

**T.C.**

**BARTIN ÜNİVERSİTESİ**

**MESLEK YÜKSEKOKULU**

**BİLGİSAYAR TEKNOLOJİLERİ BÖLÜMÜ**

**BİLGİSAYAR PROGRAMCILIĞI PROGRAMI**

**VERITABANI FINAL ÖDEVİ**

**SUNUCU İŞLETİM SİSTEMLERİ**

**TAYYİB KAAN İSEN**

**DANIŞMAN**

**ÖĞR. GÖREVLİSİ FATIH DINC**

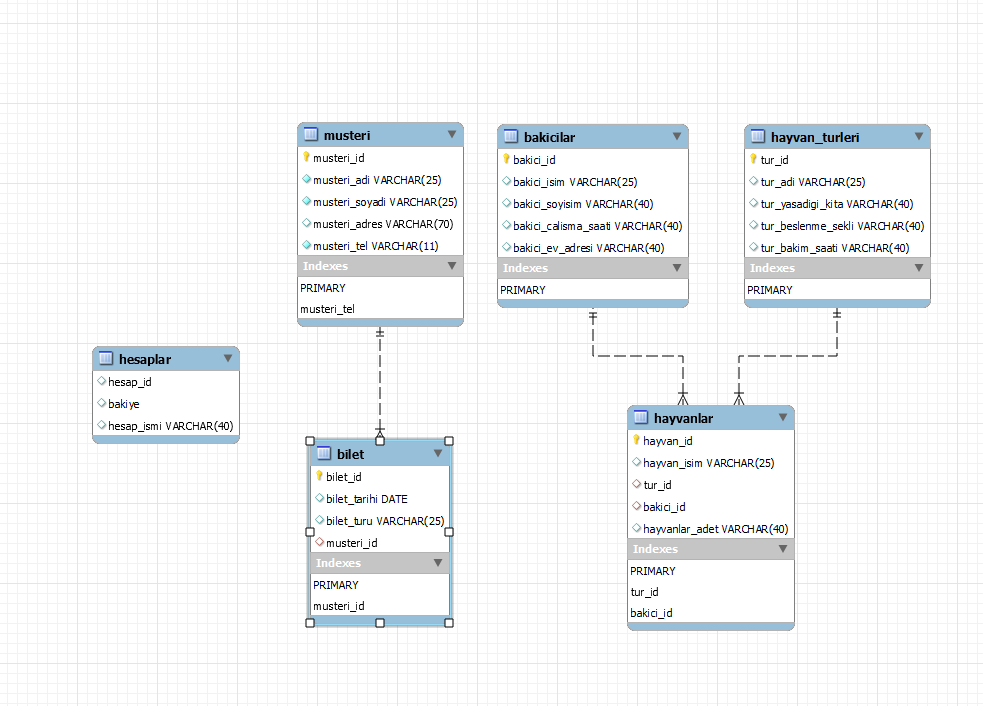
**2022**

Bu oluşturulan veritabani bir hayvnat bahçesindeki hayvan türlerini, hayvan isimlerini, adetlerini, bakicilarin isimlerini, bakicilarin calisma saatlerini ayrica muteri ve bilet bilgilerini kayit altinda tutmak için tasarlanmistir.

Ayriyeten veritabaninda müşterinin banka hesabindan hayvanat bahçesinin banka hesabina para geçiş transactionu da eklenmiştir transaction içinde müşterinin hesabinda yeterli para yoksa rollback işlemi yapiyor yeterli para varsa işlemi onaylayip para aktarimini tamamlıyor.

Veritabaninin Kodlari, er diyagrami ve kullanilan tablolarla birlikte prosedurler asagida verilmiştir:



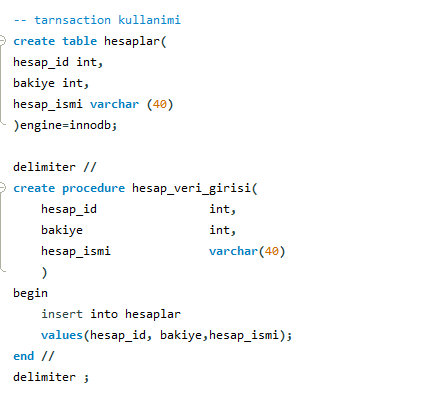


ORNEK TRANSACTION KULLANIMI

Transaction kullanim amaci müşterinin hesabinda yeterli para yoksa rollback işlemi gerçekleştirip müşterinin hesabini – değere dusurmuyor ve para transfer işlemini iptal ediyor.

