

Sakarya Üniversitesi Bilgisayar ve Bilişim Bilimleri Fakültesi
Bilgisayar Mühendisliği Bölümü Veritabanı Yönetim Sistemleri
Proje Raporu

Öğrenci Numarası: G201210011 – G201210093

Öğrenci Adı: Muhammet Ali Şahal – Mehmet Efe Erbaş

**E-posta: muhammet.sahal@ogr.sakarya.edu.tr-
efe.erbas@ogr.sakarya.edu.tr**

Uygulama Tanıtım:

Bir oyun oynama ve satın alma platformu uygulaması. Uygulama içerisinde birçok bölüm yer alıyor. Örneğin oyunlar, arkadaşlar ,hesaplar vs.

İş Kuralları:

Ücretli , ücretsiz oyun tablosundan kalıtım alır.

Bir oyun birden fazla hesapta bulunabilir , hiçbir hesapta da bulunmayabilir.

Bir hesapta hiç oyun olmayabilir ya da birden fazla oyun olabilir.

Oyunlar sadece 1 mağazada bulunur, bir mağaza çok oyun içerebilir en az bir oyun içerir.

Bir şirket birden fazla oyun yapabilir ya da hiç oyun yapmayabilir.

Bir oyunu yalnızca bir şirket yapabilir.

Bir hesabın ad, soyad ,mail, arkadaş ve bakiye bilgisi bulunmaktadır.

Bir profil yalnızca bir hesaba aittir, bir hesabın en çok bir profili vardır.

Bir hesabın en çok bir mağazası olabilir, bir mağaza çok hesapta bulunabilir.

Bir hesabın en çok bir desteği olabilir, bir destek çok hesapta bulunabilir.

Bir hesabın en çok bir envanteri olabilir, bir envanter çok hesapta bulunabilir.

Bir hesabın en çok bir kütüphanesi olabilir, bir kütüphane en çok bir hesapta bulunabilir.

Bir profilin yalnızca bir pazarı olabilir, bir pazar en çok bir profilde bulunabilir.

Bir sipariş en az bir oyun içermelidir, birden fazla oyun içerebilir.

Bir sipariş hiçbir kupon içermeyebilir ya da en fazla bir kupon içerebilir.

Bir envanterde hiç eşya olmayabilir ya da birden fazla eşya olabilir.

Bir eşya birden fazla envanterde olabilir ya da hiçbir envanterde olmayabilir.

Bir pazarda hiç eşya olmayabilir ya da birden fazla eşya olabilir.

Bir eşya hiçbir pazarda olmayabilir ya da en fazla bir pazarda bulunabilir.

Bir kütüphanede birden fazla oyun bulunabilir ya da hiç bulunmayabilir.

Bir oyun birden fazla kütüphanede bulunabilir ya da hiçbir kütüphanede bulunmayabilir.

