



T.C

SAKARYA ÜNİVERSİTESİ

BİLGİSAYAR VE BİLİŞİM BİLİMLERİ
FAKÜLTESİ

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

Ders : Veritabanı Yönetim Sistemleri
Dönem : 2022-2023 Güz Dönemi

Dilara Yavuz
2B Grubu
G191210022
dilara.yavuz@ogr.sakarya.edu.tr

Uygulama Tanıtımı

Uygulama bir sinema uygulamasıdır. Programlama dili olarak C# ve veritabanı olarak postgresql kullanıldı. Veritabanında film, salon, seans, koltuk, personel, müşteri, bilet vb gibi tablolar bulunmaktadır. Bu uygulama sayesinde bu tablolara yeni bilgiler eklenebilir, bu bilgiler güncellenebilir veya silinebilir, listelenebilir ve istediğimiz verinin araması yapılabilir. Bunların dışında müşteriye bilet satışı yapılabilir. Uygulamayı hazırlarken bir sinema şirketinin kullandığı bir uygulama gibi tasarladım ve böylece yiyecek satışını ve bilet ve yiyecek satışı sonra fatura kesme işlemlerini kapsayan bir uygulama hazırladım.

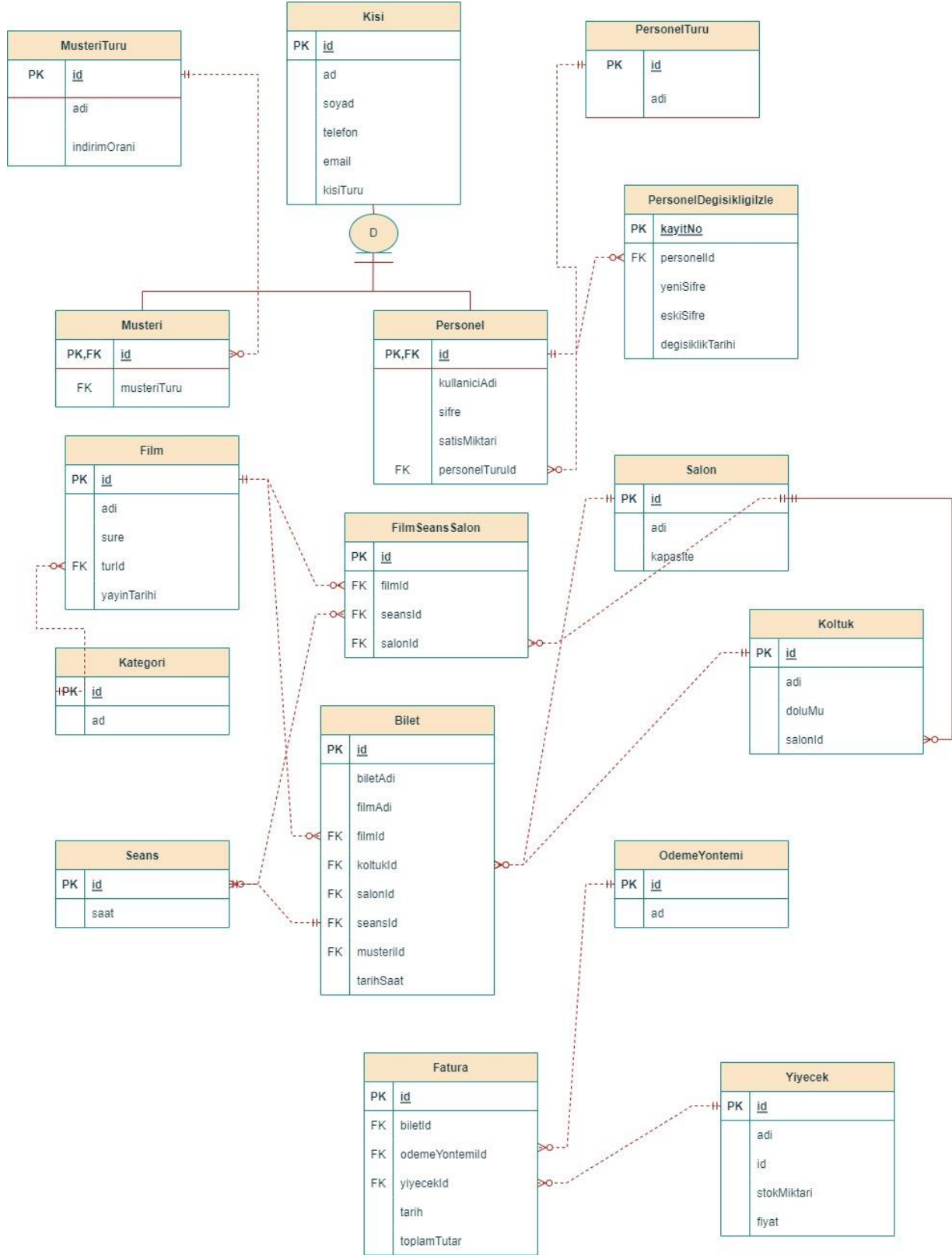
İş Kuralları

- Bir müşterinin bir bileti olabilir, bir bilet sadece bir müşteriye aittir.
- Bir filmin id'si, adı, süresi, türü, yayın tarihi bilgileri vardır.
- Film türlerinin id'si ve adı bilgileri mevcuttur.
- Bir filmin yalnızca bir türü olabilir, bir türde birden fazla film olabilir.
- Bir seansın id'si ve saat bilgileri vardır.
- Bir salonun id'si ve ad bilgileri vardır.
- Bir koltuğun id'si, adı, salon ve doluluk bilgileri vardır.
- Bir koltuk bir salona tam bağlıdır.
- Bir koltuk birden fazla salonda bulunabilir.
- Bir biletin id'si, bilet adı, film, koltuk, seans, salon, müşteri ve tarih bilgileri vardır.
- Bir yiyeceğin/içeceğin id'si adı ve stok miktarı bilgileri vardır.
- Bir faturanın id'si, tarih, toplam tutar ve ödeme yöntemi bilgileri vardır.
- Ödeme yönteminin is'si ve ad bilgileri tutulmalıdır.
- Bir faturanın bir ödeme yöntemi olur, bir ödeme yönteminde birden fazla fatura olabilir.
- Bir kişinin id'si, adı, soyadı, telefon, email ve kişi türü bilgileri vardır.
- Personel ve Müşteri olmak üzere 2 adet kişi türü vardır.
- Bir personelin kişi özelliklerinin yanında kullanıcı adı, şifre, personel türü ve satış miktarı bilgileri vardır.
- Bir müşterinin kişi bilgilerinin yanında müşteri türü bilgisi vardır.
- Müşteri türünün id'si, adı ve indirim oranı bilgileri vardır.
- Personel türünün id'si ve adı bilgileri vardır.
- Bir personel şifre değişikliği yapılması durumunda eski şifre, yeni şifre , değişim tarihi ve personel bilgileri ayrı bir tabloda tutulmalıdır.
- Bir müşteri türünde birden fazla müşteri bulunabilir, bir müşterinin sadece bir müşteri türü olabilir.
- Bir personel türünde birden fazla personel bulunabilir, bir personelin sadece bir personel türü olabilir.

İlişkisel Şema

- Kisi(**id: int**, adi: varchar(20), soyadi: varchar(20), telefon: varchar(11), email: varchar(20), kisiTuru: boolean)
- Musteri(**id: int**, adi: varchar(20), soyadi: varchar(20), telefon: varchar(11), email: varchar(20), kisiTuru varchar(1), musteriTuruId: int)
- Personel(**id: int**, adi: varchar(20), soyadi: varchar(20), telefon: varchar(11), email: varchar(20), kisiTuru varchar(1), personelTuruId: int, kullanıcıAdi: varchar(20), sifre: varchar(20), satisMiktari int)
- Bilet(**id: int**, biletAdi: varchar(20), filmAdi: varchar(20), filmId: int, koltukId: int, salonId: int, salonId: int, seansId: int, musterId: int, tarihSaat: timestamp)
- Fatura(**id: int**, biletId int, odemeYontemiId int, tarih: date, toplamTutar: money, yiyecekId: int)
- Film(**id: int**, adi: varchar(20), sure: int, turId: int, yayinTarihi: date)
- FilmSeansSalon(**id: int**, filmId: int, seansId: int, salonId: int)
- Kategori(**id: int**, ad: varchar(20))
- Koltuk(**id: int**, adi: varchar(20), doluMu: boolean, salonId int)
- MusteriTuru(**id: int**, adi: varchar(20), indirimOrani: real)
- OdemeYontemi(**id: int**, adi: varchar(20))
- PersonelDegisikligiIzle(**kayitNo: int**, degisiklikTarihi: timestamp, eskiSifre: varchar(20), yeniSifre: varchar(20), personelId: int)
- PersonelTuru(**id: int**, ad: varchar(20))
- Salon(**id: int**, adi: varchar(20), kapasite: int)
- Seans(**id: int**, saat: time)
- Yiyecek(**id: int**, adi: varchar(20), fiyat: money, stokMiktari: int)

