



SAKARYA
ÜNİVERSİTESİ

SAKARYA ÜNİVERSİTESİ

BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ (İÖ)

VERİTABANI YÖNETİM SİSTEMLERİ DERSİ

PROJE ÖDEVİ

ÖĞRENCİ ADI.....: Serdar Arıcı

ÖĞRENCİ NUMARASI...: G191210020

DERS GRUBU.....: 2B grubu

E-POSTA.....: serdar.arici1@ogr.sakarya.edu.tr

Problemin Tanıtımı

Bilet satış uygulamasında, alınan her bir bilete ait bilgilerin depolanması ve yönetilmesi istenmektedir. Biletlerin kategorileri, sipariş bilgileri, müşteri bilgileri, mekan bilgileri, mekanlarda çalışan personel bilgileri, fatura bilgileri, bilete ait etkinlik ve sanatçı bilgileri, etkinliklere ait mekanlardaki salon bilgileri, adres, il ve ilçe bilgilerinin saklanması istenmektedir.

İş Kuralları

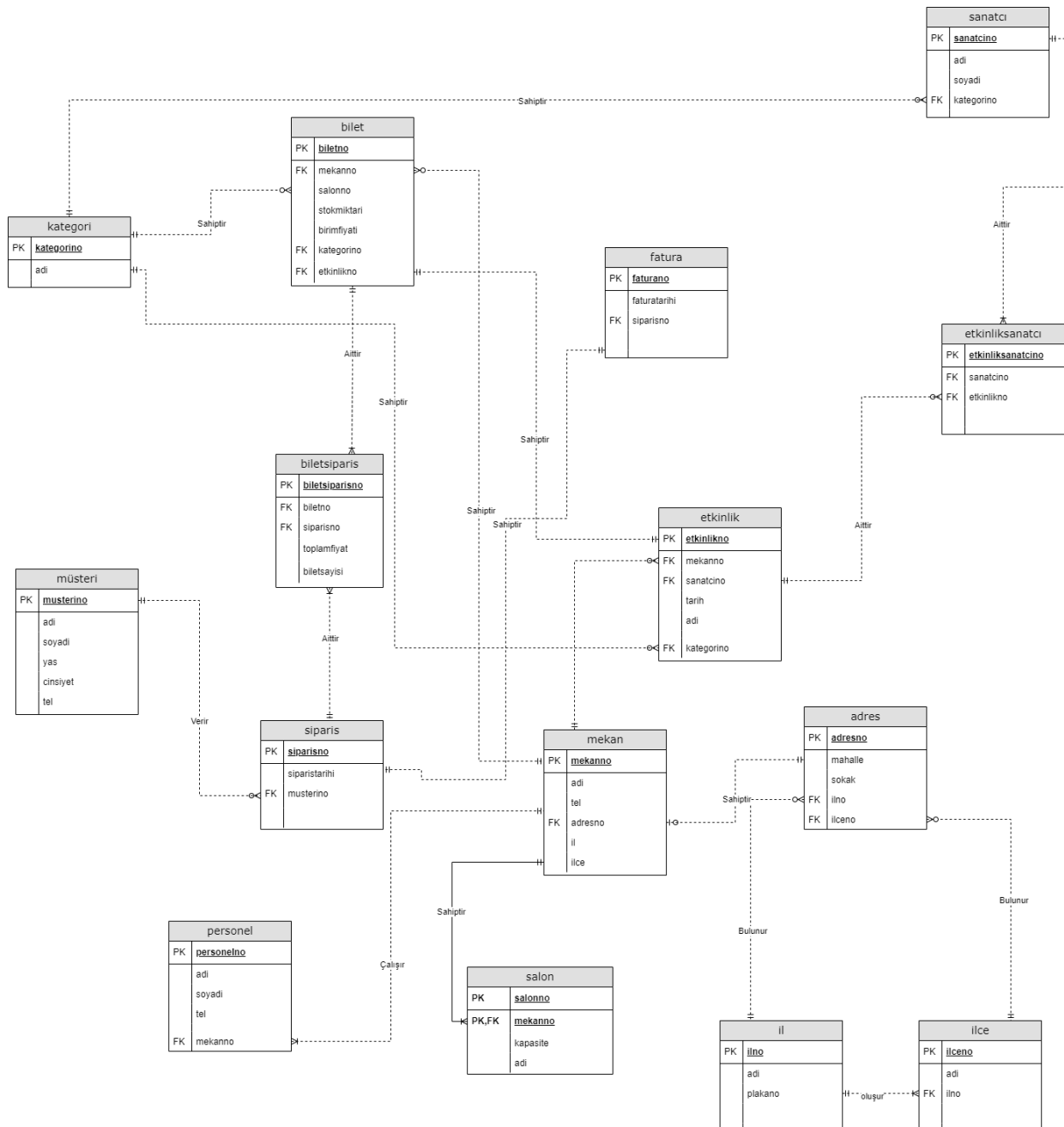
- Alınan her bir biletin numarası, mekan numarası, salon numarası, stok miktarı, birim fiyatı, kategori numarası ve etkinlik numarası mevcuttur.
- Bilet kategorilerinin kategori numarası ve adı mevcuttur.
- Verilen siparişlerin sipariş numarası, sipariş tarihi ve müşteri numarası bilgileri mevcuttur.
- Her faturanın fatura numarası, sipariş numarası ve fatura tarihi bilgileri mevcuttur.
- Etkinliklerin etkinlik numarası,adı, mekan, sanatçı, kategori ve tarih bilgileri mevcuttur.
- Sanatçıların sanatçı numarası, adı, soyadı ve kategori bilgileri mevcuttur.
- Her müşterinin müşteri numarası, adı, soyadı, yaşı, cinsiyeti ve telefon numarası bilgileri mevcuttur.
- Personellerin numarası, adı, soyadı, telefon numarası ve mekan bilgileri mevcuttur.
- Mekanların mekan numarası, adı, telefon numarası, adres, il ve ilçe bilgileri mevcuttur.
- Salonların salon numarası, adı, mekan ve kapasite bilgileri mevcuttur.
- Her bir adresin adres numarası, mahalle, sokak, il ve ilçe bilgileri mevcuttur.
- İllerin numarası, adı ve plaka bilgileri mevcuttur.
- İlçelerin numarası, adı ve il bilgileri mevcuttur.
- Bir bilet yalnızca bir kategoriye sahip olur. Bir kategoriye ait hiçbir bir bilet olmayabileceği gibi çok sayıda bilet de olabilir.
- Bir sanatçı yalnızca bir kategoriye sahip olur. Bir kategoride hiçbir sanatçı olmayabileceği gibi çok sayıda sanatçı da olabilir.
- Bir bilete ait çok sayıda sipariş olabilir. En az bir adet sipariş olmak zorundadır. Bir siparişte çok sayıda bilet olabilir. En az bir adet bilet olmak zorundadır.
- Bir müşteriye ait hiç sipariş olmayacağı gibi çok sayıda sipariş de olabilir. Bir sipariş yalnızca bir müşteriye ait olur.
- Bir siparişin yalnızca bir faturası olur. Bir fatura yalnızca bir siparişe aittir.
- Bir mekana ait hiçbir bilet olmayacağı gibi çok sayıda bilet de olabilir. Bir bilet yalnızca bir mekana sahip olur.
- Bir mekanda yalnızca bir personel çalışabileceği gibi çok sayıda personel de çalışabilir. Bir personel yalnızca bir mekanda çalışır.
- Bir mekanda hiçbir etkinlik olmayabileceği gibi çok sayıda etkinlik de olabilir. Bir etkinlik yalnızca bir mekanda olur.

- Bir mekanda yalnızca bir salon olabileceği gibi çok sayıda salon da olabilir. Bir salon yalnızca bir mekanda olur.
- Bir mekan yalnızca bir adrese sahiptir. Bir adreste hiçbir mekan olmayabileceği gibi yalnızca bir adet mekan da olabilir.
- Bir etkinliğe ait hiçbir bilet olmayabileceği gibi çok sayıda bilet de olabilir. Bir bilet yalnızca bir etkinliğe ait olur.
- Bir etkinliğe çok sayıda sanatçı ait olabileceği gibi en az bir sanatçı olur. Bir sanatçı hiçbir etkinliğe ait olmadığı gibi çok sayıda etkinliğe de ait olabilir.
- Bir etkinliğe ait yalnızca bir kategori olur. Bir kategoriye ait hiçbir etkinlik olmayabileceği gibi çok sayıda etkinlik de olabilir.
- Bir mekan yalnızca bir adrese sahip olur. Bir adreste hiçbir mekan olmayabileceği gibi yalnızca bir mekan da olabilir.
- Bir adres yalnızca bir ilde bulunur. Bir ile ait hiçbir adres olmayabileceği gibi çok sayıda adres de olabilir.
- Bir adres yalnızca bir ilçede bulunur. Bir ilçeye ait hiçbir adres olmayabileceği gibi çok sayıda adres de olabilir.
- Bir ile ait çok sayıda ilçe olabilir. En az bir ilçe olur. Bir ilçe yalnızca bir ile ait olur.

