

BMB202. Veritabanı Yönetimi

Ders 10.

ADO.Net C#

Bir Proje Uygulaması

Dersin Amacı

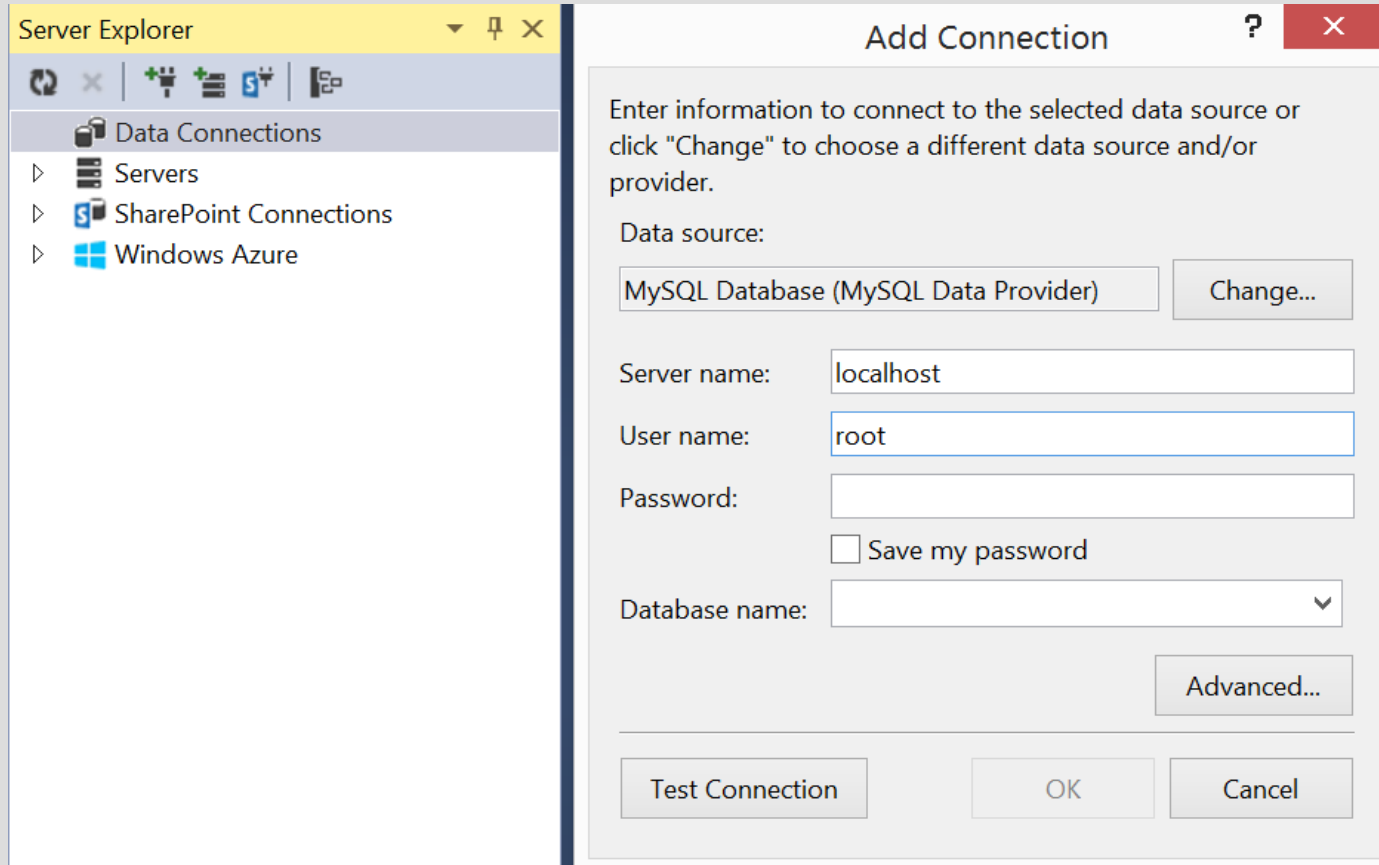
- Veritabanı uygulaması geliştirirken dikkat edilmesi gereken hususlar örnekler üzerinden anlatılacaktır.
- Uygulamada gelişmiş bir telefon rehberi uygulaması geliştirilecektir.
- MySQL veritabanı kullanılacaktır.

Projenin içeriđi

- Gelişmiş bir telefon rehberi projesi
- Kullanıcılar
 - Kullanıcı, şifre ve kullanıcı adı ile bağlansın.
 - kullanıcı türü
 - A: tüm yetkiye sahip
 - G: sadece görünüm modunda açabilir.
 - D: Deđiştirme ve silme yetkisine sahip, ama yeni kayıt yetkisi yok. Ayrıca, A D'nin yaptığı deđişiklikleri izleyebilsin. Yani sildiđi kayıtları dahi görsün.
- İsimler
- Departmanlar (Her kişinin bir departmanı olsun.)
- Telefonlar (Bir kişinin birden fazla telefonu olabilir)
- Oluşturulan tablolar sql formatında ekte verilmiştir.
“telefonrehberi” veritabanını açıp PHPMyAdmin'deki içeri aktar özelliđi tüm tablolar otomatik oluşturulabilir.

Veritabanını tasarlayalım.

Server Explorer – Data Connection



- Veritabanı işlemleri yapılırken Visual Studio'nun **Server Explorer** penceresinden yararlanmak mümkündür.
- Visual Studio'nun Server Explorer penceresi veritabanı işlemleri için kullanılan bir programın penceresi gibidir.
- Server Explorer özellikle Visual Studio 2013 versiyonlarında iyi çalışmaktadır. (MySQL için)
- Server Explorer yerine MySQL Workbench veya PHPMyAdmin uygulamaları tercih edilebilir.

Connection Ekleme

Add Connection ? X

Enter information to connect to the selected data source or click "Change" to choose a different data source and/or provider.

Data source:
MySQL Database (MySQL Data Provider) Change...

Server name: localhost

User name: root

Password: Save my password

Database name:
adys
adys_test
cdcol
demo
egitim
performance_schema
phpmyadmin
telefonrehberi
test
timesheet_unimar
webauth

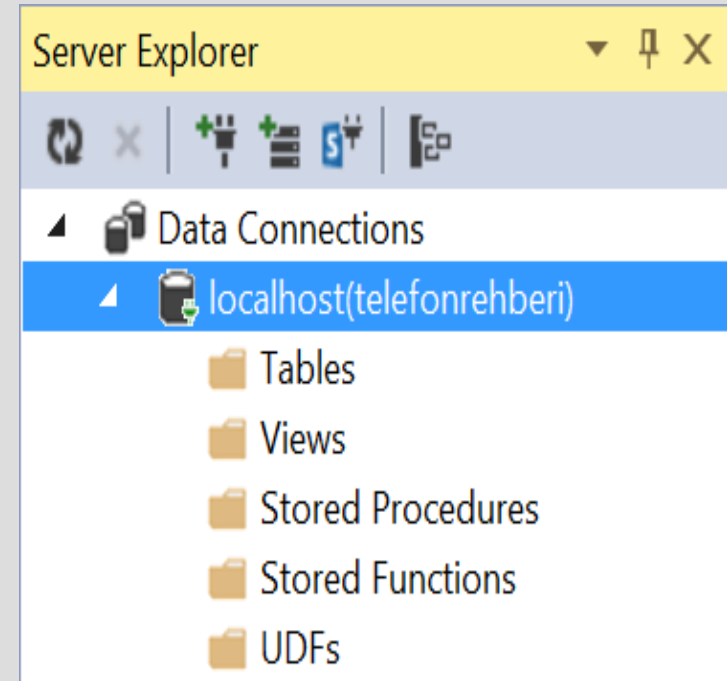
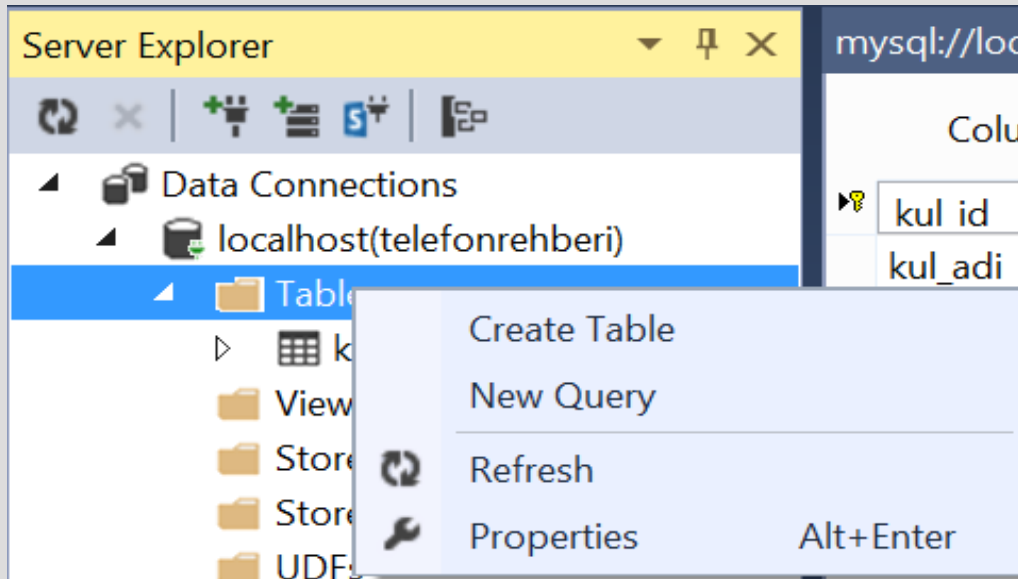
Test Connection