Varlık Bağıntı modeli (Crow's Foot, Kalıtım)



Veritabanını, içerisindeki verilerle birlikte oluşturmayı sağlayan SQL ifadeleri

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

-- Dumped from database version 15.1
-- Dumped by pg_dump version 15rc2

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: Kisi; Type: SCHEMA; Schema: -; Owner: postgres
--

CREATE SCHEMA "Kisi";

ALTER SCHEMA "Kisi" OWNER TO postgres;

--
-- Name: Sinema; Type: SCHEMA; Schema: -; Owner: postgres
--

CREATE SCHEMA "Sinema";

ALTER SCHEMA "Sinema" OWNER TO postgres;

--
-- Name: userLogin(character varying, character varying); Type: FUNCTION; Schema: Kisi;
-- Owner: postgres
--

CREATE FUNCTION "Kisi"."userLogin"(username character varying, pass character varying)
RETURNS integer
    LANGUAGE plpgsql
    AS $$
    begin
        if(select count(*) from "Kisi"."Personel" where "Personel"."kullaniciAdi"="username" and
"Personel"."sifre"="pass") > 0 then
```

```
return 1;
else
return 0;
end if;
END;
$$;
```

```
ALTER FUNCTION "Kisi"."userLogin"(username character varying, pass character varying)
OWNER TO postgres;
```

```
--
-- Name: biletEkleTR1(); Type: FUNCTION; Schema: Sinema; Owner: postgres
--
```

```
CREATE FUNCTION "Sinema"."biletEkleTR1"() RETURNS trigger
LANGUAGE plpgsql
AS $$
BEGIN
NEW."biletAdi" = UPPER( OLD."filmId"||OLD."filmAdi" || OLD."id"); -- büyük harfe
dönüştürdükten sonra ekle
NEW."biletAdi" = TRIM(NEW."biletAdi"); -- Önceki ve sonraki boşlukları temizle
RETURN NEW;
END;
$$;
```

```
ALTER FUNCTION "Sinema"."biletEkleTR1"() OWNER TO postgres;
```

```
--
-- Name: kisiIdBul(character varying); Type: FUNCTION; Schema: Sinema; Owner: postgres
--
```

```
CREATE FUNCTION "Sinema"."kisiIdBul"(username character varying) RETURNS integer
LANGUAGE plpgsql
AS $$
begin
return (select "Kisi". "Kisi"."id" from "Kisi"."Kisi" where "Kisi". "Kisi"."adi"=username);
END;
$$;
```

```
ALTER FUNCTION "Sinema"."kisiIdBul"(username character varying) OWNER TO postgres;
```

```
--
-- Name: personelSifreDegisikligi(); Type: FUNCTION; Schema: Sinema; Owner: postgres
--
```

```
CREATE FUNCTION "Sinema"."personelSifreDegisikligi"() RETURNS trigger
LANGUAGE plpgsql
AS $$
BEGIN
```

```

    IF NEW."sifre" <> OLD."sifre" THEN
        INSERT INTO "Sinema"."PersonelDegisikligiIzle"("personelId", "eskiSifre",
"yeniSifre", "degisiklikTarihi")
            VALUES(OLD."id", OLD."sifre", NEW."sifre", CURRENT_TIMESTAMP);
        END IF;

    RETURN NEW;
END;
$$;

```

```

ALTER FUNCTION "Sinema"."personelSifreDegisikligi"() OWNER TO postgres;

```

```

--
-- Name: userLogin1$$$1043_1043(character varying, character varying); Type: FUNCTION;
Schema: Sinema; Owner: postgres
--

```

```

CREATE FUNCTION "Sinema"."userLogin1$$$1043_1043"(username character varying, pass
character varying) RETURNS integer
    LANGUAGE plpgsql
    AS $$
        begin
            if(select count(*) from "Sinema"."Personel" where "kullaniciAdi"="username" and
"sifre"="pass") > 0 then
                return 1;
            else
                return 0;
            end if;
        END;
    $$;

```

```

ALTER FUNCTION "Sinema"."userLogin1$$$1043_1043"(username character varying, pass
character varying) OWNER TO postgres;

```

```

--
-- Name: yoneticisiMi(character varying); Type: FUNCTION; Schema: Sinema; Owner: postgres
--

```

```

CREATE FUNCTION "Sinema"."yoneticisiMi"(username character varying) RETURNS integer
    LANGUAGE plpgsql
    AS $$
        begin

            if(select "Kisi"."Personel"."personelTuruId" from "Kisi"."Personel" where
"Kisi"."Personel"."kullaniciAdi"="username")=1 then
                return 1;
            else
                return 0;
            end if;
        END;
    $$;

```

\$\$;

ALTER FUNCTION "Sinema"."yoneticiMi"(username character varying) OWNER TO postgres;

SET default_tablespace = '';

SET default_table_access_method = heap;

--
-- Name: Kisi; Type: TABLE; Schema: Kisi; Owner: postgres
--

CREATE TABLE "Kisi"."Kisi" (
 id integer NOT NULL,
 adi character varying(20) NOT NULL,
 soyadi character varying(20) NOT NULL,
 telefon character varying(11),
 email character varying(50),
 "kisiTuru" boolean NOT NULL
);

ALTER TABLE "Kisi"."Kisi" OWNER TO postgres;

--
-- Name: Kisi_id_seq; Type: SEQUENCE; Schema: Kisi; Owner: postgres
--

CREATE SEQUENCE "Kisi"."Kisi_id_seq"
 AS integer
 START WITH 1
 INCREMENT BY 1
 NO MINVALUE
 NO MAXVALUE
 CACHE 1;

ALTER TABLE "Kisi"."Kisi_id_seq" OWNER TO postgres;

--
-- Name: Kisi_id_seq; Type: SEQUENCE OWNED BY; Schema: Kisi; Owner: postgres
--

ALTER SEQUENCE "Kisi"."Kisi_id_seq" OWNED BY "Kisi"."Kisi".id;

--
-- Name: Musteri; Type: TABLE; Schema: Kisi; Owner: postgres
--

CREATE TABLE "Kisi"."Musteri" (


```
id integer NOT NULL,  
"musteriTuruId" integer NOT NULL  
);
```

```
ALTER TABLE "Kisi"."Musteri" OWNER TO postgres;
```

```
--  
-- Name: Personel; Type: TABLE; Schema: Kisi; Owner: postgres  
--
```

```
CREATE TABLE "Kisi"."Personel" (  
id integer NOT NULL,  
"personelTuruId" integer,  
"kullaniciAdi" character varying(20) NOT NULL,  
sifre character varying(20) NOT NULL,  
"satisMiktari" integer DEFAULT 0  
);
```

```
ALTER TABLE "Kisi"."Personel" OWNER TO postgres;
```

```
--  
-- Name: Bilet; Type: TABLE; Schema: Sinema; Owner: postgres  
--
```

```
CREATE TABLE "Sinema"."Bilet" (  
id integer NOT NULL,  
"biletAdi" character varying(20) DEFAULT ' '::character varying,  
"filmId" integer,  
"salonId" integer NOT NULL,  
"seansId" integer NOT NULL,  
"koltukId" integer NOT NULL,  
"musteriId" integer NOT NULL,  
"tarihSaat" timestamp without time zone DEFAULT CURRENT_TIMESTAMP,  
"filmAdi" character varying(30) NOT NULL,  
"personelId" integer  
);
```

```
ALTER TABLE "Sinema"."Bilet" OWNER TO postgres;
```

```
--  
-- Name: Bilet_id_seq; Type: SEQUENCE; Schema: Sinema; Owner: postgres  
--
```