İlişkisel Şema

- bilet(biletno:serial, mekanno: integer, salonno: integer, stokmiktarı:int, birimfiyatı:Money, kategorino: integer, etkinlikno: integer)
- kategori(kategoino:serial, adi: varchar)
- biletsiparis(biletsiparisno:serial, biletno: integer, siparisno: integer, toplamfiyat:Money, biletsayısı:int)
- siparis(siparisno:serial, siparistarihi:date, musterino:integer)
- muster(musterino:serial, adi:varchar, soyadi:varchar, yas:int, cinsiyet:char, tel:varchar)
- personel(personelno:serial, adi:varchar, soyadi:varchar, tel:varchar, mekanno: integer)
- mekan(mekanno:integer, adi:varchar, tel:varchar, adresno:integer, il:char, ilce:char)
- salon(salonno:serial, mekanno:integer, kapasite:int, adi:varchar)
- adres(adresno:serial, mahalle:varchar, sokak:varchar, ilno: integer, ilceno: integer)
- il(ilno:serial, adi:varchar, plakano:varchar)
- ilce(ilceno:serial, adi:varchar, ilno: integer)
- etkinlik(etkinlikno:serial, mekanno: integer, sanatcino: integer, tarih:date, adi:varchar, kategorino:integer)
- fatura(faturano:serial, faturatarihi:date, siparisno:integer)
- etkinliksanatici(etkinliksanaticino:serial, sanaticino: integer, etkinlikno:integer)
- sanatci(sanaticino:serial, adi:varchar, soyadi:varchar, kategorino:integer)

Varlık Bağntı Modeli



SQL İfadeleri

```
--
-- PostgreSQL database dump
--

-- Dumped from database version 14.1
-- Dumped by pg_dump version 14.0

SET statement_timeout = 0;
SET lock_timeout = 0;
SET idle_in_transaction_session_timeout = 0;
SET client_encoding = 'UTF8';
SET standard_conforming_strings = on;
SELECT pg_catalog.set_config('search_path', '', false);
SET check_function_bodies = false;
SET xmloption = content;
SET client_min_messages = warning;
SET row_security = off;

--
-- Name: eskimusteriTR1(); Type: FUNCTION; Schema: public; Owner: postgres
--

CREATE FUNCTION public."eskimusteriTR1"() RETURNS trigger
    LANGUAGE plpgsql
    AS $$
BEGIN
    INSERT INTO public.eskimusteri("musterino","adi","soyadi","yas","cinsiyet","tel")
    VALUES (old.musterino,old.adi,old.soyadi,old.yas,old.cinsiyet,old.tel);
```

```
    RETURN OLD;
END;
$$;
```

```
ALTER FUNCTION public."eskimusteriTR1"() OWNER TO postgres;
```

```
--
```

```
-- Name: faturaekleTR2(); Type: FUNCTION; Schema: public; Owner: postgres
```

```
--
```

```
CREATE FUNCTION public."faturaekleTR2"() RETURNS trigger
```

```
    LANGUAGE plpgsql
```

```
    AS $$
```

```
BEGIN
```

```
    INSERT INTO public.fatura("siparisno","faturaTarihi")
```

```
    VALUES (new.siparisno,CURRENT_TIMESTAMP::TIMESTAMP);
```

```
    RETURN NEW;
```

```
END;
```

```
$$;
```

```
ALTER FUNCTION public."faturaekleTR2"() OWNER TO postgres;
```

```
--
```

```
-- Name: fiyatdegisikligiTR3(); Type: FUNCTION; Schema: public; Owner: postgres
```

```
--
```

```
CREATE FUNCTION public."fiyatdegisikligiTR3"() RETURNS trigger
```

```
    LANGUAGE plpgsql
```

```

AS $$
BEGIN
    IF NEW."birimFiyati" <> OLD."birimFiyati" THEN
        INSERT INTO
public.fiyatdegisikligi("biletno","eskibirimfiyati","yenibirimfiyati","degisiklikTarihi")
        VALUES
(OLD."biletno",OLD."birimFiyati",NEW."birimFiyati",CURRENT_TIMESTAMP::TIMEST
AMP);
    END IF;

    RETURN NEW;
END;
$$;

```

```

ALTER FUNCTION public."fiyatdegisikligiTR3"() OWNER TO postgres;

```

```
--
```

```
-- Name: kayitKontrolTR4(); Type: FUNCTION; Schema: public; Owner: postgres
```

```
--
```

```

CREATE FUNCTION public."kayitKontrolTR4"() RETURNS trigger

```

```
    LANGUAGE plpgsql
```

```
    AS $$
```

```

BEGIN

```

```
    NEW."soyadi" = UPPER(NEW."soyadi");
```

```
    NEW."soyadi" = LTRIM(NEW."soyadi");
```

```
    NEW."adi" = LTRIM(NEW."adi");
```

```
    NEW."cinsiyet" = LTRIM(NEW."cinsiyet");
```

```
    IF NEW."adi" IS NULL THEN
```

```
        RAISE EXCEPTION 'Adi alanı boş olamaz';
```

```
    END IF;

```

```
IF NEW."soyadi" IS NULL THEN
    RAISE EXCEPTION 'Soyadi alanı boş olamaz';
END IF;
IF NEW."yas" IS NULL THEN
    RAISE EXCEPTION 'Yas alanı boş olamaz';
END IF;
RETURN NEW;
END;
$$;

ALTER FUNCTION public."kayitKontrolTR4"() OWNER TO postgres;

--
-- Name: maxbiletfiyati(); Type: FUNCTION; Schema: public; Owner: postgres
--

CREATE FUNCTION public.maxbiletfiyati() RETURNS money
    LANGUAGE plpgsql
    AS $$
DECLARE
    sonuc MONEY;
BEGIN
    SELECT max("birimFiyati") into sonuc from bilet;
    RETURN sonuc;
END;
$$;

ALTER FUNCTION public.maxbiletfiyati() OWNER TO postgres;
```



```
--  
-- Name: musterisayisi(); Type: FUNCTION; Schema: public; Owner: postgres  
--
```

```
CREATE FUNCTION public.musterisayisi() RETURNS integer
```

```
    LANGUAGE plpgsql
```

```
    AS $$
```

```
DECLARE
```

```
    sonuc INTEGER;
```

```
BEGIN
```

```
    SELECT count(*) into sonuc from muster;
```

```
    RETURN sonuc;
```

```
END;
```

```
$$;
```

```
ALTER FUNCTION public.musterisayisi() OWNER TO postgres;
```

```
--  
-- Name: personelara(character varying); Type: FUNCTION; Schema: public; Owner:  
postgres  
--
```

```
CREATE FUNCTION public.personelara(personeladi character varying) RETURNS  
TABLE(nosutun integer, adisutun character varying, soyadisutun character varying,  
mekannosutun integer, telsutun character varying)
```

```
    LANGUAGE plpgsql
```

```
    AS $$
```

```
BEGIN
```

```
    RETURN QUERY
```

```
SELECT personelno, adi, soyadi, mekanno, tel FROM personel WHERE adi like
personelAdi;
END;
$$;
```

```
ALTER FUNCTION public.personelara(personeladi character varying) OWNER TO
postgres;
```

```
--
```

```
-- Name: stokaralığı(integer, integer); Type: FUNCTION; Schema: public; Owner: postgres
```

```
--
```

```
CREATE FUNCTION public."stokaralığı"(altstok integer, uststok integer) RETURNS
TABLE(biletnosutun integer, mekannosutun integer, salonnosutun integer, stokmiktarisutun
integer, kategorinosutun integer, etkinliknosutun integer)
```

```
LANGUAGE plpgsql
```

```
AS $$
```

```
BEGIN
```

```
RETURN QUERY
```

```
SELECT biletno, mekanno, salonno, stokmiktari, kategorino, etkinlikno FROM bilet
WHERE stokmiktari BETWEEN altStok AND ustStok;
```

```
END;
```

```
$$;
```

```
ALTER FUNCTION public."stokaralığı"(altstok integer, uststok integer) OWNER TO
postgres;
```

```
SET default_tablespace = '';
```

```
SET default_table_access_method = heap;
```

```
--  
-- Name: kategori; Type: TABLE; Schema: public; Owner: postgres  
--
```

```
CREATE TABLE public.kategori (  
    kategorino integer NOT NULL,  
    adi character varying(40) NOT NULL  
);
```

```
ALTER TABLE public.kategori OWNER TO postgres;
```

```
--  
-- Name: Kategori_kategoriNo_seq; Type: SEQUENCE; Schema: public; Owner: postgres  
--
```

```
CREATE SEQUENCE public."Kategori_kategoriNo_seq"  
    AS integer  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1;
```

```
ALTER TABLE public."Kategori_kategoriNo_seq" OWNER TO postgres;
```

```
--  
-- Name: Kategori_kategoriNo_seq; Type: SEQUENCE OWNED BY; Schema: public;  
Owner: postgres
```

--

```
ALTER SEQUENCE public."Kategori_kategoriNo_seq" OWNED BY
public.kategori.kategorino;
```

--

-- Name: adres; Type: TABLE; Schema: public; Owner: postgres

--

```
CREATE TABLE public.adres (
    adresno integer NOT NULL,
    mahalle character varying(40),
    sokak character varying(40),
    ilno integer,
    ilceno integer
);
```

```
ALTER TABLE public.adres OWNER TO postgres;
```

--

-- Name: adres_adresno_seq; Type: SEQUENCE; Schema: public; Owner: postgres

--

```
CREATE SEQUENCE public.adres_adresno_seq
    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
```

CACHE 1;

ALTER TABLE public.adres_adresno_seq OWNER TO postgres;

--

-- Name: adres_adresno_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner: postgres

--

ALTER SEQUENCE public.adres_adresno_seq OWNED BY public.adres.adresno;

--

-- Name: bilet; Type: TABLE; Schema: public; Owner: postgres

--

CREATE TABLE public.bilet (

biletno integer NOT NULL,

mekanno integer,

salonno integer,

"birimFiyati" money,

stokmiktari integer,

kategorino integer,

etkinlikno integer

);

ALTER TABLE public.bilet OWNER TO postgres;

--

-- Name: bilet_biletno_seq; Type: SEQUENCE; Schema: public; Owner: postgres

--

CREATE SEQUENCE public.bilet_biletno_seq

AS integer

START WITH 1

INCREMENT BY 1

NO MINVALUE

NO MAXVALUE

CACHE 1;

ALTER TABLE public.bilet_biletno_seq OWNER TO postgres;

--

-- Name: bilet_biletno_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner: postgres

--

ALTER SEQUENCE public.bilet_biletno_seq OWNED BY public.bilet.biletno;

