

Matricola: _____

Cognome: _____

Nome: _____

Basi di Dati - Laboratorio

IV Appello del 28 Febbraio 2024

Avvertenze: è severamente vietato consultare libri e appunti.

Durata 1h30min

DOMANDE

a) [3] Si consideri la seguente dichiarazione in PostgreSQL:

```
CREATE TABLE Docente (  
    codice_fiscale CHAR(16),  
    nome VARCHAR(32),  
    cognome VARCHAR(32),  
    data_nascita DATE,  
    luogo_nascita VARCHAR(24),  
    nazionalità VARCHAR(32)  
);
```

```
CREATE TABLE Corso (  
    codice CHAR(10),  
    titolo CHAR(40),  
    argomento CHAR(16),  
    descrizione TEXT,  
    docente CHAR(16),  
    data_erogazione DATE,  
    ora_inizio TIMESTAMP,  
    ora_fine TIMESTAMP  
);
```

Su completi (nello schema sopra) la dichiarazione in modo che permetta di rappresentare:

- (i) la chiave primaria sull'attributo `codice_fiscale` e `codice`,
- (ii) un vincolo di integrità referenziale (chiave esportata)
`Corso.docente` → `Docente.codice_fiscale`,
- (iii) un vincolo di dominio per l'attributo `argomento` $\in \{BD, ML, PR, RT\}$
- (iv) l'obbligatorietà degli attributi `nome` e `cognome` nella tabella `Docente`
- (v) valori non nulli per `nazionalità` con valore di default uguale a `Italiana`
- (vi) un vincolo `ora_inizio < ora_fine` per la tabella `Corso`

ESERCIZI

b) **Query**

Dato il seguente schema relazionale,

PAZIENTE (CodFisc, Nome, Cognome, Nazionalità, Età)

VISITA(Codice, DataPrenotazione, Data, Ora, Medico, Paziente)

MEDICO(Matricola, Nome, Cognome, Specialità, ProvinciaResidenza)

Vincoli di integrità:

VISITA.Paziente → PAZIENTE

VISITA.Medico → MEDICO

Formulare in SQL le seguenti interrogazioni:

i. [4] Trovare il numero di visite effettuate da medici residenti nella provincia di Verona per ciascuna specialità e raggruppate per anno. Riportare il numero di visite, la specialità e l'anno. (Usare l'operatore `extract(year from your_column)` per recuperare l'anno da una data)

ii. [5] Trovare tutti i medici che non hanno eseguito visite a Gennaio 2024, riportando il nome ed il cognome del medico, la sua specialità e la data dell'ultima visita.

iii. [6] Trovare i pazienti italiani che hanno effettuato nel 2023 un numero di visite inferiore al numero medio di visite effettuate da pazienti stranieri (nazionalità \neq Italia) nel 2023. Riportare: il nome ed il cognome del paziente, il numero di visite nel 2023 e la data dell'ultima visita.

iv. [3] Trovare tutte le visite prenotate il 28/02/2024 per una visita da effettuare in una data antecedente al giorno 25/04/2025. Indicare quali indici possono essere utilizzati per ottimizzare l'esecuzione della interrogazione e spiegarne il motivo.

g) Transazioni

[6] Descrivere la differenza tra l'anomalia di aggiornamento fantasma ed inserimento fantasma. Scegliere una delle due tipologie di anomalia e scrivere due transazioni SQL che possono produrre tale anomalia. Indicare il tipo di isolamento minimo che deve essere utilizzato per ciascuna transazione per evitare questa anomalia.

h) Python/Java

[6] Si consideri una base di dati in PostgreSQL contenente la seguente tabella

`Ricevuta(numero, data, ora, codice_fiscale, nome, cognome, importo)`

scrivere il codice di un programma Python o Java che richieda all'utente un importo ed una lettera e cerchi tutte le ricevute rilasciate a favore di un cliente con un cognome che inizia per tale lettera (confronto case insensitive) ed un importo superiore a quello specificato. Riportare il nome, il cognome, il numero della ricevuta, la data della ricevuta e l'importo. Se non esistono ricevute che soddisfano i criteri di ricerca, riportare la scritta "Nessuna ricevuta rilasciata ad un cliente con cognome che inizia con la lettera X ed un importo superiore a Y", dove X e Y vanno sostituiti con i valori indicati dall'utente.