Cabir Erguven

1- 'pycoders' isimli bir server kurun. (M)
Kuruldu.

2- 'class4' database olusturun (M). Database silin (M). Ayni database yine olusturun (K)
create database class4

3- https://www.postgresqltutorial.com/postgresql-sample-database/ adresine gidin ve ER modeli inceleyin. Tablolar arasindaki en az 5 iliskiyi yazin.(Hangi tablolar arasinda ne tur bir iliski var)

- There is one and only one to zero or many relation between city and address tables. City_id is a primary key in city table but foreign key in address table.
- There is one and only one to zero or one realtion between category and film_category. category_id is a primary key in category table but foreign key in film_category table.
- There is one and only one to zero or many realtion between film and inventory tables. Film_id is the primary key in film table but foreign key in inventory table.
- There is one and only one to zero or many relation between address and staff tables. Address_id primary key in adress table but foreign key in staff table.
- There is one and only one to zero or many relation between staff and payment tables. Staff_id primary key in adress table but foreign key in staff table.

4- ER modeldeki tablolardan 3 tanesini M olusturun.
Olusturuldu.

5- ER modeldeki tablolardan 3 tanesini K olusturun.

```
create table rental        (
        rental_id int primary key not null,
        rental_date date,
        inventory_date date,
        costumer_id int,
        return_date date,
        staff_id int,
        last_update date
)

create table city          (
        city_id int primary key not null,
        city varchar(15),
        country_id int,
        costumer_id int,
        last_update date
)

create language            (
        language_id int primary key not null,
        name varchar(15),
        last_update date
)
```

6- ER modeldeki tablolardan 3 tanesini C olusturun.
(4-5-6. sorulari cozerken toblolar arasindaki iliskileri gozardi edebilirsiniz. Tum kolonlari girmek zorunda degilsiniz, en az 2 kolon olmasi yeterli.)

```
import psycopg2
conn = psycopg2.connect("dbname = class4 user=postgres password = 1111")
cur = conn.cursor()
cur.execute(
""" CREATE TABLE language (
language_id INT PRIMARY KEY,
name VARCHAR (20) NOT NULL,
last_update TIMESTAMP)""")

cur.close()
conn.commit()


import psycopg2
conn = psycopg2.connect("dbname = class4 user=postgres password = 1111")
cur = conn.cursor()
cur.execute(
""" CREATE TABLE city (
city_id INT PRIMARY KEY,
city VARCHAR (20) NOT NULL,
country_id integer NOT NULL,
last_update TIMESTAMP,
FOREIGN KEY (country_id)
    REFERENCES city (country_id)
)""")
cur.close()
conn.commit()


import psycopg2
conn = psycopg2.connect("dbname=class4 user=postgres password=1111")
cur = conn.cursor()
cur.execute(
""" CREATE TABLE payment (
 payment_id INT NOT NULL,
 payment_date TIMESTAMP,
 rental id INT NOT NULL,
 PRIMARY KEY (payment_id),
 FOREIGN KEY (rental_id)
 REFERENCES rental (rental_id))"""
)

cur.close()
conn.commit()
conn.close()
```

7- Olusturdugunuz 3 tabloya M ile 5 veri girisi yapin.
Yapildi.

8- Olusturdugunuz 3 tabloya K ile 5 veri girisi yapin.

```
insert into language  values(1, “English” , '03/03/2021')
insert into language  values(2, “French” , '04/03/2021')
insert into language values(3, “Dutch” , '05/03/2021')
insert into language values(4, “German” , '06/03/2021')
insert into language values(5, “Spainsh” , '07/03/2021')

insert into city  values(1, “Amsterdam”, 1, '07/03/2021')
insert into city  values(2, “Paris”, 2, '07/03/2021')
insert into city  values(3, “London”, 3, '07/03/2021')
insert into city  values(4, “Berlin”, 4, '07/03/2021')
insert into city  values(5, “Lahey”, 1, '07/03/2021')

insert into payment values (1, '13/04/2021',52)
insert into payment values (2, '23/04/2021',25)
insert into payment values (3, '03/04/2021',54)
insert into payment values (4, '13/04/2021',65)
insert into payment values (5, '16/04/2021',75)
```