- Connection eklerken
 - Server Name=localhost
 - User Name=root
 - Şifre varsa yazılır.
 - Database Name bölümünden mevcut database seçilir. (Yoksa veritabanını ekleyin.)
- Test Connection ile sorun olup olmadığı test edilebilir.

Connection

Table, View, SP görüntüleme ve ekleme

- Yeni Tablo Ekleme

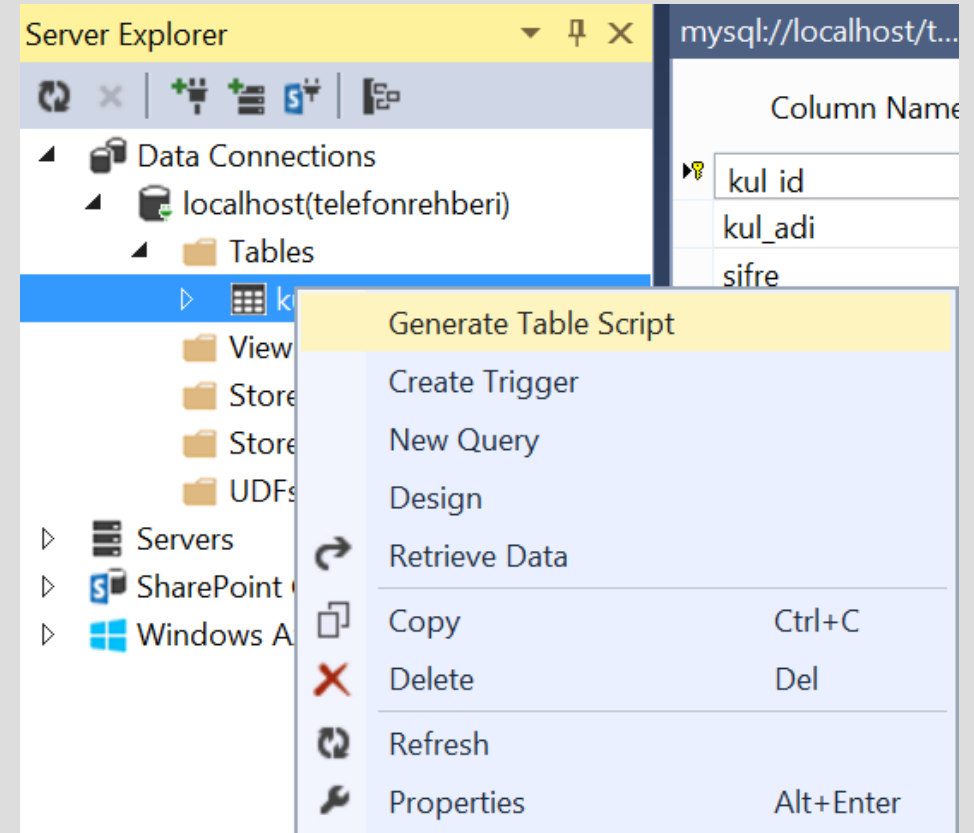


A screenshot of the SQL Server Enterprise Manager interface showing the 'Columns' pane for a table named 'kullanici'. The table has the following columns:

Column Name	Data Type	Allow Nulls
kul_id	int	<input type="checkbox"/>
kul_adi	varchar(20)	<input type="checkbox"/>
sifre	varchar(20)	<input type="checkbox"/>
Rol	varchar(1)	<input type="checkbox"/>
		<input type="checkbox"/>

Table İşlemleri

- Tablo üretme scripti
- Trigger Oluşturma
- Query Oluşturma
- Design
- Veriyi Çekme



New Query

Query3: Query(localhost) x mysql://localhost/t...fonrehberi/kullanici kullanici (telefonreh...ri): Query(localhost) Form1.cs [Design]

kullanici

* (All Columns)

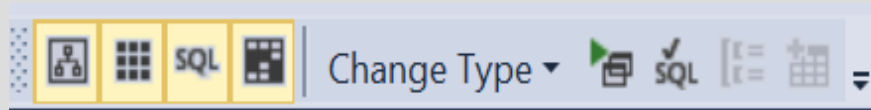
Column	Alias	Table	Outp...	Sort Type	Sort Order	Filter	Or...	Or...	Or...
			<input type="checkbox"/>						
			<input type="checkbox"/>						
			<input type="checkbox"/>						
			<input type="checkbox"/>						

```
SELECT
FROM kullanici
```

- Üst bölümden ekran görünümünü ayarlayabilir.
- Change Type bölümünden Query tipini seçebilirsiniz.
 - **Select, Insert, Update, Delete**

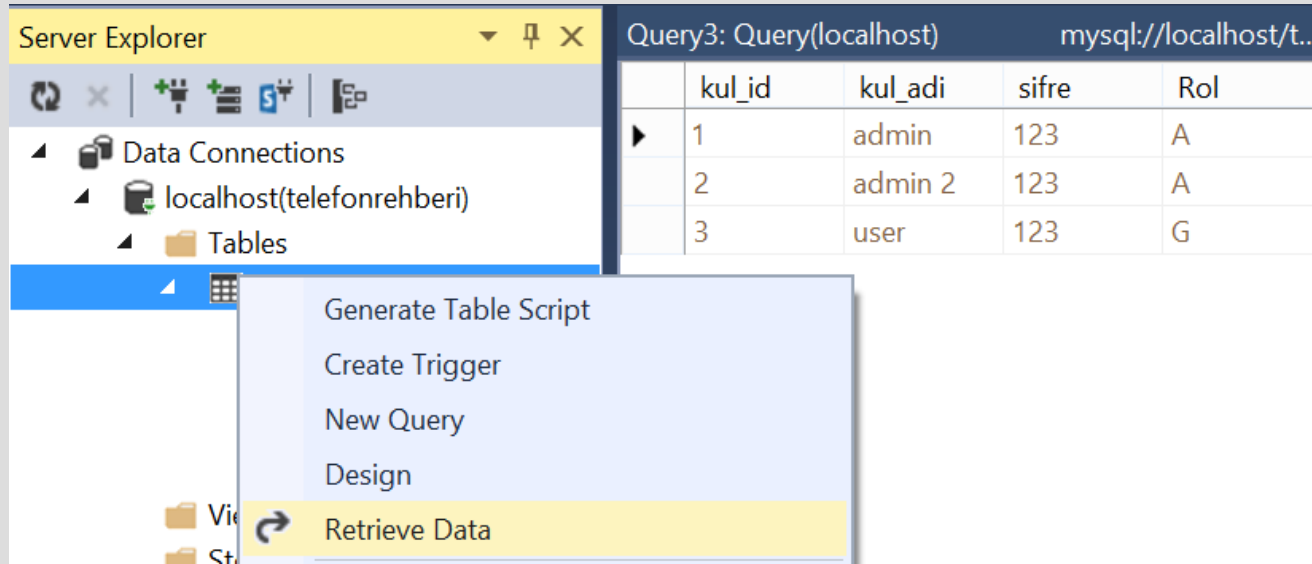
Kullanıcı Ekleme

```
INSERT INTO kullanıcı  
      (kul_adi, sifre, Rol)  
VALUES ('admin', '123', 'A')
```



- Execute SQL iconun'dan query çalıştırabilir.
- Sisteme birinci kullanıcı eklendi.

