND* Saldo < 0 *AND* (Tipo = 'R' *OR* Tipo = 'C');

d2) *SELECT* Nome, Telefono
 FROM Clienti
 WHERE CF *IN* (*SELECT* CFCliente
 FROM Depositi
 WHERE Saldo < 0 *AND* (Tipo = 'R' *OR* Tipo = 'C'));

d3) *SELECT* Nome, Telefono
 FROM Clienti, Depositi
 WHERE CF = CFCliente *AND* Saldo < 0 *AND* Tipo = 'R'
 UNION
 SELECT Nome, Telefono
 FROM Clienti, Depositi
 WHERE CF = CFClienti *AND* Saldo < 0 *AND* Tipo = 'C';

Due interrogazioni sono equivalenti se producono lo stesso risultato, ignorando l'ordine. Dire se le tre interrogazioni sono equivalenti, oppure quali di loro lo sono.

3a) Se $\text{Nome} \rightarrow \text{Telefono}$ allora Nome è una chiave, che posso verificare come segue:

```
SELECT Nome, COUNT(*)
FROM Clienti
GROUP BY Nome
HAVING COUNT(*) > 1;
```

Oppure posso usare direttamente la definizione di dipendenza funzionale:

```
SELECT c1.Nome, c1.Telefono, c2.Telefono
FROM Clienti c1, Clienti c2
WHERE c1.Nome = c2.Nome and not (c1.Telefono = c2.Telefono)
```

O anche:

```
SELECT Nome, COUNT(DISTINCT Telefono)
FROM Clienti
GROUP BY Nome
HAVING COUNT(DISTINCT Telefono) > 1;
```

```
3b) SELECT c.Nome
FROM Clienti c
WHERE NOT EXIST (SELECT * FROM Depositi d
                  WHERE c.CF = d.CFCliente AND d.Tipo = 'C' AND d.Saldo >=0);
```

```
3c) SELECT c.Nome
FROM Clienti c
WHERE NOT EXIST (SELECT * FROM Depositi d
                  WHERE c.CF = d.CFCliente AND NOT( d.Tipo = 'C' AND d.Saldo < 0));
```

ATTENZIONE: NOT(Tipo = 'C' AND Saldo < 0) ⇔ (Tipo <> 'C' **OR** Saldo >= 0)

Le due interrogazioni ritornano anche i clienti senza depositi. Per limitarsi ai clienti con qualche deposito di Tipo 'C', vanno modificate, ad esempio, come segue:

```
3b') SELECT c.Nome
FROM Clienti c
WHERE NOT EXIST (SELECT * FROM Depositi d
                  WHERE c.CF = d.CFCliente AND d.Tipo = 'C' AND d.Saldo >=0)
AND EXIST (SELECT * FROM Depositi d
            WHERE c.CF = d.CFCliente AND d.Tipo = 'C');
```

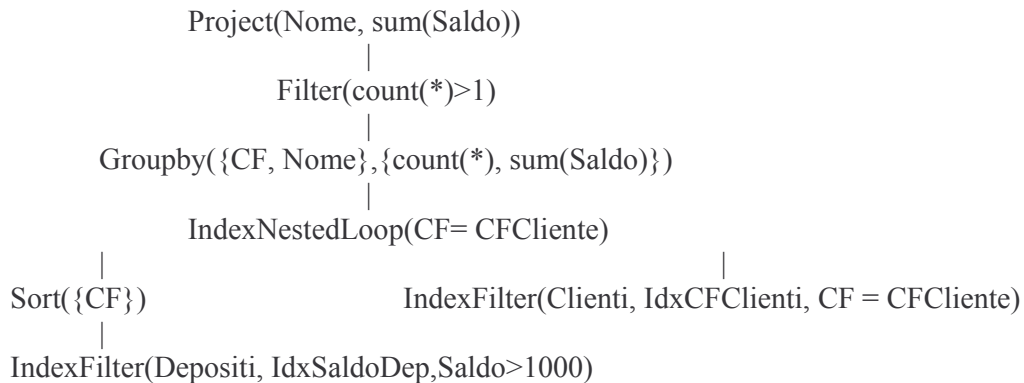
```
3b'') SELECT DISTINCT c.Nome
FROM Clienti c, Depositi d2
WHERE NOT EXIST (SELECT * FROM Depositi d
                  WHERE c.CF = d.CFCliente AND d.Tipo = 'C' AND d.Saldo >=0)
AND c.CF = d2.CFCliente AND d2.Tipo = 'C'
```

```
3b''') SELECT c.Nome
FROM Clienti c, Depositi d
WHERE c.CF = d.CFCliente AND d.Tipo = 'C'
GROUP BY c.CF, c.Nome
HAVING MAX(d.Saldo) < 0
```

