

Cognome:

Nome:

Matricola:

Basi di dati – Corso di Laurea in Informatica Umanistica

Esercitazione pre-verifica del 19.03.2012

Parte 1 – modello relazionale, SQL (A)

Si consideri il seguente schema di base di dati letteraria:

TABLE Autori

{Codice integer **PRIMARY KEY**,
Cognome char(30) **NOT NULL**,
Nome char(20) **NOT NULL**,
Sesso char(1),
AnnoNascita integer,
AnnoMorte integer,
Nazionalità char(20)}

TABLE Romanzi

{Codice integer **PRIMARY KEY**,
Titolo char(40) **NOT NULL**,
Autore integer **NOT NULL REFERENCES** Autori(Codice),
Protagonista integer **REFERENCES** Personaggi(Codice),
AnnoPubblicazione integer}

TABLE PersonaggiNeiRomanzi

{Romanzo integer **NOT NULL**
REFERENCES Romanzi(Codice),
Personaggio integer **NOT NULL**
REFERENCES Personaggi (Codice),
PRIMARY KEY(Romanzo, Personaggio)}

TABLE Personaggi

{Codice integer **PRIMARY KEY**,
Cognome char(30) **NOT NULL**,
Nome char(20) **NOT NULL**,
Sesso char(1)}

- 1) Elencare gli autori dell'ottocento, ordinati per nazionalità, che hanno scritto romanzi nel cui titolo compare un termine con radice "tempesta" (algebra oppure SQL, punti 6)

Tab1 = (Autori JOIN Romanzi ON Autori.Codice=Romanzi.Autore)

Tab2 = σ Autori.AnnoNascita \geq 1800 AND Autori.AnnoNascita $<$ 1900 AND Romanzi.Titolo LIKE '*tempest*' (Tab1)

Tab3 = (π Autori.Cognome, Autori.Nome, Autori.Nazionalità (Tab2))

Risultato = ORDER BY Autori.Nazionalità, Autori.Cognome (DISTINCT(Tab3))

```
SELECT DISTINCT Autori.Cognome, Autori.Nome, Autori.Nazionalità
FROM Autori JOIN Romanzi ON Autori.Codice=Romanzi.Autore
WHERE Autori.AnnoNascita  $\geq$  1800 AND Autori.AnnoNascita  $<$  1900 AND
      Romanzi.Titolo LIKE '*tempest*'
ORDER BY Autori.Nazionalità, Autori.Cognome
```

- 2) Elencare i romanzi degli autori russi viventi, con il relativo anno di pubblicazione (algebra oppure SQL, punti 6)

Tab1 = (Autori JOIN Romanzi ON Autori.Codice=Romanzi.Autore)

Tab2 = σ Autori.Nazionalità = 'russa' AND Autori.AnnoMorte IS NULL (Tab1)

Risultato = (π Romanzo.Titolo, Romanzi.AnnoPubblicazione, Autori.Cognome (Tab2))

```
SELECT Romanzi.Titolo, Romanzi.AnnoPubblicazione, Autori.Cognome
FROM (Autori JOIN Romanzi ON Autori.Codice=Romanzi.Autore)
WHERE Autori.Nazionalità = 'russa' AND Autori.AnnoMorte IS NULL
ORDER BY Romanzi.AnnoPubblicazione, Autori.Cognome
```

Cognome:

Nome:

Matricola:

3) Elencare i personaggi dei romanzi il cui protagonista è Maigret (algebra o SQL, punti 7)

Tab1 = (Personaggi AS Protagonisti JOIN Romanzi ON Personaggi.Codice=Romanzi.Protagonista
JOIN PersonaggiNeiRomanzi ON Romanzi.Codice=PersonaggiNeiRomanzi.Romanzo
JOIN Personaggi ON Personaggi.Codice=PersonaggiNeiRomanzi.Personaggio)

Tab2 = $\sigma_{\text{Protagonisti.Cognome} = \text{'Maigret'}}(\text{Tab1})$

Risultato = $\text{DISTINCT}(\pi_{\text{Personaggi.Cognome, Personaggi.Nome}}(\text{Tab2}))$

```
SELECT DISTINCT Personaggi.Cognome, Personaggi.Nome
FROM Personaggi AS Protagonisti JOIN Romanzi ON Protagonisti.Codice=Romanzi.Protagonista
JOIN PersonaggiNeiRomanzi ON Romanzi.Codice=PersonaggiNeiRomanzi.Romanzo
JOIN Personaggi ON Personaggi.Codice=PersonaggiNeiRomanzi.Personaggio
WHERE Protagonisti.Cognome = 'Maigret'
```

Alternativa:

```
CREATE VIEW RomanziMaigret
SELECT DISTINCT Romanzi.Codice
FROM Personaggi JOIN Romanzi ON Personaggi.Codice=Romanzi.Protagonista
WHERE Personaggi.Cognome = 'Maigret'
```

```
SELECT DISTINCT Personaggi.Cognome, Personaggi.Nome
FROM RomanziMaigret
JOIN PersonaggiNeiRomanzi ON RomanziMaigret.Codice=PersonaggiNeiRomanzi.Romanzo
JOIN Personaggi ON Personaggi.Codice=PersonaggiNeiRomanzi.Personaggio
```

4) Elencare gli autori longevi, che hanno pubblicato romanzi a distanza di oltre 40 anni l'uno dall'altro (algebra o SQL, punti 7)

Tab1 = (Autori JOIN Romanzi AS Rom1 ON Autori.Codice=Rom1.Autore
JOIN Romanzi AS Rom2 ON Autori.Codice=Rom2.Autore)

Tab2 = $\sigma_{\text{Rom1.AnnoPubblicazione} - \text{Rom2.AnnoPubblicazione} > 40}(\text{Tab1})$

