

Cognome e nome: _____

Matricola: _____

Esame di SQL (rispondere su fogli separati rispetto alla parte di teoria)

Le seguenti relazioni definiscono una base di dati “Pizzeria” per gestire gli ordini di una pizzeria. Gli attributi sottolineati sono le chiavi primarie delle relazioni.

ORDINE(Pizza, Tavolo, Data, Quantità, Note)

PIZZA(NomePizza, Prezzo, Tipo)

CONTIENE(Pizza, Ingrediente)

INGREDIENTE(NomeIngrediente, Vegetariano, SenzaGlutine)

Vincoli di integrità referenziale: ORDINE(Pizza) referencia PIZZA(NomePizza), CONTIENE(Pizza) referencia PIZZA(NomePizza), CONTIENE(Ingrediente) referencia INGREDIENTE(NomeIngrediente).

“Tipo” può assumere i valori "Normale" e “Gourmet”. “Vegetariano” e “SenzaGlutine” sono booleani.

I rimanenti attributi sono autoesplicativi. Gli attributi sono tutti NOT NULL.

Con riferimento alla base di dati "Pizzeria" esprimere in SQL le seguenti interrogazioni.

Domanda 1 (bassa complessità).

Trovare gli ordini (tavolo e data) composti da almeno due pizze con l’ananas o con le pere.

Soluzione 1.

```
SELECT DISTINCT Data, Tavolo
FROM ordine O JOIN pizza P ON (O.Pizza=P.NomePizza)
      JOIN contiene C ON (P.NomePizza=C.Pizza)
      JOIN ingrediente I ON (C.Ingrediente=I.NomeIngrediente)
WHERE O.Quantità>=2 AND (I.NomeIngrediente='Ananas' OR I.NomeIngrediente='Pera');
```

Domanda 2 (media complessità).

Mostrare le date in cui dei tavoli hanno speso più di 50 euro in pizze gourmet.

Soluzione 2.

```
SELECT DISTINCT O.Data
FROM ordine O JOIN pizza P ON (O.Pizza=P.NomePizza)
WHERE P.Tipo='Gourmet'
GROUP BY O.Data, O.Tavolo
HAVING SUM(P.Prezzo*O.Quantità)>50;
```

Domanda 3 (alta complessità).

Trovare il conto medio per l’anno 2018 degli ordini composti solo da pizze vegetariane (cioè pizze in cui tutti gli ingredienti sono vegetariani).

Soluzione 3.

```
WITH ContiVegetariani AS (
  SELECT O.Data, O.Tavolo, SUM(P.Prezzo*O.Quantità) Conto
  FROM ordine O JOIN pizza P ON (O.Pizza=P.NomePizza)
  WHERE (O.Data, O.Tavolo) NOT IN (
    SELECT O1.Data, O1.Tavolo
    FROM ordine O1 JOIN pizza P1 ON (O1.Pizza=P1.NomePizza)
      JOIN contiene C ON (P1.NomePizza=C.Pizza)
      JOIN ingrediente I ON (C.Ingrediente=I.NomeIngrediente)
    WHERE I.Vegetariano=False)
  GROUP BY O.Data, O.Tavolo)
SELECT AVG(C.Conto)
FROM ContiVegetariani C
WHERE C.Data>='01-01-2018' AND C.Data<'01-01-2019';
```

Esame di Teoria (rispondere su fogli separati rispetto alla parte di SQL)

Domanda 1.

Con riferimento alla base di dati “Pizzeria”:

A. Esprimere in Algebra Relazionale l'interrogazione

Elencare gli ingredienti che sono presenti in almeno due pizze.

B. Esprimere, nel calcolo dei predicati su tuple con dichiarazione di range, la seguente domanda:

Elencare le pizze (si richiede nome e prezzo) di tipo gourmet che hanno solo ingredienti senza glutine.

Soluzione 1.

A. Una possibile soluzione (in algebra relazionale) è la seguente:

$$\pi_{C1.Ingrediente} \left(\left(\rho_{C1 \leftarrow Contiene}(Contiene) \bowtie_{\theta_1} \rho_{C2 \leftarrow Contiene}(Contiene) \right) \right)$$

dove $\theta_1 := C1.Ingrediente = C2.Ingrediente \wedge C1.Pizza \neq C2.Pizza$

B. Una possibile soluzione è la seguente:

$$\{p.NomePizza, p.Prezzo \mid p(PIZZA) \mid p.Tipo='Gourmet' \wedge \forall c(CONTIENE)(c.Pizza=p.NomePizza \Rightarrow \exists i (INGREDIENTE)(c.Ingrediente=i.NomeIngrediente \wedge i.SenzaGlutine=True))\}$$

Domanda 2.

A. Definire il concetto di dipendenza funzionale.

Si considerino $R(A, B, C, D, E)$ e $F' = \{B \rightarrow D, D \rightarrow E\}$.

