

26.12.2023

# VERİTABANI YÖNETİM SİSTEMLERİ PROJE ÇALIŞMASI

Dr.Öğr.Üyesi İsmail Öztel

Ali Kerem Kol  
B221210042

[ali.kol@ogr.sakarya.edu.tr](mailto:ali.kol@ogr.sakarya.edu.tr)

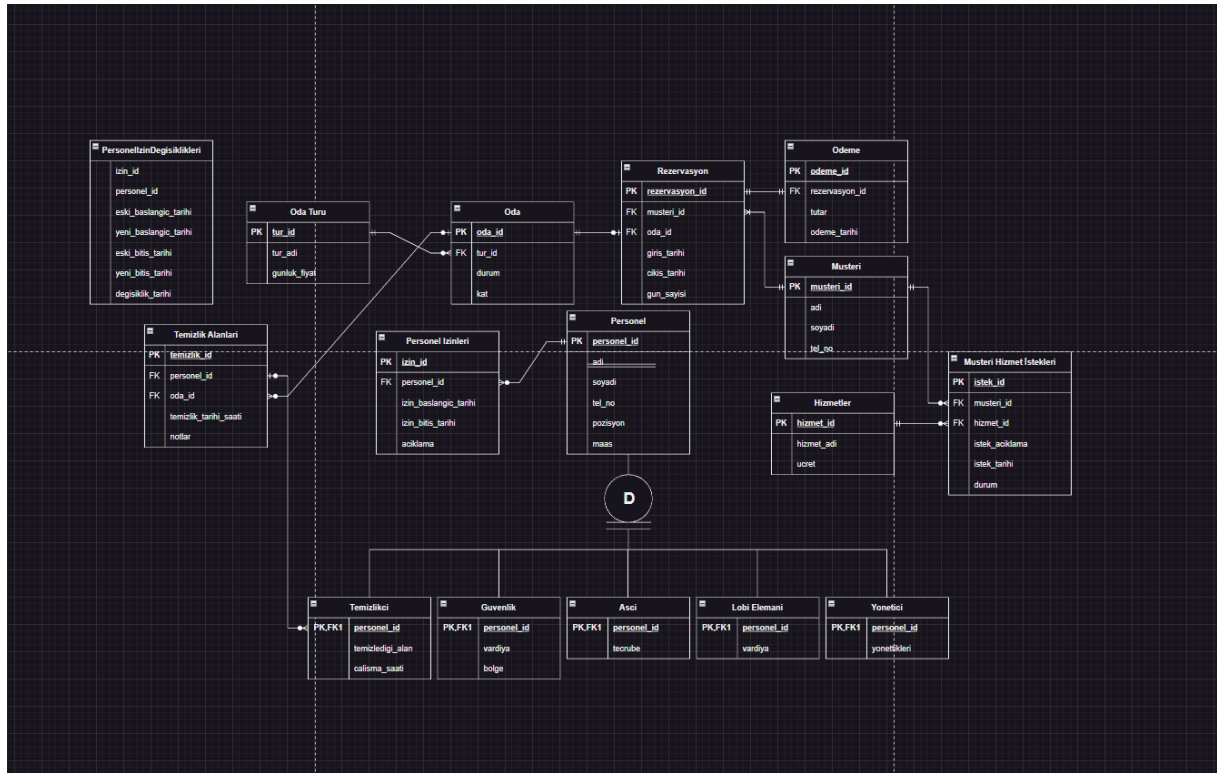
### İş Kuralları:

- 1) Bir personelin hiç izini olmayabilir ya da biden çok sayıda izini olabilir. Bir izin en az veya en çok bir personele ait olmalıdır.
- 2) Bir müşteri en az bir ya da bir den çok rezervasyon yaptırabilir. Bir rezervasyon en az veya en çok bir müşteriye ait olmalıdır.
- 3) Hiçbir müşteri veya çok sayıda müşteri hizmet alabilir. Bir hizmet çok sayıda müşteriye veya hiçbir müşteriye ait olmayabilir.
- 4) Bir rezervasyonun sadece 1 ödemesi olabilir. Bir ödemenin sadece 1 rezervasyonu olur.
- 5) Bir rezervasyonda kesinlikle bir oda bulunmak zorundadır. Bir oda herhangi bir rezervasyona ait olmak zorunda değildir.
- 6) Bir odanın en az veya en çok bir türü olmalıdır. Bir türden hiçbir oda olmayabileceği gibi çok sayıda oda da bulunabilir.
- 7) Bir personel kesinlikle ya temizlikçi ya güvenlik ya aşçı ya lobi elemanı ya da yönetici olabilir.
- 8) Bir temizlikçi hiçbir temizlik alanını temizlemeyebilir veya sadece 1 temizlik alanını temizleyebilir.
- 9) Bir temizlik alanı hiçbir temizlikçi tarafından temizlenmeyebilir veya çok sayıda temizlikçi tarafından temizlenebilir.
- 10) Bir temizlik hiçbir odaya ait olmayabilir veya sadece 1 odaya ait olabilir.
- 11) Bir oda da hiçbir temizlik işlemi olmayabilir veya birden fazla temizlik işlemi olabilir.

### İlişkisel Şema(Metinsel Gösterim):

- 1)Personel(adi:varchar , soyadi:varchar , tel\_no:varchar , pozisyon:varchar , maas:numeric , personel\_id:varchar)
- 2)Personellzinleri(izin\_id:integer , izin\_baslangic\_tarihi:date , izin\_bitis\_tarihi:date , aciklama:varchar , personel\_id:varchar)
- 3)Temizlikci(temizlediği\_alan:varchar , calisma\_saati:integer)
- 4)TemizlikAlanlari(temizlik\_id:integer , oda\_id:integer , temizlik\_tarihi\_saati:timestamp , notlar:text , personel\_id:varchar)
- 5)Guvenlik(vardiya:varchar , bolge : varchar)
- 6)Asci(tecrübe:varchar)
- 7)LobiElemani(vardiya:varchar)
- 8)Yonetici(yonettikleri:varchar)
- 9)Musteri(müşteri\_id:integer , adi:varchar , soyadi:varchar , tel\_no:varchar)
- 10)MusteriHizmetIstekleri(istek\_id:integer ,müşteri\_id:integer , hizmet\_id:integer , istek\_aciklama:text , istek\_tarihi:date , durum:varchar)
- 11)Hizmetler(hizmet\_id:integer , hizmet\_adi:varchar ,ucret:numeric)
- 12)Rezervasyon(rezervasyon\_id:integer ,musteri\_id:integer , oda\_id:integer , giriş\_tarihi:date , cikis\_tarihi:date , gun\_sayisi:interval)
- 13)Oda(oda\_id:integer , tur\_id:integer , durum:varchar , kat:integer)
- 14) OdaTur(tur\_id:integer , tur\_adi:varchar ,gunluk\_fiyat:numeric)
- 15)Odeme(odeme\_id:integer , rezervasyon\_id:integer , tutar:numeric , odeme\_tarihi:date)
- 16)PersonellzinDegisiklikleri(izin\_id:integer , personel\_id:varchar , eski\_baslangic\_tarihi:date , yeni\_baslangic\_tarihi:date , eski\_bitis\_tarihi:date , yeni\_bitis\_tarihi:date , değişiklik\_tarihi:timestamp)

## Varlık Bağıntı Modeli :



## SQL İfadeleri:

### Tablolar:

1)Personel Tablosu:

-- Table: public.Personel

-- DROP TABLE IF EXISTS public."Personel";

CREATE TABLE IF NOT EXISTS public."Personel"

(

adi character varying(50) COLLATE pg\_catalog."default",

soyadi character varying(50) COLLATE pg\_catalog."default",

tel\_no character varying(12) COLLATE pg\_catalog."default",

pozisyon character varying(20) COLLATE pg\_catalog."default",

maas numeric(10,2),

personel\_id character varying(5) COLLATE pg\_catalog."default" NOT NULL,

CONSTRAINT "Personel\_pkey" PRIMARY KEY (personel\_id)