9- Olusturdugunuz 3 tabloya C ile 5 veri girisi yapin.

```
import psycopg2
conn = psycopg2.connect("dbname = class4 user=postgres password=1111")
cur = conn.cursor()
cur.execute("insert into language values (1, “English” , '03/03/2021')")
cur.execute("insert into language values (2, “French” , '04/03/2021')")
cur.execute("insert into language values (3, “Dutch” , '05/03/2021')")
cur.execute("insert into language values (4, “German” , '06/03/2021')")
cur.execute("insert into language values (5, “Spainsh” , '07/03/2021')")
cur.close()
conn.commit()
conn.close()


import psycopg2
conn = psycopg2.connect("dbname = class4 user=postgres password=1111")
cur = conn.cursor()
cur.execute("insert into city values (1, 'Amsterdam', 1, '07/03/2021')")
cur.execute("insert into city values (2, “Paris”, 2, '07/03/2021')")
cur.execute("insert into city values (3, “London”, 3, '07/03/2021')")
cur.execute("insert into city values (4, “Berlin”, 4, '07/03/2021')")
cur.execute("insert into city values (5, “Lahey”, 1, '07/03/2021')")
cur.close()
conn.commit()
conn.close()


import psycopg2
conn = psycopg2.connect("dbname = class4 user=postgres password=1111")
cur = conn.cursor()
cur.execute("insert into payment values (1, '13/04/2021',52)")
cur.execute("insert into payment values (2, '23/04/2021',25)")
cur.execute("insert into payment values (3, '03/04/2021',54)")
cur.execute("insert into payment values (4, '13/04/2021',65)")
cur.execute("insert into payment values (5, '16/04/2021',75)")
cur.close()
conn.commit()
conn.close()
```

10- 3 tablodaki birer veriyi M ile degistirin.
Değiştirildi

11- 3 tablodaki birer veriyi K ile degistirin.

update language name=”Polish” where name =”Spanish”
update city=”Athena” where name =”Paris”
update payment payment_date=''16/05/2021” where payment_date=''16/04/2021”

12- 3 tablodaki birer veriyi C ile degistirin.

```
conn=psycopg2.connect("dbname=class4 user=postgres password=1111")
cur=conn.cursor()
cur.execute("update language name=”Polish” where name =”Spanish”)
cur.close()
conn.commit()
conn.close()

conn=psycopg2.connect("dbname=class4 user=postgres password=1111")
cur=conn.cursor()
cur.execute("city=”Athena” where name =”Paris”)
cur.close()
conn.commit()
conn.close()


conn=psycopg2.connect("dbname=class4 user=postgres password=1111")
cur=conn.cursor()
cur.execute("payment payment_date=''16/05/2021” where payment_date=''16/04/2021”)
cur.close()
conn.commit()
conn.close()
```

13- 3 tablonun son satirini M ile silin.
Silindi

14- 3 tablonun son satirini K ile silin.

delete from language where inventory_id=5
delete from city where city=”Lahey”
delete from payment where rental_id=75

15- 3 tablonun son satirini C ile silin.

```
conn=psycopg2.connect("dbname=class4 user=postgres password=1234")
cur=conn.cursor()
cur.execute("delete from language where inventory_id=5")
cur.close()
conn.commit()
conn.close()

conn=psycopg2.connect("dbname=class4 user=postgres password=1234")
cur=conn.cursor()
cur.execute("delete from city where city=”Lahey”")
```

```
cur.close()
conn.commit()
conn.close()

conn=psycopg2.connect("dbname=class4 user=postgres password=1234")
cur=conn.cursor()
cur.execute("delete from payment where rental_id=75”)
cur.close()
conn.commit()
conn.close()
```

16- 1 tabloyu M ile silin.

```
Silindi.
```

17- 1 tabloyu K ile silin.

```
Drop table film_category
```

18- 1 tabloyu C ile silin.

```
conn=psycopg2.connect("dbname=class4 user=postgres password=1111")
cur=conn.cursor()
cur.execute("drop table costumer")
cur.close()
conn.commit()
conn.close()
```

19- Kalan tablolardan 1 tanesinin 2 veya 3 sutununu K ile baska bir tablo olarak olusturun.

```
CREATE TABLE city_new AS SELECT (city_id), (city) FROM city;
```