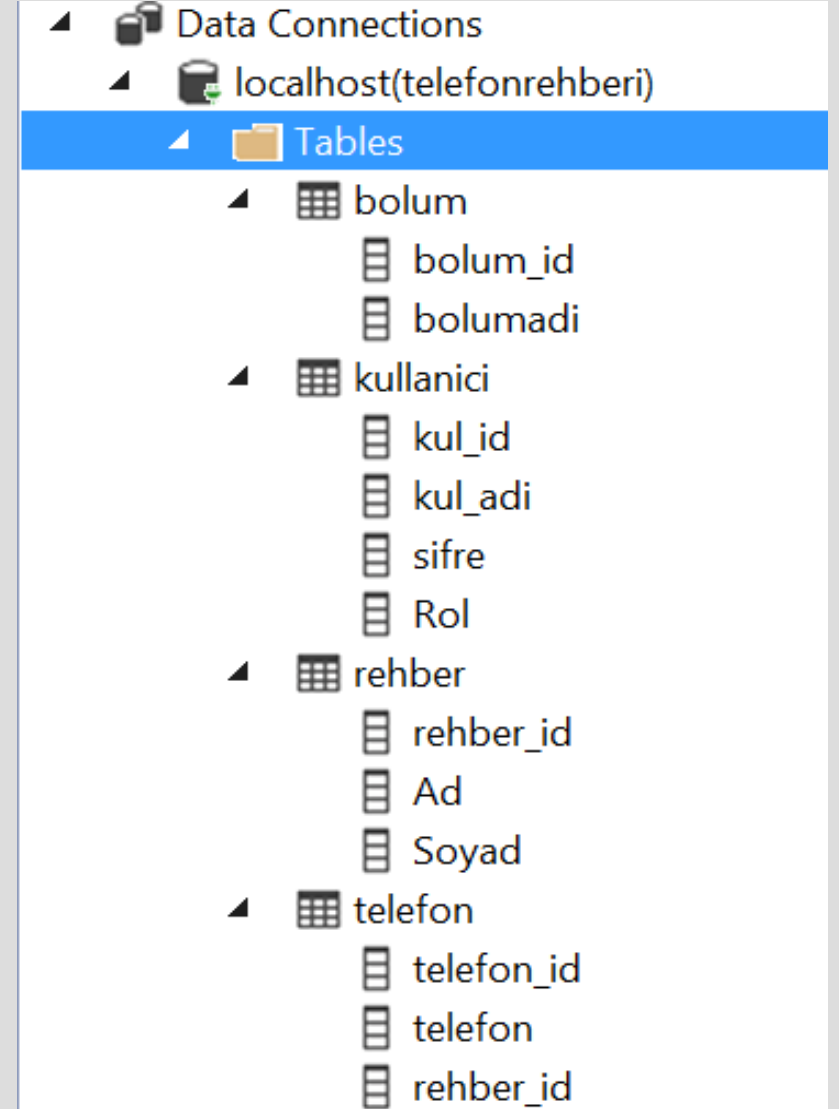
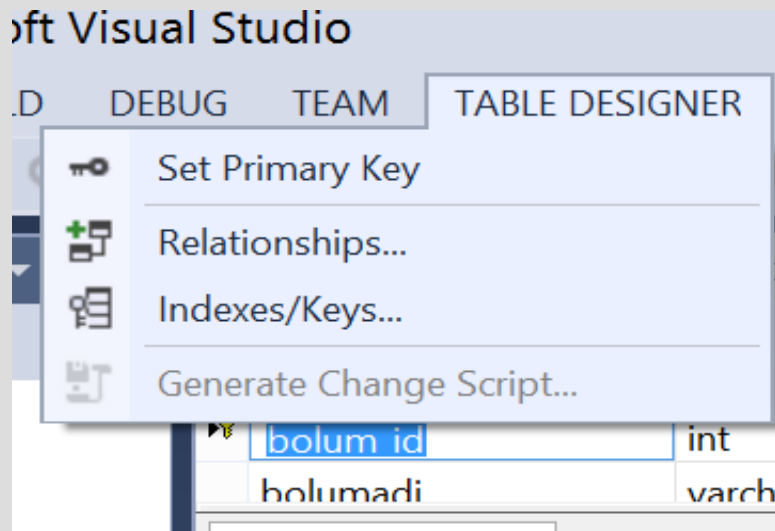
Verileri Görüntüleme



- Query bölümünden veriler sorgulanabildiği gibi “Retrieve Data” bölümünden de veriler görüntülenebilir.

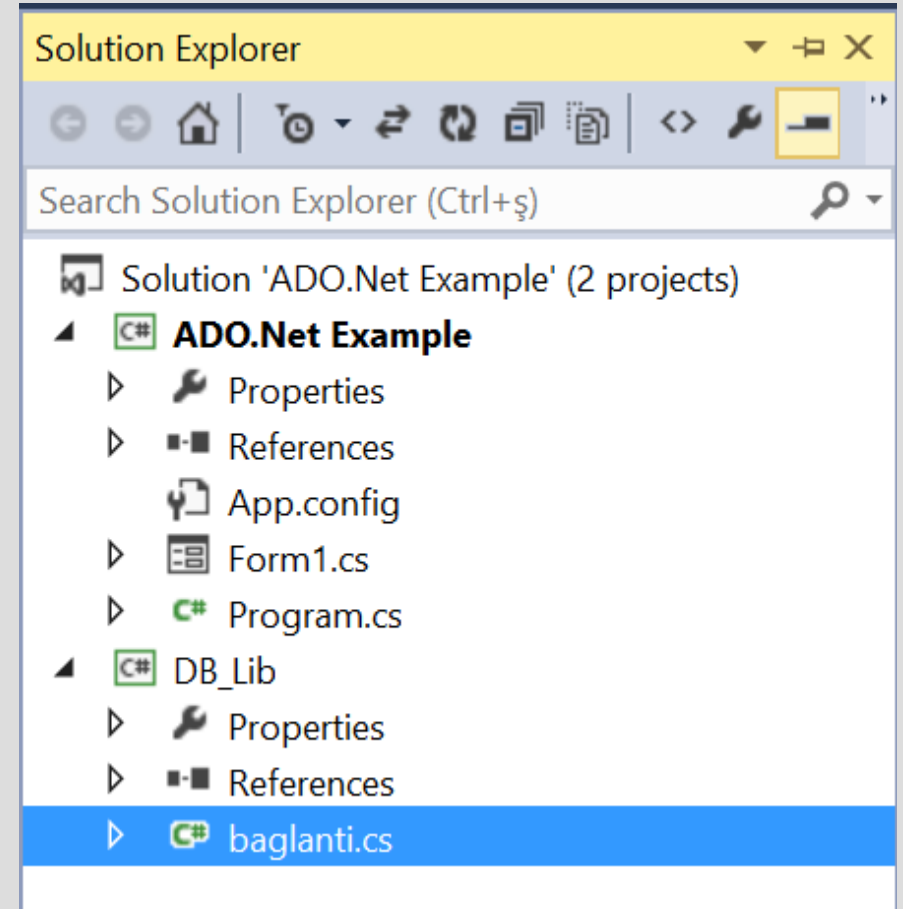
Tüm Veritabanlarını Oluşturma

- Yarıdaki tabloları ekleyelim.
- Primary Key seçerken “Table Designer” kısmını kullanın.
- İkincil anahtarların birincil anahtarla aynı olmasını sağlayın.