--

-- Name: biletsiparis; Type: TABLE; Schema: public; Owner: postgres

--

CREATE TABLE public.biletsiparis (

biletsiparisno integer NOT NULL,

biletno integer,

siparisno integer,

toplamfiyat money,

```
    biletsayisi integer NOT NULL
);
```

```
ALTER TABLE public.biletsiparis OWNER TO postgres;
```

```
--
-- Name: biletsiparis_biletsiparisno_seq; Type: SEQUENCE; Schema: public; Owner:
postgres
--
```

```
CREATE SEQUENCE public.biletsiparis_biletsiparisno_seq
    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;
```

```
ALTER TABLE public.biletsiparis_biletsiparisno_seq OWNER TO postgres;
```

```
--
-- Name: biletsiparis_biletsiparisno_seq; Type: SEQUENCE OWNED BY; Schema: public;
Owner: postgres
--
```

```
ALTER SEQUENCE public.biletsiparis_biletsiparisno_seq OWNED BY
public.biletsiparis.biletsiparisno;
```

```
--
```

```
-- Name: eskimusteri; Type: TABLE; Schema: public; Owner: postgres
```

```
--
```

```
CREATE TABLE public.eskimusteri (  
    musterino integer,  
    adi character varying(40),  
    soyadi character varying(40),  
    yas integer,  
    cinsiyet character varying(40),  
    tel character varying(40)  
);
```

```
ALTER TABLE public.eskimusteri OWNER TO postgres;
```

```
--
```

```
-- Name: etkinlik; Type: TABLE; Schema: public; Owner: postgres
```

```
--
```

```
CREATE TABLE public.etkinlik (  
    etkinlikno integer NOT NULL,  
    mekanno integer,  
    sanatcino integer,  
    tarih date,  
    kategorino integer,  
    adi character varying  
);
```

```
ALTER TABLE public.etkinlik OWNER TO postgres;
```



```
--  
-- Name: etkinlik_etkinlikno_seq; Type: SEQUENCE; Schema: public; Owner: postgres  
--
```

```
CREATE SEQUENCE public.etkinlik_etkinlikno_seq  
    AS integer  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1;
```

```
ALTER TABLE public.etkinlik_etkinlikno_seq OWNER TO postgres;
```

```
--  
-- Name: etkinlik_etkinlikno_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner:  
postgres  
--
```

```
ALTER SEQUENCE public.etkinlik_etkinlikno_seq OWNED BY public.etkinlik.etkinlikno;
```

```
--  
-- Name: etkinliksanatci; Type: TABLE; Schema: public; Owner: postgres  
--
```

```
CREATE TABLE public.etkinliksanatci (  
    etkinliksanadcino integer NOT NULL,  
    sanadcino integer,
```

```
    etkinlikno integer
);
```

```
ALTER TABLE public.etkinliksanatci OWNER TO postgres;
```

```
--
-- Name: etkinliksanatci_etkinliksanatcino_seq; Type: SEQUENCE; Schema: public; Owner:
postgres
--
```

```
CREATE SEQUENCE public.etkinliksanatci_etkinliksanatcino_seq
    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;
```

```
ALTER TABLE public.etkinliksanatci_etkinliksanatcino_seq OWNER TO postgres;
```

```
--
-- Name: etkinliksanatci_etkinliksanatcino_seq; Type: SEQUENCE OWNED BY; Schema:
public; Owner: postgres
--
```

```
ALTER SEQUENCE public.etkinliksanatci_etkinliksanatcino_seq OWNED BY
public.etkinliksanatci.etkinliksanatcino;
```

```
--
```

```
-- Name: fatura; Type: TABLE; Schema: public; Owner: postgres
```

```
--
```

```
CREATE TABLE public.fatura (  
    faturano integer NOT NULL,  
    "faturaTarihi" date,  
    siparisno integer  
);
```

```
ALTER TABLE public.fatura OWNER TO postgres;
```

```
--
```

```
-- Name: fatura_faturano_seq; Type: SEQUENCE; Schema: public; Owner: postgres
```

```
--
```

```
CREATE SEQUENCE public.fatura_faturano_seq  
    AS integer  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1;
```

```
ALTER TABLE public.fatura_faturano_seq OWNER TO postgres;
```

```
--
```

```
-- Name: fatura_faturano_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner:  
postgres
```

```
--
```

```
ALTER SEQUENCE public.fatura_faturano_seq OWNED BY public.fatura.faturano;
```

```
--
```

```
-- Name: fiyatdegisikligi; Type: TABLE; Schema: public; Owner: postgres
```

```
--
```

```
CREATE TABLE public.fiyatdegisikligi (
```

```
    kayitno integer NOT NULL,
```

```
    biletno integer,
```

```
    eskibirimfiyati money,
```

```
    yenibirimfiyati money,
```

```
    "degisiklikTarihi" date
```

```
);
```

```
ALTER TABLE public.fiyatdegisikligi OWNER TO postgres;
```

```
--
```

```
-- Name: fiyatdegisikligi_kayitno_seq; Type: SEQUENCE; Schema: public; Owner: postgres
```

```
--
```

```
CREATE SEQUENCE public.fiyatdegisikligi_kayitno_seq
```

```
    AS integer
```

```
    START WITH 1
```

```
    INCREMENT BY 1
```

```
    NO MINVALUE
```

```
    NO MAXVALUE
```

```
    CACHE 1;
```

```
ALTER TABLE public.fiyatdegisikligi_kayitno_seq OWNER TO postgres;
```

```
--
```

```
-- Name: fiyatdegisikligi_kayitno_seq; Type: SEQUENCE OWNED BY; Schema: public;  
Owner: postgres
```

```
--
```

```
ALTER SEQUENCE public.fiyatdegisikligi_kayitno_seq OWNED BY  
public.fiyatdegisikligi.kayitno;
```

```
--
```

```
-- Name: il; Type: TABLE; Schema: public; Owner: postgres
```

```
--
```

```
CREATE TABLE public.il (  
    ilno integer NOT NULL,  
    adi character varying(40),  
    plakano character varying(30)  
);
```

```
ALTER TABLE public.il OWNER TO postgres;
```

```
--
```

```
-- Name: il_ilno_seq; Type: SEQUENCE; Schema: public; Owner: postgres
```

```
--
```

```
CREATE SEQUENCE public.il_ilno_seq
```

```
AS integer
START WITH 1
INCREMENT BY 1
NO MINVALUE
NO MAXVALUE
CACHE 1;
```

```
ALTER TABLE public.il_ilno_seq OWNER TO postgres;
```

```
--
-- Name: il_ilno_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner: postgres
--
```

```
ALTER SEQUENCE public.il_ilno_seq OWNED BY public.il.ilno;
```

```
--
-- Name: ilce; Type: TABLE; Schema: public; Owner: postgres
--
```

```
CREATE TABLE public.ilce (
    ilceno integer NOT NULL,
    adi character varying(40) NOT NULL,
    ilno integer NOT NULL
);
```

```
ALTER TABLE public.ilce OWNER TO postgres;
```

```
--  
-- Name: ilce_ilceno_seq; Type: SEQUENCE; Schema: public; Owner: postgres  
--
```

```
CREATE SEQUENCE public.ilce_ilceno_seq  
    AS integer  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1;
```

```
ALTER TABLE public.ilce_ilceno_seq OWNER TO postgres;
```

```
--  
-- Name: ilce_ilceno_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner:  
postgres  
--
```

```
ALTER SEQUENCE public.ilce_ilceno_seq OWNED BY public.ilce.ilceno;
```

```
--  
-- Name: mekan; Type: TABLE; Schema: public; Owner: postgres  
--
```

```
CREATE TABLE public.mekan (  
    mekanno integer NOT NULL,  
    adi character varying(40),  
    tel character varying(40),
```

```
    adresno integer,  
    il character(40),  
    ilce character(40)  
);
```

```
ALTER TABLE public.mekan OWNER TO postgres;
```

```
--  
-- Name: mekan_mekanno_seq; Type: SEQUENCE; Schema: public; Owner: postgres  
--
```

```
CREATE SEQUENCE public.mekan_mekanno_seq  
    AS integer  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1;
```

```
ALTER TABLE public.mekan_mekanno_seq OWNER TO postgres;
```

```
--  
-- Name: mekan_mekanno_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner:  
postgres  
--
```

```
ALTER SEQUENCE public.mekan_mekanno_seq OWNED BY public.mekan.mekanno;
```



```
--  
-- Name: musteri; Type: TABLE; Schema: public; Owner: postgres  
--
```

```
CREATE TABLE public.musteri (  
    musterino integer NOT NULL,  
    adi character varying(40) NOT NULL,  
    soyadi character varying(40) NOT NULL,  
    cinsiyet character varying(40),  
    tel character varying(40),  
    yas integer DEFAULT 0  
);
```

```
ALTER TABLE public.musteri OWNER TO postgres;
```

```
--  
-- Name: musteri_musteriNo_seq; Type: SEQUENCE; Schema: public; Owner: postgres  
--
```

```
CREATE SEQUENCE public."musteri_musteriNo_seq"  
    AS integer  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1;
```

```
ALTER TABLE public."musteri_musteriNo_seq" OWNER TO postgres;
```

```
--  
-- Name: muster_i_musteriNo_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner:  
postgres
```

```
--
```

```
ALTER SEQUENCE public."muster_i_musteriNo_seq" OWNED BY  
public.musteri.musterino;
```

```
--
```

```
-- Name: personel; Type: TABLE; Schema: public; Owner: postgres
```

```
--
```

```
CREATE TABLE public.personel (  
    personelno integer NOT NULL,  
    adi character varying(40),  
    soyadi character varying(40),  
    tel character varying(40),  
    mekanno integer NOT NULL  
);
```

```
ALTER TABLE public.personel OWNER TO postgres;
```

```
--
```

```
-- Name: personel_personelno_seq; Type: SEQUENCE; Schema: public; Owner: postgres
```

```
--
```

```
CREATE SEQUENCE public.personel_personelno_seq  
    AS integer
```

```
START WITH 1
INCREMENT BY 1
NO MINVALUE
NO MAXVALUE
CACHE 1;
```

```
ALTER TABLE public.personel_personelno_seq OWNER TO postgres;
```

```
--
-- Name: personel_personelno_seq; Type: SEQUENCE OWNED BY; Schema: public;
Owner: postgres
--
```

```
ALTER SEQUENCE public.personel_personelno_seq OWNED BY
public.personel.personelno;
```

```
--
-- Name: salon; Type: TABLE; Schema: public; Owner: postgres
--
```

```
CREATE TABLE public.salon (
    salonno integer NOT NULL,
    mekanno integer,
    kapasite integer,
    adi character varying
);
```

```
ALTER TABLE public.salon OWNER TO postgres;
```

```
--  
-- Name: salon_salonno_seq; Type: SEQUENCE; Schema: public; Owner: postgres  
--
```

```
CREATE SEQUENCE public.salon_salonno_seq  
    AS integer  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1;
```

```
ALTER TABLE public.salon_salonno_seq OWNER TO postgres;
```

```
--  
-- Name: salon_salonno_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner:  
postgres  
--
```

```
ALTER SEQUENCE public.salon_salonno_seq OWNED BY public.salon.salonno;
```

```
--  
-- Name: sanatci; Type: TABLE; Schema: public; Owner: postgres  
--
```

```
CREATE TABLE public.sanatci (  
    sanatcino integer NOT NULL,  
    adi character varying(40),
```

```
    soyadi character varying(40),
    kategorino integer DEFAULT 0
);
```

```
ALTER TABLE public.sanatci OWNER TO postgres;
```

```
--
-- Name: sanatci_sanatcino_seq; Type: SEQUENCE; Schema: public; Owner: postgres
--
```

```
CREATE SEQUENCE public.sanatci_sanatcino_seq
    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;
```

```
ALTER TABLE public.sanatci_sanatcino_seq OWNER TO postgres;
```

```
--
-- Name: sanatci_sanatcino_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner:
postgres
--
```

```
ALTER SEQUENCE public.sanatci_sanatcino_seq OWNED BY public.sanatci.sanatcino;
```

```
--
```

