

PROVA COMPLETA - Basi di Dati e Lab

24/06/2019

Corso di Laurea in Ingegneria Informatica

Prof. Sonia Bergamaschi

Esercizio 1 (punti 5)

È necessario totalizzare almeno 3 punti in questo esercizio affinché il resto della prova venga valutato

Operatori di JOIN in SQL.

Esercizio 2 (punti 11)

Si vuole creare un sistema informativo per la gestione di una catena di pizzerie.

La catena ha più sedi, ogni sede è identificata da un codice univoco ed ha un indirizzo, un numero telefonico e una capienza. In ogni sede lavorano dei dipendenti. Per ogni dipendente sono riportati i dati anagrafici, inoltre è riportata la data di assunzione ed eventuale data di licenziamento. I dipendenti possono essere pizzaioli, camerieri o fattorini. Per ogni pizzaiolo viene indicato se possiede o meno un diploma di specializzazione e per ogni fattorino la tipologia di mezzo di trasporto usato. Una sede si rivolge ad un fornitore (identificato da un codice univoco e di cui si riporta il numero di telefono) per acquistare una certa quantità di un ingrediente per le sue pizze. Per ogni ingrediente viene anche allegata una descrizione.

La catena ha diversi clienti, per ciascuno dei quali vengono riportati gli usuali dati anagrafici.

Ogni sede deve gestire diverse prenotazioni da parte dei clienti. Una prenotazione può essere di due tipi: “tavolo” o “consegna a domicilio”. Per ogni prenotazione di tipo “tavolo” il cliente può effettuare diversi ordini. Ogni ordine dal tavolo viene raccolto da uno dei camerieri ed è identificato da un numero progressivo all’interno della prenotazione. Per ogni ordine viene riportato l’elenco dei prodotti presenti nel menù (uguale per tutte le sedi) che il cliente vuole ordinare (fino ad un massimo di 20). Un prodotto del menù può essere “cibo” o “bibita” e per ciascuno di questi vengono riportati il prezzo ed una descrizione. Per i prodotti di tipo “cibo” devono essere indicati anche gli ingredienti che li compongono (fino ad un massimo di 5). Per una prenotazione di tipo “consegna a domicilio” è possibile fare un solo ordine in modo analogo al tavolo fino ad un massimo di 30 prodotti. Inoltre viene indicato l’orario di consegna desiderato e il fattorino incaricato della consegna. Ciascun fattorino può effettuare al massimo 15 consegne al giorno. Se un fattorino consegna in ritardo più di 5 ordini in una settimana, viene etichettato come “inefficiente”.

Un cliente può rilasciare una o più recensioni per una certa sede indicando una valutazione da 1 a 5 ed eventualmente rilasciando un commento. Ogni recensione è identificata dalla sede e da un numero progressivo.

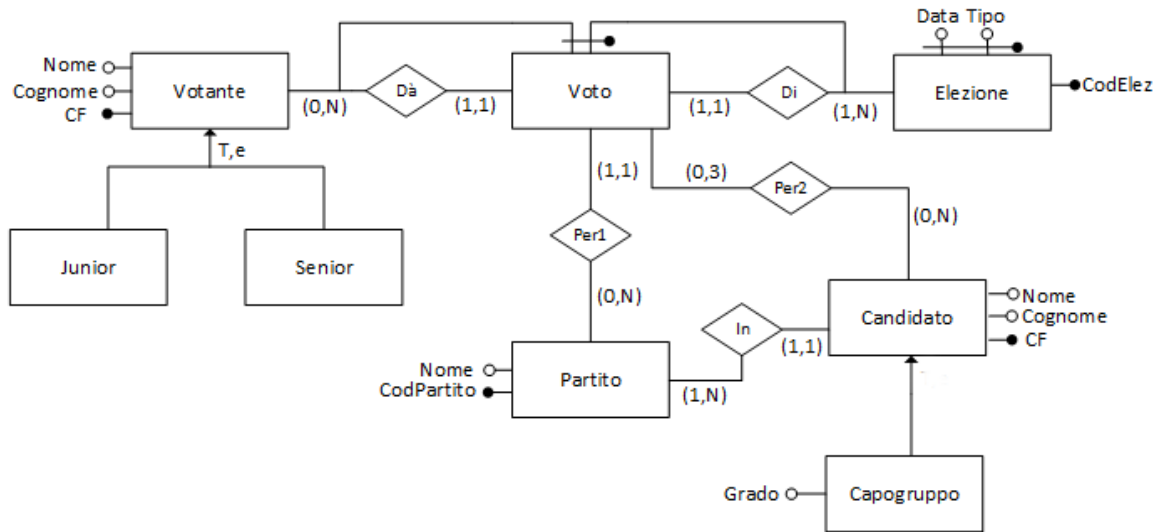
Viene richiesto di:

- a) Progettare lo schema E/R, evidenziando gli eventuali vincoli non esprimibili nel modello. **(9 punti)**
- b) Modificare lo schema precedente aggiungendo il seguente vincolo: **(2 punti)**

Nello stesso mese un cliente può rilasciare al massimo 3 recensioni totali.

Esercizio 3 (punti 7)

Sia dato il seguente schema E/R



A partire dallo schema e dalle informazioni contenute nella tabella dei volumi dei dati viene richiesto di:

1. Produrre lo schema di navigazione ed il costo della seguente operazione: **(3 punti)**
Inserimento di un nuovo partito e di tutti i suoi candidati. Inoltre devono essere inseriti i voti relativi all'ultima elezione del partito inserito (si suppongano già presenti i dati di "elezione" e "votante").

Tabella dei volumi dei dati

CONCETTO	TIPO	VOL.
Votante	E	10000
Voto	E	60000
Elezione	E	10
Partito	E	8
Candidato	E	120
Per2	R	120000

Il 30% dei votanti sono Junior, il 70% Senior. Inoltre il 10% dei candidati è anche Capogruppo.

2. Produrre lo schema logico relazionale evidenziando eventuali vincoli non esprimibili. **(3 punti)**

Risolvere l'esercizio indicando sullo schema la navigazione e completando una tabella dei costi del tipo:

CONCETTO	ACCESSI	TIPO	DESCRIZIONE / MOTIVAZIONE DELL'OPERAZIONE
Autore (ESEMPIO)	1 (ESEMPIO)	S (ESEMPIO)	Si scrive il nome dell'autore (ESEMPIO)

Esercizio 4 (punti 11)

Sia dato il seguente schema relazionale relativo alla gestione dell'organizzazione di concerti.

Artista(nome_artista, tipo, anni_attività)

Sede(id_sede, città, num_posti)

Concerto(id_concerto, nome_artista, id_sede, data, prezzo_biglietto)

AK: nome_artista, id_sede, data

FK: nome_artista REFERENCES Artista

FK: id_sede REFERENCE Sede

Persona(CF, nome, cognome, nazionalità)

Partecipa(CF, id_concerto)

FK: CF REFERENCES Persona

FK: id_concerto REFERENCES Concerto

Si richiede di scrivere in algebra relazionale e in SQL le seguenti interrogazioni:

- a) Selezionare nome e cognome delle persone che hanno partecipato solo a concerti di tipo "Rock". **(punti 2+2)**
- b) Selezionare le persone che hanno partecipato a tutti i concerti dell'artista "Vasco Rossi". **(punti 2+2)**