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Personel"

OWNER to postgres;

-- Trigger: personel\_eklendi

-- DROP TRIGGER IF EXISTS personel\_eklendi ON public."Personel";

```
CREATE OR REPLACE TRIGGER personel_eklendi
  AFTER INSERT
  ON public."Personel"
  FOR EACH ROW
  EXECUTE FUNCTION public.personel_eklendi_trigger();
```

```
-- Trigger: personel_sil_trigger
```

```
-- DROP TRIGGER IF EXISTS personel_sil_trigger ON public."Personel";
```

```
CREATE OR REPLACE TRIGGER personel_sil_trigger
  AFTER DELETE
  ON public."Personel"
  FOR EACH ROW
  EXECUTE FUNCTION public.personel_sil_tetikleyici();
```

---

2)Personellzinleri Tablosu:

```
-- Table: public.Personellzinleri
```

```
-- DROP TABLE IF EXISTS public."Personellzinleri";
```

```
CREATE TABLE IF NOT EXISTS public."Personellzinleri"
(
  izin_id integer NOT NULL,
  izin_baslangic_tarihi date,
  izin_bitis_tarihi date,
```

```
aciklama character varying(255) COLLATE pg_catalog."default",
personel_id character varying(5) COLLATE pg_catalog."default" NOT NULL,
CONSTRAINT "Personellzinleri_pkey" PRIMARY KEY (izin_id),
CONSTRAINT "Personellzinleri_personel_id_fkey" FOREIGN KEY (personel_id)
    REFERENCES public."Personel" (personel_id) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE NO ACTION
    NOT VALID
)
```

```
TABLESPACE pg_default;
```

```
ALTER TABLE IF EXISTS public."Personellzinleri"
    OWNER to postgres;
```

```
-- Trigger: personel_izin_takip_trig
```

```
-- DROP TRIGGER IF EXISTS personel_izin_takip_trig ON
public."Personellzinleri";
```

```
CREATE OR REPLACE TRIGGER personel_izin_takip_trig
    AFTER DELETE OR UPDATE
    ON public."Personellzinleri"
    FOR EACH ROW
    EXECUTE FUNCTION public.personel_izin_degisiklik_takip();
```

---

3) Temizlikci Tablosu:

-- Table: public.Temizlikci

-- DROP TABLE IF EXISTS public."Temizlikci";

CREATE TABLE IF NOT EXISTS public."Temizlikci"

(

-- Inherited from table public."Personel": adi character varying(50) COLLATE pg\_catalog."default",

-- Inherited from table public."Personel": soyadi character varying(50) COLLATE pg\_catalog."default",

-- Inherited from table public."Personel": tel\_no character varying(12) COLLATE pg\_catalog."default",

-- Inherited from table public."Personel": pozisyon character varying(20) COLLATE pg\_catalog."default",

-- Inherited from table public."Personel": maas numeric(10,2),

-- Inherited from table public."Personel": personel\_id character varying(5) COLLATE pg\_catalog."default" NOT NULL,

temizledigi\_alan character varying(20) COLLATE pg\_catalog."default",

calisma\_saati integer,

CONSTRAINT "Temizlikci\_pkey" PRIMARY KEY (personel\_id),

CONSTRAINT "Temizlikci\_personel\_id\_fkey" FOREIGN KEY (personel\_id)

REFERENCES public."Personel" (personel\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION

)

INHERITS (public."Personel")

TABLESPACE pg\_default;



```
ALTER TABLE IF EXISTS public."Temizlikci"
```

```
OWNER to postgres;
```