```
CREATE SEQUENCE "Sinema"."Bilet_id_seq"  
AS integer  
START WITH 1  
INCREMENT BY 1  
NO MINVALUE  
NO MAXVALUE
```

CACHE 1;

ALTER TABLE "Sinema"."Bilet_id_seq" OWNER TO postgres;

--
-- Name: Bilet_id_seq; Type: SEQUENCE OWNED BY; Schema: Sinema; Owner: postgres
--

ALTER SEQUENCE "Sinema"."Bilet_id_seq" OWNED BY "Sinema"."Bilet".id;

--
-- Name: Fatura; Type: TABLE; Schema: Sinema; Owner: postgres
--

CREATE TABLE "Sinema"."Fatura" (
 id integer NOT NULL,
 tarih date DEFAULT CURRENT_DATE,
 "odemeYontemiId" integer NOT NULL,
 "biletId" integer NOT NULL,
 "yiyecekId" integer NOT NULL,
 "toplamTutar" money
);

ALTER TABLE "Sinema"."Fatura" OWNER TO postgres;

--
-- Name: Fatura_id_seq; Type: SEQUENCE; Schema: Sinema; Owner: postgres
--

CREATE SEQUENCE "Sinema"."Fatura_id_seq"
 AS integer
 START WITH 1
 INCREMENT BY 1
 NO MINVALUE
 NO MAXVALUE
 CACHE 1;

ALTER TABLE "Sinema"."Fatura_id_seq" OWNER TO postgres;

--
-- Name: Fatura_id_seq; Type: SEQUENCE OWNED BY; Schema: Sinema; Owner: postgres
--

ALTER SEQUENCE "Sinema"."Fatura_id_seq" OWNED BY "Sinema"."Fatura".id;

--
-- Name: Film; Type: TABLE; Schema: Sinema; Owner: postgres

--

```
CREATE TABLE "Sinema"."Film" (  
    id integer NOT NULL,  
    adi character varying(20) NOT NULL,  
    "turAdi" integer NOT NULL,  
    sure integer NOT NULL,  
    "yayinTarihi" date,  
    fiyat money  
);
```

```
ALTER TABLE "Sinema"."Film" OWNER TO postgres;
```

--

```
-- Name: FilmSeansSalon; Type: TABLE; Schema: Sinema; Owner: postgres
```

--

```
CREATE TABLE "Sinema"."FilmSeansSalon" (  
    id integer NOT NULL,  
    "filmId" integer,  
    "salonId" integer,  
    "seansId" integer  
);
```

```
ALTER TABLE "Sinema"."FilmSeansSalon" OWNER TO postgres;
```

--

```
-- Name: FilmSeansSalon_id_seq; Type: SEQUENCE; Schema: Sinema; Owner: postgres
```

--

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

```
ALTER TABLE "Sinema"."FilmSeansSalon_id_seq" OWNER TO postgres;
```

--

```
-- Name: FilmSeansSalon_id_seq; Type: SEQUENCE OWNED BY; Schema: Sinema; Owner:  
postgres
```

--

```
ALTER SEQUENCE "Sinema"."FilmSeansSalon_id_seq" OWNED BY  
"Sinema"."FilmSeansSalon".id;
```

```
--  
-- Name: Film_id_seq; Type: SEQUENCE; Schema: Sinema; Owner: postgres  
--
```

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

```
ALTER TABLE "Sinema"."Film_id_seq" OWNER TO postgres;
```

```
--  
-- Name: Film_id_seq; Type: SEQUENCE OWNED BY; Schema: Sinema; Owner: postgres  
--
```

```
ALTER SEQUENCE "Sinema"."Film_id_seq" OWNED BY "Sinema"."Film".id;
```

```
--  
-- Name: Kategori; Type: TABLE; Schema: Sinema; Owner: postgres  
--
```

```
CREATE TABLE "Sinema"."Kategori" (  
  id integer NOT NULL,  
  adi character varying(20) NOT NULL  
);
```

```
ALTER TABLE "Sinema"."Kategori" OWNER TO postgres;
```

```
--  
-- Name: Kategori_id_seq; Type: SEQUENCE; Schema: Sinema; Owner: postgres  
--
```

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

```
ALTER TABLE "Sinema"."Kategori_id_seq" OWNER TO postgres;
```

```
--  
-- Name: Kategori_id_seq; Type: SEQUENCE OWNED BY; Schema: Sinema; Owner: postgres  
--
```

```
ALTER SEQUENCE "Sinema"."Kategori_id_seq" OWNED BY "Sinema"."Kategori".id;
```

```
--  
-- Name: Koltuk; Type: TABLE; Schema: Sinema; Owner: postgres  
--
```

```
CREATE TABLE "Sinema"."Koltuk" (  
    id integer NOT NULL,  
    "salonId" integer NOT NULL,  
    "doluMu" boolean NOT NULL,  
    adi character varying(2) NOT NULL  
);
```

```
ALTER TABLE "Sinema"."Koltuk" OWNER TO postgres;
```

```
--  
-- Name: Koltuk_id_seq; Type: SEQUENCE; Schema: Sinema; Owner: postgres  
--
```

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

```
ALTER TABLE "Sinema"."Koltuk_id_seq" OWNER TO postgres;
```

```
--  
-- Name: Koltuk_id_seq; Type: SEQUENCE OWNED BY; Schema: Sinema; Owner: postgres  
--
```

```
ALTER SEQUENCE "Sinema"."Koltuk_id_seq" OWNED BY "Sinema"."Koltuk".id;
```

```
--  
-- Name: MusteriTuru; Type: TABLE; Schema: Sinema; Owner: postgres  
--
```

```
CREATE TABLE "Sinema"."MusteriTuru" (  
    id integer NOT NULL,  
    adi character varying(20) NOT NULL,  
  
    "indirimOrani" real NOT NULL  
);
```

```
ALTER TABLE "Sinema"."MusteriTuru" OWNER TO postgres;
```

```
--  
-- Name: MusteriTuru_id_seq; Type: SEQUENCE; Schema: Sinema; Owner: postgres  
--
```

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

```
ALTER TABLE "Sinema"."MusteriTuru_id_seq" OWNER TO postgres;
```

```
--  
-- Name: MusteriTuru_id_seq; Type: SEQUENCE OWNED BY; Schema: Sinema; Owner: postgres  
--
```

```
ALTER SEQUENCE "Sinema"."MusteriTuru_id_seq" OWNED BY "Sinema"."MusteriTuru".id;
```

```
--  
-- Name: OdemeYontemi; Type: TABLE; Schema: Sinema; Owner: postgres  
--
```

```
CREATE TABLE "Sinema"."OdemeYontemi" (  
  id integer NOT NULL,  
  adi character varying(20)  
);
```

```
ALTER TABLE "Sinema"."OdemeYontemi" OWNER TO postgres;
```

```
--  
-- Name: OdemeYontemi_id_seq; Type: SEQUENCE; Schema: Sinema; Owner: postgres  
--
```

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

```
ALTER TABLE "Sinema"."OdemeYontemi_id_seq" OWNER TO postgres;
```

```
--
```

```
-- Name: OdemeYontemi_id_seq; Type: SEQUENCE OWNED BY; Schema: Sinema; Owner: postgres
```

```
--
```

```
ALTER SEQUENCE "Sinema"."OdemeYontemi_id_seq" OWNED BY "Sinema"."OdemeYontemi".id;
```

```
--
```

```
-- Name: PersonelDegisikligiIzle; Type: TABLE; Schema: Sinema; Owner: postgres
```

```
--
```

```
CREATE TABLE "Sinema"."PersonelDegisikligiIzle" (  
    "kayitNo" integer NOT NULL,  
    "personelId" smallint NOT NULL,  
    "eskiSifre" character varying(20),  
    "yeniSifre" character varying(20),  
    "degisiklikTarihi" timestamp without time zone DEFAULT CURRENT_TIMESTAMP  
);
```

```
ALTER TABLE "Sinema"."PersonelDegisikligiIzle" OWNER TO postgres;
```

```
--
```

```
-- Name: PersonelDegisikligiIzle_kayitNo_seq; Type: SEQUENCE; Schema: Sinema; Owner: postgres
```

```
--
```

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

```
ALTER TABLE "Sinema"."PersonelDegisikligiIzle_kayitNo_seq" OWNER TO postgres;
```

```
--
```

```
-- Name: PersonelDegisikligiIzle_kayitNo_seq; Type: SEQUENCE OWNED BY; Schema: Sinema; Owner: postgres
```

```
--
```

```
ALTER SEQUENCE "Sinema"."PersonelDegisikligiIzle_kayitNo_seq" OWNED BY "Sinema"."PersonelDegisikligiIzle"."kayitNo";
```

```
--
```

```
-- Name: PersonelTuru; Type: TABLE; Schema: Sinema; Owner: postgres
```

```
--
```

```
CREATE TABLE "Sinema"."PersonelTuru" (  
    id integer NOT NULL,  
    ad character varying(20) NOT NULL  
);
```

