



BLM3041 Veritabanı Yönetimi Proje Ödevi

Konu: Tapu Müdürlüğü Bilgi Sistemi

Hazırlayanlar:

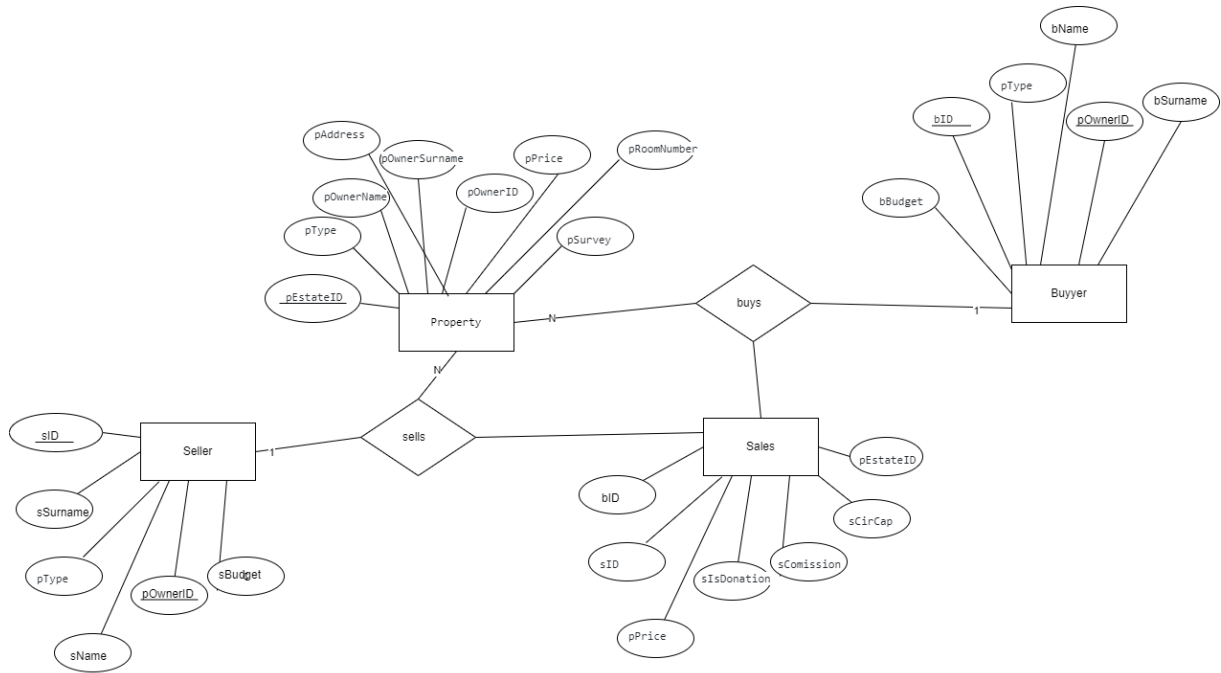
22501072 Ali Aydın Küçükçöllü

21501119 Çağrı Hakkoymaz

Ç19015045 Zeynep Özkayıkcı

G33

ER Diyagramı



Tablolar

Property Tablosu

	ptype character varying (25)	pownername character varying (30)	powersurname character varying (30)	pownerid bigint	pprice bigint	proomnumber bigint	psurvey bigint	paddress character varying (40)	pestatid [PK] bigint
1	Office	Jack	Nicholson	22054045	1350000	14	200	London,England	2892636297
2	Apartment	Benedict	Cumberbatch	12054042	3002300	5	250	Istanbul,Turkey	9284638711
3	Storehouse	Leonardo	DiCaprio	42054545	980000	[null]	1000	London,England	2846304008
4	Office	Christian	Bale	25089404	769000	8	400	Paris,France	3645938383
5	Storehouse	Natalie	Portman	14254054	450320	[null]	2500	Tokyo,Japan	6732838470
6	Apartment	Lena	Headey	10283892	320000	4	280	Istanbul,Turkey	4523843222
7	Apartment	Robert	Downey	91283765	14000000	4	350	San Francisco,USA	1823843222
8	Office	Elijah	Wood	19383422	340000	15	340	New York,USA	7523843222
9	Office	Tom	Hanks	18034035	340985	6	340	Seattle,USA	5632174590
10	Storehouse	Benedict	Cumberbatch	12054042	320000	[null]	500	San Francisco,USA	6784356411
11	Apartment	Scarlett	Johansson	9983462	1000000	3	180	London,England	1524386944
12	Office	Megan	Fox	86654042	4360000	60	4590	Tokyo,Japan	1439638466
13	Storehouse	Emma	Watson	12872350	5000000	[null]	430	Berlin,Germany	1427532099
14	Apartment	Emilia	Clarke	19283422	4000000	5	400	New York,USA	1533843222
15	Apartment	Emilia	Clarke	19283422	6800000	8	550	New York,USA	3452617843
16	Storehouse	Jennifer	Lawrence	21285422	8700000	[null]	680	New York,USA	1523843222
17	Office	Audrey	Tautou	19280923	680000	32	388	San Diego,USA	1356843222
18	Apartment	Lena	Headey	10283892	540000	2	100	Paris,France	8764037821
19	Apartment	Tom	Hanks	18034035	4000000	3	105	New York,USA	3209403233