---

4)TemizlikAlanlari Tablosu:

```
-- Table: public.TemizlikAlanlari
```

```
-- DROP TABLE IF EXISTS public."TemizlikAlanlari";
```

```
CREATE TABLE IF NOT EXISTS public."TemizlikAlanlari"
```

```
(
```

```
    temizlik_id integer NOT NULL,
```

```
    oda_id integer,
```

```
    temizlik_tarihi_saati timestamp without time zone,
```

```
    notlar text COLLATE pg_catalog."default",
```

```
    personel_id character varying COLLATE pg_catalog."default" NOT NULL,
```

```
    CONSTRAINT "TemizlikAlanlari_pkey" PRIMARY KEY (temizlik_id),
```

```
    CONSTRAINT "TemizlikAlanlari_oda_id_fkey" FOREIGN KEY (oda_id)
```

```
        REFERENCES public."Oda" (oda_id) MATCH SIMPLE
```

```
        ON UPDATE NO ACTION
```

```
        ON DELETE NO ACTION,
```

```
    CONSTRAINT "TemizlikAlanlari_personel_id_fkey" FOREIGN KEY (personel_id)
```

```
        REFERENCES public."Temizlikci" (personel_id) MATCH SIMPLE
```

```
        ON UPDATE NO ACTION
```

```
        ON DELETE NO ACTION
```

```
        NOT VALID
```

```
)
```

```
TABLESPACE pg_default;
```

```
ALTER TABLE IF EXISTS public."TemizlikAlanlari"
```

```
OWNER to postgres;
```

---

5)Guvencik Tablosu:

```
-- Table: public.Guvenlik
```

```
-- DROP TABLE IF EXISTS public."Guvenlik";
```

```
CREATE TABLE IF NOT EXISTS public."Guvenlik"
```

```
(
```

```
    -- Inherited from table public."Personel": adi character varying(50) COLLATE  
pg_catalog."default",
```

```
    -- Inherited from table public."Personel": soyadi character varying(50)  
COLLATE pg_catalog."default",
```

```
    -- Inherited from table public."Personel": tel_no character varying(12)  
COLLATE pg_catalog."default",
```

```
    -- Inherited from table public."Personel": pozisyon character varying(20)  
COLLATE pg_catalog."default",
```

```
    -- Inherited from table public."Personel": maas numeric(10,2),
```

```
    -- Inherited from table public."Personel": personel_id character varying(5)  
COLLATE pg_catalog."default" NOT NULL,
```

```
    vardiya character varying(20) COLLATE pg_catalog."default",
```

```
    bolge character varying(20) COLLATE pg_catalog."default",
```

```
    CONSTRAINT "Guvenlik_pkey" PRIMARY KEY (personel_id),
```

```
    CONSTRAINT "Guvenlik_personel_id_fkey" FOREIGN KEY (personel_id)
```

```
REFERENCES public."Personel" (personel_id) MATCH SIMPLE
```

```
        ON UPDATE NO ACTION
        ON DELETE NO ACTION
    )
    INHERITS (public."Personel")
```

```
TABLESPACE pg_default;
```

```
ALTER TABLE IF EXISTS public."Guvenlik"
    OWNER to postgres;
```

---

6)Asci Tablosu:

```
-- Table: public.Asci
```

```
-- DROP TABLE IF EXISTS public."Asci";
```

```
CREATE TABLE IF NOT EXISTS public."Asci"
```

```
(
    -- Inherited from table public."Personel": adi character varying(50) COLLATE
pg_catalog."default",
    -- Inherited from table public."Personel": soyadi character varying(50)
COLLATE pg_catalog."default",
    -- Inherited from table public."Personel": tel_no character varying(12)
COLLATE pg_catalog."default",
    -- Inherited from table public."Personel": pozisyon character varying(20)
COLLATE pg_catalog."default",
    -- Inherited from table public."Personel": maas numeric(10,2),
    -- Inherited from table public."Personel": personel_id character varying(5)
COLLATE pg_catalog."default" NOT NULL,
    tecrube character varying(20) COLLATE pg_catalog."default",
```

```
CONSTRAINT "Asci_pkey" PRIMARY KEY (personel_id),
CONSTRAINT "Asci_personel_id_fkey" FOREIGN KEY (personel_id)
REFERENCES public."Personel" (personel_id) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION
)
INHERITS (public."Personel")
```

```
TABLESPACE pg_default;
```

```
ALTER TABLE IF EXISTS public."Asci"
OWNER to postgres;
```

