

PROJE BİLGİLERİ

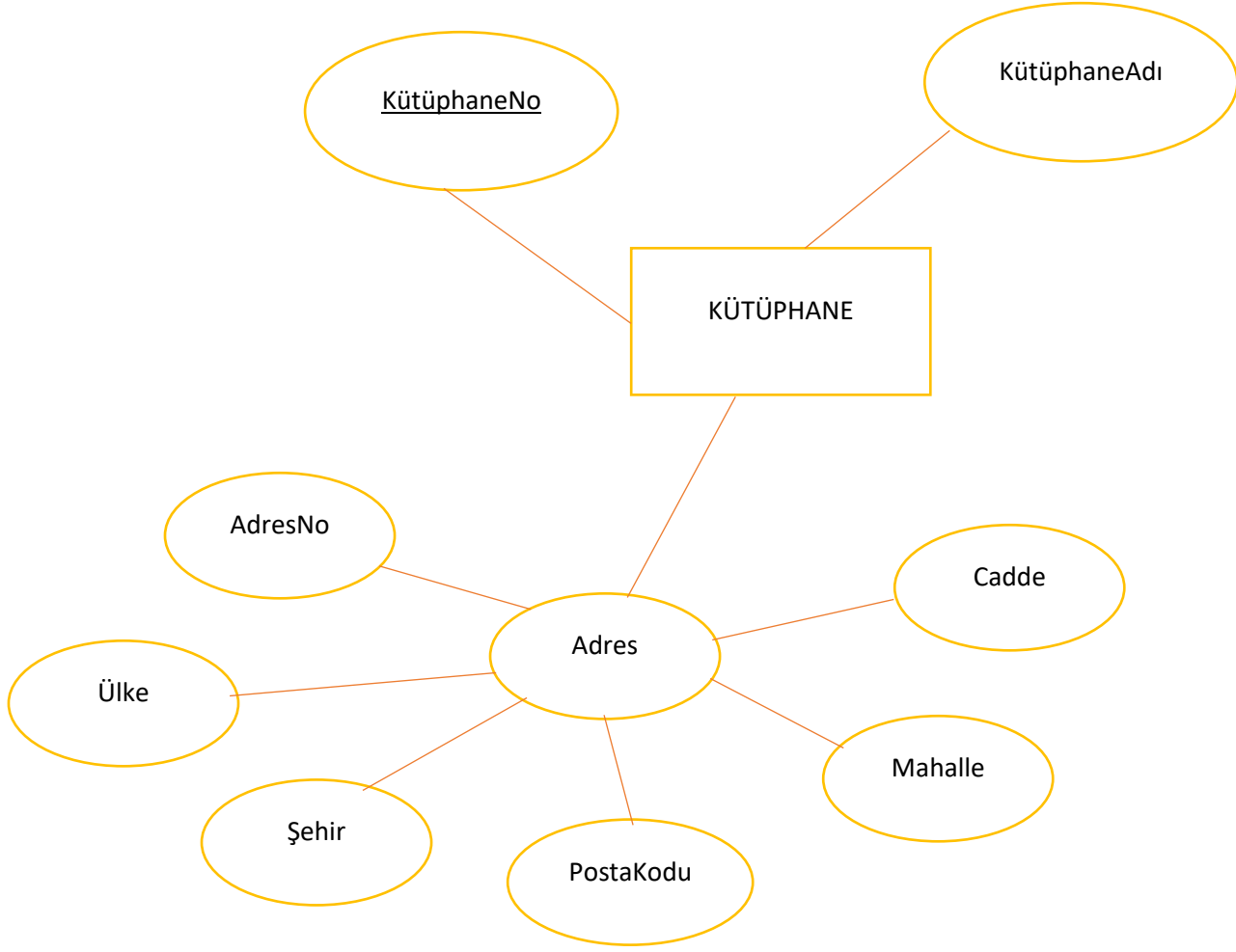
Grup No	10
Proje Adı	Kütüphane Veritabanı Sistemi
1. Öğrenci No Adı ve Soyadı	Kaan Sertel 18360859014
2. Öğrenci No Adı ve Soyadı	Aslıhan Akbulut 19360859057
3. Öğrenci No Adı ve Soyadı	İrem Sertel 18360859005

Kullanılan Teknolojiler:

HTML5 , CSS , JS , PHP

Web tabanlı olabilmesi için bu projede yukarıdaki teknolojileri kullanmayı uygun gördük. Veritabanı oluşturmak için daha kapsamlı olduğunu düşündüğümüz MySQL Workbench'i kullandık.

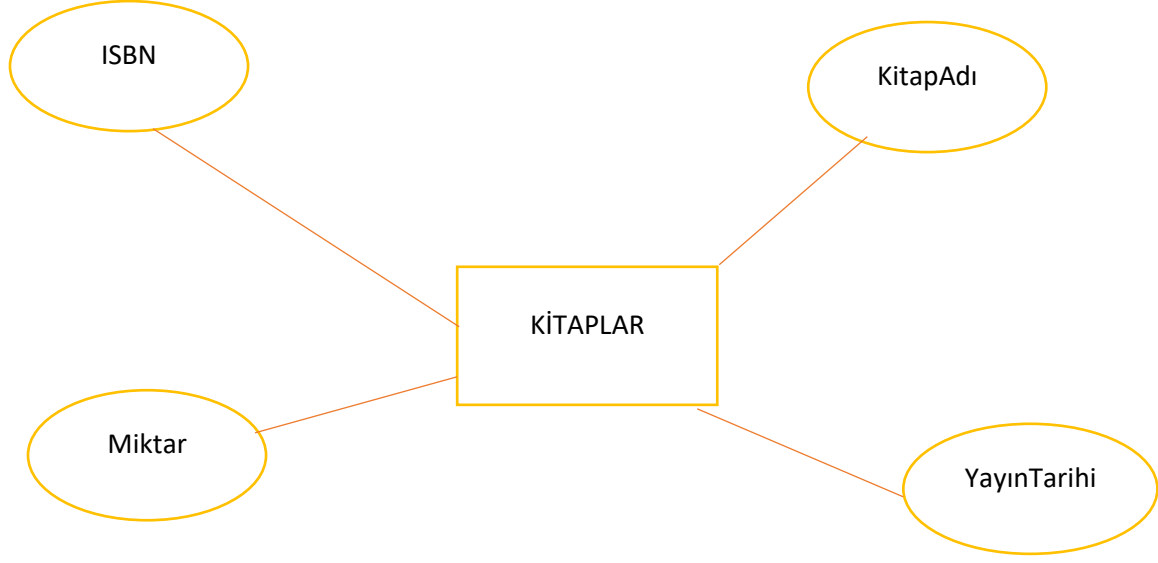
KÜTÜPHANE VARLIĞI VE NİTELİKLERİ



KütüphaneNo → Birincil Anahtar

**Kütüphanelerin adresleri ayrı bir yerde tutulacağı için
Kütüphane varlığı ile adreslerin ayrı olarak tutulduğu
Adres varlığı ilişkilendirilmiştir.**

KİTAPLAR VARLIĞI VE NİTELİKLERİ

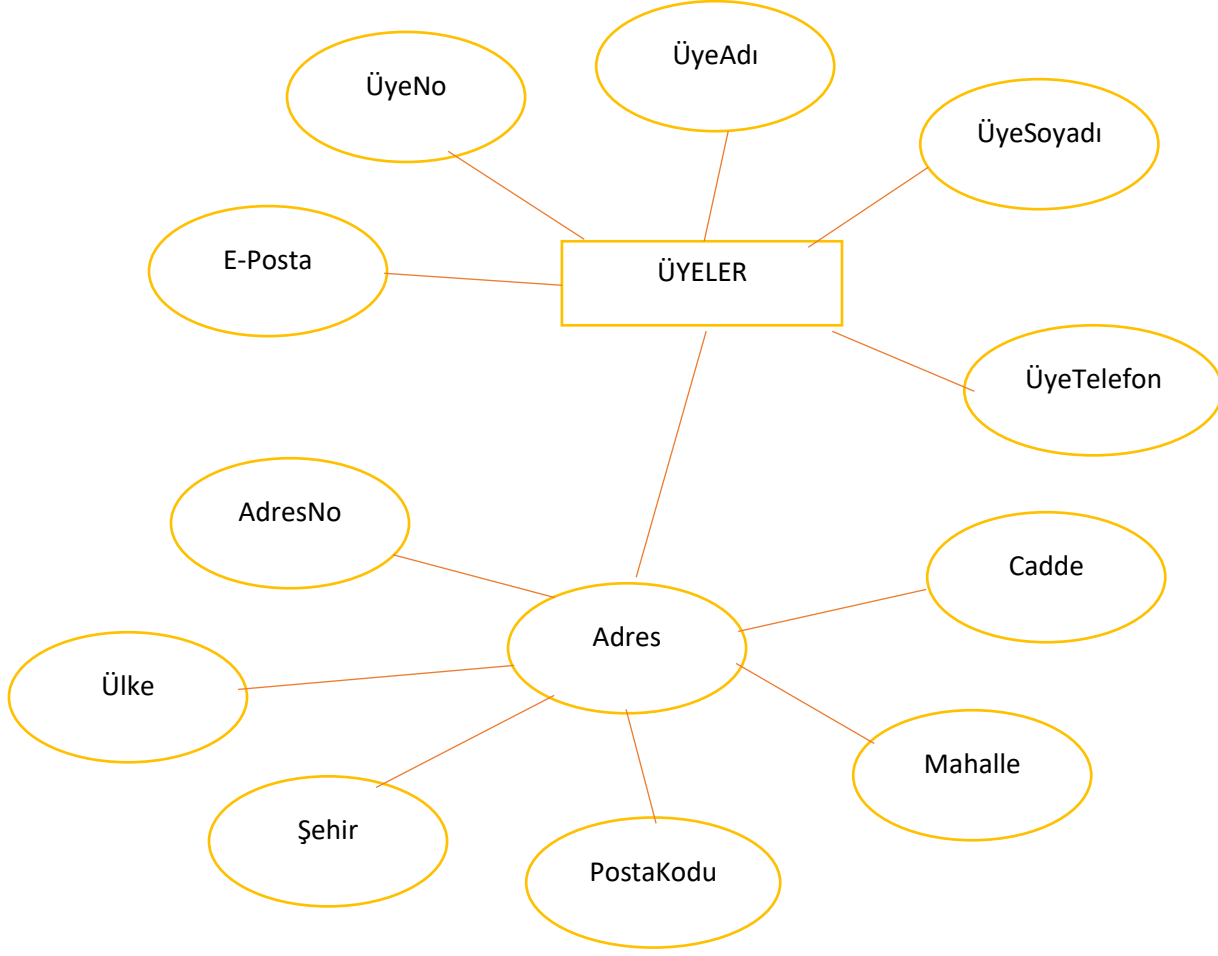


ISBN → Birincil Anahtar

Her kitap sadece kendisine ait ISBN numarasına sahiptir.

Aynı kitaptan birden çok miktarda olabilir ve bu kitapların ISBN numaraları ortaktır.