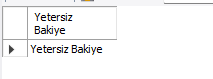
İlk once hesaplar tablosu oluşturuluyor hesaplar tablosunun içine hesap id ve banka bakiye değerleri ataniyor.

Paranin ciktigi banka hesabi , para girişi olan banka hesabi ve gönderilecek miktar in komutu ile yeni bir procedure oluşturularak ataniyor.



@bakiyeler adinda yeni bir degisken oluşturuluyor bu degisken select from ile musterinin banka hesabindaki bakiyeye baglaniyor ve gönderilen miktarin bankadaki bakiyeden kucup olup olmadigini kontrol ediyor.

Eger bankadaki bakiye gönderilen miktardan kucukse para transferini rollback komutu ile iptal ediyor.

ROLLBACK KULLANIMI 

Else durumunda ise bankadaki para gönderilen miktardan cok olduğundan dolayi transfer sirasinda banka hesap id lerini kontrol ediyor. İdler dogru girilmişse commit edip transferi gerçekleştiriyor.

COMMIT KULLANIMI tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

Kodlar asagidaki ekran goruntusunde verilmiştir.

metin içeren bir resim

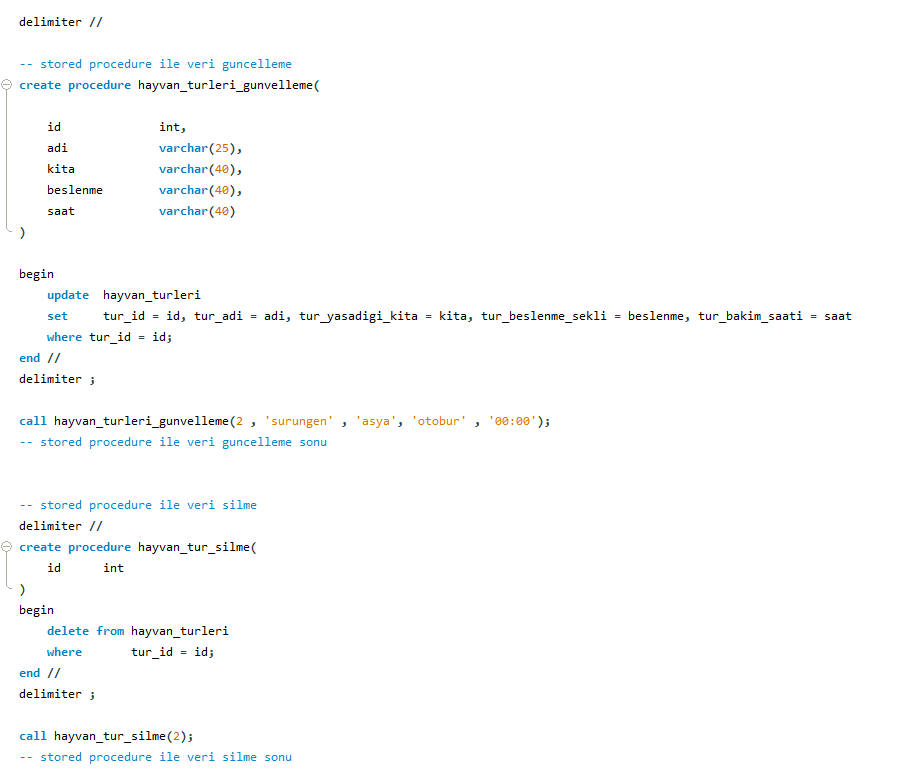
Açıklama otomatik olarak oluşturuldu

ORNEK PROCEDURE KULLANIMI

Veri girişi ve veri sorgulama procedureleri



Veri güncelleme ve veri silme procedureleri



Veritabaninin butun kodlari asagida verilmiştir:

create database bp\_20015221023;

use bp\_20015221023;

create table hayvan\_turleri(

tur\_id int primary key,

tur\_adi varchar(25)

)engine=innodb;

alter table hayvan\_turleri add tur\_yasadigi\_kita varchar(40);

alter table hayvan\_turleri add tur\_beslenme\_sekli varchar(40);

alter table hayvan\_turleri add tur\_bakim\_saati varchar(40);

create table bakicilar(

bakici\_id int primary key,

bakici\_isim varchar(25)