---

7)LobiElemani Tablosu:

```
-- Table: public.LobiElemani
```

```
-- DROP TABLE IF EXISTS public."LobiElemani";
```

```
CREATE TABLE IF NOT EXISTS public."LobiElemani"
```

```
(
```

```
-- Inherited from table public."Personel": adi character varying(50) COLLATE
pg_catalog."default",
```

```
-- Inherited from table public."Personel": soyadi character varying(50)
COLLATE pg_catalog."default",
```

```
-- Inherited from table public."Personel": tel_no character varying(12)
COLLATE pg_catalog."default",
```

```
-- Inherited from table public."Personel": pozisyon character varying(20)
COLLATE pg_catalog."default",
```

```

-- Inherited from table public."Personel": maas numeric(10,2),
-- Inherited from table public."Personel": personel_id character varying(5)
COLLATE pg_catalog."default" NOT NULL,
vardiya character varying(20) COLLATE pg_catalog."default",
CONSTRAINT "LobiElemani_pkey" PRIMARY KEY (personel_id),
CONSTRAINT "LobiElemani_personel_id_fkey" FOREIGN KEY (personel_id)
REFERENCES public."Personel" (personel_id) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION
)
INHERITS (public."Personel")

```

```

TABLESPACE pg_default;

```

```

ALTER TABLE IF EXISTS public."LobiElemani"
OWNER to postgres;

```

---

8)Yonetici Tablosu:

```

-- Table: public.Yonetici

```

```

-- DROP TABLE IF EXISTS public."Yonetici";

```

```

CREATE TABLE IF NOT EXISTS public."Yonetici"

```

```

(
-- Inherited from table public."Personel": adi character varying(50) COLLATE
pg_catalog."default",
-- Inherited from table public."Personel": soyadi character varying(50)
COLLATE pg_catalog."default",

```

```

-- Inherited from table public."Personel": tel_no character varying(12)
COLLATE pg_catalog."default",

-- Inherited from table public."Personel": pozisyon character varying(20)
COLLATE pg_catalog."default",

-- Inherited from table public."Personel": maas numeric(10,2),

-- Inherited from table public."Personel": personel_id character varying(5)
COLLATE pg_catalog."default" NOT NULL,

yonettikleri character varying(50) COLLATE pg_catalog."default",
CONSTRAINT "Yonetici_pkey" PRIMARY KEY (personel_id),
CONSTRAINT "Yonetici_personel_id_fkey" FOREIGN KEY (personel_id)
REFERENCES public."Personel" (personel_id) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION
)
INHERITS (public."Personel")

```

```

TABLESPACE pg_default;

```

```

ALTER TABLE IF EXISTS public."Yonetici"
OWNER to postgres;

```

---

9)Musteri Tablosu:

```

-- Table: public.Musteri

```

```

-- DROP TABLE IF EXISTS public."Musteri";

```

```

CREATE TABLE IF NOT EXISTS public."Musteri"
(

```

```
musteri_id integer NOT NULL,  
adi character varying(15) COLLATE pg_catalog."default",  
soyadi character varying(15) COLLATE pg_catalog."default",  
tel_no character varying(12) COLLATE pg_catalog."default",  
CONSTRAINT "Musteri_pkey" PRIMARY KEY (musteri_id)  
)
```

```
TABLESPACE pg_default;
```

```
ALTER TABLE IF EXISTS public."Musteri"  
OWNER to postgres;
```