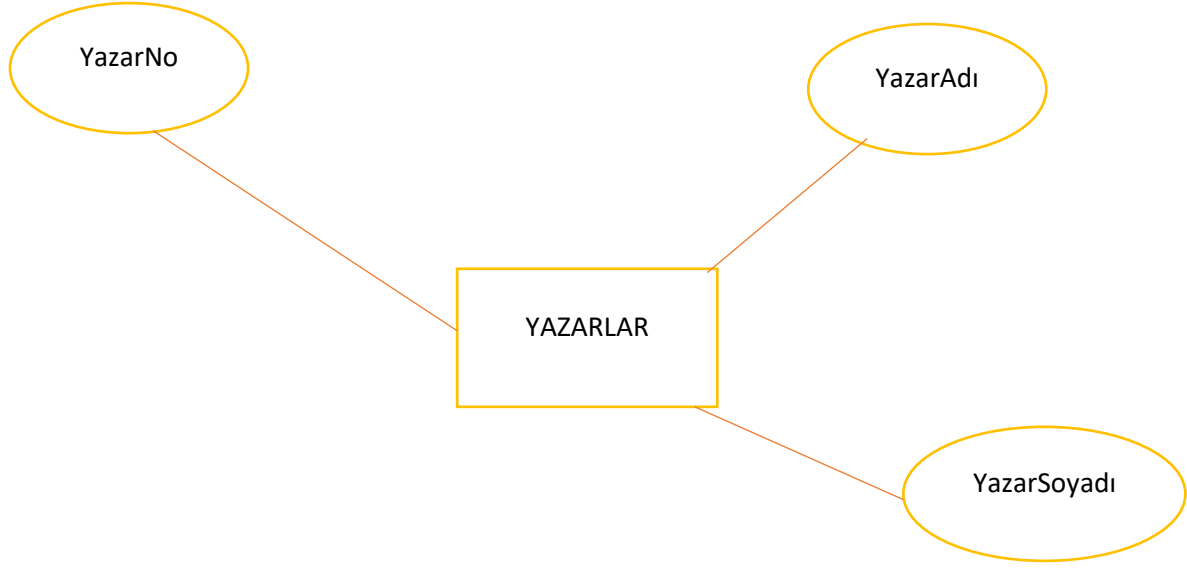
ÜYELER VARLIĞI NİTELİKLERİ



ÜyeNo → Birincil Anahtar

Üyelerin adresleri ayrı bir yerde tutulacağı için üyeler varlığı ile adreslerin ayrı olarak tutulduğu adres varlığı ilişkilendirilmiştir.

YAZARLAR VARLIĐI VE NİTELİKLERİ

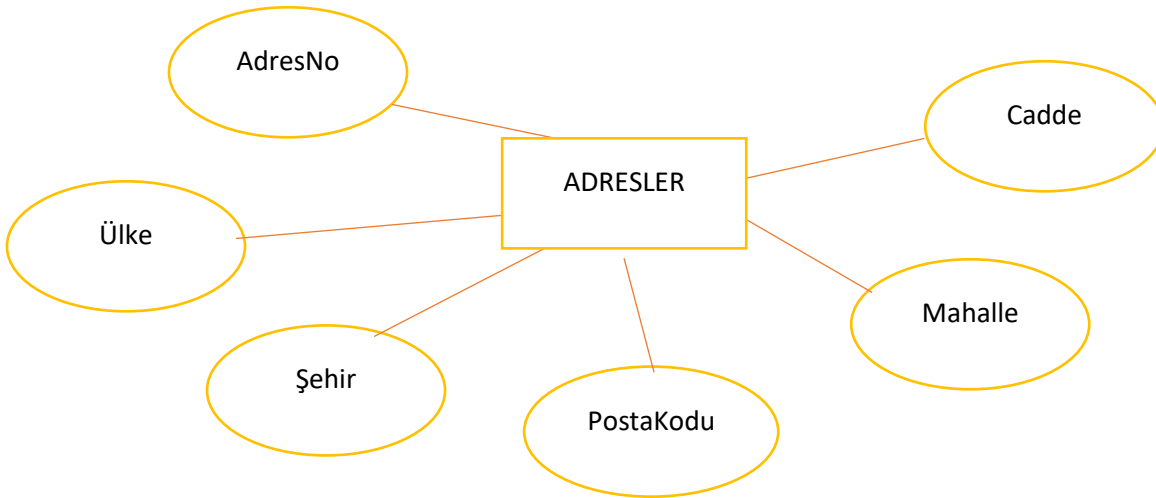


YazarNo → Birincil Anahtar

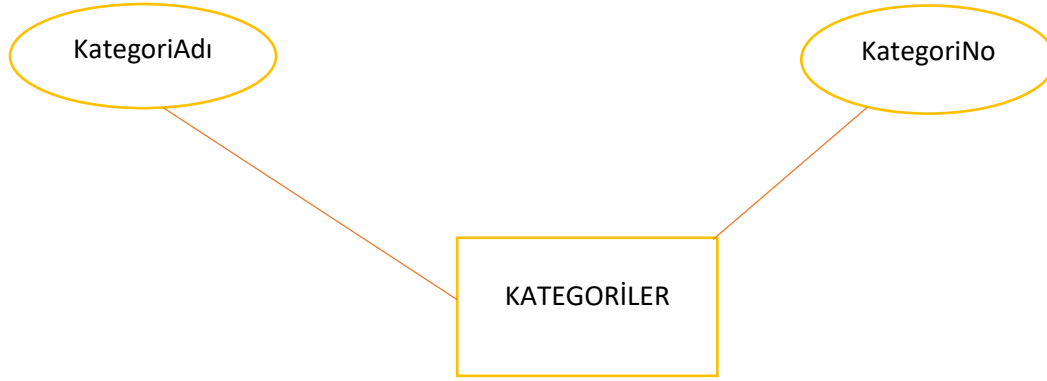
Her yazarın kendisine özel YazarNo'su vardır.

ADRESLER VARLIĐI VE NİTELİKLERİ

AdresNo → Birincil Anahtar



KATEGORİLER VARLIĞI VE NİTELİKLERİ



KategoriNo → Birincil Anahtar

VARLIKLAR ARASI İLİŞKİLER

Kütüphane-Adres



Bir kütüphanenin bir adresi olabilir.

Bir adres bir kütüphaneye ait olabilir.

Kitaplar-Kütüphane



Bir kütüphanede birden çok kitap olabilir.

Bir kitap bir çok kütüphanede olabilir.

Kitaplar-Yazarlar



Bir kitabın birden fazla yazarı olabilir.

Bir yazarın birden fazla kitabı olabilir.

Kitaplar-Kategoriler



Bir kitabın birden çok kategorisi olabilir.

Bir kategori birden çok kitaba ait olabilir.

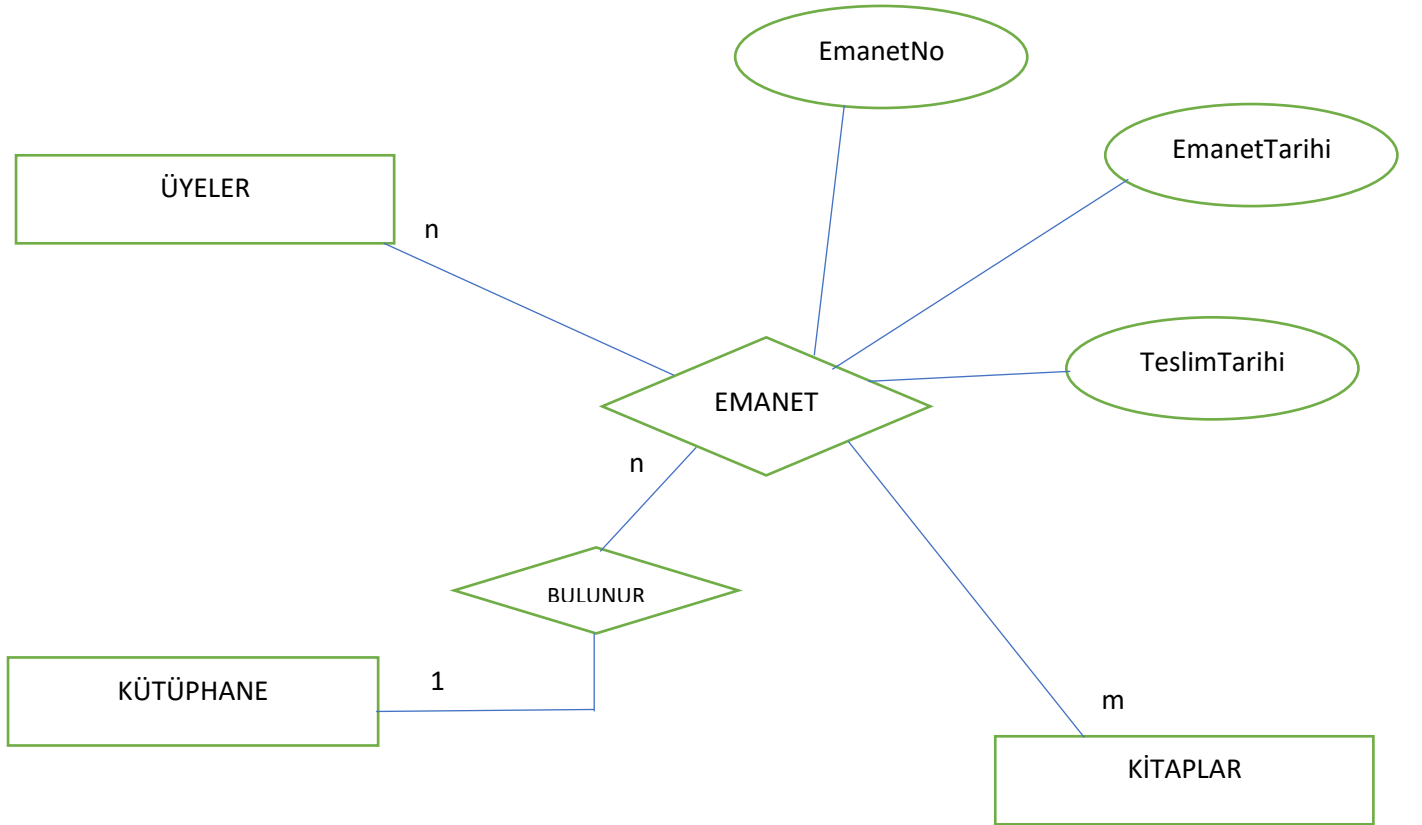
Üye-Adres



Bir üyenin bir adresi olabilir.

Bir adreste birden çok üye olabilir.

Kütüphane-Üyeler-Kitaplar



Bir kütüphaneden birden çok üye birden çok kitap ödünç alabilir.

