

Prima esercitazione su SQL

Basi di dati

A.A. 2023/24

Luca Andolfi, Maurizio Lenzerini

Usare la macchina virtuale

- Se non sei in laboratorio e hai la macchina virtuale LXLE-BIAR-4.5, accendi ed entra nella macchina virtuale stessa, dove sono già installati PostgreSQL e pgAdmin 4.
- Se non sei in laboratorio e non hai la macchina virtuale, puoi usare direttamente PostgreSQL e pgAdmin 4, che devi aver precedentemente scaricato.
- Se sei in laboratorio, apri il programma “Oracle VirtualBox” e all’interno del programma accendi la macchina virtuale di nome LXLE-BIAR-4.5, dove sono già installati PostgreSQL e pgAdmin 4.
- Dal desktop della macchina virtuale segui START > PROGRAMMING > pgAdmin 4 per mandare in esecuzione pgAdmin 4.
- Se nel menu di sinistra di pgAdmin compare il server “*local*”, apri la connessione con tale server (nella macchina virtuale lo username da usare è “*postgres*” e la password è “*biar*”). Se il server non compare occorre registrarlo premendo tasto destro su “*servers*” e specificando nome e indirizzo “*localhost*” e password “*biar*”.
- In pgAdmin 4, posiziona il mouse sull’icona del server e con il tasto destro scegli create → database; dai quindi il nome “*Esercitazione-3-2023*” al nuovo database.
- Posiziona il mouse sull’icona del nuovo database creato (se non appare, fai il refresh posizionando il mouse sull’icona del server) e clicca con il tasto destro sull’icona di tale database.
- Seleziona “*Query Tool*”.
- Nella finestra che apparirà, inserisci il contenuto del file “*script.sql*”, scaricabile dalla sezione “*Esercitazione 3*” della pagina di Moodle del corso, e dai il comando di esecuzione (icona di “*esegui*”). Il codice crea tutte le tabelle necessarie per l’esercitazione. Dalla stessa sezione di Moodle sono scaricabili queste slides.

La base di dati

persona(nome, eta, reddito, sesso, cittaNascita)

citta(nome, regione)

genia(figlio, genitore)

Osservazioni:

- il dominio di Sesso è {'M', 'F'}

Query 1

Vogliamo sapere quali sono le persone maggiorenni, con nome e regione di nascita.

Query 2

Vogliamo avere la lista senza ripetizioni delle regioni in cui si trova almeno una città in cui è nato un genitore di oltre 50 anni.

Query 3

Vogliamo tutte le coppie <genitore,figlio> nati nella stessa città.

Query 4

Vogliamo avere la lista senza ripetizioni dei nonni maschi.

Query 5

Vogliamo avere la lista senza ripetizioni delle regioni dove si trova almeno una città in cui è nato un nonno maschio che ha meno di 60 anni.

Query 6

Vogliamo tutte le coppie di fratelli (A è fratello di B se A e B hanno la stessa coppia di genitori), senza ridondanze (cioè senza <f,f> e senza <f,g> se già compare <g,f>).

Query 7

Vogliamo sapere chi è il nonno paterno e chi è la nonna paterna di ogni persona di cui entrambi tali nonni sono noti.

Query 1

-- persone maggiorenni, con nome e regione di nascita

Query 1: soluzione

-- persone maggiorenni, con nome e regione di nascita

```
select persona.nome, citta.regione  
from persona, citta  
where persona.eta > 17 and  
       persona.cittaNascita = citta.nome
```

Query 2

-- regioni dove ci sono città in cui è nato almeno un genitore di oltre 50 anni, evitando ripetizioni

Query 2: soluzione

-- regioni dove ci sono città in cui è nato almeno un genitore di oltre 50 anni, evitando ripetizioni

```
select distinct citta.regione  
from genia, persona, citta  
where genia.genitore = persona.nome and  
      persona.eta > 50 and  
      persona.cittaNascita = citta.nome
```


Query 3

-- coppie <genitore,figlio> nati nella stessa città

Query 3: soluzione

-- coppie <genitore,figlio> nati nella stessa città

select genia.genitore, genia.figlio

from persona as p1, genia, persona as p2

where p1.nome = genia.figlio and genia.genitore = p2.nome
and p1.cittaNascita = p2.cittaNascita

Query 4

-- vogliamo sapere chi sono i nonni maschi, senza ripetizioni

Query 4: soluzione

-- vogliamo sapere chi sono i nonni maschi, senza ripetizioni

```
select distinct p.nome as nonno  
from genia g1, genia g2, persona p  
where g1.genitore = g2.figlio and g2.genitore = p.nome  
and p.sesso = 'M'
```

Query 5

-- regioni dove ci sono città in cui è nato almeno un nonno (maschio) che ha meno di 60 anni, senza ripetizioni

Query 5: soluzione

-- regioni dove ci sono città in cui è nato almeno un nonno (maschio) che ha meno di 60 anni, senza ripetizioni

```
select distinct c.regione
from genia g1, genia g2, persona p, citta c
where g1.genitore = g2.figlio and g2.genitore = p.nome
      and p.sesso = 'M' and p.eta < 60 and
      p.cittaNascita = c.nome
```

Query 6

-- vogliamo tutte le coppie di fratelli (A è fratello di B se A e B hanno la stessa coppia di genitori), senza ridondanze

Query 6: soluzione

-- vogliamo tutte le coppie di fratelli (A è fratello di B se A e B hanno la stessa coppia di genitori), senza ridondanze

```
select distinct g1.figlio as fratello1, g3.figlio as fratello2
from genia g1, genia g2, genia g3, genia g4
where g1.figlio = g2.figlio and g3.figlio = g4.figlio
      and g1.genitore != g2.genitore
      and g1.genitore = g3.genitore
      and g2.genitore = g4.genitore
      and g1.figlio < g3.figlio
```


Query 7

-- vogliamo nonno paterno e nonna paterna di ogni persona di cui entrambi tali nonni sono noti.

Query 7: soluzione

-- vogliamo nonno paterno e nonna paterna di ogni persona di cui entrambi tali nonni sono noti.

```
select g1.figlio, g2.genitore as nonnopaterno, g3.genitore
      as nonnapaterna
from persona p1, genia g1, genia g2, genia g3,
      persona p2, persona p3
where p1.nome = g1.genitore and p1.sesso = 'M' and
      g1.genitore = g2.figlio and g1.genitore = g3.figlio and
      g2.genitore = p2.nome and p2.sesso = 'F' and
      g3.genitore = p3.nome and p3.sesso = 'M'
```