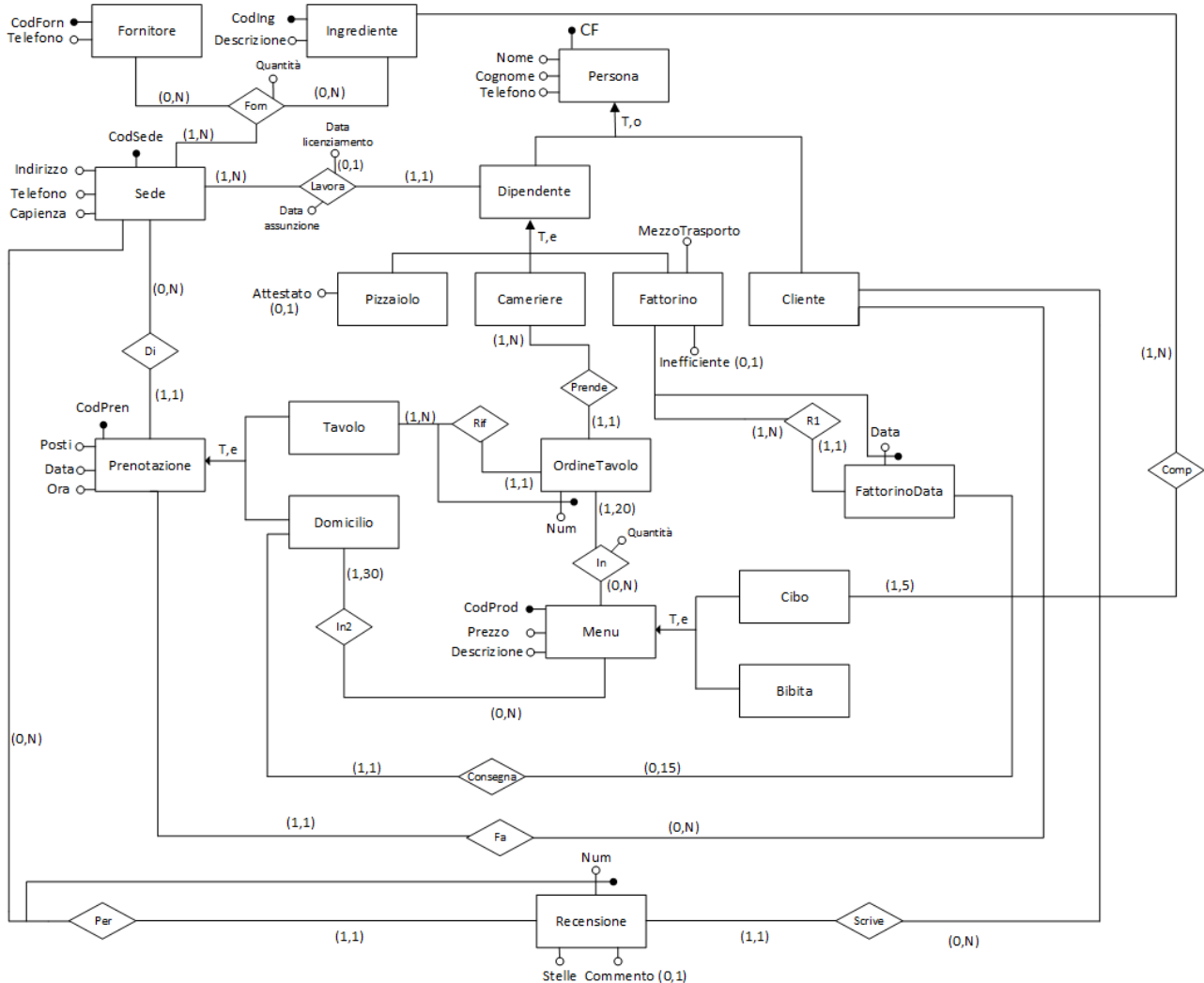
Si richiede di scrivere in SQL la seguente interrogazione (può essere risolta anche con più viste):

- c) Creare una vista che mostri per ogni artista nome dell'artista, l'incasso totale di tutti i suoi concerti, il numero di persone e il numero di nazionalità distinte dei partecipanti ai suoi concerti. Devono essere mostrati solo gli artisti con un incasso superiore a 1000000€. **(punti 3)**

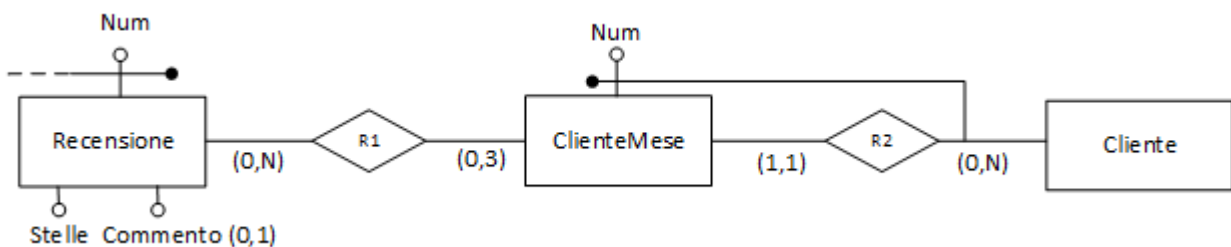
SOLUZIONE

Esercizio 2

Vincoli non esprimibili: “Se un fattorino consegna in ritardo più di 5 ordini in una settimana, viene etichettato come “inefficiente””.