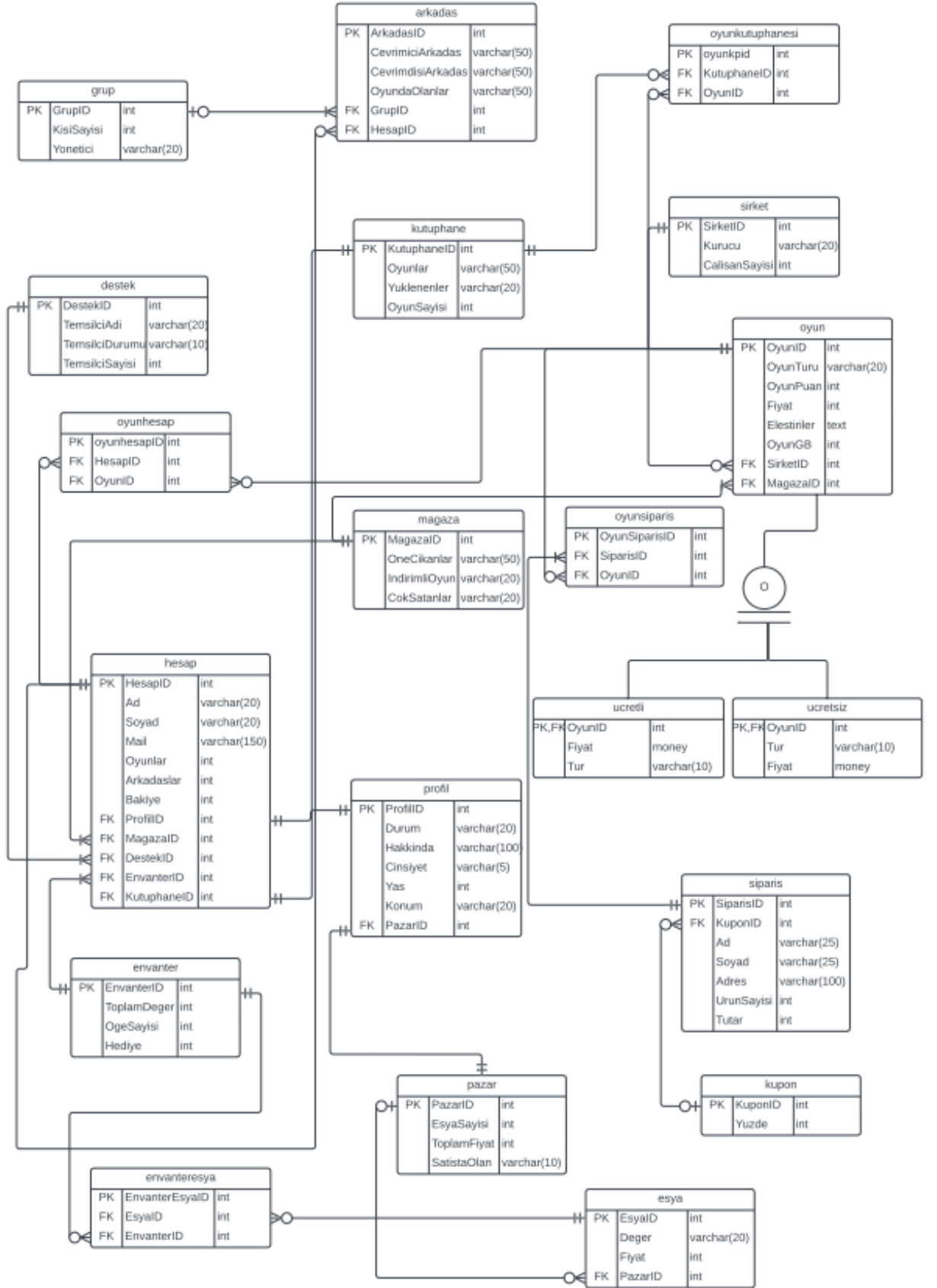
Bir hesapta birden fazla arkadaş bulunabilir ya da hiç arkadaş bulunmayabilir.

Bir arkadaş en az bir hesapta bulunabilir.

Bir grup en az bir arkadaş ya da birden çok arkadaş içerebilir.

Bir arkadaş hiçbir grupta bulunmayabilir ya da en fazla bir grupta bulunur.

Varlık Bağntı Modeli:

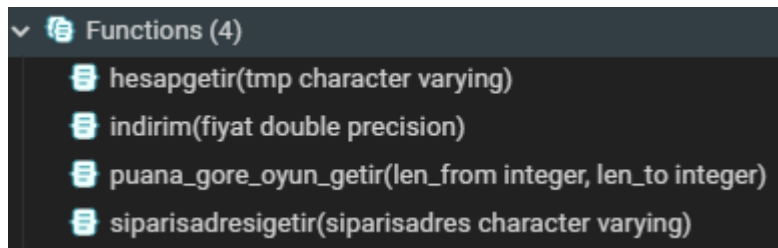


İlişkisel Şema:

- oyun(OyunId:Integer, OyunTuru:varchar, OyunPuan:Integer, Fiyat:Integer, Elestiriler:varchar, OyunGB:Integer, SirketID:Integer, MagazaID:Integer)
- hesap(HesapID:Integer, Ad:varchar, soyad:varchar, mail:varchar, Oyunlar:Integer, arkadaşlar:Integer, Bakiye:Integer, ProfillID:Integer, MagazaID:Integer, DestekID:Integer, EnvanterID:Integer ,KutuphaneID:Integer)
- arkadas(ArkadasID:Integer,CevrimiciArkadas:varchar, CevrimdisiArkadas:varchar, OyundaOlanlar:varchar, GrupID:Integer, HesapID:Integer)
- destek(DestekID:Integer, TemsilciAdi:varchar, TemsilciDurumu:varchar, TemsilciSayisi:Integer)
- envanter(EnvanterID:Integer, ToplamDeger:Integer, OgeSayisi:Integer, Hediye:Integer)
- envanteresya(EnvanterEsysID:Integer,EsysID:Integer, EnvanterID:Integer)
- esya(EsysID:Integer, EsysAdi:varchar, Fiyat:Integer, PazarID:Integer)
- grup(GrupID:Integer, KisiSayisi:Integer, Yonetici:varchar)
- kupon(KuponID:Integer, Yuzde:Integer)
- kutuphane(KutuphaneID:Integer, Oyunlar:varchar, Yuklenenler:varchar, OyunSayisi:Integer)
- magaza(MagazaID:Integer, OneCikanlar:varchar, IndirimliOyun:varchar, CokSatanlar:varchar)
- oyunhesap(OyunHesapID:Integer, HesapID:Integer, OyunID:Integer, OyunSayisi:Integer)
- oyunkutuphanesi(OyunKpID:Integer, OyunID:Integer, KutuphaneID:Integer)
- oyunsiparis(OyunSiparisID:Integer, SiparisID:Integer, OyunID:Integer, OyunSayisi:Integer, ToplamOyunFiyat:Integer, ToplamGB:Integer)
- pazar(PazarID:Integer, EsysSayisi:Integer, ToplamFiyat:Integer, SatistaOlan:varchar)