---

10)MusteriHizmetIstekleri Tablosu:

```
-- Table: public.MusteriHizmetIstekleri
```

```
-- DROP TABLE IF EXISTS public."MusteriHizmetIstekleri";
```

```
CREATE TABLE IF NOT EXISTS public."MusteriHizmetIstekleri"  
(  
    istek_id integer NOT NULL,  
    muster_i_id integer,  
    hizmet_id integer,  
    istek_aciklama text COLLATE pg_catalog."default",  
    istek_tarihi date,  
    durum character varying(10) COLLATE pg_catalog."default",  
    CONSTRAINT "MusteriHizmetIstekleri_pkey" PRIMARY KEY (istik_id),  
    CONSTRAINT "MusteriHizmetIstekleri_hizmet_id_fkey" FOREIGN KEY  
(hizmet_id)
```

```
REFERENCES public."Hizmetler" (hizmet_id) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION,
CONSTRAINT "MusteriHizmetIstekleri_musteri_id_fkey" FOREIGN KEY
(musteri_id)
REFERENCES public."Musteri" (musteri_id) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION
)
```

```
TABLESPACE pg_default;
```

```
ALTER TABLE IF EXISTS public."MusteriHizmetIstekleri"
OWNER to postgres;
```

```
-- Trigger: muster_i_hizmet_istekleri_trigger
```

```
-- DROP TRIGGER IF EXISTS muster_i_hizmet_istekleri_trigger ON
public."MusteriHizmetIstekleri";
```

```
CREATE OR REPLACE TRIGGER muster_i_hizmet_istekleri_trigger
AFTER INSERT
ON public."MusteriHizmetIstekleri"
FOR EACH ROW
EXECUTE FUNCTION public.muster_i_hizmet_istekleri_tetikleyici();
```

---

11) Hizmetler Tablosu:

```
-- Table: public.Hizmetler
```



```
-- DROP TABLE IF EXISTS public."Hizmetler";
```

```
CREATE TABLE IF NOT EXISTS public."Hizmetler"
```

```
(  
    hizmet_id integer NOT NULL,  
    hizmet_adi character varying(25) COLLATE pg_catalog."default",  
    ucuret numeric(10,2),  
    CONSTRAINT "Hizmetler_pkey" PRIMARY KEY (hizmet_id)  
)
```

```
TABLESPACE pg_default;
```

```
ALTER TABLE IF EXISTS public."Hizmetler"
```

```
    OWNER to postgres;
```

---

12)Rezervasyon Tablosu:

```
-- Table: public.Rezervasyon
```

```
-- DROP TABLE IF EXISTS public."Rezervasyon";
```

```
CREATE TABLE IF NOT EXISTS public."Rezervasyon"
```

```
(  
    rezervasyon_id integer NOT NULL,  
    musteri_id integer NOT NULL,  
    oda_id integer NOT NULL,  
    giris_tarihi date,
```

```
cikis_tarihi date,  
gun_sayisi interval(6),  
CONSTRAINT "Rezervasyon_pkey" PRIMARY KEY (rezervasyon_id),  
CONSTRAINT "Rezervasyon_musteri_id_fkey" FOREIGN KEY (musteri_id)  
    REFERENCES public."Musteri" (musteri_id) MATCH SIMPLE  
    ON UPDATE NO ACTION  
    ON DELETE CASCADE,  
CONSTRAINT "Rezervasyon_oda_id_fkey" FOREIGN KEY (oda_id)  
    REFERENCES public."Oda" (oda_id) MATCH SIMPLE  
    ON UPDATE NO ACTION  
    ON DELETE NO ACTION  
)
```

```
TABLESPACE pg_default;
```

```
ALTER TABLE IF EXISTS public."Rezervasyon"  
    OWNER to postgres;
```

```
-- Trigger: oda_durum_trig
```

```
-- DROP TRIGGER IF EXISTS oda_durum_trig ON public."Rezervasyon";
```

```
CREATE OR REPLACE TRIGGER oda_durum_trig  
    AFTER INSERT OR DELETE  
    ON public."Rezervasyon"  
    FOR EACH ROW
```

```
EXECUTE FUNCTION public.oda_durum_degisikligi();
```

---

13)Oda Tablosu:

-- Table: public.Oda

-- DROP TABLE IF EXISTS public."Oda";

CREATE TABLE IF NOT EXISTS public."Oda"

(

oda\_id integer NOT NULL,

tur\_id integer,

durum character varying(20) COLLATE pg\_catalog."default",

kat integer,

CONSTRAINT "Oda\_pkey" PRIMARY KEY (oda\_id),

CONSTRAINT "Oda\_tur\_id\_fkey" FOREIGN KEY (tur\_id)