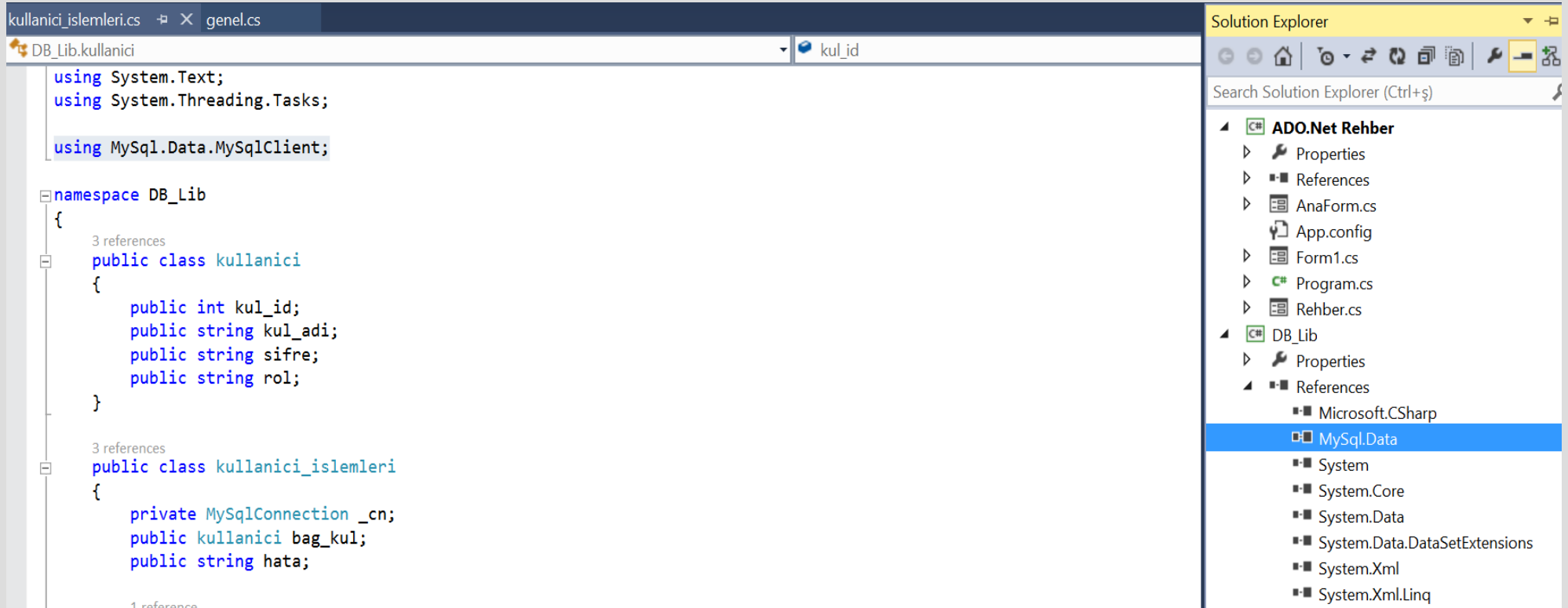
Görsel ve Veritabanı

- İki adet uygulama açalım.
 - **Windows Form Uygulaması**
 - Görsel işlemler için
 - **Class Uygulaması**
 - Veritabanı işlemleri için
- Görsel ve veritabanı bölümleri ayrı ayrı geliştirmeniz tavsiye edilir. İleride web, mobil uygulamaları geliştirmek istediğinizde veritabanı bölümü ile bu tür uygulamaları daha hızlı geliştirme şansınız olur.



Veritabanı Tarafı

- MySql.Data reference'lara ekle.
 - **MySQL Connector** kurulu olması gerekiyor.
- using'e MySql.Data.MySqlClient yazılır.



The screenshot shows a Visual Studio IDE with a C# code file named 'kullanici_islemleri.cs' open. The code is part of a namespace 'DB_Lib' and contains two classes: 'kullanici' and 'kullanici_islemleri'. The 'kullanici' class has four public properties: 'kul_id' (int), 'kul_adi' (string), 'sifre' (string), and 'rol' (string). The 'kullanici_islemleri' class has a private 'MySqlConnection _cn' and two public properties: 'bag_kul' (kullanici) and 'hata' (string). The code includes the following using statements: 'using System.Text;', 'using System.Threading.Tasks;', and 'using MySql.Data.MySqlClient;'. The Solution Explorer on the right shows the project structure, including 'ADO.Net Rehber' and 'DB_Lib'. The 'References' folder under 'DB_Lib' is expanded, showing 'Microsoft.CSharp' and 'MySql.Data' selected.

```
kullanici_islemleri.cs  X genel.cs
DB_Lib.kullanici
using System.Text;
using System.Threading.Tasks;
using MySql.Data.MySqlClient;

namespace DB_Lib
{
    3 references
    public class kullanici
    {
        public int kul_id;
        public string kul_adi;
        public string sifre;
        public string rol;
    }

    3 references
    public class kullanici_islemleri
    {
        private MySqlConnection _cn;
        public kullanici bag_kul;
        public string hata;
    }

    1 reference
}
```

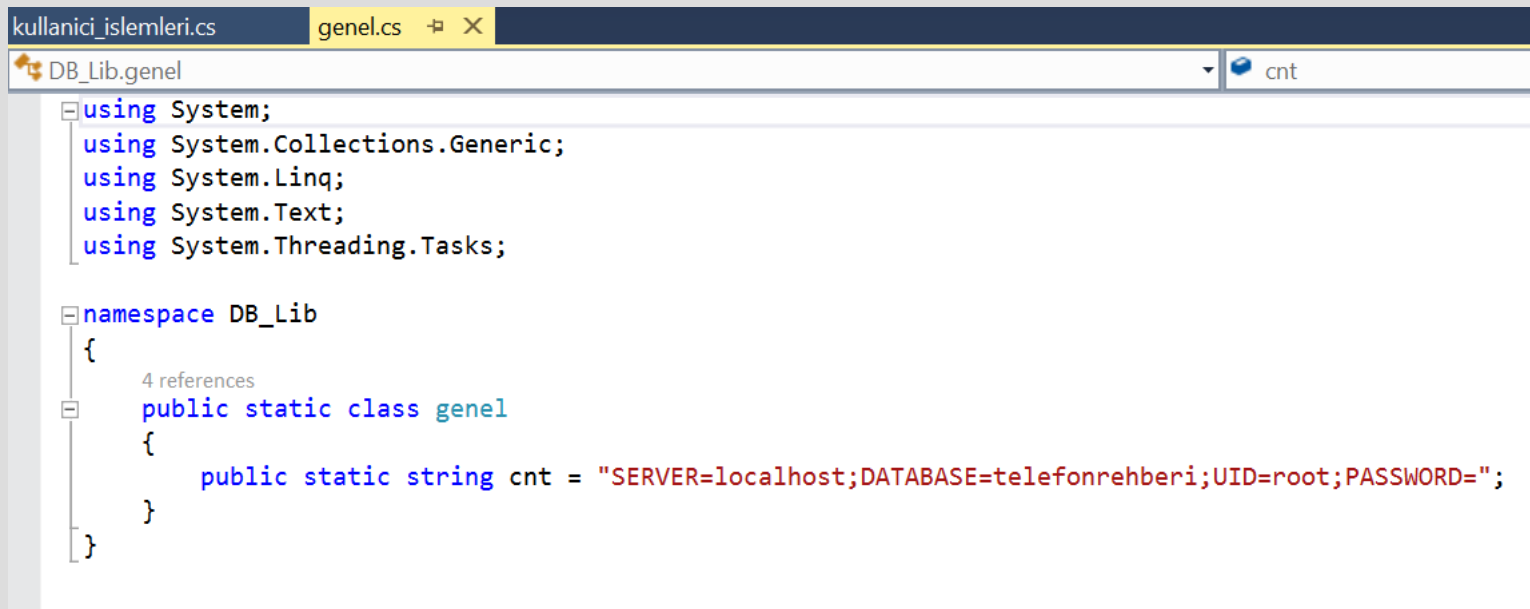
Solution Explorer

Search Solution Explorer (Ctrl+ş)