```
ALTER TABLE "Sinema"."PersonelTuru" OWNER TO postgres;
```

```
--  
-- Name: PersonelTuru_id_seq; Type: SEQUENCE; Schema: Sinema; Owner: postgres  
--
```

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

```
ALTER TABLE "Sinema"."PersonelTuru_id_seq" OWNER TO postgres;
```

```
--  
-- Name: PersonelTuru_id_seq; Type: SEQUENCE OWNED BY; Schema: Sinema; Owner:  
postgres  
--
```

```
ALTER SEQUENCE "Sinema"."PersonelTuru_id_seq" OWNED BY "Sinema"."PersonelTuru".id;
```

```
--  
-- Name: Salon; Type: TABLE; Schema: Sinema; Owner: postgres  
--
```

```
CREATE TABLE "Sinema"."Salon" (  
    id integer NOT NULL,  
    adi character varying(20) NOT NULL,  
    kapasite integer  
);
```

```
ALTER TABLE "Sinema"."Salon" OWNER TO postgres;
```

```
--  
-- Name: Salon_id_seq; Type: SEQUENCE; Schema: Sinema; Owner: postgres  
--
```

```
CREATE SEQUENCE "Sinema"."Salon_id_seq"  
    AS integer  
    START WITH 1  
    INCREMENT BY 1
```



```
NO MINVALUE
NO MAXVALUE
CACHE 1;
```

```
ALTER TABLE "Sinema"."Salon_id_seq" OWNER TO postgres;
```

```
--
-- Name: Salon_id_seq; Type: SEQUENCE OWNED BY; Schema: Sinema; Owner: postgres
--
```

```
ALTER SEQUENCE "Sinema"."Salon_id_seq" OWNED BY "Sinema"."Salon".id;
```

```
--
-- Name: Seans; Type: TABLE; Schema: Sinema; Owner: postgres
--
```

```
CREATE TABLE "Sinema"."Seans" (
  id integer NOT NULL,
  saat time without time zone NOT NULL
);
```

```
ALTER TABLE "Sinema"."Seans" OWNER TO postgres;
```

```
--
-- Name: Seans_id_seq; Type: SEQUENCE; Schema: Sinema; Owner: postgres
--
```

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

```
ALTER TABLE "Sinema"."Seans_id_seq" OWNER TO postgres;
```

```
--
-- Name: Seans_id_seq; Type: SEQUENCE OWNED BY; Schema: Sinema; Owner: postgres
--
```

```
ALTER SEQUENCE "Sinema"."Seans_id_seq" OWNED BY "Sinema"."Seans".id;
```

```
--
-- Name: Yiyecek; Type: TABLE; Schema: Sinema; Owner: postgres
--
```

```
CREATE TABLE "Sinema"."Yiyecek" (  
    id integer NOT NULL,  
    adi character varying(20) NOT NULL,  
    "stokMiktari" integer NOT NULL,  
    fiyat money  
);
```

```
ALTER TABLE "Sinema"."Yiyecek" OWNER TO postgres;
```

```
--  
-- Name: Yiyecek_id_seq; Type: SEQUENCE; Schema: Sinema; Owner: postgres  
--
```

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

```
ALTER TABLE "Sinema"."Yiyecek_id_seq" OWNER TO postgres;
```

```
--  
-- Name: Yiyecek_id_seq; Type: SEQUENCE OWNED BY; Schema: Sinema; Owner: postgres  
--
```

```
ALTER SEQUENCE "Sinema"."Yiyecek_id_seq" OWNED BY "Sinema"."Yiyecek".id;
```

```
--  
-- Name: Kisi id; Type: DEFAULT; Schema: Kisi; Owner: postgres  
--
```

```
ALTER TABLE ONLY "Kisi"."Kisi" ALTER COLUMN id SET DEFAULT  
nextval("Kisi"."Kisi_id_seq"::regclass);
```

```
--  
-- Name: Bilet id; Type: DEFAULT; Schema: Sinema; Owner: postgres  
--
```

```
ALTER TABLE ONLY "Sinema"."Bilet" ALTER COLUMN id SET DEFAULT  
nextval("Sinema"."Bilet_id_seq"::regclass);
```

```
--  
-- Name: Fatura id; Type: DEFAULT; Schema: Sinema; Owner: postgres  
--
```

```
ALTER TABLE ONLY "Sinema"."Fatura" ALTER COLUMN id SET DEFAULT
nextval('"Sinema"."Fatura_id_seq"::regclass);
```

```
--
-- Name: Film id; Type: DEFAULT; Schema: Sinema; Owner: postgres
--
```

```
ALTER TABLE ONLY "Sinema"."Film" ALTER COLUMN id SET DEFAULT
nextval('"Sinema"."Film_id_seq"::regclass);
```

```
--
-- Name: FilmSeansSalon id; Type: DEFAULT; Schema: Sinema; Owner: postgres
--
```

```
ALTER TABLE ONLY "Sinema"."FilmSeansSalon" ALTER COLUMN id SET DEFAULT
nextval('"Sinema"."FilmSeansSalon_id_seq"::regclass);
```

```
--
-- Name: Kategori id; Type: DEFAULT; Schema: Sinema; Owner: postgres
--
```

```
ALTER TABLE ONLY "Sinema"."Kategori" ALTER COLUMN id SET DEFAULT
nextval('"Sinema"."Kategori_id_seq"::regclass);
```

```
--
-- Name: Koltuk id; Type: DEFAULT; Schema: Sinema; Owner: postgres
--
```

```
ALTER TABLE ONLY "Sinema"."Koltuk" ALTER COLUMN id SET DEFAULT
nextval('"Sinema"."Koltuk_id_seq"::regclass);
```

```
--
-- Name: MusteriTuru id; Type: DEFAULT; Schema: Sinema; Owner: postgres
--
```

```
ALTER TABLE ONLY "Sinema"."MusteriTuru" ALTER COLUMN id SET DEFAULT
nextval('"Sinema"."MusteriTuru_id_seq"::regclass);
```

```
--
-- Name: OdemeYontemi id; Type: DEFAULT; Schema: Sinema; Owner: postgres
--
```

```
ALTER TABLE ONLY "Sinema"."OdemeYontemi" ALTER COLUMN id SET DEFAULT
nextval('"Sinema"."OdemeYontemi_id_seq"::regclass);
```

```
--
-- Name: PersonelDegisikligiIzle kayitNo; Type: DEFAULT; Schema: Sinema; Owner: postgres
--
```

```
ALTER TABLE ONLY "Sinema"."PersonelDegisikligiIzle" ALTER COLUMN "kayitNo" SET
DEFAULT nextval('"Sinema"."PersonelDegisikligiIzle_kayitNo_seq"::regclass);
```

```
--
-- Name: PersonelTuru id; Type: DEFAULT; Schema: Sinema; Owner: postgres
--
```

```
ALTER TABLE ONLY "Sinema"."PersonelTuru" ALTER COLUMN id SET DEFAULT
nextval('"Sinema"."PersonelTuru_id_seq"::regclass);
```

```
--
-- Name: Salon id; Type: DEFAULT; Schema: Sinema; Owner: postgres
--
```

```
ALTER TABLE ONLY "Sinema"."Salon" ALTER COLUMN id SET DEFAULT
nextval('"Sinema"."Salon_id_seq"::regclass);
```

```
--
-- Name: Seans id; Type: DEFAULT; Schema: Sinema; Owner: postgres
--
```

```
ALTER TABLE ONLY "Sinema"."Seans" ALTER COLUMN id SET DEFAULT
nextval('"Sinema"."Seans_id_seq"::regclass);
```

```
--
-- Name: Yiyecek id; Type: DEFAULT; Schema: Sinema; Owner: postgres
--
```

```
ALTER TABLE ONLY "Sinema"."Yiyecek" ALTER COLUMN id SET DEFAULT
nextval('"Sinema"."Yiyecek_id_seq"::regclass);
```

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

```
INSERT INTO "Kisi"."Kisi" VALUES
(1, 'Dilara', 'Yavuz', '01234567890', 'dilara@gmail.com', true),
(5, 'Betül', 'Yavuz', '01234567892', 'betul@gmail.com', false),
(16, 'müşteri', 'müşteri', '01234567893', 'müşteri', false),
(17, 'm', 'm', 'm', 'm', false),
(19, 'musteri', 'musteri', '01234567896', 'musteri@gmail.com', false),
(20, 'Ayşe', 'Kaya', '01234567892', 'ayse@gmail.com', true),
(21, 'Müşteri', 'Müşteri', '01234567893', 'musteri@gmail.com', false);
```