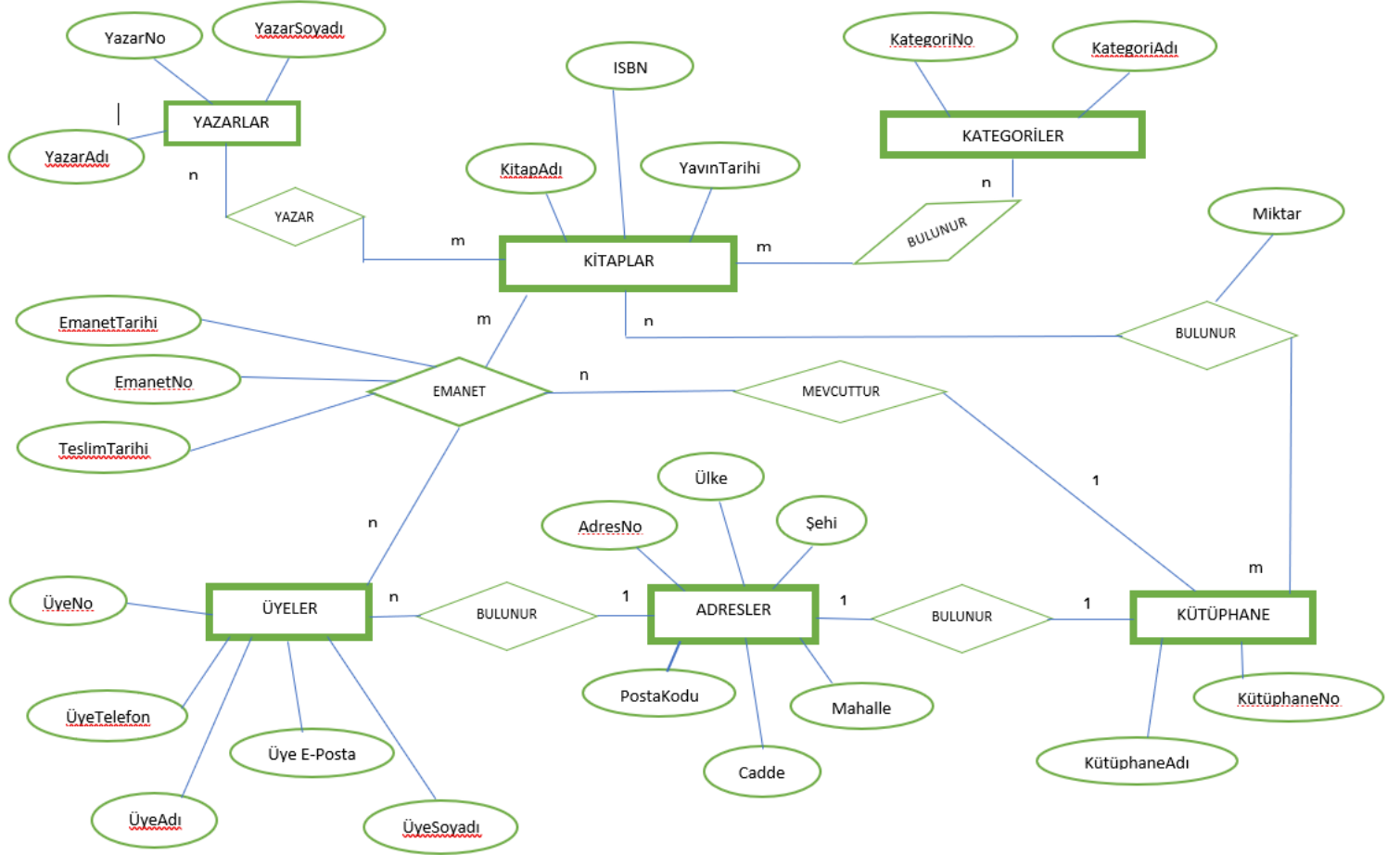
Bir üye birden çok kitap ödünç alabilir.

Bir kitap birden çok üye tarafından ödünç alınabilir.

Bir emanet bir kütüphaneden alınabilir.

Bir kütüphaneden birden çok emanet alınabilir.

Genel Varlık İlişki Diyagramı



Veri Tabanının Açıklaması (Tabloların ve Alanların Amacı)

Normalizasyon Aşamaları:

Kütüphane													
KütüphaneNo	Kütüphane Adı	Kütüphane Adres	ÜyeAd Soyad	ÜyeTelefon	ÜyeE-Posta	Üye Adres	ISBN	KitapAdı	Kitap YayınTarihi	Yazar	Kategori	EmanetTarihi	TeslimTarihi



Normalizasyonun ilk aşamasında aynı sütunda olan birden fazla nitelikleri ayırarak farklı sütunlarda gösterdik.

→ ÜyeAd ve ÜyeSoyad kısmını ayrı yerlere yazdık.

Kütüphanelerin ve üyelerin adresleri ayrı olarak tutulacağı için Adresler adı altında yeni bir tablo oluşturduk ve nitelikleri belirledik.

Bu aşamada Ana Tablodan toplamda 4 ayrı tablo elde etmiş olduk.

Kütüphane		
KütüphaneNo	KütüphaneAdı	AdresNo

Kitaplar				
ISBN	KitapAdı	KitapYayınTarihi	KitapYazarı	Kategori

Üyeler							
ÜyeNo	ÜyeAd	ÜyeSoyad	ÜyeTelefon	ÜyeE-Posta	AdresNo	EmanetTarihi	TeslimTarihi

Adresler					
AdresNo	Ülke	Şehir	PostaKodu	Mahalle	Cadde



Kitaplar tablosundan Yazar ve Kategori kısmını ayırarak Yazarlar ve Kategori tablosu diye 2 ayrı tablo oluşturduk.

→ Bir kitabın birden fazla yazarı olabileceği gibi birden fazla kategoriye de sahip olacağını düşündük.

Kütüphane		
KütüphaneNo	KütüphaneAdı	AdresNo

Kitaplar			
ISBN	KitapAdı	KitapYayın Tarihi	Miktar

Üyeler							
ÜyeNo	ÜyeAd	ÜyeSoyad	ÜyeTelefon	ÜyeE-Posta	AdresNo	EmanetTarihi	TeslimTarihi

Adresler					
AdresNo	Ülke	Şehir	PostaKodu	Mahalle	Cadde

Yazarlar		
YazarNo	YazarAd	YazarSoyad

Kategori	
KategoriNo	KategoriAdı



Son aşamada 6 ana tablo, 6 ara tablo oluşturmuş olduk.

Kütüphane , Üyeler ve Kitaplar arasında oluşan ilişkiye göre Emanet adlı tabloyu oluşturduk.

→Bu kısımdan hangi üyenin hangi kütüphaneden hangi kitabı ne zaman emanet aldığı ve ne zaman teslim ettiği bilgilerine ulaşılabilir.

Kütüphane	
KütüphaneNo	KütüphaneAdı

Kitaplar		
ISBN	KitapAdı	KitapYayınTarihi

Yazarlar		
YazarNo	YazarAd	YazarSoyad

Üyeler				
ÜyeNo	ÜyeAd	ÜyeSoyad	ÜyeTelefon	ÜyeE-Posta

Emanet					
EmanetNo	KütüphaneNo	ÜyeNo	ISBN	EmanetTarihi	TeslimTarihi

Adresler					
AdresNo	Ülke	Şehir	PostaKodu	Mahalle	Cadde

Kategori	
KategoriNo	KategoriAdı

KütüphaneAdres		
KütüphaneAdres	AdresNo	KütüphaneNo
GeçişId		

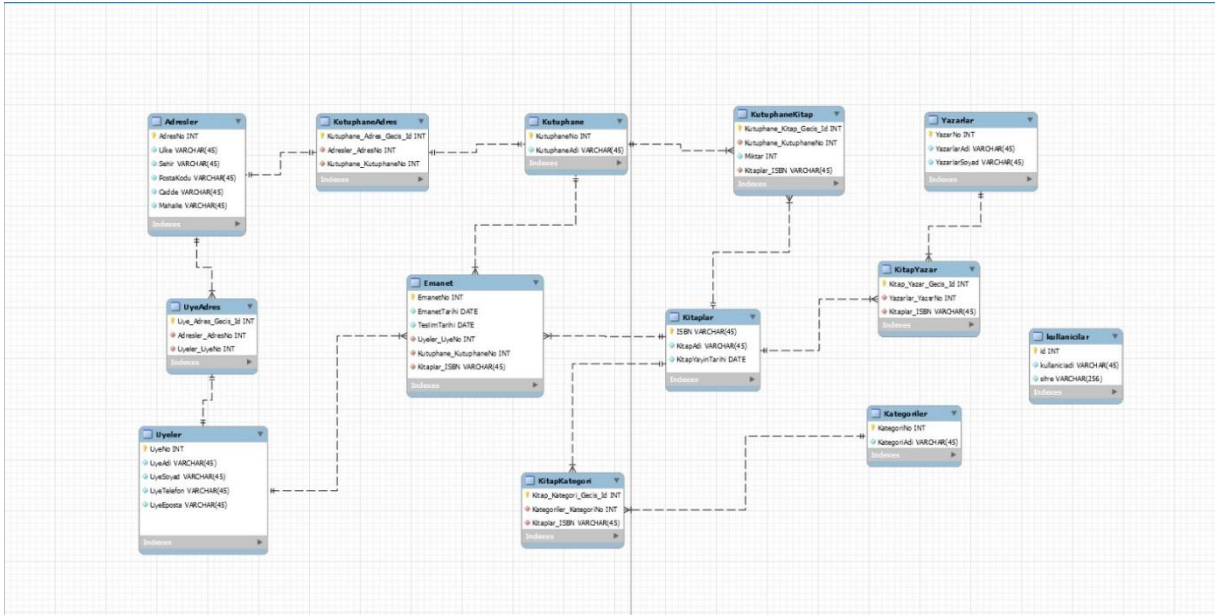
KitapKategori		
KitapKategori	ISBN	KategoriNo
GeçişId		

Üye Adres		
ÜyeAdres	Adres No	Üye No
GeçişId		

KütüphaneKitap			
KütüphaneKitap	Kütüphane No	ISBN	Miktar
GeçişId			

KitapYazar		
KitapYazar	ISBN	YazarNo
GeçişId		

Projemizin Veri Tabanı Yapısı (Veri Tabanı Şeması – ER diyagramı):



Veri Tabanını Oluşturma

CREATE SCHEMA IF NOT EXISTS `kutuphane` ;

Veri Tabanını Kullanma

USE `kutuphane` ;

Tabloları Oluşturma (Tüm Tablolar; açıklaması, ekran çıktısı)

Açıklama: Table `kutuphane`.`Kutuphane`

CREATE TABLE IF NOT EXISTS `kutuphane`.`Kutuphane` (

 `KutuphaneNo` INT NOT NULL AUTO_INCREMENT,

 `KutuphaneAdi` VARCHAR(45) NOT NULL,

 PRIMARY KEY (`KutuphaneNo`),

 UNIQUE INDEX `KutuphaneNo_UNIQUE` (`KutuphaneNo` ASC) VISIBLE)

ENGINE = InnoDB;

Açıklama: Table `kutuphane`.`Yazarlar`

```
CREATE TABLE IF NOT EXISTS `kutuphane`.`Yazarlar` (  
  `YazarNo` INT NOT NULL AUTO_INCREMENT,  
  `YazarlarAdi` VARCHAR(45) NOT NULL,  
  `YazarlarSoyad` VARCHAR(45) NOT NULL,  
  UNIQUE INDEX `YazarNo_UNIQUE` (`YazarNo` ASC) VISIBLE,  
  PRIMARY KEY (`YazarNo`))  
ENGINE = InnoDB;
```

Açıklama: Table `kutuphane`.`Uyeler`

```
CREATE TABLE IF NOT EXISTS `kutuphane`.`Uyeler` (  
  `UyeNo` INT NOT NULL AUTO_INCREMENT,  
  `UyeAdi` VARCHAR(45) NOT NULL,  
  `UyeSoyad` VARCHAR(45) NOT NULL,  
  `UyeTelefon` VARCHAR(45) NOT NULL,  
  `UyeEposta` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`UyeNo`),  
  UNIQUE INDEX `UyeNo_UNIQUE` (`UyeNo` ASC) VISIBLE)  
ENGINE = InnoDB;
```