B. Le dipendenze funzionali in F' non giustificano A come chiave di R : aggiungere a F' delle dipendenze funzionali in modo che A sia chiave di R .

C. Considerando il nuovo insieme di dipendenze funzionali ottenuto nel punto precedente, dire se R è in 3FN e se non lo è decomporla in relazioni in 3FN esplicitando tutti i passaggi. Lo schema ottenuto è BCNF?

Soluzione 2.

A. Vedere libro di testo o slide.

B. Se aggiungiamo la d.f. $A \rightarrow BC$, la chiusura di A comprende tutti gli attributi di R quindi è una superchiave ed essendo composta da un solo attributo è minimale ed è una chiave.

C. $R(A, B, C, D, E)$ e $F = \{A \rightarrow BC, B \rightarrow D, D \rightarrow E\}$ non è 3NF perché, ad esempio, nella d.f. $B \rightarrow D$ B non è una superchiave, D non è un attributo primo e la d.f. è non banale.

Calcolo l'insieme di copertura minimale.

Trasformo le d.f. mettendo un solo attributo a destra: $F'' = \{A \rightarrow B, A \rightarrow C, B \rightarrow D, D \rightarrow E\}$.

Non ci sono attributi estranei perché ogni d.f. ha un solo attributo a sinistra.

Ogni d.f. ha a destra un attributo differente, quindi ogni d.f. è necessaria per avere quell'attributo nella chiusura e non ci sono d.f. ridondanti. F'' , quindi, è una copertura minimale.

Decompongo R in $R1(\underline{A}, B, C)$, $R2(\underline{B}, D)$, $R3(\underline{D}, E)$. Non ci sono relazioni sottoinsieme di altre e $R1$ comprende la chiave di R quindi ho terminato la normalizzazione in 3NF.

Lo schema ottenuto è BCNF perché ogni d.f. è rappresentata tramite la chiave primaria delle relazioni.

Domanda 3.

- A. Spiegare in modo succinto ma preciso le principali differenze tra B-tree e B+-tree e quali vantaggi apportano i B+-tree rispetto ai B-tree nella gestione degli indici.
- B. Disegnare un B+-tree con $m=4$ contenente come chiavi i numeri da 1 a 20. Non occorre simulare gli inserimenti (le chiavi potrebbero essere state inserite in un ordine qualunque) ma soltanto rappresentare un possibile B+-tree contenente tutte le chiavi.

Soluzione 3.

- A. Vedere libro di testo o slide.
- B. È valido qualsiasi B+-tree che rispetta le proprietà dei B-tree e dei B+-tree riportate sul libro di testo e sulle slide. In particolare, tutti i nodi (eccetto la radice) devono avere due o tre chiavi, ogni chiave k dei nodi intermedi deve essere presente nella foglia più a destra del sottoramo sinistro della chiave k stessa. Se un nodo ha j chiavi, deve avere esattamente $j+1$ figli. Ogni nodo foglia ha un puntatore al nodo foglia adiacente. Inoltre, devono essere rispettate le proprietà base dei B-tree. In particolare, le foglie devono essere tutte al medesimo livello.

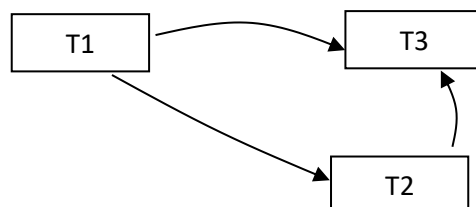
Domanda 4.

- A. Enunciare il criterio di serializzabilità.
- Considerare la seguente storia interfogliata $S = r1(x), r2(y), w2(x), r3(y), w3(x)$.
- B. La storia S è conflict-serializzabile? Giustificare la risposta.
- C. La storia S è view-serializzabile? Giustificare la risposta.
- D. (Caso 1) Nel caso la storia S sia conflict-serializzabile, è possibile aggiungere a S un'azione che la renda non conflict-serializzabile? Se sì, quale?
- (Caso 2) Nel caso la storia S non sia conflict-serializzabile, è possibile togliere da S un'azione che la renda conflict-serializzabile? Se sì, quale?
- N.B.: ovviamente, solo uno dei due casi dell'esercizio D è da svolgere.**

Soluzione 4.

- A. Vedere libro di testo o slide.

B e C. Il grafo dei conflitti è il seguente:



Poiché il grafo è aciclico, la storia è conflict-serializzabile e view-serializzabile.

- D.** Aggiungendo al termine della storia ad esempio l'azione $w1(x)$ si aggiungono al grafo dei conflitti gli archi $T2 \rightarrow T1$ e $T3 \rightarrow T1$ e si formano due cicli rendendo così la storia non conflict-serializzabile.