```
--  
-- Data for Name: Musteri; Type: TABLE DATA; Schema: Kisi; Owner: postgres  
--
```

```
INSERT INTO "Kisi"."Musteri" VALUES  
    (16, 1),  
    (17, 2),  
    (19, 1),  
    (21, 1);
```

```
--  
-- Data for Name: Personel; Type: TABLE DATA; Schema: Kisi; Owner: postgres  
--
```

```
INSERT INTO "Kisi"."Personel" VALUES  
    (1, 1, 'admin', 'admin', 0),  
    (20, 2, 'ayse kaya', 'ayse kaya', 0);
```

```
--  
-- Data for Name: Bilet; Type: TABLE DATA; Schema: Sinema; Owner: postgres  
--
```

```
INSERT INTO "Sinema"."Bilet" VALUES  
    (10, '3FORREST GUMP10', 3, 2, 5, 16, 16, '2022-12-25 16:04:35.122573', 'Forrest Gump',  
    NULL),  
    (11, '1PULP FICTION11', 1, 1, 2, 10, 17, '2022-12-25 16:08:40.812374', 'Pulp Fiction',  
    NULL),  
    (12, '1PULP FICTION12', 1, 2, 2, 16, 19, '2022-12-26 12:51:29.63832', 'Pulp Fiction',  
    NULL),  
    (13, '4MATRIX13', 4, 2, 4, 15, 21, '2022-12-26 15:54:03.106956', 'MATRIX', NULL);
```

```
--  
-- Data for Name: Fatura; Type: TABLE DATA; Schema: Sinema; Owner: postgres  
--
```

```
INSERT INTO "Sinema"."Fatura" VALUES  
    (12, '2022-12-26', 1, 12, 6, '?10,00'),  
    (13, '2022-12-26', 2, 13, 9, NULL);
```

```
--  
-- Data for Name: Film; Type: TABLE DATA; Schema: Sinema; Owner: postgres  
--
```

```
INSERT INTO "Sinema"."Film" VALUES  
    (1, 'Pulp Fiction', 2, 140, '1994-01-01', '?30,00'),  
    (2, 'Lord Of The Rings', 5, 180, '2001-01-01', '?40,00'),
```

```
(3, 'Forrest Gump', 3, 185, '1994-01-01', '?50,00'),  
(4, 'MATRIX', 5, 180, '1999-02-02', '?30,00');
```

```
--  
-- Data for Name: FilmSeansSalon; Type: TABLE DATA; Schema: Sinema; Owner: postgres  
--
```

```
--  
-- Data for Name: Kategori; Type: TABLE DATA; Schema: Sinema; Owner: postgres  
--
```

```
INSERT INTO "Sinema"."Kategori" VALUES
```

```
    (1, 'Korku'),  
    (2, 'Gerilim'),  
    (3,  
 'Komedi'),  
    (4, 'Romantik'),  
    (5, 'Fantastik');
```

```
--  
-- Data for Name: Koltuk; Type: TABLE DATA; Schema: Sinema; Owner: postgres  
--
```

```
INSERT INTO "Sinema"."Koltuk" VALUES
```

```
    (10, 1, false, 'A1'),  
    (11, 1, false, 'A2'),  
    (12, 1, false, 'B1'),  
    (13, 1, false, 'B2'),  
    (15, 2, false, 'A1'),  
    (16, 2, false, 'B2'),  
    (17, 3, false, 'A1'),  
    (18, 3, false, 'B1'),  
    (19, 4, false, 'A1'),  
    (20, 4, false, 'A2'),  
    (21, 4, false, 'B1');
```

```
--  
-- Data for Name: MusteriTuru; Type: TABLE DATA; Schema: Sinema; Owner: postgres  
--
```

```
INSERT INTO "Sinema"."MusteriTuru" VALUES
```

```
    (1, 'Öğrenci', 25),  
    (2, 'VIP', 50);
```

```
--  
-- Data for Name: OdemeYontemi; Type: TABLE DATA; Schema: Sinema; Owner: postgres
```

--

```
INSERT INTO "Sinema"."OdemeYontemi" VALUES
(1, 'Nakit'),
(2, 'Kart');
```

--

-- Data for Name: PersonelDegisikligiIzle; Type: TABLE DATA; Schema: Sinema; Owner: postgres

--

```
INSERT INTO "Sinema"."PersonelDegisikligiIzle" VALUES
(1, 2, 'rabia', 'rabiaA', '2022-12-24 23:26:36.740404'),
(2, 18, 'personel', 'personel1', '2022-12-26 12:46:11.888798'),
(3, 20, 'ayse', 'ayse kaya', '2022-12-26 14:06:33.041062');
```

--

-- Data for Name: PersonelTuru; Type: TABLE DATA; Schema: Sinema; Owner: postgres

--

```
INSERT INTO "Sinema"."PersonelTuru" VALUES
(1, 'Yönetici'),
(2, 'Biletci');
```

--

-- Data for Name: Salon; Type: TABLE DATA; Schema: Sinema; Owner: postgres

--

```
INSERT INTO "Sinema"."Salon" VALUES
(1, 'Salon1', 50),
(2, 'Salon2', 100),
(3, 'Salon3', 75),
(4, 'Salon4', 50);
```

--

-- Data for Name: Seans; Type: TABLE DATA; Schema: Sinema; Owner: postgres

--

```
INSERT INTO "Sinema"."Seans" VALUES
(2, '10:20:00'),
(3, '09:00:00'),
(4, '12:00:00'),
(5, '14:30:00'),
(6, '18:00:00');
```

--

-- Data for Name: Yiyecek; Type: TABLE DATA; Schema: Sinema; Owner: postgres

--

```
INSERT INTO "Sinema"."Yiyecek" VALUES
    (6, 'Patlamış Mısır', 10, NULL),
    (7, 'Su', 50, NULL),
    (8, 'Kola', 25, NULL),
    (9, 'Çikolata', 30, NULL);
```

```
--
-- Name: Kisi_id_seq; Type: SEQUENCE SET; Schema: Kisi; Owner: postgres
--
```

```
SELECT pg_catalog.setval('"Kisi"."Kisi_id_seq"', 21, true);
```

```
--
-- Name: Bilet_id_seq; Type: SEQUENCE SET; Schema: Sinema; Owner: postgres
--
```

```
SELECT pg_catalog.setval('"Sinema"."Bilet_id_seq"', 13, true);
```

```
--
-- Name: Fatura_id_seq; Type: SEQUENCE SET; Schema: Sinema; Owner: postgres
--
```

```
SELECT pg_catalog.setval('"Sinema"."Fatura_id_seq"', 13, true);
```

```
--
-- Name: FilmSeansSalon_id_seq; Type: SEQUENCE SET; Schema: Sinema; Owner: postgres
--
```

```
SELECT pg_catalog.setval('"Sinema"."FilmSeansSalon_id_seq"', 1, false);
```

```
--
-- Name: Film_id_seq; Type: SEQUENCE SET; Schema: Sinema; Owner: postgres
--
```

```
SELECT pg_catalog.setval('"Sinema"."Film_id_seq"', 4, true);
```

```
--
-- Name: Kategori_id_seq; Type: SEQUENCE SET; Schema: Sinema; Owner: postgres
--
```

```
SELECT pg_catalog.setval('"Sinema"."Kategori_id_seq"', 1, false);
```

```
--
-- Name: Koltuk_id_seq; Type: SEQUENCE SET; Schema: Sinema; Owner: postgres
```


--

```
SELECT pg_catalog.setval('"Sinema"."Koltuk_id_seq"', 1, false);
```

--

-- Name: MusteriTuru_id_seq; Type: SEQUENCE SET; Schema: Sinema; Owner: postgres

--

```
SELECT pg_catalog.setval('"Sinema"."MusteriTuru_id_seq"', 1, false);
```

--

-- Name: OdemeYontemi_id_seq; Type: SEQUENCE SET; Schema: Sinema; Owner: postgres

--

```
SELECT pg_catalog.setval('"Sinema"."OdemeYontemi_id_seq"', 2, true);
```

--

-- Name: PersonelDegisikligiIzle_kayitNo_seq; Type: SEQUENCE SET; Schema: Sinema; Owner: postgres

--

```
SELECT pg_catalog.setval('"Sinema"."PersonelDegisikligiIzle_kayitNo_seq"', 3, true);
```

--

-- Name: PersonelTuru_id_seq; Type: SEQUENCE SET; Schema: Sinema; Owner: postgres

--