Seller Tablosu

	sname character varying (30)	ssurname character varying (30)	sid [PK] bigint	ptype character varying (25)	pownerid [PK] bigint	sbudget bigint
1	Tom	Hanks	16034035	Apartment	18034035	6000000
2	Jack	Nicholson	13034036	Office	22054045	80000
3	Benedict	Cumberbatch	56034523	Storehouse	12054042	45000
4	Leonardo	DiCaprio	74360990	Storehouse	42054545	1000000
5	Christian	Bale	56238776	Office	25089404	280000
6	Natalie	Portman	19283432	Storehouse	14254054	35000
7	Scarlett	Johansson	13245321	Apartment	9983462	8300000
8	Megan	Fox	17716556	Office	86654042	100000
9	Emma	Watson	9098735	Storehouse	12872350	2000000
10	Emilia	Clarke	23432166	Apartment	19283422	540000

Buyyer Tablosu

	bname character varying (30)	bsurname character varying (30)	bid [PK] bigint	ptype character varying (25)	pownerid [PK] bigint	bbudget bigint
1	Leonhard	Euler	24939461	Apartment	18034035	12000000
2	Marie	Curie	2184646	Office	22054045	200000
3	Nikola	Tesla	98327676	Storehouse	12054042	450000
4	Linus	Torvalds	16243215	Storehouse	42054545	3200
5	Ada	Lovelace	67435643	Office	25089404	0
6	Alan	Turing	98765423	Storehouse	14254054	2000
7	Bjarne	Stroustrup	6785643	Apartment	9983462	5000000
8	Richard	Stallman	23421621	Office	86654042	110000
9	Dennis	Ritchie	98656774	Storehouse	12872350	650430
10	William	Shockley	31093290	Apartment	19283422	4630000

Sales tablosu

	bid bigint	sid bigint	pprice bigint	scomission bigint	scircap bigint	sisdonation boolean	pestatid bigint
1	24939461	16034035	0	4000	1000	true	5632174590
2	2184646	13034036	1350000	5400	600	false	2892636297
3	98327676	56034523	0	10000	350	true	6784356411
4	16243215	74360990	980000	8600	200	false	2846304008
5	67435643	56238776	0	1300	460	true	3645938383
6	98765423	19283432	450320	12000	940	false	6732838470
7	6785643	13245321	1000000	4380	330	false	1524386944
8	23421621	17716556	0	5250	245	true	1439638466
9	98656774	9098735	5000000	9000	465	false	1427532099
10	31093290	23432166	4000000	7500	250	false	1533843222

KOD

1. 4 tablo ve verileri.

tapu_db klasörü içerisinde yer alan database klasöründe yer almaktadır.

- tapu_db_schema.sql (Tablolar)
- tapu_db_data.sql (Veriler)

2. primary key ve foreign key kısıtları

tapu_db_schema.sql içerisinde yer alan tablolarda belirtilmiştir.

Sales tablosundaki foreign key kısıtı gösterimi:

```
pEstateID          bigint references property(pEstateID)
```

Sales tablosundaki primary key kısıtı gösterimi:

```
UNIQUE(pEstateID)
```

Seller tablosundaki primary key kısıtı gösterimi:

```
primary key(sID, pOwnerID)
```

4. insert, update ve delete işlemleri

Arayüz üzerinden yapılabilmektedir. gui.py dosyasında da gerekli fonksiyonlarda yer almaktadır. Aşağıda da kullanılan kod blokları yer almaktadır. %s ile görülen yerlere arayüz üzerinden parametrik değerler atanmaktadır.