- profil(ProfilID:Integer, Durum:varchar, Hakkinda:varchar, Cinsiyet:varchar, Yas:Integer, Konum:varchar, PazarID:Integer)
- siparis(SiparisID:Integer, KuponID:Integer, Ad:varhcar; Soyad:varchar, Adres:varchar, OyunSayisi:Integer, Tutar:Integer)
- sirket(SirketID:Integer, Kurucu:varchar; CalisanSayisi:Integer)
- ucretli(OyunID:Integer, Fiyat:Integer, TurAdi:varchar)
- ücretsiz(OyunID:Integer, TurAdi:varchar)

Fonksiyonlar:



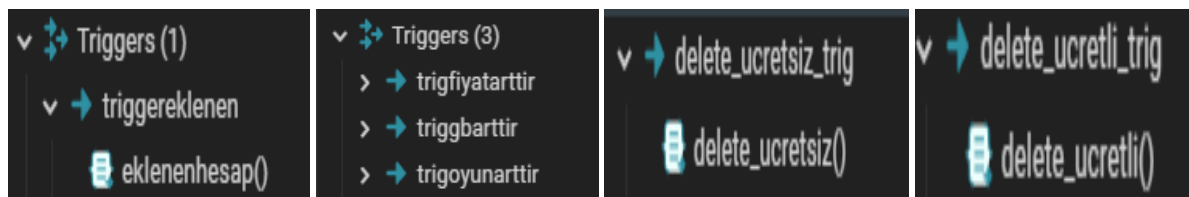
hesapgetir(): İçine yazılan isme ait hesabın bilgilerini getirir.

indirim(): İçine yazılan tutara göre indirim uygular.

puana_gore_oyun_getir():İki girilen puan aralığındaki oyun sayısını getirir.

Siparisadresigetir(): İçine girilen şehre göre siparişin bilgilerini getirir.

Triggerler:



eklenenhesap(): Bir hesap eklendiğinde eklenen hesap tablosuna eklenilmiş hesabın bilgilerini ekler.

trigfiyatarttir():Bir oyun eklendiğinde eklenen oyun fiyatını “oyunsiparis” tablosundaki “toplamoyunfiyat” kısmına ekler.

triggbarttir():Bir oyun eklendiğinde eklenen oyun gb’ını “oyunsiparis” tablosundaki “toplamb” kısmına ekler.

trigoyunarttir():Bir oyun eklendiğinde “oyunsiparis” tablosundaki “oyunsayisi” kısmını bir arttırır.

delete_ucretsiz():”ucretsiz” tablosundan bir oyun silindiğinde “oyun” tablosundan da siler.

delete_ucretli():”ucretli” tablosundan bir oyun silindiğinde “oyun” tablosundan da siler.

Ekleme/Silme/Güncelleme

Başlangıç:

Form1

oyunid	oyunturu	oyunpuan	fiyat	elestiriler	oyungb	sirketid	magazaid
8	anime	7	750	iyi	25	4	3
3	mmorpg	6	0	eh iste	4	1	1
9	mmorpg	8	0	harika	45	5	5
14	bilim kurgu	7	0	kotu	10	3	4
11	moba	7	100	harika	85	2	3
10	bilim kurgu	4	100	kotu	10	3	4
13	bilim kurgu	3	100	kotu	10	3	4
4	mmorpg	9	100	eh	10	2	3
1	aksiyon	5	50	ben begenmedim	2	1	1
15	aksiyon	8	150	iyi	50	3	4
*							