REFERENCES public."OdaTur" (tur\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Oda"

OWNER to postgres;

---

14)OdaTur Tablosu:

-- Table: public.OdaTur

```
-- DROP TABLE IF EXISTS public."OdaTur";
```

```
CREATE TABLE IF NOT EXISTS public."OdaTur"
```

```
(  
    tur_id integer NOT NULL,  
    tur_adi character varying(50) COLLATE pg_catalog."default",  
    gunluk_fiyat numeric,  
    CONSTRAINT "OdaTur_pkey" PRIMARY KEY (tur_id)  
)
```

```
TABLESPACE pg_default;
```

```
ALTER TABLE IF EXISTS public."OdaTur"
```

```
    OWNER to postgres;
```

---

15)Odeme Tablosu:

```
-- Table: public.Odeme
```

```
-- DROP TABLE IF EXISTS public."Odeme";
```

```
CREATE TABLE IF NOT EXISTS public."Odeme"
```

```
(  
    odeme_id integer NOT NULL,  
    rezervasyon_id integer NOT NULL,  
    tutar numeric(10,2),  
    odeme_tarihi date,  
    CONSTRAINT "Odeme_pkey" PRIMARY KEY (odeme_id),
```

```
CONSTRAINT "Odeme_rezervasyon_id_fkey" FOREIGN KEY (rezervasyon_id)
REFERENCES public."Rezervasyon" (rezervasyon_id) MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE CASCADE
)
```

```
TABLESPACE pg_default;
```

```
ALTER TABLE IF EXISTS public."Odeme"
OWNER to postgres;
```

---

16)PersonellzinDegisikleri Tablosu:

-- Table: public.PersonellzinDegisiklikleri

-- DROP TABLE IF EXISTS public."PersonellzinDegisiklikleri";

```
CREATE TABLE IF NOT EXISTS public."PersonellzinDegisiklikleri"
```

```
(
    izin_id integer,
    personel_id character varying(5) COLLATE pg_catalog."default",
    eski_baslangic_tarihi date,
    yeni_baslangic_tarihi date,
    eski_bitis_tarihi date,
    yeni_bitis_tarihi date,
    degisiklik_tarihi timestamp without time zone
)
```

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."PersonellzinDegisiklikleri"

OWNER to postgres;

## **Fonksiyonlar:**

1) en\_yuksek\_fiyatli\_hizmet() : "Hizmetler" tablosundan en yüksek ücrete sahip olan hizmetin adını (hizmet\_adi) ve ücretini (ucret) döndüren bir SQL fonksiyonu.

2) en\_yuksek\_fiyatli\_oda\_turu\_bul() : "OdaTur" tablosundan en yüksek günlük fiyata sahip olan oda türünü bulan ve bu türün tüm alanlarını içeren bir satırı döndüren bir SQL fonksiyonu.

3) en\_yuksek\_maasli\_personel() : "Personel" tablosundan en yüksek maaşa sahip olan personelin bazı bilgilerini döndüren bir SQL fonksiyonu.

4) gunluk\_fiyat\_gunceller() : "OdaTur" tablosundaki bir oda türünün günlük fiyatını güncellemek için kullanılır.

## **TRIGGERLAR:**

1) musteri\_hizmet\_istekleri\_tetikleyici() : Bu tetikleyici, "MusteriHizmetIstekleri" tablosuna yeni bir satır eklediğinde çalışan bir PL/pgSQL tetikleyici fonksiyonudur. Bu fonksiyonun temel işlevi, yeni eklenen hizmet isteği için belirtilen hizmetin ücretini bulmak ve bu ücreti ilgili müşterinin rezervasyonu ile ilişkilendirilmiş "Odeme" tablosundaki tutara eklemektir.

2) oda\_durum\_degisikligi() : Bu tetikleyici fonksiyonu, "Oda" tablosunda bir değişiklik olduğunda çalışan bir PL/pgSQL tetikleyici fonksiyonudur. Temel amacı, "Rezervasyon" tablosunda bir rezervasyon silindiğinde veya müşteri otelden ayrıldığında ilgili odanın durumunu "boş" olarak güncellemek ve yeni bir rezervasyon yapıldığında ilgili odanın durumunu "dolmuş" olarak güncellemektir.