Insert

```
INSERT INTO property
(pType,pOwnerName,pOwnerSurname,pOwnerID,pPrice,pRoomNumber,pSurvey,pAddress,p
EstateID) VALUES (%s,%s,%s,%s,%s,%s,%s,%s,%s)
```

Update

```
if input1 == "pOwnerName":
```

```
    query = """UPDATE property SET pOwnerName=%s where
pEstateID=%s"""
```

```
    elif input1 == "pOwnerSurname":
```

```
        query = """UPDATE property SET pOwnerSurname=%s where
pEstateID=%s"""
```

```
    elif input1 == "pOwnerID":
```

```
        query = """UPDATE property SET pOwnerID=%s where
pEstateID=%s"""
```

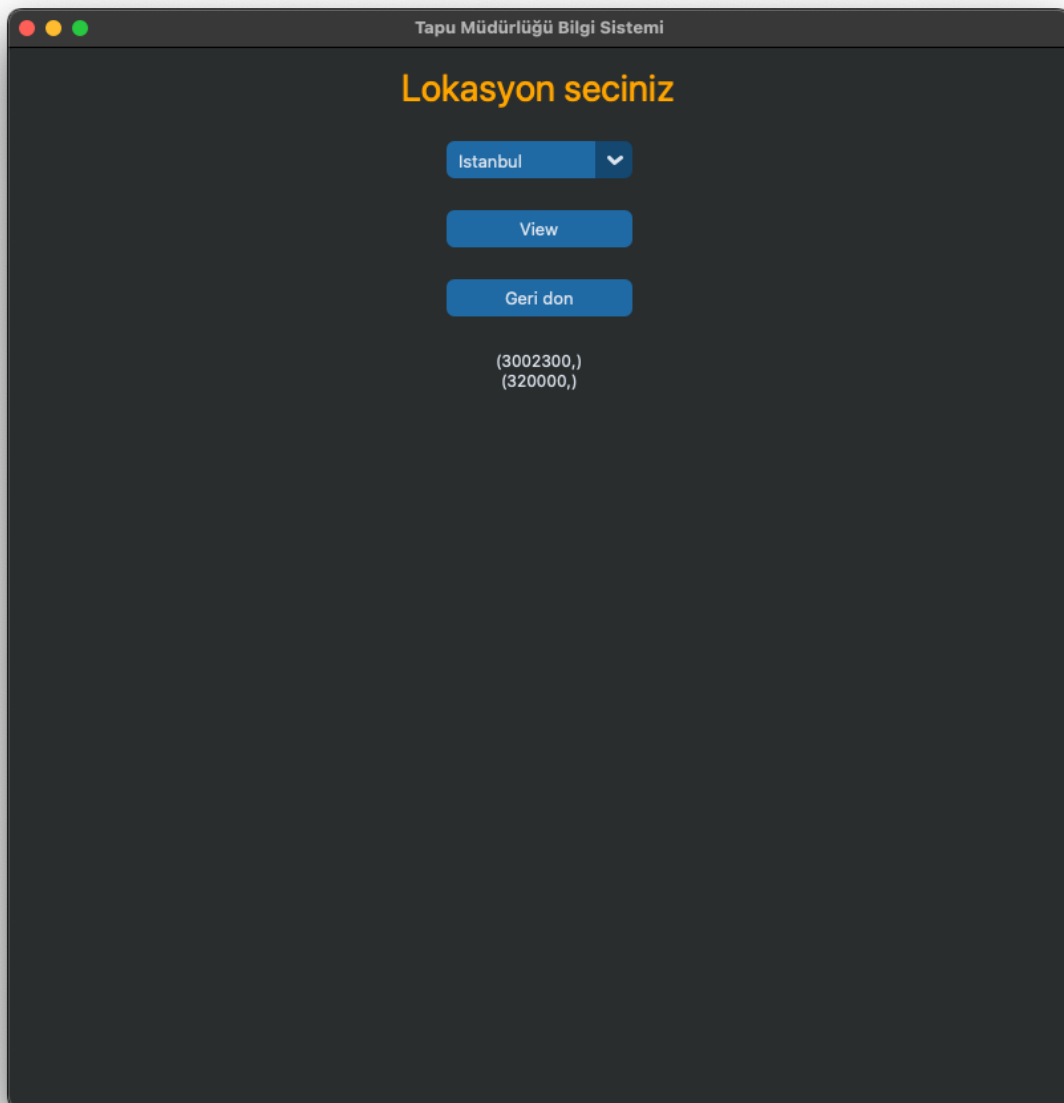
```
    elif input1 == "pPrice":
```

```
query = """"UPDATE property SET pPrice=%s where pEstateID=%s""""  
else:  
    print("error")
```