-- Name: siparis; Type: TABLE; Schema: public; Owner: postgres

--

```
CREATE TABLE public.siparis (  
    siparisno integer NOT NULL,  
    siparistarihi date,  
    musterino integer NOT NULL  
);
```

```
ALTER TABLE public.siparis OWNER TO postgres;
```

--

-- Name: siparis_siparisno_seq; Type: SEQUENCE; Schema: public; Owner: postgres

--

```
CREATE SEQUENCE public.siparis_siparisno_seq  
    AS integer  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1;
```

```
ALTER TABLE public.siparis_siparisno_seq OWNER TO postgres;
```

--

-- Name: siparis_siparisno_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner: postgres

--

```
ALTER SEQUENCE public.siparis_siparisno_seq OWNED BY public.siparis.siparisno;
```

```
--
```

```
-- Name: adres adresno; Type: DEFAULT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.adres ALTER COLUMN adresno SET DEFAULT  
nextval('public.adres_adresno_seq'::regclass);
```

```
--
```

```
-- Name: bilet biletno; Type: DEFAULT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.bilet ALTER COLUMN biletno SET DEFAULT  
nextval('public.bilet_biletno_seq'::regclass);
```

```
--
```

```
-- Name: biletsiparis biletsiparisno; Type: DEFAULT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.biletsiparis ALTER COLUMN biletsiparisno SET DEFAULT  
nextval('public.biletsiparis_biletsiparisno_seq'::regclass);
```

```
--
```

```
-- Name: etkinlik etkinlikno; Type: DEFAULT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.etkinlik ALTER COLUMN etkinlikno SET DEFAULT  
nextval('public.etkinlik_etkinlikno_seq'::regclass);
```

```
--
```

```
-- Name: etkinliksanatici etkinliksanaticino; Type: DEFAULT; Schema: public; Owner:  
postgres
```

```
--
```

```
ALTER TABLE ONLY public.etkinliksanatici ALTER COLUMN etkinliksanaticino SET  
DEFAULT nextval('public.etkinliksanatici_etkinliksanaticino_seq'::regclass);
```

```
--
```

```
-- Name: fatura faturano; Type: DEFAULT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.fatura ALTER COLUMN faturano SET DEFAULT  
nextval('public.fatura_faturano_seq'::regclass);
```

```
--
```

```
-- Name: fiyatdegisikligi kayitno; Type: DEFAULT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.fiyatdegisikligi ALTER COLUMN kayitno SET DEFAULT  
nextval('public.fiyatdegisikligi_kayitno_seq'::regclass);
```

```
--
```

```
-- Name: il ilno; Type: DEFAULT; Schema: public; Owner: postgres
```

```
--
```



```
ALTER TABLE ONLY public.il ALTER COLUMN ilno SET DEFAULT  
nextval('public.il_ilno_seq'::regclass);
```

```
--
```

```
-- Name: ilce ilceno; Type: DEFAULT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.ilce ALTER COLUMN ilceno SET DEFAULT  
nextval('public.ilce_ilceno_seq'::regclass);
```

```
--
```

```
-- Name: kategori kategorino; Type: DEFAULT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.kategori ALTER COLUMN kategorino SET DEFAULT  
nextval('public."Kategori_kategoriNo_seq"::regclass);
```

```
--
```

```
-- Name: mekan mekanno; Type: DEFAULT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.mekan ALTER COLUMN mekanno SET DEFAULT  
nextval('public.mekan_mekanno_seq'::regclass);
```

```
--
```

```
-- Name: musteri musterino; Type: DEFAULT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.musteri ALTER COLUMN musterino SET DEFAULT  
nextval('public."musteri_musteriNo_seq"::regclass);
```

```
--
```

```
-- Name: personel personelno; Type: DEFAULT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.personel ALTER COLUMN personelno SET DEFAULT  
nextval('public.personel_personelno_seq'::regclass);
```

```
--
```

```
-- Name: salon salonno; Type: DEFAULT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.salon ALTER COLUMN salonno SET DEFAULT  
nextval('public.salon_salonno_seq'::regclass);
```

```
--
```

```
-- Name: sanatici sanatcino; Type: DEFAULT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.sanatici ALTER COLUMN sanatcino SET DEFAULT  
nextval('public.sanatici_sanatcino_seq'::regclass);
```

```
--
```

```
-- Name: siparis siparisno; Type: DEFAULT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.siparis ALTER COLUMN siparisno SET DEFAULT  
nextval('public.siparis_siparisno_seq'::regclass);
```

```
--
```

```
-- Data for Name: adres; Type: TABLE DATA; Schema: public; Owner: postgres
```

```
--
```

```
INSERT INTO public.adres VALUES
```

```
(1, 'Esentepe', 'No:8', 1, 1),  
(2, 'Kocatepe', 'Çiçek', 1, 1),  
(3, 'Harbiye', 'Taşkışla', 11, 7),  
(4, 'Cihannüma', 'Akmaz Çeşme', 11, 8),  
(5, 'Cumhuriyet', 'Atatürk Blv.', 14, 5);
```

```
--
```

```
-- Data for Name: bilet; Type: TABLE DATA; Schema: public; Owner: postgres
```

```
--
```

```
INSERT INTO public.bilet VALUES
```

```
(2, 3, 3, '?15,00', 56, 1, 2),  
(3, 3, 4, '?15,00', 75, 1, 2),  
(4, 3, 5, '?15,00', 88, 1, 2),  
(5, 3, 6, '?15,00', 96, 1, 2),  
(1, 1, 1, '?150,00', 4532, 2, 1);
```

```
--
```

```
-- Data for Name: biletsiparis; Type: TABLE DATA; Schema: public; Owner: postgres
```

--

INSERT INTO public.biletsiparis VALUES

(1, 1, 1, '?450,00', 3);

--

-- Data for Name: eskimusteri; Type: TABLE DATA; Schema: public; Owner: postgres

--

INSERT INTO public.eskimusteri VALUES

(16, 'bbbbbbb', 'bbbbbbb', 25, 'kadın', '66666666'),

(15, 'aaaaaaaaaaa', 'aaaaaaaaaaa', 56, 'kadın', '555555555'),

(19, 'kontrol3', '', 26, 'erkek', '55555555555'),

(18, 'kontrol', 'KONTROL2', 54, 'erkek', '55555555555'),

(11, '', 'GÜNCEL2', 12, 'kadın', ''),

(20, 'Proje', 'DENEME', 45, 'erkek', '053214569854'),

(28, 'Silinecek', 'MÜŞTERİ', 0, 'belirtilmemiş', '33333333333');

--

-- Data for Name: etkinlik; Type: TABLE DATA; Schema: public; Owner: postgres

--

INSERT INTO public.etkinlik VALUES

(1, 1, 1, '2022-01-25', 2, 'Athena Konseri'),

(2, 3, NULL, '2021-12-17', 1, 'Spider-Man: No Way Home');

--

-- Data for Name: etkinliksanatci; Type: TABLE DATA; Schema: public; Owner: postgres

--

INSERT INTO public.etkinliksanatci VALUES

(1, 1, 1),

(2, NULL, 2);

--

-- Data for Name: fatura; Type: TABLE DATA; Schema: public; Owner: postgres

--

INSERT INTO public.fatura VALUES

(1, '2021-12-15', 1),

(3, '2021-12-15', 2),

(4, '2021-12-16', 5);

--

-- Data for Name: fiyatdegisikligi; Type: TABLE DATA; Schema: public; Owner: postgres

--

INSERT INTO public.fiyatdegisikligi VALUES

(1, 1, '?200,00', '?150,00', '2021-12-16');

