

A. Dato il seguente schema logico e i seguenti vincoli, si scriva il codice SQL che crea le tabelle corrispondenti. Si ricordi di creare le tabelle nell'ordine richiesto dal linguaggio SQL.

fotografi(id, nome, cognome, SSN, nazione)

foto(id, fotografo, modello_fotocamera, produttore_fotocamera, immagine, apertura, tempo, data)

fotocamere(modello, produttore, caratteristiche, prezzo, peso, tipo)

produttori(id, nome, telefono, sigla)

Vincoli integrità referenziale

foto.fotografo -> fotografi.id

foto.modello_fotocamera, foto.produttore_fotocamera -> fotocamere.modello, fotocamere.produttore

fotocamere.produttore -> produttori.id

Altri vincoli

A parte le chiavi, tutti gli altri campi possono essere nulli.

Il campo SSN è una stringa a lunghezza variabile che rappresenta il servizio sanitario nazionale (in Italia è il codice fiscale). Il campo SSN è unico in ciascuna nazione, ma possono esistere SSN uguali in nazioni diverse. Fare in modo che non si possano inserire due fotografi con lo stesso SSN nella stessa nazione.

Il valore di default del campo data di foto è "current_date" (la funzione SQL current_date restituisce la data corrente).

Il campo immagine di foto deve poter contenere un'immagine grande fino a 100MB.

I campi apertura e tempi di foto devono contenere numeri in virgola mobile.

Il campo prezzo delle fotocamere deve poter contenere valori con due cifre decimali fino a 99999.99

Il campo caratteristiche delle fotocamere deve contenere un testo lungo fino a 10kB.

Il campo tipo di fotocamere deve essere una stringa fra "compatta", "reflex", "mirrorless". E' anche possibile che tale campo contenga un valore nullo.

Il campo peso delle fotocamere deve contenere un valore intero

Occorre controllare che non ci siano due produttori con lo stesso numero di telefono.

Il campo sigla di produttore è lungo esattamente 5 caratteri.

Sia dato il seguente schema relazionale

libri(id, titolo, anno)
scritture(libro, autore, compenso, ruolo)
autori(id, nome, nascita, nazione)
parentele(autore, genitore)

Vincoli integrità referenziale

scritture.libro -> libri.id
scritture.autore -> autori.id
parentele.autore -> autori.id
parentele.genitore -> autori.id

B. Scrivere le seguenti interrogazioni in algebra relazionale con theta join

1. Trovare le nazioni dove vivono almeno due autori.
2. Trovare il nome degli autori che non hanno scritto libri

Scrivere le seguenti interrogazioni con il calcolo su tuple

3. Trovare il nome degli autori che non hanno partecipato alla scrittura di nessun libro ricevendo un compenso maggiore di 10.000
4. Fra tutte le coppie di autori che si possono definire combinando gli autori esistenti nell'archivio, si trovino le coppie che non hanno mai scritto un libro insieme, restituendo il nome di entrambi gli autori

**C. Scrivere le seguenti interrogazioni sia in SQL
(se necessario, definire delle viste)**

5. Trovare tutte le scritture di libri presenti nell'archivio, restituendo il titolo del libro, il nome dell'autore e il ruolo svolto. Includere anche i libri senza autore.
6. Trovare il nome e l'età dei nonni dell'autore di nome "pippo". Ordinare i risultati per l'età del nonno, dal più vecchio al più giovane. Nota: l'età di un autore può essere calcolata per differenza con la data corrente (current_date) a partire dalla data di nascita dell'autore
7. Si calcoli il numero di libri scritti da ciascun autore nato dopo il '1/1/1970', restituendo il nome dell'autore e il numero dei libri scritti. Si restituisca solo gli autori che hanno scritto almeno 2 libri in tale periodo.
8. Si calcoli la somma dei compensi erogati per ciascun anno, restituendo l'anno e il totale dei compensi
9. Usando gli operatori insiemistici (senza usare le sottointerrogazioni) si trovi il nome degli autori che non hanno mai ricevuto un compenso maggiore 5000 per la scrittura di un libro.
10. Usando le sottointerrogazioni (senza usare gli operatori insiemistici), calcolare il numero degli autori italiani presenti nell'archivio che non hanno scritto libri.
11. Trovare il numero medio e il numero minimo dei libri per gli anni presenti nell'archivio. Il risultato deve contenere una riga con i valori medio e minimo
12. Trovare il nome degli autori che hanno ricevuto sempre compensi maggiori di 10000 per ogni libro che hanno scritto.