△ Ana Sayfa

Listele

Oyun ID:

Şirket ID:

Oyun Türü:

Magaza ID:

Oyun Puanı:

Fiyat:

Eleştiri:

GB:

Ekle

Güncelle

Sil

Ücretli Oyun

Ücretli Oyunlar

ID:

Sil

Ücretsiz Oyun

Ücretsiz Oyunlar

ID:

Sil

Ekleme:

Form1

oyunid	oyunturu	oyunpuan	fiyat	elestiriler	oyungb	sirketid	magazaid
3	mmorpg	6	0	eh iste	4	1	1
9	mmorpg	8	0	harika	45	5	5
14	bilim kurgu	7	0	kotu	10	3	4
11	moba	7	100	harika	85	2	3
10	bilim kurgu	4	100	kotu	10	3	4
13	bilim kurgu	3	100	kotu	10	3	4
4	mmorpg	9	100	eh	10	2	3
1	aksiyon	5	50	ben begenmedim	2	1	1
15	aksiyon	8	150	iyi	50	3	4
16	mmorpg	8	100	eh	15	1	2

Ana Sayfa

Liste

Oyun ID: 16

Oyun Türü: mmorpg

Oyun Puanı: 8

Fiyat: 100

Eleştiri: eh

GB: 15

Ekle

Güncelle

Sil

Ücretli Oyun

Ücretli Oyunlar

ID:

Sil

Ücretsiz Oyun

Ücretsiz Oyunlar

ID:

Sil

Silme:

Silmeye tıklandıktan sonra mesaj kutusu

Form1

oyunid	oyunturu	oyunpuan	fiyat	elestiriler	oyungb	sirketid	magazaid
3	mmorpg	6	0	eh iste	4	1	1
9	mmorpg	8	0	harika	45	5	5
14	bilim kurgu	7	0	kotu	10	3	4
11	moba	7	100	harika	85	2	3
10	bilim kurgu	4	100	kotu	10	3	4
13	bilim kurgu	3	100	kotu	10	3	4
4	mmorpg	9	100	eh	10	2	3
1	aksiyon	5	50	ben begenmedim	2	1	1
15	aksiyon	8	150	iyi	50	3	4
16	mmorpg	8	100	eh	15	1	2

Ana Sayfa

Liste

Oyun ID: 16

Oyun Türü: mmorpg

Oyun Puanı: 8

Fiyat: 100

Eleştiri: eh

GB: 15

Ekle

Güncelle

Sil

Ücretli Oyun

Ücretli Oyunlar

ID:

Sil

Ücretsiz Oyun

Ücretsiz Oyunlar

ID:

Sil

Silme işlemi başarılı!

Tamam

Yeniden listeleye tıklanınca 16 id'li oyun silindi.

Form1

oyunid

oyunturu

oyunpuan

fiyat

elestiniler

oyungb

sirketid

magazaid

8	anime	7	750	iyi	25	4	3
3	mmorpg	6	0	eh iste	4	1	1
9	mmorpg	8	0	harika	45	5	5
14	bilim kurgu	7	0	kotu	10	3	4
11	moba	7	100	harika	85	2	3
10	bilim kurgu	4	100	kotu	10	3	4
13	bilim kurgu	3	100	kotu	10	3	4
4	mmorpg	9	100	eh	10	2	3
1	aksiyon	5	50	ben begenmedim	2	1	1
15	aksiyon	8	150	iyi	50	3	4
*							

Ana Sayfa

Liste

Oyun ID: 16

Oyun Türü: mmorpg

Oyun Puanı: 8

Fiyat: 100

Eleştiri: eh

GB: 15

Ekle

Güncelle

Sil

Ücretli Oyun

Ücretli Oyunlar

ID:

Sil

Ücretsiz Oyun

Ücretsiz Oyunlar

ID:

Sil

Güncelleme:

Güncelleme tuşuna basılmadan önce oyun id'si 15 olanın fiyatı 200

Form1

oyunid

oyunturu

oyunpuan

fiyat

elestiniler

oyungb

sirketid

magazaid

8	anime	7	750	iyi	25	4	3
3	mmorpg	6	0	eh iste	4	1	1
9	mmorpg	8	0	harika	45	5	5
14	bilim kurgu	7	0	kotu	10	3	4
11	moba	7	100	harika	85	2	3
10	bilim kurgu	4	100	kotu	10	3	4
13	bilim kurgu	3	100	kotu	10	3	4
4	mmorpg	9	100	eh	10	2	3
1	aksiyon	5	50	ben begenmedim	2	1	1
15	aksiyon	8	200	eh	50	3	4
*							