--

-- Data for Name: il; Type: TABLE DATA; Schema: public; Owner: postgres

--

```
INSERT INTO public.il VALUES
```

```
(1, 'Adana', '01'),  
(2, 'Adıyaman', '02'),  
(3, 'Afyon', '03'),  
(4, 'Ağrı', '04'),  
(5, 'Amasya', '05'),  
(6, 'Ankara', '06'),  
(7, 'Antalya', '07'),  
(8, 'Artvin', '08'),  
(9, 'Aydın', '09'),  
(10, 'Balıkesir', '10'),  
(11, 'İstanbul', '34'),  
(12, 'İzmir', '35'),  
(13, 'Eskişehir', '26'),  
(14, 'Sakarya', '54');
```

```
--
```

```
-- Data for Name: ilce; Type: TABLE DATA; Schema: public; Owner: postgres
```

```
--
```

```
INSERT INTO public.ilce VALUES
```

```
(1, 'Çukurova', 1),  
(2, 'Sarıçam', 1),  
(3, 'Seyhan', 1),  
(4, 'Tepebaşı', 13),  
(5, 'Adapazarı', 14),  
(6, 'Serdivan', 14),  
(7, 'Şişli', 11),  
(8, 'Beşiktaş', 11);
```

--

-- Data for Name: kategori; Type: TABLE DATA; Schema: public; Owner: postgres

--

INSERT INTO public.kategori VALUES

(1, 'Sinema'),
(2, 'Konser'),
(3, 'Tiyatro'),
(4, 'Spor'),
(5, 'Konferans'),
(6, 'Eğitim'),
(7, 'Stand-up'),
(8, 'Dans'),
(9, 'Gösteri'),
(10, 'Sirk');

--

-- Data for Name: mekan; Type: TABLE DATA; Schema: public; Owner: postgres

--

INSERT INTO public.mekan VALUES

(1, 'Harbiye Açıkhava Sahnesi', '0(212)4553900', 3, 'İstanbul',
'Şişli'),
(2, 'IF Performance Hall Beşiktaş', '05465669946', 4, 'İstanbul',
'Beşiktaş'),
(3, 'AKM Sinemaları', '(0264)282 19 99', 5, 'Sakarya', 'Adapazarı');

--

-- Data for Name: musteri; Type: TABLE DATA; Schema: public; Owner: postgres

--

INSERT INTO public.musteri VALUES

(21, 'Efe', 'DURMUŞ', 'erkek', '05364589874', 20),
(22, 'Barış', 'KıRÖMEROĞLU', 'erkek', '05364589872', 19),
(23, 'Ömer', 'UZUN', 'erkek', '05364589741', 25),
(2, 'Kemal', 'YILDIRIM', 'erkek', '05621567483', 56),
(1, 'Serdar', 'ARICI', 'erkek', '05355555555', 20),
(3, 'Ayşe', 'YILMAZ', 'kadın', '05364581972', 45),
(6, 'Ali', 'ÇAVUŞ', 'erkek', '05335698452', 19),
(9, 'Hasan', 'TÜRK', 'erkek', '05345694512', 74),
(24, 'Zeynep', 'ANADOLU', 'kadın', '05335698547', 56),
(25, 'Hasan', 'ELMA', 'erkek', '05335698216', 35),
(26, 'Hasan', 'YıLDıZ', 'erkek', '053912256847', 27),
(27, 'Ekleme', 'İŞLEMİ', 'belirtilmemiş', '00000000000', 0),
(29, 'Güncelleme', 'İŞLEMİ', 'kadın', '44444444444', 22);

--

-- Data for Name: personel; Type: TABLE DATA; Schema: public; Owner: postgres

--

INSERT INTO public.personel VALUES

(2, 'Ahmet', 'Hoca', '05362145987', 2),
(3, 'Hatice', 'Direk', '05361459852', 3),
(4, 'Osman', 'Dağ', '05321254675', 3),
(5, 'Hasan', 'Orman', '05380215281', 1);

--

-- Data for Name: salon; Type: TABLE DATA; Schema: public; Owner: postgres

--

INSERT INTO public.salon VALUES

(1, 1, 4532, 'Salon1'),

(2, 2, 500, 'Salon1'),

(11, 3, 56, 'Salon1'),

(12, 3, 75, 'Salon2'),

(13, 3, 88, 'Salon3'),

(14, 3, 96, 'Salon4');

--

-- Data for Name: sanatci; Type: TABLE DATA; Schema: public; Owner: postgres

--

INSERT INTO public.sanatci VALUES

(1, 'Athena', NULL, 2);

--

-- Data for Name: siparis; Type: TABLE DATA; Schema: public; Owner: postgres

--

INSERT INTO public.siparis VALUES

(1, '2021-12-15', 1),

(2, '2021-07-17', 3),

(5, '2021-11-25', 6);

```
--  
-- Name: Kategori_kategoriNo_seq; Type: SEQUENCE SET; Schema: public; Owner:  
postgres
```

```
--  
  
SELECT pg_catalog.setval('public."Kategori_kategoriNo_seq"', 10, true);
```

```
--  
-- Name: adres_adresno_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres
```

```
--  
  
SELECT pg_catalog.setval('public.adres_adresno_seq', 5, true);
```

```
--  
-- Name: bilet_biletno_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres
```

```
--  
  
SELECT pg_catalog.setval('public.bilet_biletno_seq', 5, true);
```

```
--  
-- Name: biletsiparis_biletsiparisno_seq; Type: SEQUENCE SET; Schema: public; Owner:  
postgres
```

```
--  
  
SELECT pg_catalog.setval('public.biletsiparis_biletsiparisno_seq', 1, true);
```

```
--  
-- Name: etkinlik_etkinlikno_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres  
--
```

```
SELECT pg_catalog.setval('public.etkinlik_etkinlikno_seq', 2, true);
```

```
--  
-- Name: etkinliksanatici_etkinliksanaticino_seq; Type: SEQUENCE SET; Schema: public;  
Owner: postgres  
--
```

```
SELECT pg_catalog.setval('public.etkinliksanatici_etkinliksanaticino_seq', 2, true);
```

```
--  
-- Name: fatura_faturano_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres  
--
```

```
SELECT pg_catalog.setval('public.fatura_faturano_seq', 4, true);
```

```
--  
-- Name: fiyatdegisikligi_kayitno_seq; Type: SEQUENCE SET; Schema: public; Owner:  
postgres  
--
```

```
SELECT pg_catalog.setval('public.fiyatdegisikligi_kayitno_seq', 1, true);
```

```
--  
-- Name: il_ilno_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres  
--
```

```
SELECT pg_catalog.setval('public.il_ilno_seq', 14, true);
```

```
--  
-- Name: ilce_ilceno_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres  
--
```

```
SELECT pg_catalog.setval('public.ilce_ilceno_seq', 8, true);
```

```
--  
-- Name: mekan_mekanno_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres  
--
```

```
SELECT pg_catalog.setval('public.mekan_mekanno_seq', 3, true);
```

```
--  
-- Name: musteri_musteriNo_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres  
--
```

```
SELECT pg_catalog.setval('public."musteri_musteriNo_seq"', 29, true);
```

```
--  
-- Name: personel_personelno_seq; Type: SEQUENCE SET; Schema: public; Owner:  
postgres
```

--

```
SELECT pg_catalog.setval('public.personel_personelno_seq', 5, true);
```

--

```
-- Name: salon_salonno_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres
```

--

```
SELECT pg_catalog.setval('public.salon_salonno_seq', 14, true);
```

--

```
-- Name: sanatci_sanatcino_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres
```

--

```
SELECT pg_catalog.setval('public.sanatci_sanatcino_seq', 2, true);
```

--

```
-- Name: siparis_siparisno_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres
```

--

```
SELECT pg_catalog.setval('public.siparis_siparisno_seq', 5, true);
```

--

```
-- Name: adres adresPK; Type: CONSTRAINT; Schema: public; Owner: postgres
```

--

```
ALTER TABLE ONLY public.adres
```

```
    ADD CONSTRAINT "adresPK" PRIMARY KEY (adresno);
```

```
--
```

```
-- Name: билет билетPK; Type: CONSTRAINT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.ilet
```

```
    ADD CONSTRAINT "iletPK" PRIMARY KEY (iletno);
```

```
--
```

```
-- Name: biletsiparis biletsiparisPK; Type: CONSTRAINT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.biletsiparis
```

```
    ADD CONSTRAINT "biletsiparisPK" PRIMARY KEY (biletsiparisno);
```

```
--
```

```
-- Name: etkinlik etkinlikPK; Type: CONSTRAINT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.etkinlik
```

```
    ADD CONSTRAINT "etkinlikPK" PRIMARY KEY (etkinlikno);
```

```
--
```

```
-- Name: etkinliksanatci etkinliksanatciPK; Type: CONSTRAINT; Schema: public; Owner: postgres
```

--

ALTER TABLE ONLY public.etkinliksanatci

ADD CONSTRAINT "etkinliksanatciPK" PRIMARY KEY (etkinliksanatcino);

--

-- Name: fatura faturaPK; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public.fatura

ADD CONSTRAINT "faturaPK" PRIMARY KEY (faturano);

--

-- Name: il ilPK; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public.il

ADD CONSTRAINT "ilPK" PRIMARY KEY (ilno);

--

-- Name: il ilUnique; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public.il