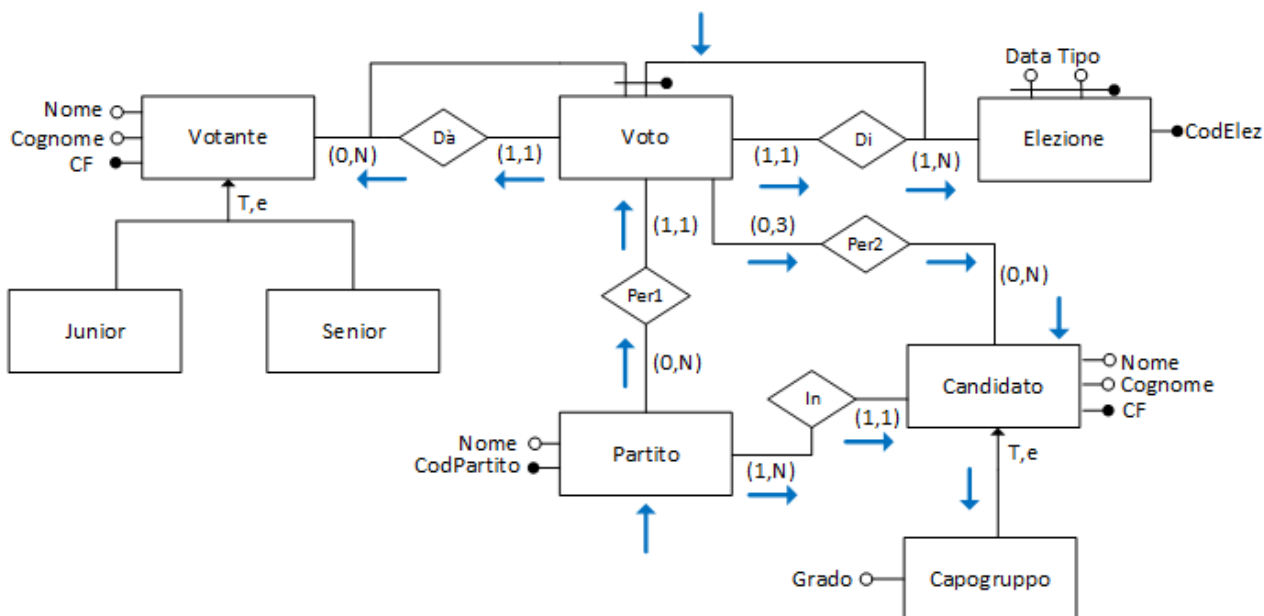


Vincolo aggiuntivo



Esercizio 3

Schema di navigazione



Calcolo costo operazione

CONCETTO	ACCESSI	TIPO	DESCRIZIONE / MOTIVAZIONE DELL'OPERAZIONE
Partito	1	L	Verifico che il partito da inserire non sia già presente.
Partito	1	S	Inserisco il partito.
In	15	S	Ci sono mediamente $120/8=15$ candidati per partito.
Candidato	15	S	Vedi riga sopra.
Capogruppo	1.5	S	Il 10% dei candidati è capogruppo quindi inserisco $15*0.1=1.5$ elementi.
Per1	750	S	Devo inserire i voti di una sola elezione. Ci sono 60000 voti e 10 elezioni quindi per ogni elezione ci sono $60000/10=6000$ voti. Non avendo informazioni sull'esito delle votazione supponiamo che i voti siano equamente distribuiti tra i partiti, quindi per un partito ci sono $6000/8=750$ voti.
Voto	750	S	Vedi riga sopra.
Dà	750	S	Vedi riga sopra.
Di	750	S	Vedi riga sopra.
Per2	1500	S	Ci sono 60000 voti e 120000 preferenze quindi in media ci sono $120000/60000=2$ preferenze per voto. In totale bisogna inserire $750*2=1500$ preferenze.

