**(M:Manuel olarak, K:SQL komutlarıyla, C:Python kodlariyla)**

**Asagidaki sorudan K ve C ile cozulmesini tasdikimizin cozumlerini (komüt kodlarini) ustte test veya altta cozumu sekilde bir dosyaya yapistirip gondermenizi istiyoruz.**

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

Kuruldu

**2- 'class4' veritabanı olusturun (M). Veritabanı silindi (M). Ayni veri tabanı yine olusturun (K)**

CREATE DATABASE class4;

**3-** [**https://www.postgresqltutorial.com/postgresql-sample-database/**](https://www.postgresqltutorial.com/postgresql-sample-database/) **gidiş ve ER modelini kullanarak. Tablolar arasindaki en az 5 iliskiyi yazin.(Hangi tablodaki arasinda ne tur bir iliski var)**

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

Yapıldı

**5- ER modeldeki tablolardan 3 tanesini K olusturun.**

**CREATE TABLE category**

(category\_id integer PRIMARY KEY NOT NULL,

name character varying(15) NOT NULL,

last\_update integer )

**CREATE TABLE film\_category**

(film\_id integer PRIMARY KEY NOT NULL,

category\_id character varying NOT NULL,

last\_update timestamp NOT NULL)

**CREATE TABLE country**

(country\_id integer PRIMARY KEY NOT NULL,

country character varying (25) NOT NULL,

last\_update timestamp)

**6- ER modeldeki tablolardan 3 tanesini K olusturun.**

(4-5-6. sorulari cozerken toblolar arasındaki iliskileri gozardi olabilirsiniz.

**7- Olusturdugunuz 3 tabloya M ile 5 veri girişi yapin.**

Girildi.

**8- Olusturdugunuz 3 tabloya K ile 5 veri girişi yapin.**

insert into category (category\_id, name, last\_update)

values (3, 'Ford', 2020,

4, 'BMW', 2022,

5, 'Nissan', 2015,

6, 'Toyota', 2009,

7, 'Honda', 2017)

insert into country (country\_id, country)

values (90, ‘Turkey’,

31, ‘Nederland’,

30, ‘Greece’,

49, ‘Germany’,

32, ‘Belguim’)

insert into film\_category (film\_id, category\_id)

values (1, ‘Korku’,

2, Aksiyon,

3, ‘komedi’,

4, ‘Bilim Kurgu’,

5, ‘Müzikal’)

**9- Olusturdugunuz 3 tabloya C ile 5 veri girişi yapin.**

conn = psycopg2.connect(“dbname=class4 user=postgres password=1”)

cur=conn.cursor()

cur.execute(“INSERT INTO category VALUES(%s,%s,%s)”,(8, “Mercedes”,”2000”,

9, “Tofaş”, “1999”,

10, “Kartal”, “1989”,

11,”Opel”,”2005”

12, “Volswogen”, “2013”)

cur.close()

conn.comit()

cur.close()

conn = psycopg2.connect(“dbname=class4 user=postgres password=1”)

cur=conn.cursor()

cur.execute(“INSERT INTO country VALUES(%s,%s)”,(1, “Amerika”,

2, “Rusya”,

3, “Çin”,

4, “Fransa”,

5, “İngiltere”)

cur.close()

conn.comit()

cur.close()

cur.execute(“INSERT INTO film\_categoryVALUES(%s,%s)”,(1, “Komedi”,

2, “Kurgu”,

3, “Deneysel”,

4, “Drama”,

5, “Suç”)

cur.close()

conn.comit()

cur.close()

**10- 3 tablodaki birer veriyi M ile degistirin.**

Yapıldı

**11- 3 tablodaki birer veriyi K ile degistirin.**

UPDATE category SET name=’BMW’ WHERE category\_id=1

UPDATE film\_category SET category\_id=’Aksiyon’ WHERE film\_id=2

UPDATE country SET country ='Yunanistan', WHERE country\_id=31

**12- 3 tablodaki birer veriyi C ile degistirin.**

**conn= psycopg2.connect(“dbname=class4 user=postgres password=1”)**

cur = conn.cursor()

cur.execute(‘’'UPDATE category SET name=’Araba’ WHERE category\_id=1’’’)

cur.close()

conn.commit()

conn.close()

**conn= psycopg2.connect(“dbname=class4 user=postgres password=1”)**

cur = conn.cursor()

cur.execute(‘’'UPDATE film\_category SET category\_id=’Korku’ WHERE film\_id=1’’’)

cur.close()

conn.commit()

conn.close()

**conn= psycopg2.connect(“dbname=class4 user=postgres password=1”)**

cur = conn.cursor()

cur.execute('''UPDATE country

SET country='Turkey' WHERE country\_id=90''')

cur.close()

conn.commit()

conn.close()

**13- 3 tablonun son satirini M ile silinmiş.**

OK

**14- 3 tablonun son satirini K ile silinmiş.**

**DELETE** FROM category

WHERE category\_id = (SELECT category\_id FROM category

ORDER BY category\_id DESC LIMIT 1)

**DELETE** FROM film\_category

WHERE film\_id = (SELECT film\_id FROM film\_category

ORDER BY film\_id DESC LIMIT 1)

**DELETE** FROM country

WHERE country\_id = (SELECT country\_id FROM country

ORDER BY country\_id DESC LIMIT 1)

**15- 3 tablonun son satirini C ile silinmiş.**

**conn= psycopg2.connect(“dbname=class4 user=postgres password=1”)**

cur = conn.cursor()

cur.execute('''DELETE FROM film\_category

WHERE film\_id = (SELECT film\_id FROM film\_category

ORDER BY film\_id DESC LIMIT 1)''')

cur.close()

conn.commit()

conn.close()