3) personel\_eklendi\_trigger() : Bu tetikleyici fonksiyonu, "Personel" tablosuna yeni bir personel eklendiğinde çalışan bir PL/pgSQL tetikleyici fonksiyonudur. Fonksiyonun temel amacı, eklenen personelin "personel\_id" değerine göre ilgili

alt tabloya (örneğin, "Temizlikci", "Guvencik", "Asci", "LobiElemani", "Yoneticisi") eklenmesidir.

4) personel\_izin\_degisiklik\_takip() : Bu tetikleyici fonksiyonu, "Personellzin" tablosundaki bir izin kaydının güncellendiğinde veya silindiğinde çalışan bir PL/pgSQL tetikleyici fonksiyonudur. Bu fonksiyonun temel amacı, izin değişikliklerini takip etmek ve bu değişiklikleri "PersonellzinDegisiklikleri" tablosuna kaydetmektir.

5) personel\_sil\_tetikleyici() : Bu tetikleyici fonksiyonu, "Personel" tablosundan bir personel kaydı silindiğinde çalışan bir PL/pgSQL tetikleyici fonksiyonudur. Bu fonksiyonun temel amacı, ilgili personelin silinmesi durumunda bazı bağlı tablolarındaki kayıtları da silmektir.

## Ekleme İşlemi:

```
INSERT INTO "Musteri" ("musteri_id","adi", "soyadi", "tel_no")
VALUES (1,'Ali Kerem', 'Kol', '05445620724');
```

Data Output Messages Notifications

INSERT 0 1

Query returned successfully in 42 msec.

Total rows: 1 of 1 Query complete 00:00:00.042 Ln 23, Col 1

	musteri_id [PK] integer	adi character varying (15)	soyadi character varying (15)	tel_no character varying (12)
1	1	Ali Kerem	Kol	05445620724

Total rows: 1 of 1 Query complete 00:00:00.060 Ln 26, Col 1

## Güncelleme İşlemi:

Data Output Messages Notifications

UPDATE 1

Query returned successfully in 47 msec.

Total rows: 0 of 0 Query complete 00:00:00.047 Ln 28, Col 1

```
UPDATE "Musteri" SET adi = 'Mehmet', soyadi = 'Yılmaz',tel_no = '05399692557' WHERE muster_i_d = 1;
```

	musteri_id [PK] integer	adi character varying (15)	soyadi character varying (15)	tel_no character varying (12)
1	1	Mehmet	Yılmaz	05399692557

✓ Successfully run. Total query runtime: 105 msec. 1 row affected.

## Silme İşlemi:

```
DELETE FROM "Musteri" WHERE muster_i_id=1;
```

Data Output	Messages	Notifications
DELETE 1		
Query returned successfully in 45 msec.		
Total rows: 1 of 1    Query complete 00:00:00.045    Ln 28, Col 1		
Data Output	Messages	Notifications
<div><div>muster_i_id [PK] integer</div><div>adi character varying (15)</div><div>soyadi character varying (15)</div><div>tel_no character varying (12)</div></div>		
Total rows: 0 of 0    Query complete 00:00:00.068    Ln 26, Col 1		

## Select İşlemi:

```
select * from "Musteri";
```

Data Output	Messages	Notifications
<div><div>muster_i_id [PK] integer</div><div>adi character varying (15)</div><div>soyadi character varying (15)</div><div>tel_no character varying (12)</div></div>		
1	Mehmet	Yilmaz
2	Ali Kerem	Kol
3	Kaan	Kazgan
4	Volkan	Demirel
Total rows: 4 of 4    Query complete 00:00:00.120    Ln 26, Col 1		

## Uygulamanın Kaynak Kodları:

<https://github.com/Ali-Kerem-Kol/Veri-Tabani-Yonetim-Sistemleri-Proje-Uygulaması>