Soluzioni esercizio A

Create table fotografi(

id serial not null primary key,
nome varchar(30),
cognome varchar(30),
SSN varchar(20),
nazione varchar(20),
unique(nazione,SSN))

create table produttori(

id serial not null primary key,
nome varchar(30),
telefono varchar(20) unique,
sigla char(5))

create table fotocamere(

modello varchar(20) not null,
produttore int not null references produttori(id),
caratteristiche clob(10k),
prezzo decimal(7,2),
peso int,
tipo varchar(20) check(tipo is null or tipo='compatta' or tipo= 'reflex' or tipo= 'mirrorless'),
primary key(modello,produttore)
)

Create table foto(

id serial not null primary key,
fotografo int references fotografi(id),
modello_fotocamera varchar(20),
produttore_fotocamera int,
immagine blob(100M),
apertura float,
tempo float,
data date default current_date,
foreign key (modello_fotocamera, produttore_fotocamera)
references fotocamere(modello, produttore))

Soluzioni B

Scrivere le seguenti interrogazioni in algebra relazionale con theta join

1. Trovare le nazioni dove vivono almeno due autori.

$$\text{autori2} = \delta_{\text{id2}, \text{nome2}, \text{nascita2}, \text{nazione2} \leftarrow \text{id}, \text{nome}, \text{nascita}, \text{nazione}}(\text{autori})$$
$$\pi_{\text{nazione}}(\text{autori} \bowtie \triangleleft_{\text{id} \neq \text{id2} \wedge \text{nazione} = \text{nazione2}} \text{autori2})$$

2. Trovare il nome degli autori che non hanno scritto libri

$$\pi_{\text{nome}}(\pi_{\text{nome}, \text{id}}(\text{autori}) - \pi_{\text{nome}, \text{id}}(\text{scritture} \bowtie \triangleleft_{\text{id} = \text{autore}} \text{autori}))$$

Soluzioni C

Scrivere le seguenti interrogazioni con il calcolo su tuple

3. Trovare il nome degli autori che non hanno partecipato alla scrittura di nessun libro ricevendo un compenso maggiore di 10.000

$$\{a.\text{nome} \mid a(\text{autori}) \mid \neg \exists s(\text{scritture}) s.\text{autore} = a.\text{id} \wedge s.\text{compenso} < 10000\}$$

4. Fra tutte le coppie di autori che si possono definire combinando gli autori esistenti nell'archivio, si trovino le coppie che non hanno mai scritto un libro insieme, restituendo il nome di entrambi gli autori

$$\{\text{nome1}: a.\text{nome}, \text{nome2}: a.\text{nome} \mid a(\text{autori}), a2(\text{autori}) \mid$$
$$\neg \exists s(\text{scritture}) \exists s2(\text{scritture}) s.\text{autore} = a.\text{id} \wedge s2.\text{autore} = a2.\text{id} \wedge s.\text{libro} = s2.\text{libro}\}$$

5. Trovare tutte le scritture di libri presenti nell'archivio, restituendo il titolo del libro, il nome dell'autore e il ruolo svolto. Includere anche i libri senza autore.

```
select libri.titolo, scritture.ruolo, autori.nome
from libri left join scritture on libri.id=scritture.libro
      left join autori on scritture.autore=autori.id
```

6. Trovare il nome e l'età dei nonni dell'autore di nome "pippo". Ordinare i risultati per l'età del nonno, dal più vecchio al più giovane. Nota: l'età di un autore può essere calcolata per differenza con la data corrente (current_date) a partire dalla data di nascita dell'autore

```
select nonni.nome, nonni.eta
from autori as nonni, autori, parentele as par1, parentele as par2
where nonni.id=par1.genitore and par1.autore=par2.genitore and par2.autore=autori.id
      autori.nome='pippo'
```

7. Si calcoli il numero di libri scritti da ciascun autore nato dopo il '1/1/1970', restituendo il nome dell'autore e il numero dei libri scritti. Si restituisca solo gli autori che hanno scritto almeno 2 libri in tale periodo.

```
select autori.nome, count(*) as numero_libri
from autori, scritture
where nascita > '1/1/1970' and scritture.autore=autori.id
group by autori.id, autori.nome
having count(*) >=2
```

8. Si calcoli la somma dei compensi erogati per ciascun anno, restituendo l'anno e il totale dei compensi

```
Select libri.anno, sum(compenso) as somma_compensi
from scritte, libri
where scritte.libro=libri.id
group by libri.anno
```

9. Usando gli operatori insiemistici (senza usare le sottointerrogazioni) si trovi il nome degli autori che non hanno mai ricevuto un compenso maggiore 5000 per la scrittura di un libro.

```
Select autori.nome
from autori
except
Select autori.nome
from autori, scritte
where autori.id=scritte.autore and scritte.compenso>5000
```

10. Usando le sottointerrogazioni (senza usare gli operatori insiemistici), calcolare il numero degli autori italiani presenti nell'archivio che non hanno scritto libri.

```
select count(*) as numero_autori
from autori
where not exists( select from scritte where autori.id=scritte.autore)
```

11. Trovare il numero medio e il numero minimo dei libri per gli anni presenti nell'archivio. Il risultato deve contenere una sola riga con i valori medio e minimo

```
create view numero_libri(anno,numero)
as select anno, count(*)
from scritte, libri
where scritte.libro=libri.id
group by anno

select avg(numero) as media, min(numero) as minimo
from numero_libri
```

12. Trovare il nome degli autori che hanno ricevuto sempre compensi maggiori di 10000 per ogni libro che hanno scritto.

```
select nome
from autori
where 10000 < all (select compenso from scritte where scritte.autore=autori.id)
```