Totale accessi in lettura: 1

Totale accessi in scrittura: $1+15+15+1.5+750+750+750+750+1500=4532.5$

Costo totale operazione: $1 + 4532.5*2 = 9065$

Schema logico (punti 3)

Partito(CodPartito, Nome)

Candidato(CF, Nome, Cognome, CodPartito)

FK: CodPartito **REFERENCES** Partito

Capogruppo(CF, Grado)

FK: CF **REFERENCES** Candidato

Elezione(CodElez, Data, Tipo)

AK: CodElez

Voto(CodElez, CF, CodPartito)

FK: CF **REFERENCES** Votante

FK: CodElez **REFERENCES** Elezione

FK: CodPartito **REFERENCES** Partito

Per2(CodElez, CFVot, CFCand)

FK: CF **REFERENCES** Votante

FK: CodElez **REFERENCES** Elezione

FK: CFCand **REFERENCES** Candidato

Votante(CF, Nome, Cognome, SelJuniorSenior)

Non esprimibile: preferenza voto a massimo 3 candidati.

Esercizio 4

Artista(nome_artista, tipo, anni_attività)

Sede(id_sede, città, num_posti)

Concerto(id_concerto, nome_artista, id_sede, data, prezzo_biglietto)

AK: nome_artista, id_sede, data

FK: nome_artista **REFERENCES** Artista

Persona(CF, nome, cognome, nazionalità)

Partecipa(CF, id_concerto)

FK: CF **REFERENCES** Persona

FK: id_concerto **REFERENCES** Concerto

- a) Selezionare nome e cognome delle persone che hanno partecipato solo a concerti di artisti di tipo “Rock”. (punti 2+2)

$$\begin{aligned} S1 &= \pi_{CF}(Persona \bowtie Partecipa \bowtie \sigma_{tipo='Rock'}(Artista)) \\ S2 &= \pi_{CF}(Persona \bowtie Partecipa \bowtie \sigma_{tipo \neq 'Rock'}(Artista)) \\ &\pi_{nome,cognome}(Persona \bowtie (S1 - S2)) \end{aligned}$$

```
SELECT DISTINCT P.nome, P.cognome
FROM Persona P
JOIN Partecipa Pa ON P.CF=Pa.CF
JOIN Concerto C ON Pa.id_concerto=C.id_concerto
JOIN Artista A ON Pa.nome_artista=A.nome_artista
WHERE A.tipo = 'Rock'
AND P.CF NOT IN (
    SELECT P1.CF
    FROM Persona P1
    JOIN Partecipa Pa1 ON P1.CF=Pa1.CF
    JOIN Artista A1 ON Pa1.nome_artista=A1.nome_artista
    WHERE A1.tipo <> 'Rock'
)
```

- b) Selezionare le persone che hanno partecipato a tutti i concerti dell'artista “Vasco Rossi”. (punti 2+2)

$$\begin{aligned} S1 &= \pi_{id_concerto}(\sigma_{nome_artista='Vasco Rossi'}(Concerto)) \\ &Persona \bowtie (Partecipa \div S1) \end{aligned}$$

```
SELECT *
FROM Persona P
WHERE NOT EXISTS (
    SELECT *
    FROM Concerto C
    WHERE C.nome_artista = 'Vasco Rossi'
    AND NOT EXISTS (
```

```

        SELECT *
        FROM Partecipa Pa
        WHERE C.id_concerto = Pa.id_concerto AND P.CF = Pa.CF
    )
)

```

- c) Creare una vista che mostri per ogni artista l'incasso totale di tutti i suoi concerti, il numero di persone e di nazionalità distinte che hanno partecipato ad almeno uno dei suoi concerti. Devono essere mostrati solo gli artisti con un incasso superiore a 1000000. **(punti 3)**

```

CREATE VIEW incasso_totale_artista AS
SELECT C.nome_artista, SUM(C.prezzo_biglietto) AS incasso_totale,
COUNT(DISTINCT nazionalità) AS num_nazionalità
FROM Concerto C
JOIN Partecipa Pa ON C.id_concerto = Pa.id_concerto
JOIN Persona P ON P.CF = Pa.CF
GROUP BY C.nome_artista
HAVING SUM(C.prezzo_biglietto) > 1000000

```