Açıklama: Table `kutuphane`.`Kitaplar`

```
CREATE TABLE IF NOT EXISTS `kutuphane`.`Kitaplar` (  
  `ISBN` VARCHAR(45) NOT NULL,  
  `KitapAdi` VARCHAR(45) NOT NULL,  
  `KitapYayinTarihi` DATE NOT NULL,  
  PRIMARY KEY (`ISBN`),  
  UNIQUE INDEX `ISBN_UNIQUE` (`ISBN` ASC) VISIBLE)  
ENGINE = InnoDB;
```

Açıklama: Table `kutuphane`.`Adresler`

```
CREATE TABLE IF NOT EXISTS `kutuphane`.`Adresler` (  
  `AdresNo` INT NOT NULL AUTO_INCREMENT,  
  `Ulke` VARCHAR(45) NOT NULL,  
  `Sehir` VARCHAR(45) NOT NULL,  
  `PostaKodu` VARCHAR(45) NOT NULL,  
  `Cadde` VARCHAR(45) NOT NULL,  
  `Mahalle` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`AdresNo`),  
  UNIQUE INDEX `AdresNo_UNIQUE` (`AdresNo` ASC) VISIBLE)  
ENGINE = InnoDB;
```

Açıklama: Table `kutuphane`.`Kategoriler`

```
CREATE TABLE IF NOT EXISTS `kutuphane`.`Kategoriler` (  
  `KategoriNo` INT NOT NULL AUTO_INCREMENT,  
  `KategoriAdi` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`KategoriNo`),  
  UNIQUE INDEX `KategoriNo_UNIQUE` (`KategoriNo` ASC) VISIBLE)  
ENGINE = InnoDB;
```


Açıklama: Table `kutuphane`.`KitapYazar`

```
CREATE TABLE IF NOT EXISTS `kutuphane`.`KitapYazar` (  
  `Kitap_Yazar_Gecis_Id` INT NOT NULL AUTO_INCREMENT,  
  `Yazarlar_YazarNo` INT NOT NULL,  
  `Kitaplar_ISBN` VARCHAR(45) NOT NULL,  
  INDEX `fk_Yazarlar_has_Kitaplar_Yazarlar_idx` (`Yazarlar_YazarNo` ASC) VISIBLE,  
  PRIMARY KEY (`Kitap_Yazar_Gecis_Id`),  
  UNIQUE INDEX `Kitap_Yazar_Gecis_Id_UNIQUE` (`Kitap_Yazar_Gecis_Id` ASC) VISIBLE,  
  INDEX `fk_KitapYazar_Kitaplar1_idx` (`Kitaplar_ISBN` ASC) VISIBLE,  
  CONSTRAINT `fk_Yazarlar_has_Kitaplar_Yazarlar`  
    FOREIGN KEY (`Yazarlar_YazarNo`)  
    REFERENCES `kutuphane`.`Yazarlar` (`YazarNo`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_KitapYazar_Kitaplar1`  
    FOREIGN KEY (`Kitaplar_ISBN`)  
    REFERENCES `kutuphane`.`Kitaplar` (`ISBN`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

Açıklama: Table `kutuphane`.`KitapKategori`

```
CREATE TABLE IF NOT EXISTS `kutuphane`.`KitapKategori` (  
  `Kitap_Kategori_Gecis_Id` INT NOT NULL AUTO_INCREMENT,  
  `Kategoriler_KategoriNo` INT NOT NULL,  
  `Kitaplar_ISBN` VARCHAR(45) NOT NULL,  
  INDEX `fk_Kitaplar_has_Kategoriler_Kategoriler1_idx` (`Kategoriler_KategoriNo` ASC)  
  VISIBLE,  
  PRIMARY KEY (`Kitap_Kategori_Gecis_Id`),  
  UNIQUE INDEX `Kitap_Kategori_Gecis_Id_UNIQUE` (`Kitap_Kategori_Gecis_Id` ASC)  
  VISIBLE,  
  INDEX `fk_KitapKategori_Kitaplar1_idx` (`Kitaplar_ISBN` ASC) VISIBLE,  
  CONSTRAINT `fk_Kitaplar_has_Kategoriler_Kategoriler1`  
    FOREIGN KEY (`Kategoriler_KategoriNo`)  
    REFERENCES `kutuphane`.`Kategoriler` (`KategoriNo`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_KitapKategori_Kitaplar1`  
    FOREIGN KEY (`Kitaplar_ISBN`)  
    REFERENCES `kutuphane`.`Kitaplar` (`ISBN`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

Açıklama: Table `kutuphane`.`KutuphaneKitap`

```
CREATE TABLE IF NOT EXISTS `kutuphane`.`KutuphaneKitap` (  
  `Kutuphane_Kitap_Gecis_Id` INT NOT NULL AUTO_INCREMENT,  
  `Kutuphane_KutuphaneNo` INT NOT NULL,  
  `Miktar` INT NOT NULL,  
  `Kitaplar_ISBN` VARCHAR(45) NOT NULL,  
  INDEX `fk_Kitaplar_has_Kutuphane_Kutuphane1_idx` (`Kutuphane_KutuphaneNo` ASC)  
  VISIBLE,  
  PRIMARY KEY (`Kutuphane_Kitap_Gecis_Id`),  
  UNIQUE INDEX `Kutuphane_Kitap_Gecis_Id_UNIQUE` (`Kutuphane_Kitap_Gecis_Id` ASC)  
  VISIBLE,  
  INDEX `fk_KutuphaneKitap_Kitaplar1_idx` (`Kitaplar_ISBN` ASC) VISIBLE,  
  CONSTRAINT `fk_Kitaplar_has_Kutuphane_Kutuphane1`  
    FOREIGN KEY (`Kutuphane_KutuphaneNo`)  
    REFERENCES `kutuphane`.`Kutuphane` (`KutuphaneNo`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_KutuphaneKitap_Kitaplar1`  
    FOREIGN KEY (`Kitaplar_ISBN`)  
    REFERENCES `kutuphane`.`Kitaplar` (`ISBN`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

Açıklama: Table `kutuphane`.`KutuphaneAdres`

```
CREATE TABLE IF NOT EXISTS `kutuphane`.`KutuphaneAdres` (  
  `Kutuphane_Adres_Gecis_Id` INT NOT NULL AUTO_INCREMENT,  
  `Adresler_AdresNo` INT NOT NULL,  
  `Kutuphane_KutuphaneNo` INT NOT NULL,  
  PRIMARY KEY (`Kutuphane_Adres_Gecis_Id`),  
  UNIQUE INDEX `Kutuphane_Adres_Gecis_Id_UNIQUE` (`Kutuphane_Adres_Gecis_Id` ASC)  
  VISIBLE,  
  INDEX `fk_KutuphaneAdres_Adresler1_idx` (`Adresler_AdresNo` ASC) VISIBLE,  
  INDEX `fk_KutuphaneAdres_Kutuphane1_idx` (`Kutuphane_KutuphaneNo` ASC) VISIBLE,  
  CONSTRAINT `fk_KutuphaneAdres_Adresler1`  
    FOREIGN KEY (`Adresler_AdresNo`)  
    REFERENCES `kutuphane`.`Adresler` (`AdresNo`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_KutuphaneAdres_Kutuphane1`  
    FOREIGN KEY (`Kutuphane_KutuphaneNo`)  
    REFERENCES `kutuphane`.`Kutuphane` (`KutuphaneNo`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

Açıklama: Table `kutuphane`.`Emanet`

```
CREATE TABLE IF NOT EXISTS `kutuphane`.`Emanet` (  
  `EmanetNo` INT NOT NULL AUTO_INCREMENT,  
  `EmanetTarihi` DATE NOT NULL,  
  `TeslimTarihi` DATE NOT NULL,  
  `Uyeler_UyeNo` INT NOT NULL,  
  `Kutuphane_KutuphaneNo` INT NOT NULL,  
  `Kitaplar_ISBN` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`EmanetNo`),  
  UNIQUE INDEX `EmanetNo_UNIQUE` (`EmanetNo` ASC) VISIBLE,  
  INDEX `fk_Emanet_Uyeler1_idx` (`Uyeler_UyeNo` ASC) VISIBLE,  
  INDEX `fk_Emanet_Kutuphane1_idx` (`Kutuphane_KutuphaneNo` ASC) VISIBLE,  
  INDEX `fk_Emanet_Kitaplar1_idx` (`Kitaplar_ISBN` ASC) VISIBLE,  
  CONSTRAINT `fk_Emanet_Uyeler1`  
    FOREIGN KEY (`Uyeler_UyeNo`)  
    REFERENCES `kutuphane`.`Uyeler` (`UyeNo`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_Emanet_Kutuphane1`  
    FOREIGN KEY (`Kutuphane_KutuphaneNo`)  
    REFERENCES `kutuphane`.`Kutuphane` (`KutuphaneNo`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_Emanet_Kitaplar1`  
    FOREIGN KEY (`Kitaplar_ISBN`)  
    REFERENCES `kutuphane`.`Kitaplar` (`ISBN`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

Açıklama: Table `kutuphane`.`UyeAdres`

```
CREATE TABLE IF NOT EXISTS `kutuphane`.`UyeAdres` (  
  `Uye_Adres_Gecis_Id` INT NOT NULL AUTO_INCREMENT,  
  `Adresler_AdresNo` INT NOT NULL,  
  `Uyeler_UyeNo` INT NOT NULL,  
  PRIMARY KEY (`Uye_Adres_Gecis_Id`),  
  UNIQUE INDEX `Uye_Adres_Gecis_Id_UNIQUE` (`Uye_Adres_Gecis_Id` ASC) VISIBLE,  
  INDEX `fk_UyeAdres_Adresler1_idx` (`Adresler_AdresNo` ASC) VISIBLE,  
  INDEX `fk_UyeAdres_Uyeler1_idx` (`Uyeler_UyeNo` ASC) VISIBLE,  
  CONSTRAINT `fk_UyeAdres_Adresler1`  
    FOREIGN KEY (`Adresler_AdresNo`)  
    REFERENCES `kutuphane`.`Adresler` (`AdresNo`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_UyeAdres_Uyeler1`  
    FOREIGN KEY (`Uyeler_UyeNo`)  
    REFERENCES `kutuphane`.`Uyeler` (`UyeNo`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

Açıklama: Table `kutuphane`.`kullaniciilar`

```
CREATE TABLE IF NOT EXISTS `kutuphane`.`kullaniciilar` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `kullaniciadi` VARCHAR(45) NOT NULL,  
  `sifre` VARCHAR(256) NOT NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE INDEX `id_UNIQUE` (`id` ASC) VISIBLE)  
ENGINE = InnoDB;
```

Program ile Veritabanı Arasındaki Bağlantının Kurulması

PHP ile veritabanı bağlantısının kurulması için nesneye yönelik desteği ve hızlı olduğu için **MySQLi** kütüphanesi kullanılmıştır.

mysqli_connect() → Veritabanı ile bağlantı kurulması için MySQLi kütüphanesinin **connect()** fonksiyonu kullanılmıştır.

mysqli_connect_error() → Veritabanı ile kurulan bağlantı hatasını döndüren MySQLi kütüphanesinin **connect_error()** fonksiyonu kullanılmıştır.

mysqli_query() → Veritabanı üzerinde bir sorgu çalıştırmak için MySQLi kütüphanesinin **query()** fonksiyonu kullanılmıştır.

mysqli_error() → Veritabanının döndürdüğü hatayı göstermek için MySQLi kütüphanesinin **error()** fonksiyonu kullanılmıştır.

mysqli_fetch_array() → Sorgudan dönen her bir cevap değişkenini teker teker alıp diziye atabilmek için MySQLi kütüphanesinin **fetch_array()** fonksiyonu kullanılmıştır.

mysqli_close() → Veritabanı bağlantısının kapatılması için MySQLi kütüphanesinin **close()** fonksiyonu kullanılmıştır.