Ana Sayfa

Liste

Oyun ID: 15

Oyun Türü: mmorpg

Oyun Puanı: 8

Fiyat: 400

Eleştiri: eh

GB: 15

Ekle

Güncelle

Sil

Ücretli Oyun

Ücretli Oyunlar

ID:

Sil

Ücretsiz Oyun

Ücretsiz Oyunlar

ID:

Sil

Güncelleme yapıldıktan sonra fiyatı 400 oldu.

oyunid	oyunturu	oyunpuan	fiyat	elestiriler	oyungb	sirketid	magazaid
8	anime	7	750	iyi	25	4	3
3	mmorpg	6	0	eh iste	4	1	1
9	mmorpg	8	0	harika	45	5	5
14	bilim kurgu	7	0	kotu	10	3	4
11	moba	7	100	harika	85	2	3
10	bilim kurgu	4	100	kotu	10	3	4
13	bilim kurgu	3	100	kotu	10	3	4
4	mmorpg	9	100	eh	10	2	3
1	aksiyon	5	50	ben begenmedim	2	1	1
15	aksiyon	8	400	eh	50	3	4

[Ana Sayfa](#)
[Listele](#)

Oyun ID: 15
Oyun Türü: mmorpg
Oyun Puanı: 8
Fiyat: 400
Eleştiri: eh
GB: 15

Şirket ID: 1
Mağaza ID: 2

Ekle **Güncelle** **Sil**

Ücretli Oyun
Ücretli Oyunlar
ID:
Sil

Ücretsiz Oyun
Ücretsiz Oyunlar
ID:
Sil

Github Linkleri:

1. <https://github.com/meferbas/Database-SQL-Project>
2. <https://github.com/Muhammet-842/SQLProje>