**conn= psycopg2.connect(“dbname=class4 user=postgres password=1”)**

cur = conn.cursor()

cur.execute('''DELETE FROM country

WHERE country\_id = (SELECT country\_id FROM country

ORDER BY country\_id DESC LIMIT 1)''')

cur.close()

conn.commit()

conn.close()

**conn= psycopg2.connect(“dbname=class4 user=postgres password=1”)**

cur = conn.cursor()

cur.execute('''DELETE FROM category

WHERE category \_id = (SELECT category \_id FROM category

ORDER BY category \_id DESC LIMIT 1)''')

cur.close()

conn.commit()

conn.close()

**16- 1 tabloyu M ile silindi.**

OK

**17- 1 tabloyu K ile silinmiş.**

DROP TABLE category

**18- 1 tabloyu C ile silinmiş.**

conn= psycopg2.connect(“dbname=class4 user=postgres password=1”)

cur = conn.cursor()

cur.execute('''DROP TABLE country’’’)

cur.close()

conn.commit()

conn.close()

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

CREATE TABLE actor\_new AS

SELECT actor\_id,first\_name,last\_name FROM actor

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

conn= psycopg2.connect(“dbname=class4 user=postgres password=1”)

cur = conn.cursor()

cur.execute('''CREATE TABLE newfilm\_actor AS

SELECT actor\_id, film\_id FROM film\_actor ''')

cur.close()

conn.commit()

conn.close()

**21- Tablolardan 1 tanesini M ile truncate edin.**

OK

**22- Tablolardan 1 tanesini K ile truncate edin.**

TRUNCATE TABLE newfilm\_actor

**23- Tablolardan 1 tanesini C ile truncate edin.**

conn= psycopg2.connect(“dbname=class4 user=postgres password=1”)

cur = conn.cursor()

cur.execute('''TRUNCATE TABLE actor\_new''')

cur.close()

conn.commit()

conn.close()

**24- Kesil edilmis tablolari M ile silinmiş.**

OK

**25- 2 tabloyu K ile silindi.**

DROP TABLE film\_actor

DROP TABLE actor

**26- 2 tabloyu silin.**

conn= psycopg2.connect(“dbname=class4 user=postgres password=1”)

cur = conn.cursor()

cur.execute(‘’’DROP TABLE city’’’)

cur.execute(‘’’DROP TABLE language’’’)

cur.close()

conn.commit()

conn.close()

**27- Elimizde veri olan 1 tablo kalmis olmasi lazim. Tabloyu csv olarak bilgisayarınızı yukleyin.**

OK

**28- Postgresql diziuzundeki son tabloyu da K ile silinmiş.**

DROP TABLE inventory

**29- Bilgisayarınızdaki csv yi arayuze ithalat edin.**

OK

**30- İthalatın bu tabloyu C ile silinmiş.**

conn= psycopg2.connect(“dbname=class4 user=postgres password=1”)

cur = conn.cursor()

cur.execute(‘’’DROP TABLE inventory’’)

cur.close()

conn.commit()

conn.close()

**31-** [**https://www.postgresqltutorial.com/postgresql-sample-database/**](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 tablodaki hangi kolonun PK ve FK olduğunu yazin. sorgular? (Asagi'deki sorularıin sayfasınıini ve bu cedakibi bulurken kullandiginiz kodlari yazin)**

TABLE 'city' --> city\_id-pk; country\_id-fk

TABLE 'customer' --> customer\_id-pk; address\_id-fk

TABLE 'film' --> film\_id-pk; language\_id-fk

TABLE 'actor' --> actor\_id-pk

TABLE 'address' --> address\_id-pk; city\_id-fk

**33- Aksiyon filmlerinin ortalama suresi ne kadar?**

SELECT avg(length) FROM film

WHERE film\_id IN (SELECT film\_id FROM film\_category

WHERE category\_id=(SELECT category\_id FROM category WHERE name='Action'))

**34- En cok personel olan mağaza mı?**

SELECT store\_id, COUNT(store\_id) FROM customer GROUP BY store\_id ORDER BY COUNT(store\_id) DESC LIMIT 1

**35- 'Gene Willis' adli aktörün filmlerin reytingi nedir?**

SELECT rating FROM film WHERE

film\_id IN (SELECT film\_id FROM film\_actor WHERE

actor\_id=(SELECT actor\_id FROM actor WHERE

first\_name='Gene' AND last\_name='Willis'))

**36- Aktif müşteri sayisi nedir?**

SELECT count(\*) FROM customer WHERE active=1

**37-'C' harfiyle baslayan müziktir?**

SELECT title FROM film WHERE title LIKE 'C%'

**38- 4$ den az odeme yapan musterilerin e-posta edresleri nedir?**

SELECT email FROM customer WHERE

customer\_id IN (SELECT customer\_id FROM payment WHERE amount<4 )

**39- Moskova'da ikamet eden personel ve müşteri tablosu? (sadece isim/soyisim sütün olsun)**

SELECT first\_name, last\_name FROM customer WHERE

address\_id IN (SELECT address\_id FROM address WHERE

city\_id = (SELECT city\_id FROM city WHERE city='Moscow'))

**40- En az kiralanan 5 film?**

SELECT title FROM film WHERE film\_id IN

(SELECT film\_id FROM inventory GROUP BY film\_id ORDER BY count(\*) LIMIT 5)

**41- 2006 yilinda yayinlanan film filmlerdir?**

SELECT title FROM film WHERE release\_year=2006 and

language\_id = (SELECT language\_id FROM language WHERE name='English')