```
SELECT pg_catalog.setval('"Sinema"."PersonelTuru_id_seq"', 1, false);
```

--

-- Name: Salon_id_seq; Type: SEQUENCE SET; Schema: Sinema; Owner: postgres

--

```
SELECT pg_catalog.setval('"Sinema"."Salon_id_seq"', 1, false);
```

--

-- Name: Seans_id_seq; Type: SEQUENCE SET; Schema: Sinema; Owner: postgres

--

```
SELECT pg_catalog.setval('"Sinema"."Seans_id_seq"', 1, false);
```

--

-- Name: Yiyecek_id_seq; Type: SEQUENCE SET; Schema: Sinema; Owner: postgres

--

```
SELECT pg_catalog.setval('"Sinema"."Yiyecek_id_seq"', 9, true);
```

```
--  
-- Name: Kisi Kisi_pkey; Type: CONSTRAINT; Schema: Kisi; Owner: postgres  
--
```

```
ALTER TABLE ONLY "Kisi"."Kisi"  
  ADD CONSTRAINT "Kisi_pkey" PRIMARY KEY (id);
```

```
--  
-- Name: Musteri Musteri_pkey; Type: CONSTRAINT; Schema: Kisi; Owner: postgres  
--
```

```
ALTER TABLE ONLY "Kisi"."Musteri"  
  ADD CONSTRAINT "Musteri_pkey" PRIMARY KEY (id);
```

```
--  
-- Name: Personel Personel_pkey; Type: CONSTRAINT; Schema: Kisi; Owner: postgres  
--
```

```
ALTER TABLE ONLY "Kisi"."Personel"  
  ADD CONSTRAINT "Personel_pkey" PRIMARY KEY (id);
```

```
--  
-- Name: PersonelDegisikligiIzle PK; Type: CONSTRAINT; Schema: Sinema; Owner: postgres  
--
```

```
ALTER TABLE ONLY "Sinema"."PersonelDegisikligiIzle"  
  ADD CONSTRAINT "PK" PRIMARY KEY ("kayitNo");
```

```
--  
-- Name: Bilet biletPK; Type: CONSTRAINT; Schema: Sinema; Owner: postgres  
--
```

```
ALTER TABLE ONLY "Sinema"."Bilet"  
  ADD CONSTRAINT "biletPK" PRIMARY KEY (id);
```

```
--  
-- Name: Fatura faturaPK; Type: CONSTRAINT; Schema: Sinema; Owner: postgres  
--
```

```
ALTER TABLE ONLY "Sinema"."Fatura"  
  ADD CONSTRAINT "faturaPK" PRIMARY KEY (id);
```

```
--
```

-- Name: Film filmPK; Type: CONSTRAINT; Schema: Sinema; Owner: postgres

--

```
ALTER TABLE ONLY "Sinema"."Film"  
  ADD CONSTRAINT "filmPK" PRIMARY KEY (id);
```

--

-- Name: FilmSeansSalon filmSeansSalonPK; Type: CONSTRAINT; Schema: Sinema; Owner: postgres

--

```
ALTER TABLE ONLY "Sinema"."FilmSeansSalon"  
  ADD CONSTRAINT "filmSeansSalonPK" PRIMARY KEY (id);
```

--

-- Name: Kategori kategoriPK; Type: CONSTRAINT; Schema: Sinema; Owner: postgres

--

```
ALTER TABLE ONLY "Sinema"."Kategori"  
  ADD CONSTRAINT "kategoriPK" PRIMARY KEY (id);
```

--

-- Name: Koltuk koltukPK; Type: CONSTRAINT; Schema: Sinema; Owner: postgres

--

```
ALTER TABLE ONLY "Sinema"."Koltuk"  
  ADD CONSTRAINT "koltukPK" PRIMARY KEY (id);
```

--

-- Name: OdemeYontemi odemeYontemiPK; Type: CONSTRAINT; Schema: Sinema; Owner: postgres

--

```
ALTER TABLE ONLY "Sinema"."OdemeYontemi"  
  ADD CONSTRAINT "odemeYontemiPK" PRIMARY KEY (id);
```

--

-- Name: MusteriTuru personelPK; Type: CONSTRAINT; Schema: Sinema; Owner: postgres

--

```
ALTER TABLE ONLY "Sinema"."MusteriTuru"  
  ADD CONSTRAINT "personelPK" PRIMARY KEY (id);
```

--

-- Name: PersonelTuru personelTuruPK; Type: CONSTRAINT; Schema: Sinema; Owner: postgres

--

```
ALTER TABLE ONLY "Sinema"."PersonelTuru"
  ADD CONSTRAINT "personelTuruPK" PRIMARY KEY (id);

--
-- Name: Salon salonPK; Type: CONSTRAINT; Schema: Sinema; Owner: postgres
--

ALTER TABLE ONLY "Sinema"."Salon"
  ADD CONSTRAINT "salonPK" PRIMARY KEY (id);

--
-- Name: Seans seansPK; Type: CONSTRAINT; Schema: Sinema; Owner: postgres
--

ALTER TABLE ONLY "Sinema"."Seans"
  ADD CONSTRAINT "seansPK" PRIMARY KEY (id);

--
-- Name: Yiyecek yiyecekPK; Type: CONSTRAINT; Schema: Sinema; Owner: postgres
--

ALTER TABLE ONLY "Sinema"."Yiyecek"
  ADD CONSTRAINT "yiyecekPK" PRIMARY KEY (id);

--
-- Name: index_email; Type: INDEX; Schema: Kisi; Owner: postgres
--

CREATE INDEX index_email ON "Kisi"."Kisi" USING btree (email);

--
-- Name: index_telefon; Type: INDEX; Schema: Kisi; Owner: postgres
--

CREATE INDEX index_telefon ON "Kisi"."Kisi" USING btree (telefon);

--
-- Name: index_adi; Type: INDEX; Schema: Sinema; Owner: postgres
--

CREATE INDEX index_adi ON "Sinema"."Koltuk" USING btree (adi);

--
-- Name: Personel personelBilgiDegisimi; Type: TRIGGER; Schema: Kisi; Owner: postgres
```

--

```
CREATE TRIGGER "personelBilgiDegisimi" BEFORE UPDATE ON "Kisi"."Personel" FOR
EACH ROW EXECUTE FUNCTION "Sinema"."personelSifreDegisikligi"();
```

--

-- Name: Bilet biletKontrol; Type: TRIGGER; Schema: Sinema; Owner: postgres

--

```
CREATE TRIGGER "biletKontrol" AFTER INSERT OR UPDATE ON "Sinema"."Bilet" FOR
EACH ROW EXECUTE FUNCTION "Sinema"."biletEkleTR1"();
```

--

-- Name: Bilet biletKontroll; Type: TRIGGER; Schema: Sinema; Owner: postgres

--

```
CREATE TRIGGER "biletKontroll" AFTER INSERT OR UPDATE ON "Sinema"."Bilet" FOR
EACH ROW EXECUTE FUNCTION "Sinema"."biletEkleTR1"();
```

--

-- Name: Bilet biletKontrolu; Type: TRIGGER; Schema: Sinema; Owner: postgres

--

```
CREATE TRIGGER "biletKontrolu" BEFORE INSERT OR UPDATE ON "Sinema"."Bilet" FOR
EACH ROW EXECUTE FUNCTION "Sinema"."biletEkleTR1"();
```

--

-- Name: Musteri musteriKisi1; Type: FK CONSTRAINT; Schema: Kisi; Owner: postgres

--

```
ALTER TABLE ONLY "Kisi"."Musteri"
```

```
ADD CONSTRAINT "musteriKisi1" FOREIGN KEY (id) REFERENCES "Kisi"."Kisi"(id) ON
UPDATE CASCADE ON DELETE CASCADE;
```

--

-- Name: Musteri musteriTuruFK; Type: FK CONSTRAINT; Schema: Kisi; Owner: postgres

--

```
ALTER TABLE ONLY "Kisi"."Musteri"
```

```
ADD CONSTRAINT "musteriTuruFK" FOREIGN KEY ("musteriTuruId") REFERENCES
"Sinema"."MusteriTuru"(id) MATCH FULL ON UPDATE CASCADE ON DELETE CASCADE;
```

--

-- Name: Personel personelKisi1; Type: FK CONSTRAINT; Schema: Kisi; Owner: postgres

--

```
ALTER TABLE ONLY "Kisi"."Personel"
```

```
ADD CONSTRAINT "personelKisi1" FOREIGN KEY (id) REFERENCES "Kisi"."Kisi"(id) ON  
UPDATE CASCADE ON DELETE CASCADE;
```