20- Kalan tablolardan 1 tanesinin 2 veya 3 sutununu C ile baska bir tablo olarak olusturun.

```
conn=psycopg2.connect("dbname=class4 user=postgres password=1234")
cur=conn.cursor()
cur.execute("CREATE TABLE city_new AS SELECT (city_id), (city) FROM city ")
cur.close()
conn.commit()
conn.close()
```

21- Tablolardan 1 tanesini M ile truncate edin.

```
Edildi.
```

22- Tablolardan 1 tanesini K ile truncate edin.

```
Truncate store
```

23- Tablolardan 1 tanesini C ile truncate edin.

```
conn=psycopg2.connect("dbname=class4 user=postgres password=1111")
cur=conn.cursor()
cur.execute("truncate staff")
cur.close()
conn.commit()
conn.close()
```

24- Truncate edilmis tablolari M ile silin.

```
Silindi.
```

25- 2 tabloyu K ile silin.

drop table film, inventory

26- 2 tabloyu C ile silin.

```
conn=psycopg2.connect("dbname=class4 user=postgres password=1111")
cur=conn.cursor()
cur.execute("drop table address, category")
cur.close()
conn.commit()
conn.close()
```

27- Elimizde veri olan 1 tablo kalmis olmasi lazim. Bu tabloyu csv olarak bilgisayariniza yukleyin.
Ok.

28- Postgresql arayuzundeki son tabloyu da K ile silin.
drop table payment

29- Bilgisayarinizdaki csv yi arayuze import edin.
Ok.

30- Import ettiginiz bu tabloyu C ile silin.

```
conn=psycopg2.connect("dbname=class4 user=postgres password=1111")
cur=conn.cursor()
cur.execute("drop table city")
cur.close()
conn.commit()
conn.close()
```

31- https://www.postgresqltutorial.com/postgresql-sample-database/ linkindeki ornek DB yi bilgisayariniza indirin ve arayuze yukleyin.
Ok.

32- DB nizde 15 adet tablo olmasi lazim. Her tabloyu teker teker goruntuleyin ve kolon isimlerine bakarak, 5 tabloda hangi kolonun PK ve FK oldugunu yazin.

- Actor: PK: actor_id FK:
- Adress PK: adress_pkey FK: fk_address_city
- Category PK: category_pkey FK:
- City PK: city_pkey FK: fk_city
- Country PK: country_pkey
- Customer PK: customer_pkey FK: customer_address_id_fkey
- Film PK: film_pkey FK: film_language_id_fkey
- Film_actor PK: film_actor_pkey FK: film_actor_actor_id_fkey, film_actor_film_id_fkey
- Film_category PK: film_category_pkey FK: film_category_category_id_fkey,
- Inventory PK: inventory_pkey FK: inventory_film_id_fkey
- Language PK: language_pkey FK:
- Payment PK: payment_pkey FK: payment_customer_id_fkey, payment_rental_id_fkey, payment_staff_id_fkey
- Rental PK: rental_pkey FK: rental_customer_id_fkey, rental_inventory_id_fkey, rental_staff_id_key
- Staff PK: staff_pkey FK: staff_address_id_fkey
- Store PK: store_pkey FK: store_address_id_fkey, store_manager_staff_id_fkey

Sorgular? (Asagidaki sorularin cevaplarini ve bu cevabi bulurken kullandiginiz kodlari yazin)
33- Action filmlerinin ortalama suresi ne kadar?
SELECT AVG(length) FROM film where (description) like '%Action%';

select avg(length) from film where film_id in (select film_id from film_category where category_id=1)

34- En cok staff olan store hangisidir?
SELECT store_id, count (store_id) FROM staff GROUP BY store_id ORDER BY count (store_id) DESC limit 1

35- 'Gene Willis' adli actorun oynadigi filmlerin ratingi nedir?

36- Aktif customer sayisi nedir?
SELECT count(*) from customer where active=1

37- 'C' harfiyle baslayan filmler hangileridir?
SELECT * from film where title like 'C%'

38- 4$ den az odeme yapan musterilerin e-mail edresleri nedir?
SELECT email from customer where customer_id in (select payment.customer_id from payment where amount<4)

39- Moscow'da ikamet eden staff ve customer tablosu? (sadece isim/soyisim sutunu olsun)

40- En az kiralanan 5 film hangisidir?

41- 2006 yilinda yayinlanan ingilizce filmler hangileridir?

select title from film where release_year=2006 and language_id in (select language_id from language where name='English')