Kodların Dokümantasyonu ve Çalışma Mantığı

Admin Giriş Paneli:



```
session_start();
require ('_mysqlbaglan.php');

if (isset($_POST['username']) and isset($_POST['password'])) {
    extract($_POST);

    $password=hash('sha256', $password);
    $sql = "SELECT * FROM kullanicilar WHERE kullanciadi='$username' and sifre='$password'";

    $cevap = mysqli_query($baglanti,$sql);
    // eger cevap FALSE ise HATA yazdiriyoruz.
    if (!$cevap) {
        echo "<br>Hata: " . mysqli_error($baglanti);
    }
    //veritabanından dönen satır sayısını bulma
    $say = mysqli_num_rows($cevap);
    if ($say == 1) {
        $_SESSION['username'] = $username;
    }else{
        $mesaj="<h1>Hatalı kullanıcı adı veya şifre! </h1>";
    }
}
```

Admin girişi için forma girdiğimiz username ve şifrelenmiş password değişkenlerinin "SELECT * FROM kullanicilar WHERE kullanciadi='\$username' and sifre='\$password'"; sorgusu kullanılarak kullanıcı adı ve şifrenin veritabanındaki bilgilerle eşleşip eşleşmediği kontrol edilerek oturum açılması sağlanmıştır.

Admin Ekleme Sayfası:



```
//Kullanıcı adı ve şifresinin bos veya dolu oldugunu kontrol etme
if (!empty($_POST['username']) && !empty($_POST['password'])) {
    extract($_POST);

    $password=$_POST['password'];
    $password=hash('sha256', $password);

    $sql = "INSERT INTO kullanicilar".
        "(kullanciadi,sifre)".
        "VALUES ('$username','$password')";

    $cevap = mysqli_query($baglanti,$sql);
}
```

"INSERT INTO kullanicilar". "(kullanciadi,sifre)".

"VALUES ('\$username','\$password')";

Sorgusu kullanılarak username ve passwordun kullanicilar tablosuna eklenmesi sağlanmıştır.

Üye Ekleme Sayfası:

Üye Bilgileri

Adı:

Soyadı:

Telefon Numarası:

E-Posta:

Üye Adres Bilgileri

Ülke:

Şehir:

Posta Kodu:

Cadde:

Mahalle:

KAYDET

Geri Dön

```
//Üye ekleme sorgusunu hazırlıyoruz
$sql1 = "INSERT INTO Uyeler" .
        "(UyeAdi,UyeSoyad,UyeTelefon,UyeEposta)" .
        "VALUES ('$UyeAdi','$UyeSoyad','$UyeTelefon','$UyeEposta')";

//Adres ekleme sorgusunu hazırlıyoruz
$sql2= "INSERT INTO Adresler" .
        "(Ulke,Sehir,Postakodu,Cadde,Mahalle)" .
        "VALUES ('$Ulke','$Sehir','$Postakodu','$Cadde','$Mahalle')";

//Sorguları veritabanına gönderiyoruz
$cevap1 = mysqli_query($baglanti,$sql1);
$cevap2= mysqli_query($baglanti,$sql2);

//Eger cevap FALSE ise HATA yazdırıyoruz
if (($cevap1 && ($cevap2))) {
    echo "<div>HATA: ", mysqli_error($baglanti);
}

if ($cevap1 && $cevap2) {
    $sql3 = "SELECT UyeNo FROM Uyeler WHERE UyeAdi='$UyeAdi' AND UyeSoyad='$UyeSoyad' AND UyeTelefon='$UyeTelefon' AND UyeEposta='$UyeEposta'";
    $cevap3 = mysqli_query($baglanti,$sql3);
    $sql4 = "SELECT AdresNo FROM Adresler WHERE Ulke='$Ulke' AND Sehir='$Sehir' AND Postakodu='$Postakodu' AND Cadde='$Cadde' AND Mahalle='$Mahalle'";
    $cevap4 = mysqli_query($baglanti,$sql4);
    //veritabanından gelen değerleri satır satır alma
    $gelen=mysqli_fetch_array($cevap3);
    $UyeNo=(int)$gelen["UyeNo"];

    $gelen2=mysqli_fetch_array($cevap4);
    $AdresNo=(int)$gelen2["AdresNo"];

    $sql5 = "INSERT INTO UyeAdres" .
            "(Adresler.AdresNo,Uyeler.UyeNo)" .
            "VALUES ($Adres,$UyeNo)";

    $cevap5 = mysqli_query($baglanti,$sql5);
    echo "<div>Üye Bilgileri Veritabanına Eklendi.</div>";
}
```

Formdan girilen veriler INSERT INTO kullanılarak önce üyeler ve adresler tablosuna eklenmiş sonrasında SELECT sorgusunda bu bilgilerin üyeler ve adresler tablosunda eşleştiği “UyeNo” ve “AdresNo” alınarak UyeAdres tablosuna eklenmiştir.

Üye Bilgileri Listeleme:

```
//sorguyu yazıyoruz
$sql1 = " SELECT Uyeler.UyeNo, Uyeler.UyeAdi, Uyeler.UyeSoyad, Uyeler.UyeTelefon, Uyeler.UyeEposta, Adresler.AdresNo, Adresler.Ulke, Adresler.Sehir, Adresler.PostaKodu, Adresler.Cadde, Adresler.Mahalle
FROM ((Uyeler
INNER JOIN UyeAdres ON Uyeler.UyeNo = UyeAdres.Uyeler.UyeNo)
INNER JOIN Adresler ON UyeAdres.Adresler_AdresNo = Adresler.AdresNo)";
```

Üye Bilgileri										
Üye No	Üye Adı	Üye Soyadı	Üye Telefon Numarası	Üye E-posta	Üye AdresNo	Ülke	Şehir	PostaKodu	Cadde	Mahalle
3	Fulya	Deniz	232323233	deneme22@gmail.com	4	Türkiye	Antalya	12121	Millet	Kars
4	Kaan	Sertel	555555555	deneme321@gmail.com	5	Türkiye	Bursa	16270	Yıldırım	Millet
5	Ece	Yılmaz	222222222	deneme2e@gmail.com	6	Türkiye	Tokat	11270	Özgür	Park

INNER JOIN ile üç tablonun kesişim verilerine göre tablolardaki bilgiler alınarak üye ve üyenin adres bilgileri listelenmiştir.

Üye Bilgilerini Silme ve Güncelleme:

Üye Bilgileri												
Üye No	Üye Adı	Üye Soyadı	Üye Telefon Numarası	Üye E-posta	Üye AdresNo	Ülke	Şehir	PostaKodu	Cadde	Mahalle		
3	Fulya	Deniz	232323233	deneme22@gmail.com	4	Türkiye	Antalya	12121	Millet	Kars	Sil	Güncelle
4	Kaan	Sertel	555555555	deneme321@gmail.com	5	Türkiye	Bursa	16270	Yıldırım	Millet	Sil	Güncelle
5	Ece	Yılmaz	222222222	denem1e@gmail.com	6	Türkiye	Tokat	11270	Özgür	Park	Sil	Güncelle

```
while($gelen=mysqli_fetch_array($cevap)){
    // veritabanından gelen değerler ile tablo satırları oluşturalım
    echo "<tr><td>".$gelen['UyeNo']. "</td>";
    echo "<td>".$gelen['UyeAdi']. "</td>";
    echo "<td>".$gelen['UyeSoyad']. "</td>";
    echo "<td>".$gelen['UyeTelefon']. "</td>";
    echo "<td>".$gelen['UyeEposta']. "</td>";
    echo "<td>".$gelen['AdresNo']. "</td>";
    echo "<td>".$gelen['Ulke']. "</td>";
    echo "<td>".$gelen['Sehir']. "</td>";
    echo "<td>".$gelen['PostaKodu']. "</td>";
    echo "<td>".$gelen['Cadde']. "</td>";
    echo "<td>".$gelen['Mahalle']. "</td>";
    // sil linki oluşturma
    echo "<td><a class='button' href=kayitsil.php?id=";
    echo $gelen['UyeNo'];
    echo "&no=";
    echo $gelen['AdresNo'];
    echo ">Sil</a></td>";
    // Güncelleştirme
    echo "<td><a class='button' href=guncelle.php?id=";
    echo $gelen['UyeNo'];
    echo "&no=";
    echo $gelen['AdresNo'];
    echo ">Güncelle</a></td></tr>";
}
```

INNER JOIN ile üç tablonun kesişim verilerine göre tablolardaki bilgiler alınarak üye ve üyenin adres bilgileri listelenmiş ve “UyeNo” ve “AdresNo” sil ve güncelle sayfalarına gönderilmiştir.

```
// sorgu yazma
$sql1 = "DELETE FROM Uyeler WHERE UyeNo=".$$_GET['id'];
$sql2 = "DELETE FROM Adresler WHERE AdresNo=".$$_GET['no'];
$sql3 = "DELETE FROM UyeAdres WHERE Uye_Adres_Gecis_Id=".$$_GET['id'];

// sorguyu veritabanına gönderme
$cevap3 = mysqli_query($baglanti,$sql3);
$cevap = mysqli_query($baglanti,$sql1);
$cevap2 = mysqli_query($baglanti,$sql2);
```

DELETE sorgusuyla önce geçiş tablosundaki veriler silinmiş sonrasında ana tablolardaki ilgili veriler silinmiştir.

Adi:

Soyad:

Telefon Numarası:

E-Posta:

Ulke:

Sehir:

Posta Kodu:

Cadde:

Mahalle:

KAYDET

İPTAL

```
// sorgu yazma
$sql = "SELECT * FROM Uyeler WHERE UyeNo=".$GET['id'];
$sql2 = "SELECT * FROM Adresler WHERE AdresNo=".$GET['no'];

//sorguyu veritabanına gönderme
$cevap = mysqli_query($baglanti,$sql);
$cevap2 = mysqli_query($baglanti,$sql2);

//Eger cevap FALSE ise HATA yazdırma
if ( (!$cevap) && (!$cevap2) ) {
    echo "<br>Hata: " . mysqli_error($baglanti);
}

//veritabanından gelen cevabi alma
$gelen=mysqli_fetch_array($cevap);
$gelen2=mysqli_fetch_array($cevap2);

$id=(int)$GET['id'];
$no=(int)$GET['no'];

echo "<html>";
echo "<form class= 'form' action='guncelkaydet.php?id=$id&no=$no'";
```

Güncelle sayfasına gönderilen UyeNo(id) ve AdresNo(no) ile SELECT sorgusu kullanılarak form üzerinde veriler yazdırılmıştır.

```
//sorgu hazırlama
$sql = "UPDATE Uyeler SET UyeAdi='$UyeAdi', UyeSoyad='$UyeSoyad', UyeTelefon='$UyeTelefon', UyeEposta='$UyeEposta' WHERE UyeNo=".$GET['id'];

$sql2 = "UPDATE Adresler SET Ulke='$Ulke', Sehir='$Sehir', PostaKodu='$PostaKodu', Cadde='$Cadde', Mahalle='$Mahalle' WHERE AdresNo=".$GET['no'];
```

UPDATE sorgusuyla üye bilgileri ve üyenin adres bilgileri formdan gelen verilerle güncellenmiştir.