- ADO.Net Rehber
 - Properties
 - References
 - AnaForm.cs
 - App.config
 - Form1.cs
 - Program.cs
 - Rehber.cs
- DB_Lib
 - Properties
 - References
 - Microsoft.CSharp
 - MySql.Data
 - System
 - System.Core
 - System.Data
 - System.Data.DataSetExtensions
 - System.Xml
 - System.Xml.Linq

Connection String

- Connection String veritabanına bağlanmak için kullanacağımız katarıdır. Bu katarı oluştururken genel bir noktadan ulaşılmasını sağlayın. Özellikle aşağıdaki gibi statik fonksiyon kullanmanızı tavsiye ederim.



```
kullanici_islemleri.cs genel.cs X
DB_Lib.genel cnt
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace DB_Lib
{
    4 references
    public static class genel
    {
        public static string cnt = "SERVER=localhost;DATABASE=telefonrehberi;UID=root;PASSWORD=";
    }
}
```

Login İşlemi

- Login işlemi için
 - Hangi SQL ile yapılır?
 - Datareader mı?
 - DataAdaptor mü?
 - Rol bilgisi nasıl taşınacak?
 - Kullanıcı bilgileri gerekir mi?

Login İşlemi

- Login işlemi için
 - Hangi SQL ile yapılır?
 - `SELECT kul_id, rol FROM rehber WHERE kul_adi=@kul_adi and sifre=@sifre`
 - @kul_adi ve @sifre dışarıdan girdi olarak gelecek.
 - Sadece kul_id ve rol geri döndürmek yeterlidir.
 - Bu bilgileri bir sınıf içinde saklanabilir.

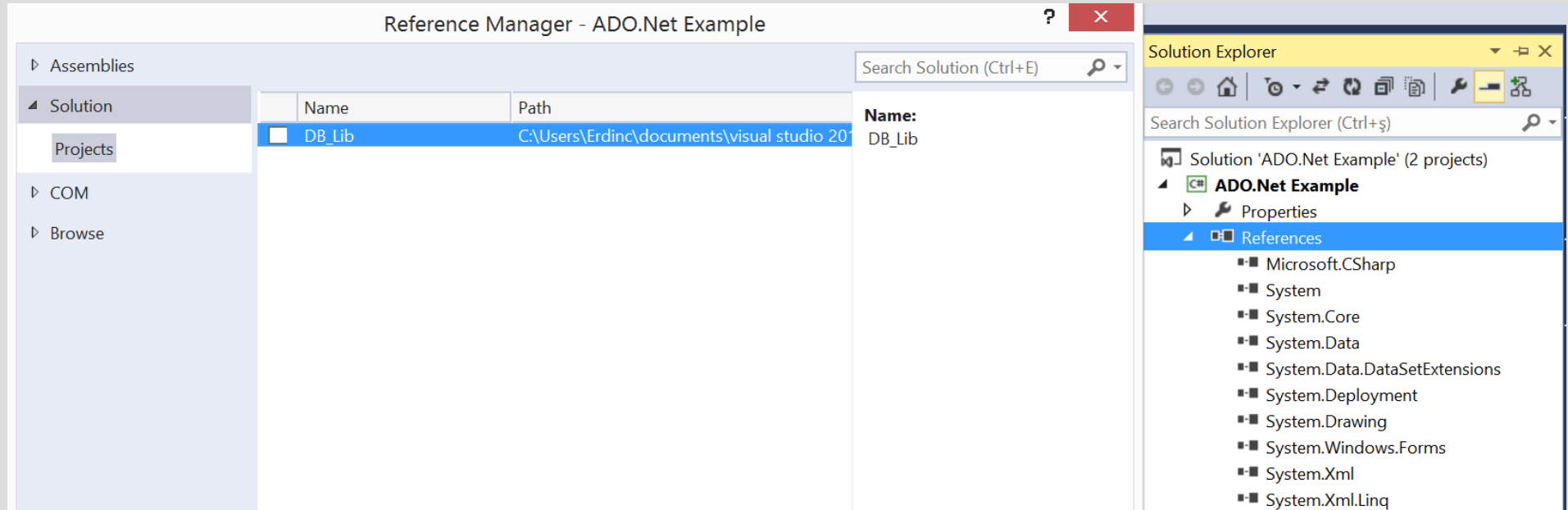
Login İşlemi

- Login işlemi için
 - Datareader mı?
 - DataAdaptor mü?
- Datareader daha hızlı olduğu için böyle bir kontrol için en uygundur.
- O table'a özel sınıf oluşturmanız önerilir.

```
2 references
public class kullanıcı
{
    public int kul_id;
    public string kul_adi;
    public string sifre;
    public string rol;
}
```

Aynı projedeki iki farklı uygulamayı haberleştirme

- Add Reference bölümünden Solution->Projects seçilip bağlanılmak istenen uygulama seçilir.
- Windows Form bölümüne MySQL.Data eklemeye gerek yoktur. Tüm işlemler DB_Lib üzerinden yönetilecektir.



MySQL tarafı

- Genel ayarlar için static yaptık.

```
using MySql.Data.MySqlClient;

namespace DB_Lib
{
    //Ayarlar için
    1 reference
    public static class Baglanti
    {
        public static string cn_str = "SERVER=localhost;DATABASE=;UID=root;PASSWORD=";
    }
}
```

Login İşlemleri

Constructor bağlantı ve diğer değişkenler için başlangıç ayarları yapılıyor.

```
using MySql.Data.MySqlClient;

namespace DB_Lib
{
    2 references
    public class kullanici
    {
        public int kul_id;
        public string kul_adi;
        public string sifre;
        public string rol;
    }

    3 references
    public class kullanici_islemleri
    {
        private MySqlConnection _cn;
        public string hata;
        private kullanici bag_kul;
        1 reference
        public kullanici_islemleri()
        {
            _cn = new MySqlConnection(Baglanti.cn_str);
            bag_kul = new kullanici();
            hata = "";
        }

        1 reference
        public bool login_islemi(string kul_adi,string sifre)...
```

Login İşlemleri

Parametre Kullanımı

bag_kul içinde kullanıcı ayarları tutuluyor. Şifre tutulmuyor.

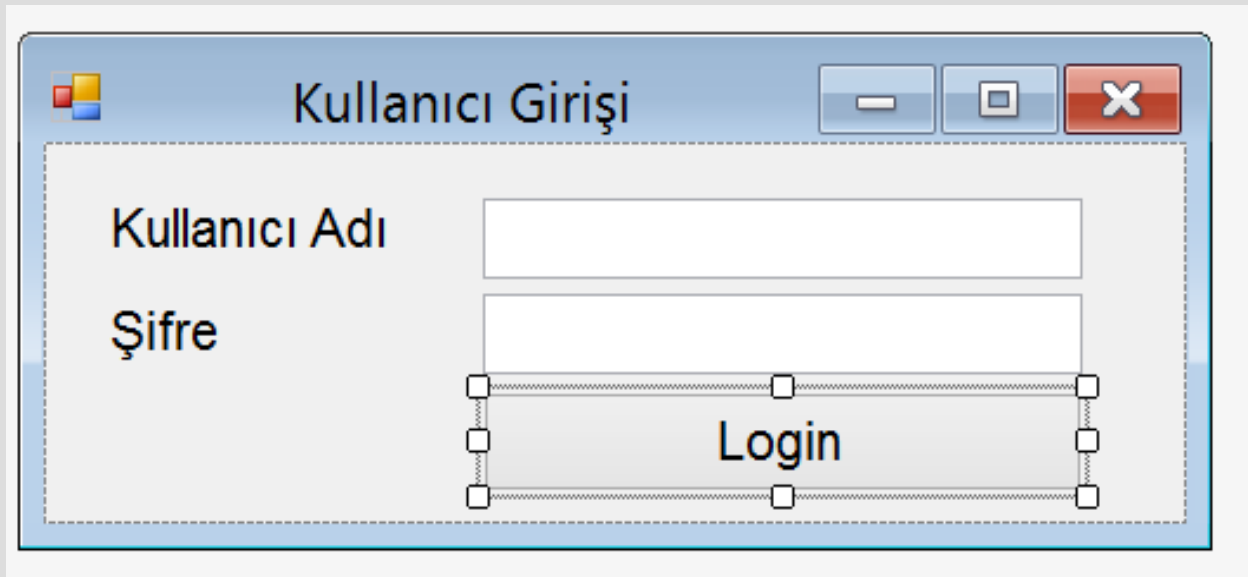
Hasrows dr içinde satır olup olmadığı kontrol ediliyor. Satır yoksa kullanıcı yoktur.