ADD CONSTRAINT "ilUnique" UNIQUE (plakano);

```
--  
-- Name: ilce ilcePK; Type: CONSTRAINT; Schema: public; Owner: postgres  
--
```

```
ALTER TABLE ONLY public.ilce  
    ADD CONSTRAINT "ilcePK" PRIMARY KEY (ilceno);
```

```
--  
-- Name: kategori kategoriPK; Type: CONSTRAINT; Schema: public; Owner: postgres  
--
```

```
ALTER TABLE ONLY public.kategori  
    ADD CONSTRAINT "kategoriPK" PRIMARY KEY (kategorino);
```

```
--  
-- Name: fiyatdegisikligi kayitPK; Type: CONSTRAINT; Schema: public; Owner: postgres  
--
```

```
ALTER TABLE ONLY public.fiyatdegisikligi  
    ADD CONSTRAINT "kayitPK" PRIMARY KEY (kayitno);
```

```
--  
-- Name: mekan mekanPK; Type: CONSTRAINT; Schema: public; Owner: postgres  
--
```

```
ALTER TABLE ONLY public.mekan  
    ADD CONSTRAINT "mekanPK" PRIMARY KEY (mekanno);
```



```
--  
-- Name: musteri musteriPK; Type: CONSTRAINT; Schema: public; Owner: postgres  
--
```

```
ALTER TABLE ONLY public.musteri  
    ADD CONSTRAINT "musteriPK" PRIMARY KEY (musterino);
```

```
--  
-- Name: personel personelPK; Type: CONSTRAINT; Schema: public; Owner: postgres  
--
```

```
ALTER TABLE ONLY public.personel  
    ADD CONSTRAINT "personelPK" PRIMARY KEY (personelno);
```

```
--  
-- Name: salon salonPK; Type: CONSTRAINT; Schema: public; Owner: postgres  
--
```

```
ALTER TABLE ONLY public.salon  
    ADD CONSTRAINT "salonPK" PRIMARY KEY (salonno);
```

```
--  
-- Name: sanatici sanaticiPK; Type: CONSTRAINT; Schema: public; Owner: postgres  
--
```

```
ALTER TABLE ONLY public.sanatici
```

```
ADD CONSTRAINT "sanaticiPK" PRIMARY KEY (sanaticino);
```

```
--
```

```
-- Name: siparis siparisPK; Type: CONSTRAINT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.siparis
```

```
ADD CONSTRAINT "siparisPK" PRIMARY KEY (siparisno);
```

```
--
```

```
-- Name: siparis faturaekleTR2; Type: TRIGGER; Schema: public; Owner: postgres
```

```
--
```

```
CREATE TRIGGER "faturaekleTR2" AFTER INSERT ON public.siparis FOR EACH ROW  
EXECUTE FUNCTION public."faturaekleTR2"();
```

```
--
```

```
-- Name: bilet fiyatdegistiginde; Type: TRIGGER; Schema: public; Owner: postgres
```

```
--
```

```
CREATE TRIGGER fiyatdegistiginde BEFORE UPDATE ON public.bilet FOR EACH  
ROW EXECUTE FUNCTION public."fiyatdegisikligiTR3"();
```

```
--
```

```
-- Name: musteriyayitKontrol; Type: TRIGGER; Schema: public; Owner: postgres
```

```
--
```

```
CREATE TRIGGER "kayitKontrol" BEFORE INSERT OR UPDATE ON public.musteri
FOR EACH ROW EXECUTE FUNCTION public."kayitKontrolTR4"();
```

```
--
```

```
-- Name: musterisilindiginde; Type: TRIGGER; Schema: public; Owner: postgres
```

```
--
```

```
CREATE TRIGGER musterisilindiginde AFTER DELETE ON public.musteri FOR EACH
ROW EXECUTE FUNCTION public."eskimusteriTR1"();
```

```
--
```

```
-- Name: mekan adresFK; Type: FK CONSTRAINT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.mekan
```

```
ADD CONSTRAINT "adresFK" FOREIGN KEY (adresno) REFERENCES
public.adres(adresno);
```

```
--
```

```
-- Name: biletsiparis biletk; Type: FK CONSTRAINT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.biletsiparis
```

```
ADD CONSTRAINT "biletk" FOREIGN KEY (biletk) REFERENCES
public.bilet(biletk) ON UPDATE CASCADE ON DELETE CASCADE;
```

```
--
```

-- Name: bilet etkinlikFK; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public.bilet

ADD CONSTRAINT "etkinlikFK" FOREIGN KEY (etkinlikno) REFERENCES
public.etkinlik(etkinlikno);

--

-- Name: etkinliksanatici etkinlikFK; Type: FK CONSTRAINT; Schema: public; Owner:
postgres

--

ALTER TABLE ONLY public.etkinliksanatici

ADD CONSTRAINT "etkinlikFK" FOREIGN KEY (etkinlikno) REFERENCES
public.etkinlik(etkinlikno);

--

-- Name: adres ilFK; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public.adres

ADD CONSTRAINT "ilFK" FOREIGN KEY (ilno) REFERENCES public.il(ilno);

--

-- Name: ilce ilFK; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public.ilce

ADD CONSTRAINT "ilFK" FOREIGN KEY (ilno) REFERENCES public.il(ilno);

--

-- Name: adres ilceFK; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public.adres

ADD CONSTRAINT "ilceFK" FOREIGN KEY (ilceno) REFERENCES
public.ilce(ilceno);

--

-- Name: bilet kategoriFK; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public.bilet

ADD CONSTRAINT "kategoriFK" FOREIGN KEY (kategorino) REFERENCES
public.kategori(kategorino);

--

-- Name: sanatci kategoriFK; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public.sanatci

ADD CONSTRAINT "kategoriFK" FOREIGN KEY (kategorino) REFERENCES
public.kategori(kategorino);

--

-- Name: bilet mekanFK; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

```
ALTER TABLE ONLY public.bilet
```

```
    ADD CONSTRAINT "mekanFK" FOREIGN KEY (mekanno) REFERENCES  
public.mekan(mekanno);
```

```
--
```

```
-- Name: salon mekanFK; Type: FK CONSTRAINT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.salon
```

```
    ADD CONSTRAINT "mekanFK" FOREIGN KEY (mekanno) REFERENCES  
public.mekan(mekanno);
```

```
--
```

```
-- Name: personel mekanFK; Type: FK CONSTRAINT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.personel
```

```
    ADD CONSTRAINT "mekanFK" FOREIGN KEY (mekanno) REFERENCES  
public.mekan(mekanno);
```

```
--
```

```
-- Name: etkinlik mekanFK; Type: FK CONSTRAINT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.etkinlik
```

```
    ADD CONSTRAINT "mekanFK" FOREIGN KEY (mekanno) REFERENCES  
public.mekan(mekanno);
```

--

-- Name: siparis musterFK; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public.siparis

ADD CONSTRAINT "musterFK" FOREIGN KEY (musterino) REFERENCES
public.muster(musterino);

--

-- Name: etkinlik sanatciFK; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public.etkinlik

ADD CONSTRAINT "sanatciFK" FOREIGN KEY (sanatcino) REFERENCES
public.sanatci(sanatcino);

--

-- Name: etkinliksanatci sanatciFK; Type: FK CONSTRAINT; Schema: public; Owner:
postgres

--

ALTER TABLE ONLY public.etkinliksanatci

ADD CONSTRAINT "sanatciFK" FOREIGN KEY (sanatcino) REFERENCES
public.sanatci(sanatcino);

--

-- Name: biletsiparis siparisFK; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

```
ALTER TABLE ONLY public.biletsiparis
```

```
    ADD CONSTRAINT "siparisFK" FOREIGN KEY (siparisno) REFERENCES  
public.siparis(siparisno) ON UPDATE CASCADE ON DELETE CASCADE;
```

```
--
```

```
-- Name: fatura siparisFK; Type: FK CONSTRAINT; Schema: public; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY public.fatura
```

```
    ADD CONSTRAINT "siparisFK" FOREIGN KEY (siparisno) REFERENCES  
public.siparis(siparisno) ON UPDATE CASCADE ON DELETE CASCADE;
```

```
--
```

```
-- PostgreSQL database dump complete
```

```
--
```


Saklı Yordam ve Tetikleyiciler

- **Saklı Yordam**

1. PersonelAra()

Create or replace Function personelAra(personelAdi varchar)

returns table

(

noSutun int,

adiSutun varchar(40),

soyadiSutun varchar(40),

mekannoSutun int,

telSutun varchar(40)

)

AS

\$\$

BEGIN

RETURN QUERY

SELECT personelno, adi, soyadi, mekanno, tel FROM personel WHERE adi
like personelAdi;

END;

\$\$

language "plpgsql";

----- kontrol kodu -----

Select * from personelAra('%a%');

2. musterisayisi()

CREATE OR REPLACE FUNCTION musterisayisi()

RETURNS INTEGER

AS

\$\$

DECLARE

```

        sonuc INTEGER;

BEGIN

    SELECT count(*) into sonuc from musteri;

    RETURN sonuc;

END;

$$

LANGUAGE 'plpgsql';

----- kontrol kodu -----

select musteriSayisi();

```

3. maxBiletFiyati()

```

CREATE OR REPLACE FUNCTION maxBiletFiyati()
RETURNS MONEY
AS
$$
DECLARE
    sonuc MONEY;
BEGIN
    SELECT max("birimFiyati") into sonuc from bilet;

    RETURN sonuc;

END;

$$

LANGUAGE 'plpgsql';

----- kontrol kodu -----

select maxBiletFiyati();

```

4. stokAralığı()

```

Create or replace Function stokAralığı(altStok INTEGER, ustStok INTEGER)
returns table
(

```

```

biletNoSutun int,
mekanNoSutun int,
salonNoSutun int,
stokMiktariSutun INTEGER,
kategoriNoSutun INTEGER,
etkinlikNoSutun INTEGER
)
as
$$
BEGIN

RETURN QUERY

SELECT biletno, mekanno, salonno, stokmiktari, kategorino, etkinlikno
FROM bilet WHERE stokmiktari BETWEEN altStok AND ustStok;
END;
$$
language "plpgsql";

----- kontrol kodu -----

Select * from stokAralığı(80,5000);

```

- **Tetikleyiciler**

- 1. musterisilindiginde**

```

CREATE OR REPLACE FUNCTION "eskimusteriTR1"()
RETURNS TRIGGER
AS
$$
BEGIN

INSERT INTO
public.eskimusteri("musterino","adi","soyadi","yas","cinsiyet","tel")

VALUES (old.musterino,old.adi,old.soyadi,old.yas,old.cinsiyet,old.tel);

```

```

        RETURN OLD;

END;

$$

LANGUAGE "plpgsql";

CREATE TRIGGER "musterisilindiginde"
after delete ON "musteri"
FOR EACH ROW
EXECUTE PROCEDURE "eskimusteriTR1"();

----- kontrol kodu -----

DELETE FROM "musteri"
WHERE "musterino" = '16';