Kütüphane Ekleme:

Kutüphane Bilgileri

Adi:

Kutüphane Adres Bilgileri

Ulke:

Şehir:

Posta Kodu:

Cadde:

Mahalle:

KAYDET

Geri Dön

```
//Kutüphane ekleme sorgusunu hazırlıyoruz
$sql = "INSERT INTO Kutüphane".
    "(KutüphaneAdi)".
    "VALUES ( '$KutüphaneAdi' )";

//Adres ekleme sorgusunu hazırlıyoruz
$sql2= "INSERT INTO Adresler".
    "(Ulke,Sehir,PostaKodu,Cadde,Mahalle)".
    "VALUES ('".$Ulke."','".$Sehir."','".$PostaKodu."','".$Cadde."','".$Mahalle."')";

//Sorguları veritabanına gönderiyoruz
$cevap = mysqli_query($baglanti,$sql);
$cevap2= mysqli_query($baglanti,$sql2);

//Eger cevap FALSE ise HATA yazdırıyoruz
if ( ( !$cevap && !$cevap2 ) ) {
    echo "Hata: " . mysqli_error($baglanti);
}

if ($cevap && $cevap2) {
    $sql3 = "SELECT KutüphaneNo FROM Kutüphane WHERE KutüphaneAdi= '$KutüphaneAdi'";
    $cevap3 = mysqli_query($baglanti,$sql3);
    $sql4 = "SELECT AdresNo FROM Adresler WHERE Ulke= '$Ulke' AND Sehir= '$Sehir' AND PostaKodu= '$PostaKodu' AND Cadde= '$Cadde' AND Mahalle= '$Mahalle'";
    $cevap4 = mysqli_query($baglanti,$sql4);
    //veritabanından gelen değerleri satır alma
    $gelen=mysqli_fetch_array($cevap3);
    $No=(int)$gelen[ 'KutüphaneNo' ];
    $gelen2=mysqli_fetch_array($cevap4);
    $Adres=(int)$gelen2[ 'AdresNo' ];
    $sql5 = "INSERT INTO KutüphaneAdres".
        "(Adresler_AdresNo,Kutüphane_KutüphaneNo)".
        "VALUES ( $Adres, $No )";
```

Formdan alınan bilgiler INSERT INTO ile gerekli tablolara eklenmiştir.

Kütüphane Listele:

Kutuphane Bilgileri							
Kutuphane No	Kutuphane Adı	Kutuphane AdresNo	Ulke	Sehir	PostaKodu	Cadde	Mahalle
1	Bilgin	2	Türkiye	Bursa	16270	Özgür	Millet
3	Saygun	8	Türkiye	Ankara	22321	Yıldırım	Kars
4	Özgün	9	Türkiye	Bursa	16270	Millet	Hak

```
//sorguyu yaziyoruz
$sql = " SELECT Kutuphane.KutuphaneNo, Kutuphane.KutuphaneAdi, Adresler.AdresNo, Adresler.Ulke, Adresler.Sehir, Adresler.PostaKodu, Adresler.Cadde, Adresler.Mahalle
FROM ((Kutuphane
INNER JOIN KutuphaneAdres ON Kutuphane.KutuphaneNo = KutuphaneAdres.Kutuphane_KutuphaneNo)
INNER JOIN Adresler ON KutuphaneAdres.Adresler_AdresNo = Adresler.AdresNo);";
```

INNER JOIN ile üç tablonun kesişim verilerine göre tablolardaki bilgiler alınarak kütüphane ve kütüphanenin adres bilgileri listelenmiştir.

Kütüphane Sil ve Güncelle:

Kutuphane Bilgileri									
Kutuphane No	Kutuphane Adı	Kutuphane AdresNo	Ulke	Sehir	PostaKodu	Cadde	Mahalle		
1	Bilgin	2	Türkiye	Bursa	16270	Özgür	Millet	Sil	Güncelle
3	Saygun	8	Türkiye	Ankara	22321	Yıldırım	Kars	Sil	Güncelle
4	Özgün	9	Türkiye	Bursa	16270	Millet	Hak	Sil	Güncelle

```
while($gelen=mysqli_fetch_array($cevap)){
    // veritabanından gelen değerler ile tablo satirlari olusturalim
    echo "<tr><td>".$gelen['KutuphaneNo']. "</td>";
    echo "<td>".$gelen['KutuphaneAdi']. "</td>";
    echo "<td>".$gelen['AdresNo']. "</td>";
    echo "<td>".$gelen['Ulke']. "</td>";
    echo "<td>".$gelen['Sehir']. "</td>";
    echo "<td>".$gelen['PostaKodu']. "</td>";
    echo "<td>".$gelen['Cadde']. "</td>";
    echo "<td>".$gelen['Mahalle']. "</td>";
    // sil linki olusturma
    echo "<td><a class ='button' href=kutuphanesil.php?id=";
    echo $gelen['KutuphaneNo'];
    echo "&no=";
    echo $gelen['AdresNo'];
    echo ">Sil</a></td>";
    // guncellestirme
    echo "<td><a class ='button' href=kutuphaneguncelle.php?id=";
    echo $gelen['KutuphaneNo'];
    echo "&no=";
    echo $gelen['AdresNo'];
    echo ">Güncelle</a></td></tr>";
}
```

INNER JOIN ile üç tablonun kesişim verilerine göre tablolardaki bilgiler alınarak kütüphane ve kütüphane adres bilgileri listelenmiş ve “KutuphaneNo” ve “AdresNo” sil ve güncelle sayfalarına gönderilmiştir.

```
// sorgu yazma
$sql = "DELETE FROM Kutuphane WHERE KutuphaneNo=".$_GET['id'];
$sql2 = "DELETE FROM Adresler WHERE AdresNo=".$_GET['no'];
$sql3 = "DELETE FROM KutuphaneAdres WHERE Kutuphane_Adres_Gecis_Id=".$_GET['id'];

// sorguyu veritabanina gönderme
$cevap3 = mysqli_query($baglanti,$sql3);
$cevap = mysqli_query($baglanti,$sql);
$cevap2 = mysqli_query($baglanti,$sql2);
```

DELETE sorgusuyla önce geçiş tablosundaki veriler silinmiş sonrasında ana tablolardaki ilgili veriler silinmiştir.

Adı:

Ülke:

Şehir:

Posta Kodu:

Cadde:

Mahalle:

KAYDET

IPTAL

```
// sorgu yazma
$sql = "SELECT * FROM Kutuphane WHERE KutuphaneNo=".$_GET['id'];
$sql2 = "SELECT * FROM Adresler WHERE AdresNo=".$_GET['no'];

//sorguyu veritabanına gönderme
$cevap = mysqli_query($baglanti,$sql);
$cevap2 = mysqli_query($baglanti,$sql2);

//Eger cevap FALSE ise HATA yazdırma
if ( (!$cevap) && (!$cevap2) ) {
    echo "<br>Hata: " . mysqli_error($baglanti);
}

//veritabanından gelen cevabı alma
$gelen=mysqli_fetch_array($cevap);
$gelen2=mysqli_fetch_array($cevap2);

$id=(int)$_GET['id'];
$no=(int)$_GET['no'];

echo "<html>";
echo "<form class= 'form' action='kutuphaneguncelkaydet.php?id=$id&no=$no'";
```

Güncelle sayfasına gönderilen KutuphaneNo(id) ve AdresNo(no) ile SELECT sorgusu kullanılarak form üzerinde veriler yazdırılmıştır.

```
//sorgu hazırlama
$sql = "UPDATE Kutuphane SET KutuphaneAdi='$KutuphaneAdi' WHERE KutuphaneNo=".$_GET['id'];

$sql2 = "UPDATE Adresler SET Ulke='$Ulke', Sehir='$Sehir', PostaKodu='$PostaKodu', Cadde='$Cadde', Mahalle='$Mahalle' WHERE AdresNo=".$_GET['no'];
```

UPDATE sorgusuyla kütüphane bilgileri ve kütüphanenin adres bilgileri formdan gelen verilerle güncellenmiştir.

Kategori Ekleme:

Kategori Ekleme

Kategori Adı:

KAYDET

Geri Dön

```
//Kategori adinin bos veya dolu oldugunu kontrol etme
if (!empty($_POST['KategoriAdi'])) {
    $KategoriAdi=$_POST["KategoriAdi"];

    $sql = "INSERT INTO Kategoriler".
    "(KategoriAdi)".
    "VALUES ('$KategoriAdi')";
```

INSERT INTO ile Kategoriler tablosuna “KategoriAdi” eklenmiştir.

Kategori Listeleme, Silme ve Güncelleme:

Kategori Bilgileri

Kategori No	Kategori Adı		
1	Felsefe	Sil	Güncelle
2	Aşk	Sil	Güncelle
3	Dram	Sil	Güncelle
4	Bilim	Sil	Güncelle

```
//sorguyu yaziyoruz
$sql = " SELECT * FROM Kategoriler ";
```

```
while($gelen=mysqli_fetch_array($cevap)){
    // veritabanından gelen değerler ile tablo satırları oluşturalım
    echo "<tr><td>".$gelen['KategoriNo'].</td>";
    echo "<td>".$gelen['KategoriAdi'].</td>";
    // sil linki oluşturma
    echo "<td><a class='button' href=kategorisil.php?id=";
    echo $gelen['KategoriNo'];
    echo ">Sil</a></td>";
    // Güncelleştirme
    echo "<td><a class='button' href=kategoriguncelle.php?id=";
    echo $gelen['KategoriNo'];
    echo ">Güncelle</a></td></tr>";
}
```

SELECT sorgusu kullanarak kategori bilgilerini çekiyoruz. mysqli_fetch_array() kullanarak tablo içinde listeliyoruz ve sil ve güncelle sayfalarına "KategoriNo" yu id olarak gönderiyoruz.

```
// sorgu yazma
$sql = "DELETE FROM Kategoriler WHERE KategoriNo=".$_GET['id'];
```

DELETE sorgusuyla sil sayfasına gönderilen id ile eşleşen KategoriNo'ya sahip satırı siliyoruz.

```
// sorgu yazma
$sql = "SELECT * FROM Kategoriler WHERE KategoriNo=".$_GET['id'];
```

SELECT sorgusuyla kategori adı bölümüne KategoriNo ile eşleşen kategori adını yani felsefeyi forma yazdırıyoruz.

```
//sorgu hazırlama
$sql = "UPDATE Kategoriler SET KategoriAdi='".$KategoriAdi' WHERE KategoriNo=".$_GET['id'];
```

Form değişkenine girilen veri UPDATE sorgusuyla güncellenmiştir.