Kaynak Kodları:

```
BEGIN;
```

```
CREATE TABLE IF NOT EXISTS public.arkadas
```

```
(
```

```
  "ArkadasID" integer NOT NULL,
```

```
"CevrimiciArkadas" character varying(50) COLLATE pg_catalog."default",
"CevrimdisiArkadas" character varying(50) COLLATE pg_catalog."default",
"OyundaOlanlar" character varying(50) COLLATE pg_catalog."default",
"GrupID" integer NOT NULL,
"HesapID" integer NOT NULL,
CONSTRAINT arkadas_pkey PRIMARY KEY ("ArkadasID")
);
```

```
CREATE TABLE IF NOT EXISTS public.destek
(
    "DestekID" integer NOT NULL,
    "TemsilciAdi" character varying(20) COLLATE pg_catalog."default" NOT NULL,
    "TemsilciDurumu" character varying(10) COLLATE pg_catalog."default" NOT NULL,
    "TemsilciSayisi" integer NOT NULL,
    CONSTRAINT destek_pkey PRIMARY KEY ("DestekID")
);
```

```
CREATE TABLE IF NOT EXISTS public.eklenenhesap
(
    hesap integer NOT NULL,
    hesapadi character varying(50) COLLATE pg_catalog."default" NOT NULL,
    hesapsoyadi character varying(50) COLLATE pg_catalog."default" NOT NULL,
    mail character varying(100) COLLATE pg_catalog."default" NOT NULL,
    oynusayi integer NOT NULL,
    sayiarkadas integer NOT NULL,
    para integer NOT NULL,
    profil integer NOT NULL,
    magaza integer NOT NULL,
    destek integer NOT NULL,
```

```
    envanter integer NOT NULL,  
    kutuphane integer NOT NULL  
);
```

```
CREATE TABLE IF NOT EXISTS public.envanter  
(  
    "EnvanterID" integer NOT NULL,  
    "ToplamDeger" integer,  
    "OgeSayisi" integer,  
    "Hediye" integer,  
    CONSTRAINT envanter_pkey PRIMARY KEY ("EnvanterID")  
);
```

```
CREATE TABLE IF NOT EXISTS public.envanteresya  
(  
    "EnvanterEsysaID" integer NOT NULL,  
    "EsysaID" integer,  
    "EnvanterID" integer,  
    CONSTRAINT envanteresya_pkey PRIMARY KEY ("EnvanterEsysaID")  
);
```

```
CREATE TABLE IF NOT EXISTS public.esya  
(  
    "EsysaID" integer NOT NULL,  
    "EsysaAdi" character varying(30) COLLATE pg_catalog."default" NOT NULL,  
    "Fiyat" integer,  
    "PazarID" integer,  
    CONSTRAINT esya_pkey PRIMARY KEY ("EsysaID")  
);
```

```
CREATE TABLE IF NOT EXISTS public.grup
(
    "GrupID" integer NOT NULL,
    "KisiSayisi" integer,
    "Yonetici" character varying(20) COLLATE pg_catalog."default" NOT NULL,
    CONSTRAINT grup_pkey PRIMARY KEY ("GrupID")
);
```

```
CREATE TABLE IF NOT EXISTS public.hesap
(
    hesapid integer NOT NULL,
    ad character varying(20) COLLATE pg_catalog."default" NOT NULL,
    soyad character varying(20) COLLATE pg_catalog."default" NOT NULL,
    mail character varying(150) COLLATE pg_catalog."default" NOT NULL,
    oyunlar integer,
    arkadaslar integer,
    bakiye integer,
    profilid integer NOT NULL,
    magazaid integer NOT NULL,
    destekid integer NOT NULL,
    envanterid integer NOT NULL,
    kutuphaneid integer NOT NULL,
    CONSTRAINT hesap_pkey PRIMARY KEY (hesapid)
);
```

```
CREATE TABLE IF NOT EXISTS public.kupon
(
    "KuponID" integer NOT NULL,
```

```
"Yuzde" integer NOT NULL,  
CONSTRAINT kupon_pkey PRIMARY KEY ("KuponID")  
);
```

```
CREATE TABLE IF NOT EXISTS public.kutuphane  
(  
    "KutuphaneID" integer NOT NULL,  
    "Oyunlar" character varying(50) COLLATE pg_catalog."default",  
    "Yuklenenler" character varying(50) COLLATE pg_catalog."default",  
    "OyunSayisi" integer,  
    CONSTRAINT kutuphane_pkey PRIMARY KEY ("KutuphaneID")  
);
```

```
CREATE TABLE IF NOT EXISTS public.magaza  
(  
    "MagazaID" integer NOT NULL,  
    "OneCikanlar" character varying(50) COLLATE pg_catalog."default",  
    "IndirimliOyun" character varying(20) COLLATE pg_catalog."default",  
    "CokSatanlar" character varying(20) COLLATE pg_catalog."default",  
    CONSTRAINT magaza_pkey PRIMARY KEY ("MagazaID")  
);
```

```
CREATE TABLE IF NOT EXISTS public.oyun  
(  
    oyunid integer NOT NULL,  
    oyunturu character varying(20) COLLATE pg_catalog."default" NOT NULL,  