Delete

```
DELETE FROM property WHERE pestateid = %s",(deletion,)
```

5. Arayüzden girilen ile göre fiyatlar listelenmektedir.



6. Kullanıcı tarafından yazılan şehirlerdeki evlerin fiyatını veren view

```
CREATE VIEW places_money AS  
SELECT pPrice  
FROM sales  
WHERE pAdress="%user entry%"
```

7. Sequence oluşturuldu ve arayüzden yapılacak insert ile ilgili sütundaki değerlerin otomatik olarak atanması sağlandı

```
CREATE SEQUENCE sequence_1  
MINVALUE 7  
MAXVALUE 77  
INCREMENT BY 1
```

```
INSERT INTO buyyer(bname,bSurname,ssn) VALUES ('Johnny','Depp', nextval('sequence_1'))
```

8.UNION veya INTERSECT veya EXCEPT kullanımı

Sistemde kayıtlı olan tüm kişileri gösteren sorgu için UNION kullanılmıştır.

```
select bname from buyyer  
union  
select sname from seller
```

9.Aggregate fonksiyon ve having ifadesi kullanımı.

Sistemde kayıtlı olan evleri oda sayılarına göre gruplayarak her oda sayısı için ortalama fiyatı veren sorguda HAVING ifadesi ve aggregate function kullanılmıştır.

```
select roomnumber,avg(pprice)  
from property  
GROUP BY roomnumber  
HAVING roomnumber IS NOT NULL
```

10.Functions

Fonksiyon 1

Kullanıcıdan minimum oda sayısı ve yüzölçümü verileri alınarak kullanıcıya bu şartları sağlayan evlerin ortalama fiyatını döndülür.

```
CREATE OR REPLACE FUNCTION Avg_PRICE(minroomnumber bigint,minsurvey bigint)
RETURNS bigint
LANGUAGE PLPGSQL
AS $$
DECLARE flat bigint;
BEGIN
select avg(pprice) into flat
from property
where property.proomnumber >=minroomnumber and property.psurvey>= minsurvey;
RETURN flat;
END;
$$
```

Fonksiyon 2

Kullanıcıdan minimum oda sayısı ve yüzölçümü verileri alınarak kullanıcıya bu şartları sağlayan evlerin tablosu döndülür.

```
CREATE OR REPLACE FUNCTION Good_Flat(minroomnumber bigint,minsurvey bigint)
RETURNS TABLE(price bigint,roomnumber bigint,survey bigint )
LANGUAGE PLPGSQL
AS $$
BEGIN
RETURN QUERY SELECT pprice,proomnumber,psurvey
from property
where property.proomnumber >=minroomnumber and property.psurvey>= minsurvey;
END;
$$
```


Fonksiyon 3

Satın alım yapan kişilerin listesini döndürür.

CREATE OR REPLACE FUNCTION declare_buyers()

RETURNS TRIGGER AS

\$\$

DECLARE

cseller_rec record;

cseller_cur CURSOR for

SELECT bid, bname

FROM buyyer;

BEGIN

OPEN cseller_cur;

LOOP

FETCH cseller_cur into cseller_rec;

EXIT WHEN not found;

raise notice '% %', cseller_rec.bid,cseller_rec.bname;

END LOOP;

close cseller_cur;

RETURN void;

END;

\$\$

language plpgsql;

Fonksiyon 4

Satış tablosunda bulunan satışlar için satıcı ve alıcının bütçesinden komisyon alır.

```
CREATE OR REPLACE FUNCTION take_commission()
```

```
RETURNS TRIGGER
```

```
LANGUAGE PLPGSQL
```

```
AS $$
```

```
BEGIN
```

```
UPDATE seller SET sbudget=sbudget-pprice*0.5;
```

```
UPDATE buyer SET bbudget=bbudget-pprice*0.5;
```

```
RETURN NEW;
```

```
END;
```

```
$$
```

11.

TRIGGERS

TRIGGER 1

Satış işlemi sonrası kullanıcılardan komisyon alan fonksiyonu çağıran TRIGGER işlemi tanımlanmıştır.

```
CREATE OR REPLACE TRIGGER after_sales
```

```
AFTER INSERT ON sales
```

```
FOR EACH ROW
```

```
EXECUTE PROCEDURE take_commission()
```

TRIGGER 1

Satış alım yapan kişi listesine ekleme yapıldığında satın alan yapan kişilerin listesini döndüren TRIGGER tanımlanmıştır.

```
CREATE OR REPLACE TRIGGER declare_B
```

```
AFTER INSERT ON buyer
```

```
FOR EACH ROW
```

```
EXECUTE PROCEDURE declare_buyers()
```