Yazar Ekleme:

Yazar Bilgileri

Adı:

Soyadı:

KAYDET

Geri Dön

```
//Yazar ekleme sorgusunu hazirliyoruz
$sql = "INSERT INTO Yazarlar".
      "(YazarlarAdi,YazarlarSoyad)".
      "VALUES ('".$YazarlarAdi."','".$YazarlarSoyad."')";
```

Formdan alınan veriler INSERT INTO ile Yazarlar tablosuna eklenmiştir.

Yazar Listeleme ve Güncelleme:

Yazar No	Yazar Adı	Yazar Soyadı	
1	Sahakattin	Ali	Güncelle
2	Ömer	Seyfettin	Güncelle
3	Rifat	Ilgaz	Güncelle
4	Oğuz	Atay	Güncelle

```
//sorguyu yaziyoruz
$sql = " SELECT * FROM Yazarlar";
```

```
while($gelen=mysqli_fetch_array($cevap)){
    // veritabanından gelen değerler ile tablo satirlari olusturalim
    echo "<tr><td>".$gelen['YazarNo']." </td>";
    echo "<td>".$gelen['YazarlarAdi']."</td>";
    echo "<td>".$gelen['YazarlarSoyad']."</td>";
    // Guncellestirme
    echo "<td><a class = 'button' href=yazarguncelle.php?id=";
    echo $gelen['YazarNo'];
    echo ">Güncelle</a></td></tr>";
```

SELECT sorgusuyla Yazarlar tablosundaki bilgiler listelenmiştir.

Adı:

Soyadı:

KAYDET

IPTAL

```
// sorgu yazma
$sql = "SELECT * FROM Yazarlar WHERE YazarNo=".$_GET['id'];
```

SELECT sorgusuyla formun içine ilgili YazarNo ile eşleşen Yazarlar tablosundaki bilgiler yazdırıldı.

```
//sorgu hazirlama
$sql = "UPDATE Yazarlar SET YazarlarAdi='".$YazarlarAdi"', YazarlarSoyad='".$YazarlarSoyad"' WHERE YazarNo=".$_GET['id'];
```

Formdan gelen veri ile UPDATE sorgusu kullanılarak yazar adı ve soyadı güncellendi.

Kitap Ekleme:

Kitap Bilgileri

ISBN:

Kitap Adı:

Yayın Tarihi:

Kategori:

Felsefe

Felsefe

Aşk

Dram

Bilim

Ali

Yazarı:

Geri Dön

KAYDET

```
//sorguyu yazıyoruz
$sql = "SELECT * FROM Kategoriler";

//sorguyu veritabanına gönderme
$cevap = mysqli_query($baglanti,$sql);

// eger cevap FALSE ise HATA yazdırıyoruz
if (!$cevap) {
    echo '<br>Hata: ' . mysqli_error($baglanti);
}

echo "<select name='KategoriNo'>";
while($gelen=mysqli_fetch_array($cevap)){
    echo "<option ";
    echo "value=";
    echo $gelen['KategoriNo'];
    echo ">";
    echo $gelen['KategoriAdi'];
    echo "</option>";
}
echo "</select>";
echo "<br /> <br /> ";
```

SELECT sorgusuyla kategori adı bilgilerini alıyoruz ve kitap eklerken kullanıcıya bunlardan birini seçme imkanı tanıyoruz.

```
//sorguyu yazıyoruz
$sql3 = "SELECT * FROM Yazarlar";

//sorguyu veritabanına gönderme
$cevap3 = mysqli_query($baglanti,$sql3);

// eger cevap FALSE ise HATA yazdırıyoruz
if (!$cevap3) {
    echo '<br>Hata: ' . mysqli_error($baglanti);
}

echo "<select name='YazarNo'>";
while($gelen3=mysqli_fetch_array($cevap3)){
    echo "<option ";
    echo "value=";
    echo $gelen3['YazarNo'];
    echo ">";
    echo $gelen3['YazarlarAdi'];
    echo " ";
    echo $gelen3['YazarlarSoyad'];
    echo "</option>";
}
echo "</select>";
echo "<br>";
```

Bu kod ile aynı şekilde SELECT sorgusuyla yazar bilgilerini alıp seçme imkanı tanıyoruz.

```
//Kitap ekleme sorgusunu hazırlıyoruz
$sql = "INSERT INTO Kitaplar".
    "(ISBN,KitapAdi,KitapYayinTarihi)".
    "VALUES ('$ISBN','$KitapAdi','$KitapYayinTarihi')";

//Sorguları veritabanına gönderiyoruz
$cevap = mysqli_query($baglanti,$sql);

//Eger cevap FALSE ise HATA yazdırıyoruz
if (!$cevap) {
    echo '<br>Hata: ' . mysqli_error($baglanti);
}if ($cevap) {

    //KitapKategori ekleme sorgusunu hazırlıyoruz
    $sql2= "INSERT INTO KitapKategori".
        "(Kitaplar_ISBN,Kategoriler_KategoriNo)".
        "VALUES ('$ISBN','$KategoriNo')";

    $cevap2= mysqli_query($baglanti,$sql2);

    //KitapYazar ekleme sorgusunu hazırlıyoruz
    $sql3= "INSERT INTO KitapYazar".
        "(Kitaplar_ISBN,Yazarlar_YazarNo)".
        "VALUES ('$ISBN','$YazarNo')";
```

INSERT INTO ile formdan alınan bilgiler dahilinde kitap, kitapkategori ve kitapyazar bilgilerini ilgili tablolara ekliyoruz.

Kitap Kategorisi Ekleme:

Bir kitabın birden çok kategorisi olabileceği için oluşturulan kitaba ek olarak kategori eklenebiliyor.

Kitap Kategorisi Ekleme

Kitap: 10000000000002-İçimizdeki Şeytan

Kategori: Felsefe

[Geri Dön](#) [KAYDET](#)

```
//KitapKategori ekleme sorgusunu hazırlıyoruz
$sql= "INSERT INTO KitapKategori".
      "(Kitaplar_ISBN,Kategoriler_KategoriNo)".
      "VALUES ('$ISBN','$KategoriNo')";
```

Kitap Yazar Ekleme:

Bir kitabın birden çok yazarı olabileceği için oluşturulan kitaba ek olarak yazar eklenebiliyor.

Kitap Yazar Ekleme

Kitap: 10000000000001-Kürk Mantolu Madonna

Yazar: Sabahattin Ali

[Geri Dön](#) [KAYDET](#)

```
//KitapYazar ekleme sorgusunu hazırlıyoruz
$sql= "INSERT INTO KitapYazar".
      "(Kitaplar_ISBN,Yazarlar_YazarNo)".
      "VALUES ('$ISBN','$YazarNo')";
```

Kitap Bilgileri Listeleme:

Kitap Bilgileri					
ISBN	Kitap Adı	Kitap Yayın Tarihi	Yazar Adı	Yazar Soyadı	Kategori
10000000000001	Kürk Mantolu Madonna	1943-12-18	Sabahattin	Ali	Dram
10000000000001	Kürk Mantolu Madonna	1943-12-18	Sabahattin	Ali	Aşk
10000000000002	İçimizdeki Şeytan	1940-12-18	Sabahattin	Ali	Bilim
10000000000002	İçimizdeki Şeytan	1940-12-18	Ömer	Sevket	Bilim

```
//sorguyu yazıyoruz
$sql = "SELECT Kitaplar.ISBN, Kitaplar.KitapAdi, Kitaplar.KitapYayinTarihi, Yazarlar.YazarlarAdi, Yazarlar.YazarlarSoyad, Kategoriler.KategoriAdi
FROM (((Kitaplar
      INNER JOIN KitapYazar ON Kitaplar.ISBN = KitapYazar.Kitaplar_ISBN)
      INNER JOIN Yazarlar ON KitapYazar.Yazarlar_YazarNo = Yazarlar.YazarNo)
      INNER JOIN KitapKategori ON Kitaplar.ISBN = KitapKategori.Kitaplar_ISBN)
      INNER JOIN Kategoriler ON KitapKategori.Kategoriler_KategoriNo = Kategoriler.KategoriNo)
ORDER BY ISBN";
```

ORDER BY ile ISBN numarasına göre sıralanmış INNER JOIN kullanarak 5 tablonun verilerinin kesiştiği alandaki Kitap bilgilerini yazdırıyoruz.

Kitap Silme ve Güncelleme:

Kitap Bilgileri							
ISBN	Kitap Adı	Kitap Yayın Tarihi	Yazar Adı	Yazar Soyadı	Kategori		
1000000000001	Kürk Mantolu Madonna	1943-12-18	Sabahattin	Ali	Dram	Sil	Güncelle
1000000000001	Kürk Mantolu Madonna	1943-12-18	Sabahattin	Ali	Aşk	Sil	Güncelle
1000000000002	İçimizdeki Şeytan	1940-12-18	Sabahattin	Ali	Bilim	Sil	Güncelle
1000000000002	İçimizdeki Şeytan	1940-12-18	Ömer	Seyfettin	Bilim	Sil	Güncelle

```
while($gelen=mysqli_fetch_array($cevap)){
    // veritabanından gelen değerler ile tablo satırları oluşturalım
    echo "<tr><td>".$gelen['ISBN']. "</td>";
    echo "<td>".$gelen['KitapAdi']. "</td>";
    echo "<td>".$gelen['KitapYayinTarihi']. "</td>";
    echo "<td>".$gelen['YazarlarAdi']. "</td>";
    echo "<td>".$gelen['YazarlarSoyad']. "</td>";
    echo "<td>".$gelen['KategoriAdi']. "</td>";

    // sil linki oluşturma
    echo "<td><a class='button' href=kitapsil.php?id=";
    echo $gelen['ISBN'];
    echo "&KategoriNo=";
    echo $gelen['Kitap_Kategori_Gecis_Id'];
    echo "&YazarNo=";
    echo $gelen['Kitap_Yazar_Gecis_Id'];
    echo ">Sil</a></td>";

    // Güncelleştirme
    echo "<td><a class='button' href=kitapguncelle.php?id=";
    echo $gelen['ISBN'];
    echo "&KategoriNo=";
    echo $gelen['Kitap_Kategori_Gecis_Id'];
    echo "&YazarNo=";
    echo $gelen['Kitap_Yazar_Gecis_Id'];
    echo ">Güncelle</a></td></tr>";
}
```

DELETE sorgularıyla önce geçiş tabloları sonra ana tablo siliniyor.

```
// sorgu yazma
$sql1 = "DELETE FROM Kitaplar WHERE ISBN=".$_GET['id'];
$sql2 = "DELETE FROM KitapKategori WHERE Kitaplar_ISBN=".$_GET['id'];
$sql3 = "DELETE FROM KitapYazar WHERE Kitaplar_ISBN=".$_GET['id'];
```

Kitap Bilgileri

Kitap Adı:

*Doldurulması zorunludur.