    oyunpuan integer,  
    fiyat integer,  
    elestiriler text COLLATE pg_catalog."default" NOT NULL,
```

```
    oyungb integer NOT NULL,  
    sirketid integer NOT NULL,  
    magazaid integer NOT NULL,  
    CONSTRAINT oyun_pkey PRIMARY KEY (oyunid)  
);
```

```
CREATE TABLE IF NOT EXISTS public.oyunhesap  
(  
    "oyunhesapID" integer NOT NULL,  
    "HesapID" integer NOT NULL,  
    "OyunID" integer NOT NULL,  
    oyunsayisi integer,  
    CONSTRAINT oyunhesap_pkey PRIMARY KEY ("oyunhesapID")  
);
```

```
CREATE TABLE IF NOT EXISTS public.oyunkutuphanesi  
(  
    oyunkpid integer NOT NULL,  
    "KutuphaneID" integer NOT NULL,  
    "OyunID" integer NOT NULL,  
    CONSTRAINT oyunkutuphanesi_pkey PRIMARY KEY (oyunkpid)  
);
```

```
CREATE TABLE IF NOT EXISTS public.oyunsiparis  
(  
    "OyunSiparisID" integer NOT NULL,  
    "SiparisID" integer,  
    "OyunID" integer,  
    oyunsayisi integer,
```

```
toplamoynfiyat integer,  
toplamgb integer,  
CONSTRAINT oyunsiparis_pkey PRIMARY KEY ("OyunSiparisID")  
);
```

```
CREATE TABLE IF NOT EXISTS public.pazar  
(  
    pazarid integer NOT NULL,  
    esyasayisi integer,  
    toplamfiyat integer,  
    satistaolan character varying(10) COLLATE pg_catalog."default",  
    CONSTRAINT pazar_pkey PRIMARY KEY (pazarid)  
);
```

```
CREATE TABLE IF NOT EXISTS public.profil  
(  
    "ProfilID" integer NOT NULL,  
    "Durum" character varying(20) COLLATE pg_catalog."default" NOT NULL,  
    "Hakkinda" character varying(100) COLLATE pg_catalog."default" NOT NULL,  
    "Cinsiyet" character varying(5) COLLATE pg_catalog."default" NOT NULL,  
    "Yas" integer NOT NULL,  
    "Konum" character varying(20) COLLATE pg_catalog."default" NOT NULL,  
    "PazarID" integer NOT NULL,  
    CONSTRAINT profil_pkey PRIMARY KEY ("ProfilID")  
);
```

```
CREATE TABLE IF NOT EXISTS public.siparis  
(  
    siparisid integer NOT NULL,
```

```
kuponid integer,  
ad character varying(25) COLLATE pg_catalog."default",  
soyad character varying(25) COLLATE pg_catalog."default",  
adres character varying(100) COLLATE pg_catalog."default",  
oyunsayisi integer NOT NULL,  
tutar integer,  
CONSTRAINT siparis_pkey PRIMARY KEY (siparisid)  
);
```

```
CREATE TABLE IF NOT EXISTS public.sirket  
(  
    "SirketID" integer NOT NULL,  
    "Kurucu" character varying(20) COLLATE pg_catalog."default" NOT NULL,  
    "CalisanSayisi" integer,  
    CONSTRAINT sirket_pkey PRIMARY KEY ("SirketID")  
);
```

```
CREATE TABLE IF NOT EXISTS public.ucretli  
(  
    oyunid integer NOT NULL,  
    fiyat integer NOT NULL,  
    tur character varying(20) COLLATE pg_catalog."default" NOT NULL,  
    adi character varying(40) COLLATE pg_catalog."default" NOT NULL,  
    CONSTRAINT ucretli_pkey PRIMARY KEY (oyunid)  
);
```

```
CREATE TABLE IF NOT EXISTS public.ucretsiz  
(  
    oyunid integer NOT NULL,
```



```
tur character varying(20) COLLATE pg_catalog."default" NOT NULL,  
adi character varying(20) COLLATE pg_catalog."default" NOT NULL,  
CONSTRAINT ucretsiz_pkey PRIMARY KEY (oyunid)  
);
```

```
ALTER TABLE IF EXISTS public.arkadas  
ADD CONSTRAINT "FK_arkadas.GrupID" FOREIGN KEY ("GrupID")  
REFERENCES public.grup ("GrupID") MATCH SIMPLE  
ON UPDATE NO ACTION  
ON DELETE NO ACTION;
```

```
ALTER TABLE IF EXISTS public.arkadas  
ADD CONSTRAINT "FK_arkadas.HesapID" FOREIGN KEY ("HesapID")  
REFERENCES public.hesap (hesapid) MATCH SIMPLE  
ON UPDATE NO ACTION  
ON DELETE NO ACTION;
```

```
ALTER TABLE IF EXISTS public.envanteresya  
ADD CONSTRAINT "FK_envanteresya.EnvanterID" FOREIGN KEY ("EnvanterID")  
REFERENCES public.envanter ("EnvanterID") MATCH SIMPLE  
ON UPDATE NO ACTION  
ON DELETE NO ACTION;
```

```
ALTER TABLE IF EXISTS public.envanteresya  
ADD CONSTRAINT "FK_envanteresya.EsyaID" FOREIGN KEY ("EsyaID")  
REFERENCES public.esya ("EsyaID") MATCH SIMPLE
```