Hata değişkeninde hata tutulmaktadır.

```
1 reference
public bool login_islemi(string kul_adi,string sifre)
{
    try
    {
        _cn.Open();
        string cmd_sql = "SELECT kul_id, rol FROM kullanıcı WHERE kul_adi=@kul_adi and sifre=@sifre";
        MySqlCommand _cmd = new MySqlCommand(cmd_sql, _cn);
        _cmd.Parameters.AddWithValue("@kul_adi", kul_adi);
        _cmd.Parameters.AddWithValue("@sifre", sifre);
        MySqlDataReader _dr = _cmd.ExecuteReader();
        if (_dr.HasRows){
            _dr.Read();
            bag_kul.kul_id = _dr.GetInt32(0);
            bag_kul.rol = _dr.GetString(1);
            bag_kul.kul_adi = kul_adi;
            _dr.Close();
            _cn.Close();
            return true;
        }
        else{
            hata = "Kullanıcı hatalı";
            _cn.Close();
            return false;
        }
    }
    catch (Exception ex)
    {
        hata = ex.Message;
        return false;
    }
}
```

Windows Form Tarafı

- TextBox, Label, Button eklenmiş bir form.
- Özellikle TextBox isimlerini ve Button isimleri anlaşılacak şekilde verin.



The image shows a Windows Form titled "Kullanıcı Giriş" (User Login). The form has a standard Windows window title bar with a minimize button, a maximize button, and a close button. Inside the form, there are two text boxes: one labeled "Kullanıcı Adı" (Username) and one labeled "Şifre" (Password). Below the password box is a "Login" button. The form is currently in a design state, as indicated by the dashed border and the small square handles around the text boxes and the button.

Kontrol İşlemi

- İkinci form AnaForm'dur. Birinci form this ile visible false yapılıyor.
- AnaForm'dan bir tane yaratıp çağrılıyor.
- Bu form'un adresi direkt AnaForm içine taşınıyor. AnaForm'dan geri döndüğünde bu form kullanılacaktır.

1 reference

```
private void BTN_Login_Click(object sender, EventArgs e)
{
    kullanıcı_islemleri _ki = new kullanıcı_islemleri();
    bool sonuc = _ki.login_islemi(TXT_Kul_Adi.Text, TXT_Sifre.Text);

    if (sonuc)
    {
        this.Visible = false;
        AnaForm _af = new AnaForm();
        _af._f = this;
        _af.ShowDialog();
    }
    else
        LBL_Hata.Text = _ki.hata;
}
```

Kullanıcıların Görsel Yönetimi

- Sistemde üç kullanıcı tanımlamıştık.
 - Admin, 123, A: Tüm yetkiye sahip
 - User, 123, D: Değişirme ve Silme yetkisine sahip. Ama silmesi kalıcı değil.
 - Demo, 123, G: Sadece görünüm modu açık.
- Tüm kullanıcılar için ayrı tasarımlar yapmak zaman alıcı ve kodun hep tekrarladığı yazılımlardır. Bunun yerine tek ekran tasarlanıp ekrandaki görünümler bağlanan kullanıcının rolüne göre düzenlenebilir.

Kullanıcıların Görsel Yönetimi

- Admin, Değişiklik Modu, Görünüm Modu için sırasıyla üç ekran.

AnaForm

Sorgu

A

Ekle Değiştir Sil

Ad	Soyad	Bölüm Adı	id	Durum
Metin 2	Tekin	Bilgisayar	1	<input type="checkbox"/>
Feyyaz	Uçar	Bilgisayar	2	<input type="checkbox"/>
Ali	Gültekin	Muhasebe	3	<input type="checkbox"/>
Test	1	Bilgisayar	4	<input checked="" type="checkbox"/>
demo	test trigger	Muhasebe	5	<input type="checkbox"/>

Telefon

04535
5456

AnaForm

Sorgu

D

Değiştir Sil

Ad	Soyad	Bölüm Adı	id
Metin 2	Tekin	Bilgisayar	1
Feyyaz	Uçar	Bilgisayar	2
Ali	Gültekin	Muhasebe	3
demo	test trigger	Muhasebe	5

Telefon

04535
5456

AnaForm

Sorgu

G

Ad	Soyad	Bölüm Adı	id
Metin 2	Tekin	Bilgisayar	1
Feyyaz	Uçar	Bilgisayar	2
Ali	Gültekin	Muhasebe	3
demo	test trigger	Muhasebe	5

Telefon

04535
5456

Rehber DataGridView Görüntülenmesi

- DataGridView DataSource Özelliği üzerinden bir datatable yüklenebilir. DataTable, DataAdapter ve DataSet üzerinden üretilebilmektedir. DataBase bölümünde, bu sebepten dolayı aşağıdaki kod kullanılabilir.

```
1 reference
public DataTable Get_Rehber(string rol, string aranan)
{
    try
    {
        string sql = "select r.rehber_id, r.ad, r.soyad, b.bolum_adi, r.silindimi from rehber r INNER JOIN bolum b ON b.bolum_id=r.bolum_id ";

        if (rol == "D" || rol == "G")
        {
            sql += " WHERE (silindimi=false)";
            if (aranan != "")
                sql += " AND (r.ad like '%" + aranan + "%' OR r.soyad like '%" + aranan + "%' OR b.bolum_adi like '%" + aranan + "%')";
        }
        else
        {
            if (aranan != "")
                sql += " WHERE (r.ad like '%" + aranan + "%' OR r.soyad like '%" + aranan + "%' OR b.bolum_adi like '%" + aranan + "%')";
        }

        MySqlDataAdapter _da = new MySqlDataAdapter(sql, _cn);
        DataSet _ds = new DataSet();
        _da.Fill(_ds);

        return _ds.Tables[0];
    }
    catch
    {
        return null;
    }
}
```

Doğru SQL ile çok işlem yaptırma

```
string sql = "select r.rehber_id, r.ad, r.soyad, b.bolum_adi, r.silindimi from rehber r INNER JOIN bolum b ON b.bolum_id=r.bolum_id ";  
  
if (rol == "D" || rol == "G")  
{  
    sql += " WHERE (silindimi=false)";  
    if (aranan != "")  
        sql += " AND (r.ad like '%" + aranan + "%' OR r.soyad like '%" + aranan + "%' OR b.bolum_adi like '%" + aranan + "%')";  
}  
else  
{  
    if (aranan != "")  
        sql += " WHERE (r.ad like '%" + aranan + "%' OR r.soyad like '%" + aranan + "%' OR b.bolum_adi like '%" + aranan + "%')";  
}
```

- Bir önceki kodta yazılan SQL önemlidir. Kullanıcı tipine göre SQL hazırlanır. Ayrıca aranan bir string ise SQL eklenti yapılır.
- Silindimi alanı verinin D tipli kullanıcı tarafından silinip silinmediği tutar. (Geçici silme ve kalıcı silme... ileriki bölümde)
- Where bölümü şekillenmesi dikkat edilmelidir. İki Where olmayacağından AND ile bağlama yapılır. Eğer hiç Where yoksa Where ile başlanır.
- Yukarıda tüm alanlarda like sorgusu ile arama yapılmaktadır. Bu durumda büyük verilerde yavaş bir sorgu olacağını hesaba katın. Tek alana uygun olarak sorgu düzenlenebilir. Bu sorgu süresini hızlandırır. Ayrıca indeksleme yapılarak sorgu daha da hızlandırılabilir.