)engine=innodb;

alter table bakicilar add bakici\_soyisim varchar(40);

alter table bakicilar add bakici\_calisma\_saati varchar(40);

alter table bakicilar add bakici\_ev\_adresi varchar(40);

create table hayvanlar(

hayvan\_id int primary key,

hayvan\_isim varchar (25),

tur\_id int,

foreign key (tur\_id)

references hayvan\_turleri(tur\_id)

on delete cascade,

bakici\_id int,

foreign key(bakici\_id)

references bakicilar(bakici\_id)

on delete cascade

)engine=InnoDB;

alter table hayvanlar add hayvanlar\_adet varchar(40);

create table musteri(

musteri\_id int auto\_increment primary key,

musteri\_adi varchar(25) not null,

musteri\_soyadi varchar(25) not null,

musteri\_adres varchar (70) null,

musteri\_tel varchar(11) not null,

unique(musteri\_tel)

)engine=innodb;

create table bilet(

bilet\_id int auto\_increment not null primary key,

bilet\_tarihi date,

bilet\_turu varchar(25),

musteri\_id int,

foreign key(musteri\_id)

references musteri(musteri\_id)

on delete cascade

)engine=innodb;

-- stored procedure ile veri girisi

delimiter //

create procedure tur\_ekle(

tur\_id int,

tur\_adi varchar(25),

tur\_yasadigi\_kita varchar(40),

tur\_beslenme\_sekli varchar(40),

tur\_bakim\_saati varchar(40)

)

begin

insert into hayvan\_turleri

values(tur\_id, tur\_adi, tur\_yasadigi\_kita, tur\_beslenme\_sekli, tur\_bakim\_saati);

end //

delimiter ;

call tur\_ekle (1 , 'kanatli\_hayvan', 'afrika', 'et', '12:00');

-- stored procedure ile veri girisi sonu

-- stored procedure ile veri sorgulama

DELIMITER //

CREATE PROCEDURE tur\_adi\_ara(in tur\_adi\_gir varchar(25) )

BEGIN

select \* from hayvan\_turleri where tur\_adi = tur\_adi\_gir;

END //

DELIMITER ;

call tur\_adi\_ara('kanatli\_hayvan');

-- stored procedure ile veri sorgulama sonu

delimiter //

-- stored procedure ile veri guncelleme

create procedure hayvan\_turleri\_gunvelleme(

id int,

adi varchar(25),

kita varchar(40),

beslenme varchar(40),

saat varchar(40)

)

begin

update hayvan\_turleri

set tur\_id = id, tur\_adi = adi, tur\_yasadigi\_kita = kita, tur\_beslenme\_sekli = beslenme, tur\_bakim\_saati = saat

where tur\_id = id;

end //

delimiter ;

call hayvan\_turleri\_gunvelleme(2 , 'surungen' , 'asya', 'otobur' , '00:00');

-- stored procedure ile veri guncelleme sonu

-- stored procedure ile veri silme

delimiter //

create procedure hayvan\_tur\_silme(

id int

)

begin

delete from hayvan\_turleri

where tur\_id = id;

end //

delimiter ;

call hayvan\_tur\_silme(2);

-- stored procedure ile veri silme sonu

-- tarnsaction kullanimi

create table hesaplar(

hesap\_id int,

bakiye int,

hesap\_ismi varchar (40)

)engine=innodb;

delimiter //

create procedure hesap\_veri\_girisi(

hesap\_id int,

bakiye int,

hesap\_ismi varchar(40)

)

begin

insert into hesaplar

values(hesap\_id, bakiye,hesap\_ismi);

end //

delimiter ;

call hesap\_veri\_girisi (0,'1000','musteri\_hesap');

call hesap\_veri\_girisi (1, '2000','bahce\_hesap');

delimiter //

create procedure para\_transferi(in gidenH int, in gelenH int, in miktar int)

begin

start transaction;

select bakiye into @bakiyeler from hesaplar where hesap\_id = gidenH;

if @bakiyeler < miktar then

select"Yetersiz Bakiye";

else

update hesaplar set bakiye = bakiye-miktar where hesap\_id = gidenH;

update hesaplar set bakiye = bakiye+miktar where hesap\_id = gelenH;

commit;

end if;

rollback;

end //

delimiter ;

SET SQL\_SAFE\_UPDATES = 0;

call para\_transferi(0,1,100);

select \* from hesaplar;