Tab3 = $\text{DISTINCT}(\pi_{\text{Autori.Cognome, Autori.Nome, Autori.Nazionalit\`a}}(\text{Tab2}))$

Risultato = ORDER BY Autori.Cognome (Tab3)

```
SELECT DISTINCT Autori.Cognome, Autori.Nome
FROM Autori JOIN Romanzi AS Rom1 ON Autori.Codice=Rom1.Autore
JOIN Romanzi AS Rom2 ON Autori.Codice=Rom2.Autore
WHERE Rom1.AnnoPubblicazione - Rom2.AnnoPubblicazione > 40
ORDER BY Autori.Cognome
```

Cognome:

Nome:

Matricola:

5) Modificare lo schema del database letterario per rappresentare le case editrici dei romanzi (punti 7)

```
TABLE Romanzi
{Codice integer PRIMARY KEY,
Titolo char(40) NOT NULL,
Autore integer NOT NULL REFERENCES Autori(Codice),
Protagonista integer REFERENCES Personaggi(Codice),
CasaEditrice integer REFERENCES CaseEditrici(Codice),
AnnoPubblicazione integer}
```

```
TABLE CaseEditrici
{Codice integer PRIMARY KEY,
Nome char(30) NOT NULL,
Città char(20),
Nazionalità char(20)}
```

6) Elencare i romanzi il cui protagonista è Maigret (algebra o SQL, punti 7)

Tab1 = (Personaggi JOIN Romanzi ON Personaggi.Codice=Romanzi.Protagonista)

Tab2 = $\sigma_{\text{Personaggi.Cognome} = \text{'Maigret'}}(\text{Tab1})$

Risultato = $\text{DISTINCT}(\pi_{\text{Romanzi.Titolo, Romanzi.AnnoPubblicazione}}(\text{Tab2}))$

```
SELECT DISTINCT Romanzi.Titolo, Romanzi.AnnoPubblicazione
FROM Personaggi JOIN Romanzi ON Personaggi.Codice=Romanzi.Protagonista
WHERE Personaggi.Cognome = 'Maigret'
```

7) Elencare gli autori che hanno scritto romanzi sia nell'800 che nel 900 (algebra o SQL, punti 7)

```
SELECT DISTINCT Autori.Cognome, Autori.Nome
FROM Autori JOIN Romanzi ON Autori.Codice=Romanzi.Autore
WHERE Romanzi.AnnoPubblicazione < 1900 AND Romanzi.AnnoPubblicazione >= 1900
```

ERRORE! Perché? Qual è il risultato che si ottiene? Alternativa corretta:

```
SELECT DISTINCT Autori.Cognome, Autori.Nome
FROM Autori JOIN Romanzi ON Autori.Codice=Romanzi.Autore
WHERE Romanzi.AnnoPubblicazione > 1799 AND Romanzi.AnnoPubblicazione < 1900
```

INTERSECT

```
SELECT DISTINCT Autori.Cognome, Autori.Nome
FROM Autori JOIN Romanzi ON Autori.Codice=Romanzi.Autore
WHERE Romanzi.AnnoPubblicazione > 1899 AND Romanzi.AnnoPubblicazione < 2000
```

Cognome:

Nome:

Matricola:

Altra possibilità:

```
SELECT DISTINCT Autori.Cognome, Autori.Nome
FROM  Autori JOIN Romanzi AS Romanzi1800 ON Autori.Codice=Romanzi1800.Autore
      JOIN Romanzi AS Romanzi1900 ON Autori.Codice=Rom1900.Autore
WHERE Romanzi1800.AnnoPubblicazione > 1799 AND Romanzi1800.AnnoPubblicazione <1900 AND
      Romanzi1900.AnnoPubblicazione > 1899 AND Romanzi1900.AnnoPubblicazione < 2000
ORDER BY Autori.Cognome, Autori.Nome
```

8) Elencare gli autori che hanno scritto romanzi esclusivamente nell'800 (algebra o SQL, punti 7)

```
SELECT DISTINCT Autori.Cognome, Autori.Nome
FROM  Autori JOIN Romanzi ON Autori.Codice=Romanzi.Autore
WHERE Romanzi.AnnoPubblicazione > 1799 AND Romanzi.AnnoPubblicazione < 1900
```

EXCEPT

```
SELECT DISTINCT Autori.Cognome, Autori.Nome
FROM  Autori JOIN Romanzi ON Autori.Codice=Romanzi.Autore
WHERE Romanzi.AnnoPubblicazione < 1800 OR Romanzi.AnnoPubblicazione > 1899
```

Cognome:

Nome:

Matricola:

Elencare gli autori longevi, che hanno pubblicato romanzi a distanza di oltre 40 anni.

```
CREATE VIEW AutoriLongevi
SELECT DISTINCT Autori.Cognome, Autori.Nome, Autori.Codice
FROM Autori JOIN Romanzi ON Autori.Codice=Romanzi.Autore
      JOIN Romanzi AS Romanzi1 ON Autori.Codice=Romanzi1.Autore
WHERE Romanzi1.AnnoPubblicazione – Romanzi.AnnoPubblicazione > 40
ORDER BY Autori.Cognome, Autori.Nome
```

Elencare tutti i romanzi degli autori longevi, che hanno pubblicato romanzi a distanza di oltre 40 anni.

```
SELECT DISTINCT AutoriLongevi.Cognome, Romanzi.Titolo, AnnoPubblicazione
FROM AutoriLongevi JOIN Romanzi ON AutoriLongevi.Codice=Romanzi.Autore
ORDER BY AutoriLongevi.Cognome, Romanzi.AnnoPubblicazione
```