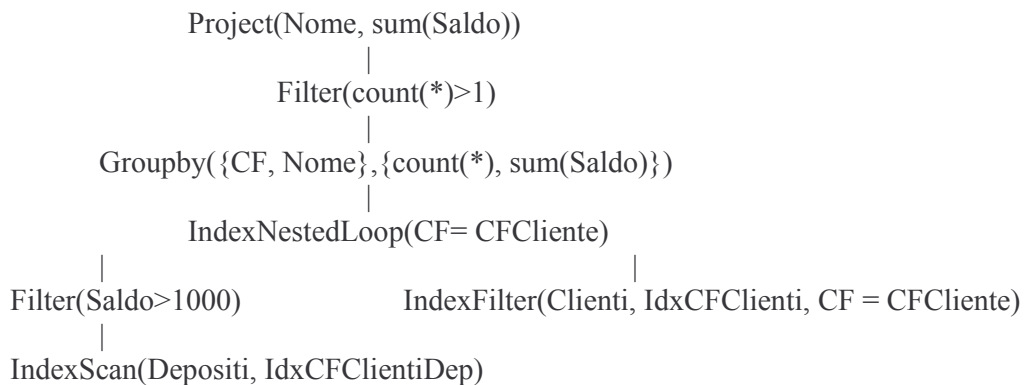
```

    Project(Nome, sum(Saldo))
      |
    Filter(count(*)>1)
      |
    Groupby({CF, Nome},{count(*), sum(Saldo)})
      |
    SortMerge(CF= CFCliente)
      |               |
      |               | Sort({CFCliente})
      |               |   |
      |               |   | Filter(Saldo>1000)
      |               |   |   |
      |               |   |   |
    Sort({CF})       |   |   |
      |               |   |   |
    TableScan(Clienti) TableScan(Depositi)
  
```

4b) Con indice su Depositi.Saldo e su Clienti.CF:



Oppure, con indice su Depositi.CFCliente e su Clienti.CF, posso usare l'indice sui depositi per evitare l'operazione di ordinamento:



5) (opzionale) Si consideri il seguente schema, che rappresenta informazioni relative alle partenze da un aeroporto; un volo è identificato dalla coppia Compagnia,Codice:

Partenze(Compagnia, Codice, Ora, Destinazione)

Si esprimano le seguenti affermazioni come dipendenze funzionali, se possibile:

- a) Non è possibile che due voli della stessa compagnia partano alla stessa ora
- b) Due voli che partono alla stessa ora hanno due destinazioni diverse
- c) Due voli della stessa compagnia possono avere la stessa destinazione

Suggerimento: tenete conto che un'unica dipendenza funzionale (ad es. Compagnia,Codice → Destinazione) si può esprimere almeno in tre modi:

(Diretta) Sinistra uguale implica destra uguale:

Cod uguale, Comp uguale ⇒ Dest uguale

“fissata la compagnia ed il codice, la destinazione è unica”

(Inversa) Destra diverso implica sinistra non completamente uguale:

Cod uguale, Dest **diversa** $\Rightarrow$ Comp **diversa**)

“quando il codice è uguale ma la destinazione è diversa, i voli hanno compagnie diverse”

(Per assurdo) È impossibile avere sinistra uguale e destra diverso:

Not(Cod uguale $\wedge$ Comp uguale $\wedge$ Dest **diversa**)

“non è possibile avere lo stesso codice e la stessa compagnia, ma destinazione diversa”

La forma inversa si trasforma in quella diretta usando l'equivalenza:

$$(A = \wedge B_{\neq}) \Rightarrow C_{\neq} \Leftrightarrow (A = \wedge C =) \Rightarrow B = \quad (2 \Leftrightarrow 1)$$

La forma per assurdo si trasforma in quella diretta usando l'equivalenza:

$$\text{Not}(A = \wedge B = \wedge C_{\neq}) \Leftrightarrow (A = \wedge B =) \Rightarrow C = \quad (3 \Leftrightarrow 1)$$

a) **Not**(Volo **diverso** $\wedge$ Compagnia uguale $\wedge$ Ora uguale) (per assurdo, uso $(3 \Leftrightarrow 1)$)

$\Leftrightarrow$ Compagnia uguale $\wedge$ Ora uguale $\Rightarrow$ Volo uguale

$\Leftrightarrow$ Compagnia, Ora $\rightarrow$ Codice

b) Volo **diverso** $\wedge$ Ora uguale $\Rightarrow$ Destinazione **diversa** (inversa, uso $(2 \Leftrightarrow 1)$)

$\Leftrightarrow$ Destinazione uguale $\wedge$ Ora uguale $\Rightarrow$ Volo uguale

$\Leftrightarrow$ Destinazione, Ora $\rightarrow$ Compagnia, Codice

c) Non è un vincolo (“*possono avere...*”)

Basi di Dati, Dicembre 2002, secondo compito B, soluzioni, Versione 0.2

- 5) (opzionale) Si consideri il seguente schema, che rappresenta informazioni relative alle partenze da un aeroporto; un volo è identificato dalla coppia Compagnia,Codice:

Partenze(Compagnia, Codice, Ora, Destinazione)

Si esprimano le seguenti affermazioni come dipendenze funzionali, se possibile:

- a) Non è possibile che due voli che partono alla stessa ora abbiano la stessa destinazione*
- b) Due voli con la stessa destinazione possono essere della stessa compagnia*
- c) Se due voli sono della stessa compagnia, allora partono ad ore diverse*

- a) **Not**(Volo **diverso** $\wedge$ Ora uguale $\wedge$ Destinazione uguale) (per assurdo, uso ($3 \Leftrightarrow 1$))
 - $\Leftrightarrow$ Destinazione uguale $\wedge$ Ora uguale $\Rightarrow$ Volo uguale
 - $\Leftrightarrow$ Destinazione, Ora $\rightarrow$ Codice
- b) Non è un vincolo (“*possono essere...*”)
- c) Volo **diverso** $\wedge$ Compagnia uguale $\Rightarrow$ Ora **diversa** (inversa, uso ($2 \Leftrightarrow 1$))
 - $\Leftrightarrow$ Compagnia uguale $\wedge$ Ora uguale $\Rightarrow$ Volo uguale
 - $\Leftrightarrow$ Compagnia, Ora $\rightarrow$ Codice