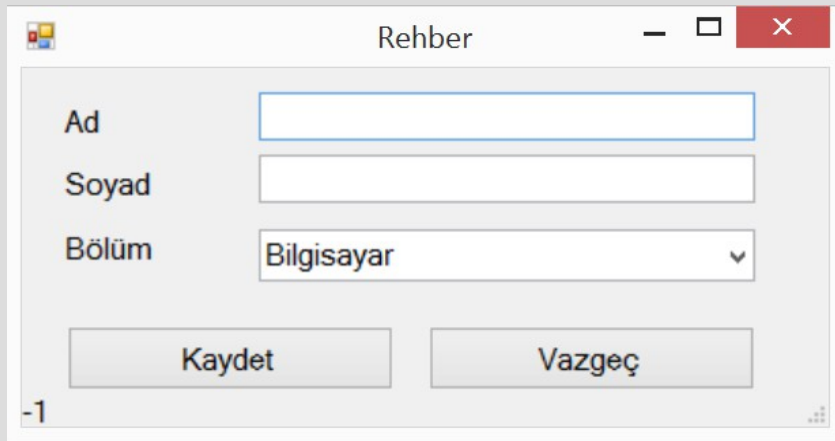
Kullanıcıların Görsel Yönetimi

- Farklı role sahip kullanıcıların ekran görünümleri için kod.
- Görüldüğü gibi üç form yerine tek form ile farklı üç kullanıcı basit bir şekilde yönetilebilir.

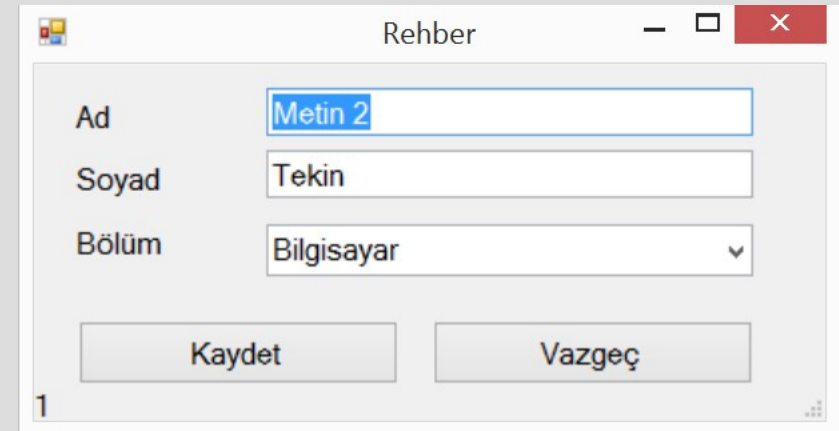
```
1 reference
private void gorunum()
{
    if (LBL_Rol.Text == "D")
    {
        BTN_Ekle_Reh.Visible = false;
        dataGridView_Rehber.Columns["silindimi"].Visible = false;
    }
    else if (LBL_Rol.Text == "G")
    {
        BTN_Deg_Reh.Visible = false;
        BTN_Ekle_Reh.Visible = false;
        BTN_Sil_Reh.Visible = false;
        dataGridView_Rehber.Columns["silindimi"].Visible = false;
    }
}
```

Rehber Kullanıcı Ekleme / Deęiřtirme

- Rehber Ekleme ve Deęiřtirme İřlemleri iin tek ekran yeterlidir.
- Anaform'dan gelen bilgiye gre ekleme veya deęiřtirme iřlemi ynetilebilir. Kodta -1 geldięinde ekleme dięer durumlarda ise deęiřtirme iřlemi yapılmaktadır.



The screenshot shows a window titled 'Rehber' with a standard Windows title bar. The form contains three input fields: 'Ad' (empty), 'Soyad' (empty), and 'Bölüm' (a dropdown menu with 'Bilgisayar' selected). Below the fields are two buttons: 'Kaydet' and 'Vazge'. In the bottom-left corner, the code '-1' is displayed.



The screenshot shows the same 'Rehber' window. The 'Ad' field now contains the text 'Metin 2', 'Soyad' contains 'Tekin', and 'Bölüm' remains 'Bilgisayar'. The 'Kaydet' and 'Vazge' buttons are still present. In the bottom-left corner, the code '1' is displayed.

Yeni Kayıt Ekleme İçin Form Çağırılması

1 reference

```
private void BTN_Ekle_Reh_Click(object sender, EventArgs e)
{
    Rehber _r_form = new Rehber();

    bolum _b = new bolum();
    _r_form.CB_Bolum.DataSource = _b.Get_bolum();
    _r_form.CB_Bolum.DisplayMember = "bolum_adi";
    _r_form.CB_Bolum.ValueMember = "bolum_id";

    //refresh olsun diye anaform ile bağlantı
    _r_form._af = this;

    //insert mi? (-1) update mi? (>0)
    _r_form.LBL_Rehber_id.Text = "-1";
    _r_form.ShowDialog();
}
```

Değiştirme için Form Çağırılması

1 reference

```
private void BTN_Deg_Reh_Click(object sender, EventArgs e)
{
    Rehber _r_form = new Rehber();
    _r_form.TXT_Ad.Text = dataGridView_Rehber.CurrentRow.Cells["ad"].Value.ToString();
    _r_form.TXT_Soyad.Text = dataGridView_Rehber.CurrentRow.Cells["soyad"].Value.ToString();
    _r_form.LBL_Rehber_id.Text = dataGridView_Rehber.CurrentRow.Cells["rehber_id"].Value.ToString();

    bolum _b = new bolum();
    _r_form.CB_Bolum.DataSource = _b.Get_bolum();
    _r_form.CB_Bolum.DisplayMember = "bolum_adi";
    _r_form.CB_Bolum.ValueMember = "bolum_id";
    _r_form.CB_Bolum.Text = dataGridView_Rehber.CurrentRow.Cells["bolum_adi"].Value.ToString();

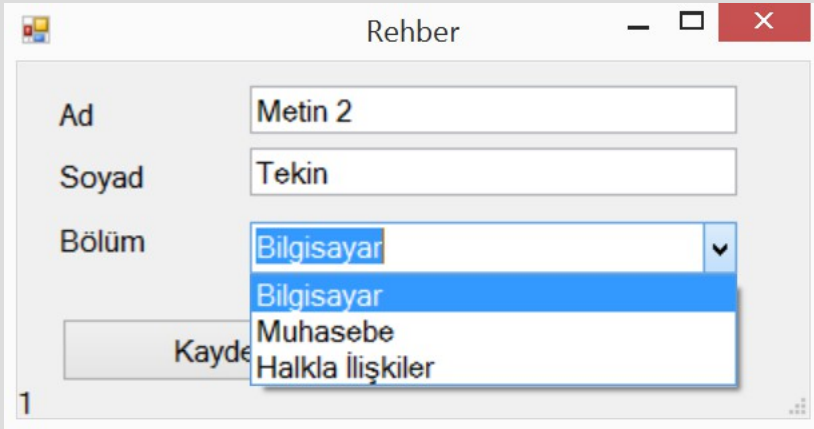
    //refresh olsun diye anaform ile bağlantı
    _r_form._af = this;

    _r_form.ShowDialog();
}
```

DataGridView'deki verilere ulaşma

- `CurrentRow` ile satır Seçilir.
- `Cells["x"]` ile hücre seçilir ve veriye ulaşılır. Veri object modunda olduğu için `toString()` fonksiyonu kullanılmıştır.