```
--
```

```
-- Name: Personel personelTuruFK; Type: FK CONSTRAINT; Schema: Kisi; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY "Kisi"."Personel"
```

```
ADD CONSTRAINT "personelTuruFK" FOREIGN KEY ("personelTuruId") REFERENCES  
"Sinema"."PersonelTuru"(id) MATCH FULL ON UPDATE CASCADE ON DELETE CASCADE;
```

```
--
```

```
-- Name: Bilet biletFK1; Type: FK CONSTRAINT; Schema: Sinema; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY "Sinema"."Bilet"
```

```
ADD CONSTRAINT "biletFK1" FOREIGN KEY ("filmId") REFERENCES  
"Sinema"."Film"(id);
```

```
--
```

```
-- Name: Bilet biletFK2; Type: FK CONSTRAINT; Schema: Sinema; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY "Sinema"."Bilet"
```

```
ADD CONSTRAINT "biletFK2" FOREIGN KEY ("salonId") REFERENCES  
"Sinema"."Salon"(id);
```

```
--
```

```
-- Name: Bilet biletFK3; Type: FK CONSTRAINT; Schema: Sinema; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY "Sinema"."Bilet"
```

```
ADD CONSTRAINT "biletFK3" FOREIGN KEY ("seansId") REFERENCES  
"Sinema"."Seans"(id);
```

```
--
```

```
-- Name: Bilet biletFK4; Type: FK CONSTRAINT;  
Schema: Sinema; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY "Sinema"."Bilet"
```

```
ADD CONSTRAINT "biletFK4" FOREIGN KEY ("koltukId") REFERENCES  
"Sinema"."Koltuk"(id);
```

```
--
```

-- Name: Fatura faturaFK; Type: FK CONSTRAINT; Schema: Sinema; Owner: postgres

--

ALTER TABLE ONLY "Sinema"."Fatura"

ADD CONSTRAINT "faturaFK" FOREIGN KEY ("odemeYontemiId") REFERENCES
"Sinema"."OdemeYontemi"(id);

--

-- Name: Film filmFK; Type: FK CONSTRAINT; Schema: Sinema; Owner: postgres

--

ALTER TABLE ONLY "Sinema"."Film"

ADD CONSTRAINT "filmFK" FOREIGN KEY ("turAdi") REFERENCES
"Sinema"."Kategori"(id);

--

-- Name: Fatura filmFK1; Type: FK CONSTRAINT; Schema: Sinema; Owner: postgres

--

ALTER TABLE ONLY "Sinema"."Fatura"

ADD CONSTRAINT "filmFK1" FOREIGN KEY ("biletId") REFERENCES
"Sinema"."Bilet"(id);

--

-- Name: Fatura filmFK2; Type: FK CONSTRAINT; Schema: Sinema; Owner: postgres

--

ALTER TABLE ONLY "Sinema"."Fatura"

ADD CONSTRAINT "filmFK2" FOREIGN KEY ("yiyecekId") REFERENCES
"Sinema"."Yiyecek"(id);

--

-- Name: FilmSeansSalon filmSeansSalonFK1; Type: FK CONSTRAINT; Schema: Sinema;
Owner: postgres

--

ALTER TABLE ONLY "Sinema"."FilmSeansSalon"

ADD CONSTRAINT "filmSeansSalonFK1" FOREIGN KEY ("filmId") REFERENCES
"Sinema"."Film"(id);

--

-- Name: FilmSeansSalon filmSeansSalonFK2; Type: FK CONSTRAINT; Schema: Sinema;
Owner: postgres

--

ALTER TABLE ONLY "Sinema"."FilmSeansSalon"

ADD CONSTRAINT "filmSeansSalonFK2" FOREIGN KEY ("salonId") REFERENCES

```
"Sinema"."Salon"(id);
```

```
--
```

```
-- Name: FilmSeansSalon filmSeansSalonFK3; Type: FK CONSTRAINT; Schema: Sinema;  
Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY "Sinema"."FilmSeansSalon"
```

```
ADD CONSTRAINT "filmSeansSalonFK3" FOREIGN KEY ("seansId") REFERENCES  
"Sinema"."Seans"(id);
```

```
--
```

```
-- Name: Koltuk koltukFK; Type: FK CONSTRAINT; Schema: Sinema; Owner: postgres
```

```
--
```

```
ALTER TABLE ONLY "Sinema"."Koltuk"
```

```
ADD CONSTRAINT "koltukFK" FOREIGN KEY ("salonId") REFERENCES  
"Sinema"."Salon"(id);
```

```
--
```

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

```
--
```


Saklı Yordam/ Fonksiyonlar ve Triggerlar

Saklı Yordam/ Fonksiyonlar

- -- CREATE FUNCTION "userLogin(varchar, varchar)" -----

```
CREATE OR REPLACE FUNCTION "Kisi"."userLogin"(username character varying, pass
character varying)
RETURNS integer
LANGUAGE plpgsql
AS $function$
begin
    if(select count(*) from "Kisi"."Personel" where
"Personel"."kullaniciAdi"="username" and "Personel"."sifre"="pass") > 0 then
        return 1;
    else
        return 0;
    end if;
END;
$function$;
```

userLogin fonksiyonu uygulama login kısmı sırasında girilen kullanıcı adı ve şifre bilgilerinin herhangi bir personele ait olup olmadığını ve doğruluğunu kontrol eder. Bilgilerin doğru olması durumunda 1 değerini, olmaması durumunda ise 0 değerini döndürür.

- -- CREATE FUNCTION "kisiIdBul(varchar)" -----

```
CREATE OR REPLACE FUNCTION "Sinema"."kisiIdBul"(username character varying)
RETURNS integer
LANGUAGE plpgsql
AS $function$
begin
    return (select "Kisi". "Kisi"."id" from "Kisi"."Kisi" where "Kisi".
"Kisi"."adi"=username);
END;
$function$;
```

kisiIdBul fonksiyonu uygulamanın bir çok yerinde tablo oluşturmak, spesifik bir kişinin bilgilerini getirmek için kullanıldı. Girilen parametredeki veriyle uyuşan kişinin id'sini bize geri döndürür.

- -- CREATE FUNCTION "yoneticiMi(varchar)" -----

```
CREATE OR REPLACE FUNCTION "Sinema"."yoneticiMi"(username character varying)
RETURNS integer
LANGUAGE plpgsql
AS $function$
begin

    if(select "Kisi"."Personel"."personelTuruId" from "Kisi"."Personel" where
"Kisi"."Personel"."kullaniciAdi"="username")=1 then
        return 1;
    else
        return 0;
    end if;
END;
$function$;
```

yoneticiMi fonksiyonu uygulamanın login bölümünde giriş yapan personelin yönetici mi yoksa bir biletçi mi olup olmadığını kontrol eder. Yönetici ise 1, biletçi ise 0 değerini döndürür. Giriş yapan personelin yönetici olmaması durumunda personelin işlemlerini kısıtlamak amacıyla direkt olarak uygulamanın satış kısmına yönlendirilir.

Triggerlar

- -- CREATE FUNCTION "biletEkleTR1()" -----

```
CREATE OR REPLACE FUNCTION "Sinema"."biletEkleTR1"()
RETURNS trigger
LANGUAGE plpgsql
AS $function$
BEGIN
    NEW."biletAdi" = UPPER( OLD."filmId"||OLD."filmAdi" || OLD."id"); -- büyük harfe
dönüştürdükten sonra ekle
    NEW."biletAdi" = TRIM(NEW."biletAdi"); -- Önceki ve sonraki boşlukları temizle
    RETURN NEW;
END;
$function$;
```

biletEkleTR1, bilet adını otomatik olarak oluşturmayı sağlayan bir triggerdır. Bilet oluşturulduktan sonra film adı, film id ve bilet id bilgilerini birleştirerek bilet için bir isim oluşturur. Bu isimin bütün harflerinin büyük olması için upper ve boşlukları kaldırmak için ise trim fonksiyonu kullanıldı.

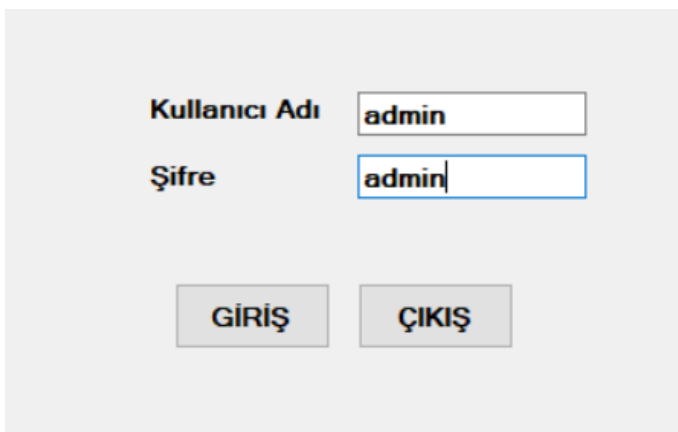
- -- CREATE FUNCTION "personelSifreDegisikligi()" -----