Yayın Tarihi:

Yazar:

Kategori:

UPDATE sorgularıyla veriler güncelleniyor.

```
//sorgu hazırlama
$sql1 = "UPDATE Kitaplar SET KitapAdi='$KitapAdi', KitapYayinTarihi='$KitapYayinTarihi' WHERE ISBN=".$_GET['id'];
$sql2 = "UPDATE KitapKategori SET Kategoriler_KategoriNo='$KategoriNo' WHERE Kitap_Kategori_Gecis_Id=".$_GET['no'];
$sql3 = "UPDATE KitapYazar SET Yazarlar_YazarNo='$YazarNo' WHERE Kitap_Yazar_Gecis_Id=".$_GET['no2'];
```

Kütüphaneye Kitap Ekleme:

Kütüphaneye Eklenecek Kitaplar

Kütüphane:

Kitap:

Miktar:

```
//Kitap ekleme sorgusunu hazırlıyoruz
$sql1 = "INSERT INTO KutuphaneKitap".
    "(Kitaplar_ISBN,Kutuphane_KutuphaneNo,Miktar)".
    "VALUES ('$ISBN','$KutuphaneNo','$Miktar')";
```

Sistemde olan kütüphanelerden birini seçerek aynı zamanda sistemde bulunan kitaplardan birini seçerek ve miktar belirleyerek INSERT INTO sorgusunu kullanarak kütüphaneye kitap ekleme işlemi yapıyoruz.

Kütüphane Kitap Bilgileri Listeleme, Silme ve Güncelleme:

Kütüphaneler

Lütfen Kitap Bilgilerinin Listeleneceği Kütüphaneyi Seçiniz!!!

☐ Bilgi ☐ Sayın ☐ Örgün

[Listele](#)

Kitap Bilgileri

ISBN	Kitap Adı	Kitap Yayın Tarihi	Kutuphane Adı	Kitap Miktarı		
1000000000001	Kürk Mantolu Madonna	1943-12-18	Örgün	20	Sil	Güncelle
1000000000002	İçimizdeki Şeytan	1940-12-18	Örgün	10	Sil	Güncelle

```
//sorguyu yazıyoruz
$sql = "SELECT * FROM Kutuphane";

//sorguyu veritabanına gönderme
$cevap = mysqli_query($baglanti,$sql);

// eger cevap FALSE ise HATA yazdırıyoruz
if (!$cevap) {
    echo "<br>Hata:" . mysqli_error($baglanti);
}

while($gelen=mysqli_fetch_array($cevap)){
    echo "<input ";
    echo "type='radio' ";
    echo "name=";
    echo "KutuphaneNo";
    echo " ";
    echo "value=";
    echo $gelen['KutuphaneNo'];
    echo " />";
    echo $gelen['KutuphaneAdi'];
    echo " ";
}
```

```
while($gelen=mysqli_fetch_array($cevap)){
    // veritabanından gelen değerler ile tablo satırları oluşturalım
    echo "<tr><td>.$gelen['ISBN'].</td>";
    echo "<td>.$gelen['KitapAdi'].</td>";
    echo "<td>.$gelen['KitapYayinTarihi'].</td>";
    echo "<td>.$gelen['KutuphaneAdi'].</td>";
    echo "<td>.$gelen['Miktar'].</td>";

    // sil linki oluşturma
    echo "<td><a class='button' href=kutuphanekitapsil.php?id=";
    echo $gelen['ISBN'];
    echo "&KutuphaneNo=";
    echo $No;
    echo "&GecisId=";
    echo $gelen['Kutuphane_Kitap_Gecis_Id'];
    echo ">Sil</a></td>";

    // Güncelleme
    echo "<td><a class='button' href=kutuphanekitapguncelle.php?id=";
    echo $gelen['ISBN'];
    echo "&KutuphaneNo=";
    echo $No;
    echo "&GecisId=";
    echo $gelen['Kutuphane_Kitap_Gecis_Id'];
    echo ">Güncelle</a></td></tr>";
}
```

```
//sorguyu yazıyoruz
$sql = "SELECT Kitaplar.ISBN, Kitaplar.KitapAdi, Kitaplar.KitapYayinTarihi, Kutuphane.KutuphaneAdi, KutuphaneKitap.Miktar, KutuphaneKitap.Kutuphane_Kitap_Gecis_Id
FROM ((Kitaplar
INNER JOIN KutuphaneKitap ON Kitaplar.ISBN = KutuphaneKitap.Kitaplar_ISBN)
INNER JOIN Kutuphane ON KutuphaneKitap.Kutuphane_KutuphaneNo = Kutuphane.KutuphaneNo)
WHERE KutuphaneNo=$No";
```

Sistemde bulunan kütüphaneler listelendi ve seçilen kütüphaneye göre kütüphanedeki kitap verileri ve kütüphanenin adı tablo şeklinde listeleniyor ve DELETE ve UPDATE sorgularıyla silme ve güncelleme işlemleri yapılabiliyor.

```
// sorgu yazma
$sql = "DELETE FROM KutuphaneKitap WHERE Kutuphane_Kitap_Gecis_Id=".$$_GET['GecisId'];
```

```
//sorgu hazırlama
$sql = "UPDATE KutuphaneKitap SET Miktar=$Miktar WHERE Kutuphane_Kitap_Gecis_Id=".$$_GET['id'];
```

Emanet Ekle:

Lütfen Görevli Olduğunuz Kütüphaneyi Seçiniz!!!

☐ Bilgin ☐ Saygın ☒ Özgün

[Listele](#)

Üye No:

Emanet Tarihi:

Teslim Tarihi:

Kitap:

[KAYDET](#)

```
//sorguyu yazıyoruz
$sql = "SELECT Kitaplar_ISBN FROM KutuphaneKitap WHERE Kutuphane_KutuphaneNo=' $No' ORDER BY Kitaplar_ISBN";

//sorguyu veritabanına gönderme
$cevap = mysqli_query($baglanti,$sql);

// eger cevap FALSE ise HATA yazdırıyoruz
if (!$cevap) {
    echo "<br>Hata: " . mysqli_error($baglanti);
}
echo "<br>Kitap:<br>";
echo "<select name='ISBN' >";
while ($gelen=mysqli_fetch_array($cevap)) {
    $no2 = $gelen['Kitaplar_ISBN'];
    $sql2 = "SELECT * FROM Kitaplar WHERE ISBN= '$no2'";
    //sorguyu veritabanına gönderme
    $cevap2 = mysqli_query($baglanti,$sql2);
    while($gelen2=mysqli_fetch_array($cevap2)){
        echo "<option ";
        echo "<value=";
        echo $gelen2['ISBN'] ;
        echo ">";
        echo $gelen2['ISBN'] ;
        echo "<";
        echo $gelen2['KitapAdi'] ;
        echo "</option>";
    }
}
```

```
//Emanet ekleme sorgusunu hazırlıyoruz
$sql = "INSERT INTO Emanet".
    "(EmanetTarihi,TeslimTarihi,Kitaplar_ISBN,Uyeler_UyeNo,Kutuphane_KutuphaneNo)".
    "VALUES ('$EmanetTarihi','$TeslimTarihi','$ISBN','$UyeNo','$KutuphaneNo')";
```

İstenilen kütüphaneden gerekli bilgiler girilerek ve o kütüphanede bulunan kitaplardan birini seçerek emanet alma işlemi gerçekleştirilir.

Emanet Listele , Sil ve Güncelle:

Kütüphaneler

Lütfen Ödünç Alma Bilgilerinin Listeleneceği Kütüphaneyi Seçiniz!!!

☐ Bilgin ☐ Saygın ☒ Özgün

[Listele](#)

Ödünç Alma Bilgileri

Emanet No	Uye No	Uye Adı	Uye Soyadı	Uye Telefon Numarası	Kutuphane	ISBN	Kitap Adı	Emanet Tarihi	Teslim Tarihi	
5	4	Kaan	Sertel	555555555	Özgün	1000000000002	İçimizdeki Şeytan	2021-06-17	0001-01-01	Sil Güncelle

```
//sorguyu yazıyoruz
$sql = "SELECT Emanet.EmanetNo, Emanet.Uyeler_UyeNo, Uyeler.UyeAdi, Uyeler.UyeSoyad, Uyeler.UyeTelefon, Kutuphane.KutuphaneAdi, Kitaplar.ISBN, Kitaplar.KitapAdi, Emanet.EmanetTarihi, Emanet.TeslimTarihi FROM ((Emanet INNER JOIN Kitaplar ON Kitaplar.ISBN = Emanet.Kitaplar_ISBN) INNER JOIN Kutuphane ON Emanet.Kutuphane_KutuphaneNo = Kutuphane.KutuphaneNo) INNER JOIN Uyeler ON Uyeler.UyeNo = Emanet.Uyeler_UyeNo) WHERE KutuphaneNo=' $No' ORDER BY EmanetNo";
```

```
while($gelen=mysqli_fetch_array($cevap)){
    // veritabanından gelen değerler ile tablo satırları olusturalım
    echo "<tr><td class='tablo'>". $gelen['EmanetNo']. "</td>";
    echo "<td>". $gelen['Uyeler_UyeNo']. "</td>";
    echo "<td>". $gelen['UyeAdi']. "</td>";
    echo "<td>". $gelen['UyeSoyad']. "</td>";
    echo "<td>". $gelen['UyeTelefon']. "</td>";
    echo "<td>". $gelen['KutuphaneAdi']. "</td>";
    echo "<td>". $gelen['ISBN']. "</td>";
    echo "<td>". $gelen['KitapAdi']. "</td>";
    echo "<td>". $gelen['EmanetTarihi']. "</td>";
    echo "<td>". $gelen['TeslimTarihi']. "</td>";

    // sil linki oluşturma
    echo "<td><a class='button' href=emanetsil.php?id=";
    echo $gelen['EmanetNo'];
    echo ">Sil</a></td>";

    // Güncelleştirme
    echo "<td><a class='button' href=emanetguncelle.php?id=";
    echo $gelen['EmanetNo'];
    echo ">Güncelle</a></td></tr>";
}
```

İstenilen kütüphaneden, alınan emanetlerin bilgisi listeleniyor. DELETE sorgusuyla emanet silme işlemi, UPDATE sorgusuyla emanet bilgilerini güncelleme işlemi yapıyor.

```
// sorgu yazma
$sql = "DELETE FROM Emanet WHERE EmanetNo=".$_GET['id'];
```

```
//sorgu hazırlama
$sql = "UPDATE Emanet SET EmanetTarihi='$EmanetTarihi', TeslimTarihi='$TeslimTarihi' WHERE EmanetNo=".$_GET['id'];
```

Sonuç:

Üye ve kütüphanenin adres bilgileri adresler tablosunda tutulmaktadır. Ayrıca adres bilgileri üye adres ve kütüphane adres tablolarında ayrı ayrı tutulmaktadır. Bir kitabın birden çok yazarı ve kategorisi bulunabilmektedir. Kitap kategori bilgileri kitap kategori tablosunda, kitap yazar bilgileri kitap yazar tablosunda tutulmaktadır. Bir kitap bir ya da birden çok kütüphanede olabileceği gibi hiçbir kütüphanede de bulunamayabilir. Bu yüzden öncelikle kitap bilgileri sisteme eklenmektedir. Daha sonra istenilen kütüphaneye sistemde bulunan kitaplar miktar verilerek eklenebilir. O kütüphaneye eklenmemiş kitap emanet alınamaz. Bir üye birçok kütüphaneden birçok emanet alabilir. Her kitabın kendine özel ISBN numarası bulunmaktadır.