```

2. faturaekleTR2

```

CREATE OR REPLACE FUNCTION "faturaekleTR2"()
RETURNS TRIGGER
AS
$$
BEGIN
    INSERT INTO public.fatura("siparisno","faturaTarihi")
    VALUES (new.siparisno,CURRENT_TIMESTAMP::TIMESTAMP);
    RETURN NEW;
END;
$$

LANGUAGE "plpgsql";

CREATE TRIGGER "faturaekleTR2"
after Insert ON "siparis"
FOR EACH ROW
EXECUTE PROCEDURE "faturaekleTR2"();

```

----- kontrol kodu -----

```
INSERT INTO "siparis"  
("musterino","siparistarihi")  
VALUES ('6','2021-11-25');
```

3. fiyatdegistiginde

```
CREATE OR REPLACE FUNCTION "fiyatdegisikligiTR3"()  
RETURNS TRIGGER  
AS  
$$  
BEGIN  
    IF NEW."birimFiyati" <> OLD."birimFiyati" THEN  
        INSERT INTO  
public.fiyatdegisikligi("biletno","eskibirimfiyati","yenibirimfiyati","degisik  
likTarihi")  
        VALUES  
(OLD."biletno",OLD."birimFiyati",NEW."birimFiyati",CURRENT_TIME  
STAMP::TIMESTAMP);  
    END IF;  
  
    RETURN NEW;  
END;  
$$  
LANGUAGE "plpgsql";
```

```
CREATE TRIGGER "fiyatdegistiginde"  
before update ON "bilet"  
FOR EACH ROW  
EXECUTE PROCEDURE "fiyatdegisikligiTR3"();
```

----- kontrol kodu -----

```
UPDATE "bilet" SET "birimFiyati" = '150' WHERE "biletno" = '1';
```

4. **kayitKontrol**

```
CREATE OR REPLACE FUNCTION "kayitKontrolTR4"()
RETURNS TRIGGER
AS
$$
BEGIN
    NEW."soyadi" = UPPER(NEW."soyadi");
    NEW."soyadi" = LTRIM(NEW."soyadi");
    NEW."adi" = LTRIM(NEW."adi");
    NEW."cinsiyet" = LTRIM(NEW."cinsiyet");
END;
$$
LANGUAGE "plpgsql";
```

```
CREATE TRIGGER "kayitKontrol"
BEFORE INSERT OR UPDATE ON "musteri"
FOR EACH ROW
EXECUTE PROCEDURE "kayitKontrolTR4"();
```

Arama, Ekleme, Silme, Güncelleme İşlemlerine Ait Ekran Görüntüleri

Arama İşlemi

Arama Yapmadan Önce

[← Geri](#)

Müşteri İşlemleri

Aranacak Kişi:

Ara

Müşteri No:

(Sadece Silme ve Güncelleme işlemleri için giriniz!)

Adı:

Soyadı:

Yaş:

Cinsiyet:

Telefon:

Müşteri Tablosu

Listele

musterino	adi	soyadi	cinsiyet	tel	yas
21	Efe	DURMUŞ	erkek	05364589874	20
22	Baş	KRÖMEROĞLU	erkek	05364589872	19
23	Ömer	UZUN	erkek	05364589741	25
2	Kemal	YILDIRIM	erkek	05621567483	56
1	Serdar	ARICI	erkek	05355555555	20
3	Ayşe	YILMAZ	kadın	05364581972	45
6	Ali	ÇAVUŞ	erkek	05335698452	19
9	Hasan	TÜRK	erkek	05345694512	74
24	Zeynep	ANADOLU	kadın	05335698547	56
25	Hasan	ELMA	erkek	05335698216	35
26	Hasan	YILDIZ	erkek	053912256847	27
27	Ekleme	İŞLEMI	belirtilmemiş	000000000	0
29	Güncelleme	İŞLEMI	kadın	4444444444	22

Ekle

Sil

Güncelle

Arama Yapıldığında

[← Geri](#)

Müşteri İşlemleri

Aranacak Kişi: Hasan

Ara

Müşteri No:

(Sadece Silme ve Güncelleme işlemleri için giriniz!)

Adı:

Soyadı:

Yaş:

Cinsiyet:

Telefon:

Müşteri Tablosu

Listele

musterino	adi	soyadi	cinsiyet	tel	yas
9	Hasan	TÜRK	erkek	05345694512	74
25	Hasan	ELMA	erkek	05335698216	35
26	Hasan	YILDIZ	erkek	053912256847	27

Ekle

Sil

Güncelle

Ekleme İşlemi

Ekleme Yapılırken

[Geri](#)

Müşteri İşlemleri

Müşteri No:

(Sadece Silme ve Güncelleme işlemleri için giriniz!)

Aranacak Kişi:

Ara

Müşteri No:

Adı:

Soyadı:

Yaş:

Cinsiyet:

Telefon:

Müşteri Tablosu

Listele

Bilgi

Müşteri Eklendi.

Tamam

musterino	adi	soyadi	cinsiyet	tel	yas
21	Efe	DURMUŞ	erkek	05364589874	20
22	Banş	KİROMEROĞLU	erkek	05364589872	19
23	Ömer	UZUN	erkek	05364589741	25
2	Kemal	YILDIRIM	erkek	05621567483	56
1	Serdar	ARICI	erkek	05355555555	20
3	Ayşe	YILMAZ	kadın	05364581972	45
6	Ali	ÇAVUŞ	erkek	05335698452	19
9	Hasan	TÜRK	erkek	05345694512	74
24	Zeynep	ANADOLU	kadın	05335698547	56
25	Hasan	ELMA	erkek	05335698216	35
26	Hasan	YILDIZ	erkek	053912256847	27
*					

Ekle

Sil

Güncelle

Ekleme Yapıldığında

[Geri](#)

Müşteri İşlemleri

Müşteri No:

(Sadece Silme ve Güncelleme işlemleri için giriniz!)

Aranacak Kişi:

Ara

Müşteri No:

Adı:

Soyadı:

Yaş:

Cinsiyet:

Telefon:

Müşteri Tablosu

Listele

musterino	adi	soyadi	cinsiyet	tel	yas
21	Efe	DURMUŞ	erkek	05364589874	20
22	Banş	KİROMEROĞLU	erkek	05364589872	19
23	Ömer	UZUN	erkek	05364589741	25
2	Kemal	YILDIRIM	erkek	05621567483	56
1	Serdar	ARICI	erkek	05355555555	20
3	Ayşe	YILMAZ	kadın	05364581972	45
6	Ali	ÇAVUŞ	erkek	05335698452	19
9	Hasan	TÜRK	erkek	05345694512	74
24	Zeynep	ANADOLU	kadın	05335698547	56
25	Hasan	ELMA	erkek	05335698216	35
26	Hasan	YILDIZ	erkek	053912256847	27
27	Ekleme	İŞLEME	belirtilmemiş	0000000000	0
*					

Ekle

Sil

Güncelle

Silme İşlemi

Silme İşlemi Yapılmadan Önce

← Geri

Müşteri İşlemleri

Müşteri No:

(Sadece Silme ve Güncelleme işlemleri için giriniz!)

Aranacak Kişi:

Ara

Müşteri Tablosu

Listele

Adı:

Soyadı:

Yaş:

Cinsiyet:

Telefon:

Ekle

Sil

Güncelle

musterino	adi	soyadi	cinsiyet	tel	yas
21	Efe	DURMUŞ	erkek	05364589874	20
22	Bağış	KROMEROĞLU	erkek	05364589872	19
23	Ömer	UZUN	erkek	05364589741	25
2	Kemal	YILDIRIM	erkek	05621567483	56
1	Serdar	ARICI	erkek	05355555555	20
3	Ayşe	YILMAZ	kadın	05364581972	45
6	Ali	ÇAVUŞ	erkek	05335698452	19
9	Hasan	TÜRK	erkek	05345694512	74
24	Zeynep	ANADOLU	kadın	05335698547	56
25	Hasan	ELMA	erkek	05335698216	35
26	Hasan	YILDIZ	erkek	053912256847	27
27	Ekleme	İŞLEMI	belirtilmemiş	0000000000	0
28	Silinecek	MÜŞTERİ	belirtilmemiş	33333333333	0

Silme İşlemi Yapılırken

← Geri

Müşteri İşlemleri

Müşteri No:

28

(Sadece Silme ve Güncelleme işlemleri için giriniz!)

Aranacak Kişi:

Ara

Müşteri Tablosu

Listele

Adı:

Soyadı:

Yaş:

Cinsiyet:

Telefon:

Ekle

Sil

Güncelle

musterino	adi	soyadi	cinsiyet	tel	yas
21	Efe	DURMUŞ	erkek	05364589874	20
22	Bağış	KROMEROĞLU	erkek	05364589872	19
23	Ömer	UZUN	erkek	05364589741	25
2	Kemal	YILDIRIM	erkek	05621567483	56
1	Serdar	ARICI	erkek	05355555555	20
3	Ayşe	YILMAZ	kadın	05364581972	45
6	Ali	ÇAVUŞ	erkek	05335698452	19
9	Hasan	TÜRK	erkek	05345694512	74
24	Zeynep	ANADOLU	kadın	05335698547	56
25	Hasan	ELMA	erkek	05335698216	35
26	Hasan	YILDIZ	erkek	053912256847	27
27	Ekleme	İŞLEMI	belirtilmemiş	0000000000	0
28	Silinecek	MÜŞTERİ	belirtilmemiş	33333333333	0

Bilgi
Müşteri Silindi
Tamam

Silme İşlemi Yapıldıktan Sonra

← Geri

Müşteri İşlemleri

Müşteri No: 28
(Sadece Silme ve Güncelleme işlemleri için giriniz!)