```
CREATE OR REPLACE FUNCTION "Sinema"."personelSifreDegisikligi"()
RETURNS trigger
LANGUAGE plpgsql
AS $function$
BEGIN
    IF NEW."sifre" <> OLD."sifre" THEN
        INSERT INTO "Sinema"."PersonelDegisikligiIzle"("personelId", "eskiSifre",
"yeniSifre", "degisiklikTarihi")
        VALUES(OLD."id", OLD."sifre", NEW."sifre", CURRENT_TIMESTAMP);
    END IF;

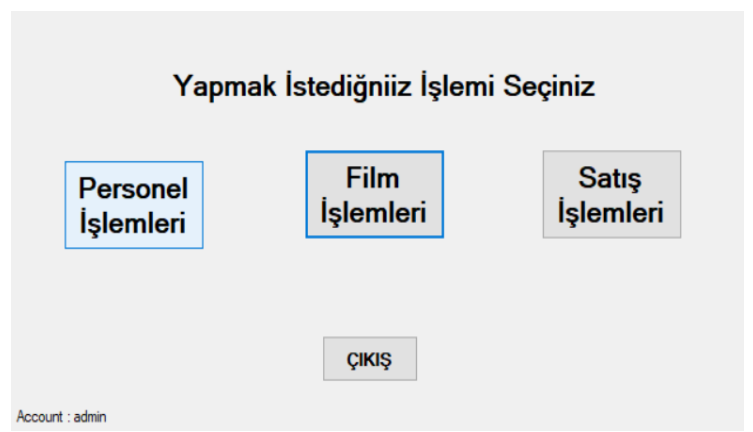
    RETURN NEW;
END;
$function$;
```

personelSifreDegisikligi, herhangi bir personelin şifre güncellemesi işleminden sonra otomatik olarak çalışan bir triggerdir. PersonelDegisikligiIzle tablosuna eski, yeni şifre ve diğer alanları ekler. Bu sayeye eski şifrelerin hepsi veritabanında bir tabloda tutulmuş olur.

GÖRÜNTÜLER



A screenshot of a login form. It has two input fields: 'Kullanıcı Adı' (Username) with the value 'admin' and 'Şifre' (Password) with the value 'admin'. Below the fields are two buttons: 'GİRİŞ' (Login) and 'ÇIKIŞ' (Logout).



A screenshot of a main menu titled 'Yapmak İstedığınız İşlemi Seçiniz' (Select the operation you want to perform). It features three buttons: 'Personel İşlemleri' (Personnel Operations), 'Film İşlemleri' (Film Operations), and 'Satış İşlemleri' (Sales Operations). Below these is a 'ÇIKIŞ' (Logout) button. At the bottom left, it says 'Account : admin'.

GERİ

ÇIKIŞ

Müşteri Bilgileri

Adı

Soyadı

E-Mail

Telefon

Müşteri Türü Öğrenci

Film Bilgileri

Film Adı Pulp Fiction

Seans 10:20:00

Salon Salon1

Koltuk A1

Yiyecek - İçecek

Seçiniz Patlamış Mısır

Ödeme Yöntemi Nakit

Koltuk

| | salonId | adi | doluMu |
|---|---------|-----|--------------------------|
| ▶ | 1 | A1 | <input type="checkbox"/> |
| | 1 | A2 | <input type="checkbox"/> |
| | 1 | B1 | <input type="checkbox"/> |
| | 1 | B2 | <input type="checkbox"/> |
| * | | | <input type="checkbox"/> |

Satış

Account : ayse kaya

FormBiletFatura

GERİ

ÇIKIŞ

Bilet ve Fatura Bilgileri

| | Bilet ID | Bilet Adı | Film Adı | Salon Adı | Seans Saati | Koltuk |
|---|----------|-----------|----------|-----------|-------------|--------|
| ▶ | 13 | 4MATRIX13 | MATRIX | Salon2 | 12:00:00 | A1 |

Account : ayse kaya

ARAMA

GERİ

ÇIKIŞ

Personel Ara

| | id | adi | soyadi | telefon |
|---|----|----------|----------|---------|
| ► | 1 | Dilara | Yavuz | 0123456 |
| | 18 | personel | personel | 0123456 |
| * | | | | |

Personel ID

Adı

Soyadı

Telefon

E-Mail

Kullanıcı Adı

Şifre

Personel Türü

Yönetici

Listele

Ekle

Güncelle

Sil

Account : admin

GERİ

ÇIKIŞ

Dila

| | id | adi | soyadi | telefon |
|---|----|--------|--------|---------|
| ► | 1 | Dilara | Yavuz | 0123456 |
| * | | | | |

Personel ID

Adı

Soyadı

Telefon

E-Mail

Kullanıcı Adı

Şifre

Personel Türü

Yönetici

Listele

Ekle

Güncelle

Sil

Account : admin

SİLME


GERİ

ÇIKIŞ

| | id | adi | soyadi | telefon |
|---|----|----------|----------|---------|
| ► | 1 | Dilara | Yavuz | 0123456 |
| * | 18 | personel | personel | 0123456 |
| | | | | |

Bilgi

×

 Personel silme işlemi başarılı bir şekilde gerçekleşti.

Tamam

Personel ID

18

Adı

Soyadı

Telefon

E-Mail

Kullanıcı Adı

Şifre

Personel Türü

Yönetici

Listele

Ekle

Güncelle

Sil

Account : admin

GERİ

ÇIKIŞ

| | id | adi | soyadi | telefon |
|---|----|--------|--------|---------|
| ► | 1 | Dilara | Yavuz | 0123456 |
| * | | | | |
| | | | | |

Personel ID

Adı

Soyadı

Telefon

E-Mail

Kullanıcı Adı

Şifre

Personel Türü

Yönetici

Listele

Ekle

Güncelle

Sil

Account : admin

EKLEME

[GERİ](#)[ÇIKIŞ](#)

| | id | adi | soyadi | telefon |
|---|----|--------|--------|---------|
| ► | 1 | Dilara | Yavuz | 0123456 |
| * | | | | |

Bilgi

Personel ekleme işlemi başarı ile gerçekleştirildi.

Tamam

Personel ID

Adı

Soyadı

Telefon

E-Mail

Kullanıcı Adı

Şifre

Personel Türü

Listele

Ekle

Güncelle

Sil

Account : admin

[GERİ](#)[ÇIKIŞ](#)

| | id | adi | soyadi | telefon |
|---|----|--------|--------|---------|
| ► | 1 | Dilara | Yavuz | 0123456 |
| | 20 | Ayşe | Kaya | 0123456 |
| * | | | | |

Personel ID

Adı

Soyadı

Telefon

E-Mail

Kullanıcı Adı

Şifre

Personel Türü

Listele

Ekle

Güncelle

Sil

Account : admin

GÜNCELLEME

GERİ

ÇIKIŞ

| | id | adi | soyadi | telefon |
|---|----|--------|--------|---------|
| ► | 1 | Dilara | Yavuz | 0123456 |
| | 20 | Ayşe | Kaya | 0123456 |
| * | | | | |

Bilgi

×

!

Personel güncelleme işlemi başarı ile gerçekleşti.

Tamam

Personel ID

20

Adı

Ayşe

Soyadı

Kaya

Telefon

01234567892

E-Mail

ayse@gmail.com

Kullanıcı Adı

ayse kaya

Şifre

ayse kaya

Personel Türü

Biletci

Listele

Ekle

Güncelle

Sil

Account : admin

GERİ

ÇIKIŞ

Personel Ara

| | ersonelTuruld | kullaniciAdi | sifre | satısMiktari |
|---|---------------|--------------|-----------|--------------|
| ► | | admin | admin | 0 |
| | | ayse kaya | ayse kaya | 0 |
| * | | | | |

Personel ID

Adı

Soyadı

Telefon

E-Mail

Kullanıcı Adı

Şifre

Personel Türü

Yönetici

Listele

Ekle

Güncelle

Sil

Account : admin