ON UPDATE NO ACTION

ON DELETE NO ACTION;

ALTER TABLE IF EXISTS public.esya

ADD CONSTRAINT "FK_esya.PazarID" FOREIGN KEY ("PazarID")

REFERENCES public.pazar (pazarid) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION;

ALTER TABLE IF EXISTS public.hesap

ADD CONSTRAINT "FK_hesap.DestekID" FOREIGN KEY (destekid)

REFERENCES public.destek ("DestekID") MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION;

ALTER TABLE IF EXISTS public.hesap

ADD CONSTRAINT "FK_hesap.EnvanterID" FOREIGN KEY (envanterid)

REFERENCES public.envanter ("EnvanterID") MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION;

ALTER TABLE IF EXISTS public.hesap

ADD CONSTRAINT "FK_hesap.KutuphaneID" FOREIGN KEY (kutuphaneid)

REFERENCES public.kutuphane ("KutuphaneID") MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION;

ALTER TABLE IF EXISTS public.hesap

ADD CONSTRAINT "FK_hesap.MagazaID" FOREIGN KEY (magazaid)

REFERENCES public.magaza ("MagazaID") MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION;

ALTER TABLE IF EXISTS public.hesap

ADD CONSTRAINT "FK_hesap.ProfilID" FOREIGN KEY (profilid)

REFERENCES public.profil ("ProfilID") MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION;

ALTER TABLE IF EXISTS public.oyun

ADD CONSTRAINT "FK_oyun.MagazaID" FOREIGN KEY (magazaid)

REFERENCES public.magaza ("MagazaID") MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION;

ALTER TABLE IF EXISTS public.oyun

ADD CONSTRAINT "FK_oyun.SirketID" FOREIGN KEY (sirketid)

REFERENCES public.sirket ("SirketID") MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION;

```
ALTER TABLE IF EXISTS public.oyunhesap  
    ADD CONSTRAINT "FK_oyunhesap.HesapID" FOREIGN KEY ("HesapID")  
    REFERENCES public.hesap (hesapid) MATCH SIMPLE  
    ON UPDATE NO ACTION  
    ON DELETE NO ACTION;
```

```
ALTER TABLE IF EXISTS public.oyunhesap  
    ADD CONSTRAINT "FK_oyunhesap.OyunID" FOREIGN KEY ("OyunID")  
    REFERENCES public.oyun (oyunid) MATCH SIMPLE  
    ON UPDATE NO ACTION  
    ON DELETE NO ACTION;
```

```
ALTER TABLE IF EXISTS public.oyunkutuphanesi  
    ADD CONSTRAINT "FK_oyunkutuphanesi.KutuphaneID" FOREIGN KEY ("KutuphaneID")  
    REFERENCES public.kutuphane ("KutuphaneID") MATCH SIMPLE  
    ON UPDATE NO ACTION  
    ON DELETE NO ACTION;
```

```
ALTER TABLE IF EXISTS public.oyunkutuphanesi  
    ADD CONSTRAINT "FK_oyunkutuphanesi.OyunID" FOREIGN KEY ("OyunID")  
    REFERENCES public.oyun (oyunid) MATCH SIMPLE  
    ON UPDATE NO ACTION  
    ON DELETE NO ACTION;
```

```
ALTER TABLE IF EXISTS public.oyunsiparis
```

```
ADD CONSTRAINT "FK_oyunsiparis.OyunID" FOREIGN KEY ("OyunID")
```

```
REFERENCES public.oyun (oyunid) MATCH SIMPLE
```

```
ON UPDATE NO ACTION
```

```
ON DELETE NO ACTION;
```

```
ALTER TABLE IF EXISTS public.oyunsiparis
```

```
ADD CONSTRAINT "FK_oyunsiparis.SiparisID" FOREIGN KEY ("SiparisID")
```

```
REFERENCES public.siparis (siparisid) MATCH SIMPLE
```

```
ON UPDATE NO ACTION
```

```
ON DELETE NO ACTION;
```

```
ALTER TABLE IF EXISTS public.profil
```

```
ADD CONSTRAINT "FK_profil.PazarID" FOREIGN KEY ("PazarID")
```

```
REFERENCES public.pazar (pazarid) MATCH SIMPLE
```

```
ON UPDATE NO ACTION
```

```
ON DELETE NO ACTION;
```

```
ALTER TABLE IF EXISTS public.siparis
```

```
ADD CONSTRAINT "FK_siparis.KuponID" FOREIGN KEY (kuponid)
```

```
REFERENCES public.kupon ("KuponID") MATCH SIMPLE
```

```
ON UPDATE NO ACTION
```

```
ON DELETE NO ACTION;
```

```
ALTER TABLE IF EXISTS public.ucretli  
    ADD CONSTRAINT "ucretli_OyunID_fkey" FOREIGN KEY (oyunid)  
    REFERENCES public.oyun (oyunid) MATCH SIMPLE  
    ON UPDATE NO ACTION  
    ON DELETE NO ACTION;  
CREATE INDEX IF NOT EXISTS ucretli_pkey  
    ON public.ucretli(oyunid);
```

```
ALTER TABLE IF EXISTS public.ucretsiz  
    ADD CONSTRAINT "ucretsiz_OyunID_fkey" FOREIGN KEY (oyunid)  
    REFERENCES public.oyun (oyunid) MATCH SIMPLE  
    ON UPDATE NO ACTION  
    ON DELETE NO ACTION;  
CREATE INDEX IF NOT EXISTS ucretsiz_pkey  
    ON public.ucretsiz(oyunid);
```

```
END;
```