Aranacak Kişi:

Ara

Müşteri Tablosu

Listele

Adı:

Soyadı:

Yaş:

Cinsiyet:

Telefon:

Ekle

Sil

Güncelle

musterino	adi	soyadi	cinsiyet	tel	yas
21	Efe	DURMUŞ	erkek	05364589874	20
22	Baş	KİROMEROĞLU	erkek	05364589872	19
23	Ömer	UZUN	erkek	05364589741	25
2	Kemal	YILDIRIM	erkek	05621567483	56
1	Serdar	ARICI	erkek	05355555555	20
3	Ayşe	YILMAZ	kadın	05364581972	45
6	Ali	ÇAVUŞ	erkek	05335698452	19
9	Hasan	TÜRK	erkek	05345694512	74
24	Zeynep	ANADOLU	kadın	05335698547	56
25	Hasan	ELMA	erkek	05335698216	35
26	Hasan	YILDIZ	erkek	053912256847	27
27	Ekleme	İŞLEMI	belirtilmemiş	0000000000	0

Güncelleme İşlemi

Güncelleme İşlemi Yapılmadan Önce

← Geri

Müşteri İşlemleri

Müşteri No:

(Sadece Silme ve Güncelleme işlemleri için giriniz!)

Aranacak Kişi:

Ara

Müşteri Tablosu

Listele

Adı:

Soyadı:

Yaş:

Cinsiyet:

Telefon:

Ekle

Sil

Güncelle

musterino	adi	soyadi	cinsiyet	tel	yas
21	Efe	DURMUŞ	erkek	05364589874	20
22	Baş	KİROMEROĞLU	erkek	05364589872	19
23	Ömer	UZUN	erkek	05364589741	25
2	Kemal	YILDIRIM	erkek	05621567483	56
1	Serdar	ARICI	erkek	05355555555	20
3	Ayşe	YILMAZ	kadın	05364581972	45
6	Ali	ÇAVUŞ	erkek	05335698452	19
9	Hasan	TÜRK	erkek	05345694512	74
24	Zeynep	ANADOLU	kadın	05335698547	56
25	Hasan	ELMA	erkek	05335698216	35
26	Hasan	YILDIZ	erkek	053912256847	27
27	Ekleme	İŞLEMI	belirtilmemiş	0000000000	0
29	Güncelleme	İŞLEMI	erkek	7777777777	55

Güncelleme İşlemi Yapılırken

[Geri](#)

Müşteri İşlemleri

Müşteri No: 29
(Sadece Silme ve Güncelleme işlemleri için giriniz!)

Aranacak Kişi:

Ara

Müşteri Tablosu

Bilgi

Müşteri güncelleme işlemi başarılı bir şekilde gerçekleşti.

Tamam

musterino	adi	soyadı	cinsiyet	tel	yas
21	Efe	DURMUŞ	erkek	05364589874	20
22	Bağış	KRÖMEROĞLU	erkek	05364589872	19
23	Ömer	UZUN	erkek	05364589741	25
2	Kemal	YILDIRIM	erkek	05621567483	56
1	Serdar	ARICI	erkek	05355555555	20
3	Ayşe	YILMAZ	kadın	05364581972	45
6	Ali	ÇAVUŞ	erkek	05335698452	19
9	Hasan	TÜRK	erkek	05345694512	74
24	Zeynep	ANADOLU	kadın	05335698547	56
25	Hasan	ELMA	erkek	05335698216	35
26	Hasan	YILDIZ	erkek	053912256847	27
27	Ekleme	İŞLEME	belirtilmemiş	0000000000	0
29	Güncelleme	İŞLEME	erkek	7777777777	55

Ekle

Sil

Güncelle

Güncelleme İşlemi Yapıldıktan Sonra

[Geri](#)

Müşteri İşlemleri

Müşteri No: 29
(Sadece Silme ve Güncelleme işlemleri için giriniz!)

Aranacak Kişi:

Ara

Müşteri Tablosu

Listele

musterino	adi	soyadı	cinsiyet	tel	yas
21	Efe	DURMUŞ	erkek	05364589874	20
22	Bağış	KRÖMEROĞLU	erkek	05364589872	19
23	Ömer	UZUN	erkek	05364589741	25
2	Kemal	YILDIRIM	erkek	05621567483	56
1	Serdar	ARICI	erkek	05355555555	20
3	Ayşe	YILMAZ	kadın	05364581972	45
6	Ali	ÇAVUŞ	erkek	05335698452	19
9	Hasan	TÜRK	erkek	05345694512	74
24	Zeynep	ANADOLU	kadın	05335698547	56
25	Hasan	ELMA	erkek	05335698216	35
26	Hasan	YILDIZ	erkek	053912256847	27
27	Ekleme	İŞLEME	belirtilmemiş	0000000000	0
29	Güncelleme	İŞLEME	kadın	4444444444	22

Ekle

Sil

Güncelle

Uygulamanın Kaynak Kodları

Form1:

```
using Npgsql;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace biletSatisUygulamasi
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            NpgsqlConnection baglanti = new NpgsqlConnection("server=localhost; port=5432;
            Database=BiletSatisUygulamasi; user ID=postgres; password=12345");

            private void Form1_Load(object sender, EventArgs e)
            {
            }

            private void musteribtn_Click(object sender, EventArgs e)
            {
                Musteri yeni = new Musteri();
                yeni.Show();
                this.Hide();
            }

            private void buttoncikis_Click(object sender, EventArgs e)
            {
                Close();
            }

            private void biletbtn_Click(object sender, EventArgs e)
            {
                Bilet yeni = new Bilet();
                yeni.Show();
                this.Hide();
            }

            private void personelbtn_Click(object sender, EventArgs e)
            {
                Personel yeni = new Personel();
                yeni.Show();
                this.Hide();
            }

            private void etkinlikbtn_Click(object sender, EventArgs e)
            {
            }
        }
    }
}
```

```

        Etkinlik yeni = new Etkinlik();
        yeni.Show();
        this.Hide();
    }
}

```

FormMusteri:

```

using Npgsql;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace BiletSatisUygulamasi
{
    public partial class Musteri : Form
    {
        public Musteri()
        {
            InitializeComponent();

            NpgsqlConnection baglanti = new NpgsqlConnection("server=localhost; port=5432;
Database=BiletSatisUygulamasi; user ID=postgres; password=12345");
            private void button1_Click(object sender, EventArgs e)
            {
                string sorgu = "select * from musteriler";
                NpgsqlDataAdapter da = new NpgsqlDataAdapter(sorgu, baglanti);
                DataSet ds = new DataSet();
                da.Fill(ds);
                dataGridView1.DataSource = ds.Tables[0];
            }

            private void button2_Click(object sender, EventArgs e)
            {
                baglanti.Open();
                NpgsqlCommand komut = new NpgsqlCommand("insert into musteriler
(adi,soyadi,yas,cinsiyet,tel) values (@p1,@p2,@p3,@p4,@p5)", baglanti);
                komut.Parameters.AddWithValue("@p1", TxtAdi.Text);
                komut.Parameters.AddWithValue("@p2", TxtSoyadi.Text);
                komut.Parameters.AddWithValue("@p3", int.Parse(TxtYas.Text));
                komut.Parameters.AddWithValue("@p4", comboBox1.SelectedItem.ToString());
                komut.Parameters.AddWithValue("@p5", TxtTelefon.Text);
                komut.ExecuteNonQuery();
                baglanti.Close();
                MessageBox.Show("Müşteri Eklendi.", "Bilgi", MessageBoxButtons.OK,
MessageBoxIcon.Information);
            }

            private void buttonSil_Click(object sender, EventArgs e)
            {
                DialogResult dr = new DialogResult();
                dr = MessageBox.Show("Silmek istediğinize emin misiniz?", "Uyarı",
MessageBoxButtons.YesNo, MessageBoxIcon.Question);
            }
        }
    }
}

```

```

        if (dr == DialogResult.Yes)
        {
            baglanti.Open();
            NpgsqlCommand komut2 = new NpgsqlCommand("Delete from musteriler where
musterino=@p1", baglanti);
            komut2.Parameters.AddWithValue("@p1", int.Parse(TxtNo.Text));
            komut2.ExecuteNonQuery();
            MessageBox.Show("Müşteri Silindi", "Bilgi", MessageBoxButtons.OK,
MessageBoxIcon.Information);
            baglanti.Close();
        }
    }

    private void button3_Click(object sender, EventArgs e)
    {
        baglanti.Open();
        NpgsqlCommand komut3 = new NpgsqlCommand("update musteriler set
adi=@p1,soyadi=@p2,yas=@p3,cinsiyet=@p4,tel=@p5 where musterino=@p6", baglanti);
        komut3.Parameters.AddWithValue("@p1", TxtAdi.Text);
        komut3.Parameters.AddWithValue("@p2", TxtSoyadi.Text);
        komut3.Parameters.AddWithValue("@p3", int.Parse(TxtYas.Text));
        komut3.Parameters.AddWithValue("@p4", comboBox1.SelectedItem.ToString());
        komut3.Parameters.AddWithValue("@p5", TxtTelefon.Text);
        komut3.Parameters.AddWithValue("@p6", int.Parse(TxtNo.Text));
        komut3.ExecuteNonQuery();
        baglanti.Close();
        MessageBox.Show("Müşteri güncelleme işlemi başarılı bir şekilde
gerçekleşti.", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);
    }

    private void buttonAra_Click(object sender, EventArgs e)
    {
        baglanti.Open();
        NpgsqlCommand komut4 = new NpgsqlCommand("select * from musteriler where adi
like '%" + TxtAra.Text + "%'", baglanti);

        NpgsqlDataAdapter da = new NpgsqlDataAdapter(komut4);
        DataSet ds = new DataSet();
        da.Fill(ds);
        dataGridView1.DataSource = ds.Tables[0];

        baglanti.Close();
    }

    private void buttonGeri_Click(object sender, EventArgs e)
    {
        Form1 yeni = new Form1();
        yeni.Show();
        this.Hide();
    }
}

```

Video Adresi: <https://youtu.be/ItjSZI5j5ho>