Combobox Kullanma



```
bolum_b = new bolum();
_r_form.CB_Bolum.DataSource = _b.Get_bolum();
_r_form.CB_Bolum.DisplayMember = "bolum_adi";
_r_form.CB_Bolum.ValueMember = "bolum_id";
_r_form.CB_Bolum.Text = dataGridView_Rehber.CurrentRow.Cells["bolum_adi"].Value.ToString();
```

- Bir form'daki farklı özelliklere ulaşmak için Modifiers bölümlerinin public olması gerektiğini unutmayın.
- Combobox DataSource Özelliği vardır. DataGridView çok benzer şekildedir. DataTable bekler.
- DisplayMember gösterilen değeri ve ValueMember veritabanına yazılacak değeri belirler.
- SQL kısmında iki kısımda olmalıdır. "bolum_adi" ve "bolum_id".

Bölümleri datatable aktaran sınıf

bolum.cs

```
namespace DB_Lib
{
    5 references
    public class bolum
    {
        private MySqlConnection _cn;

        2 references
        public bolum()
        {
            _cn = new MySqlConnection(genel.cnt);
        }

        2 references
        public DataTable Get_bolum()
        {
            try
            {
                string sql = "select bolum_id, bolum_adi from bolum";
                MySqlDataAdapter _da = new MySqlDataAdapter(sql, _cn);
                DataSet _ds = new DataSet();
                _da.Fill(_ds);

                return _ds.Tables[0];
            }
            catch
            {
                return null;
            }
        }
    }
}
```

Rehber Kaydetme İşlemi

- Eğer -1 ise insert yoksa update işlemi çağrılmaktadır. İşlem doğru bir şekilde biterse form close olmaktadır. (This.Close())
- Bu arada af.Refresh_Rehber ile alt form görüntüsü yenilenmektedir. Bu fonksiyonun public olmasına dikkat!!!

1 reference

```
private void BTN_Kaydet_Click(object sender, EventArgs e)
{
    rehber _r = new rehber();
    int s = 0;

    if (LBL_Rehber_id.Text == "-1")
    {
        s = _r.insert_rehber(TXT_Ad.Text, TXT_Soyad.Text, Convert.ToInt32(CB_Bolum.SelectedValue));
    }
    else
    {
        s = _r.update_rehber(TXT_Ad.Text, TXT_Soyad.Text, Convert.ToInt32(CB_Bolum.SelectedValue), Convert.ToInt32(LBL_Rehber_id.Text));
    }

    if (s > 0)
    {
        _af.Refresh_Rehber();
        this.Close();
    }
}
```

Ekleme İşlemi Veritabanı Kod Tarafı

- Ekleme işleminde dışarıdan 3 adet veri gelmektedir. Çıktı update sonucudur. (1 çıkması beklenir. Eğer olağanüstü bir durum olmazsa. Mesela, SQL hatalı olabilir.)
- Parametre kullanımı önemlidir. (Geçen haftaya bakın ve googling)

```
1 reference
public int insert_rehber(string ad, string soyad, int bolum_id)
{
    try
    {
        _cn.Open();
        string sql = "INSERT INTO Rehber(ad,soyad,bolum_id) VALUES(@ad,@soyad,@bolum_id)";
        MySqlCommand _cmd = new MySqlCommand(sql, _cn);
        _cmd.Parameters.AddWithValue("@ad", ad);
        _cmd.Parameters.AddWithValue("@soyad", soyad);
        _cmd.Parameters.AddWithValue("@bolum_id", bolum_id);

        int s = _cmd.ExecuteNonQuery();
        return s;
    }
    catch (Exception ex)
    {
        _cn.Close();
        return 0;
    }
}
```

Güncelleme İşlemi Veritabanı Kod Tarafı

- Güncelle işleminde 4 adet veri gelmektedir. Son veri hangi kayıt'ın güncellemesi hakkındadır. rehber_id primary key olduğu için bu değer üzerinden güncelleme yapılır.

```
1 reference
public int update_rehber(string ad, string soyad, int bolum_id, int rehber_id)
{
    try
    {
        _cn.Open();
        string sql = "UPDATE Rehber SET ad=@ad, soyad=@soyad, bolum_id=@bolum_id WHERE rehber_id=@rehber_id";
        MySqlCommand _cmd = new MySqlCommand(sql, _cn);
        _cmd.Parameters.AddWithValue("@ad", ad);
        _cmd.Parameters.AddWithValue("@soyad", soyad);
        _cmd.Parameters.AddWithValue("@bolum_id", bolum_id);
        _cmd.Parameters.AddWithValue("@rehber_id", rehber_id);

        int s = _cmd.ExecuteNonQuery();
        return s;
    }
    catch (Exception ex)
    {
        _cn.Close();
        return 0;
    }
}
```

Rehber Silme İşlemi

- Silme işlemi iki türüdür. Eğer kullanıcı admin ise kalıcı silme yapar. Eğer D tipi bir role sahipse geçici silme yapar. Geçici silinen kayıtları admin görebilir. Ama diğer kullanıcılar göremez.
- Aşağıda CurrentRow üzerinde rehber_id alınmaktadır. Rehber_id'ye göre silme işlemi yapılır.

1 reference

```
private void BTN_Sil_Reh_Click(object sender, EventArgs e)
{
    rehber _r = new rehber();
    int rehber_id = Convert.ToInt32(dataGridView_Rehber.CurrentRow.Cells["rehber_id"].Value.ToString());

    if (LBL_Rol.Text == "A")
        _r.sil_kalici_rehber(rehber_id);
    else if (LBL_Rol.Text == "D")
        _r.sil_gecici_rehber(rehber_id);

    Refresh_Rehber();
}
```

Rehber Silme İşlemi

- Geçici Silme: Geçici silme aslında bir update işlemidir. Update ile silindimi=true değeri alır. Ve admin hariç diğer kullanıcılar silindimi=false değerlerini görür. Çöp kutusu mantalitesine benzetebilirsiniz.
- Admin bölümüne tam tersi bir kod yazılarak kayıt kurtarması sağlanabilir. (Size bıraktım :))

```
1 reference
public int sil_gecici_rehber(int rehber_id)
{
    try
    {
        _cn.Open();
        string sql = "UPDATE Rehber SET silindimi=true WHERE rehber_id=@rehber_id";
        MySqlCommand _cmd = new MySqlCommand(sql, _cn);
        _cmd.Parameters.AddWithValue("@rehber_id", rehber_id);

        int s = _cmd.ExecuteNonQuery();
        return s;
    }
    catch (Exception ex)
    {
        _cn.Close();
        return 0;
    }
}
```

Rehber Silme İşlemi

- Kalıcı silmeyi sadece admin'e yapabilme yetkisi verdik.
- SQL Delete ile silme yapılır ve kayda artık ulaşamaz. Çöp Kutusu temizlendi :)

```
1 reference
public int sil_kalici_rehber(int rehber_id)
{
    try
    {
        _cn.Open();
        string sql = "DELETE FROM Rehber WHERE rehber_id=@rehber_id";
        MySqlCommand _cmd = new MySqlCommand(sql, _cn);
        _cmd.Parameters.AddWithValue("@rehber_id", rehber_id);

        int s = _cmd.ExecuteNonQuery();
        return s;
    }
    catch (Exception ex)
    {
        _cn.Close();
        return 0;
    }
}
```


Telefon Bölümü

- Kullanıcıya ait telefonların görüntülenmesi basit bir bire-çok ilişkidir. İkinci grid'te telefonları göstermek için birinci grid'ten hangi kişinin olduğu bulunması gerekir. Bunun içinde yine "rehber_id" kullanılır.
- Buradaki örnekte rowenter olayı (event) kullanılmıştır. Her satıra giriş yapıldığında bu yordam çalışır.
- Aşağıda, o satır uygun cell seçilir ve o değer okuyor. Yine Datatable DataSource içine gönderiliyor.

1 reference

```
private void dataGridView_Rehber_RowEnter(object sender, DataGridViewCellEventArgs e)
{
    string rehber_id = dataGridView_Rehber.Rows[e.RowIndex].Cells["rehber_id"].Value.ToString();
    telefon _t = new telefon();
    dataGridView_Telefon.DataSource = _t.Get_Telefon(rehber_id);
}
```

Telefon Veritabanı Tarafı

```
using System.Data;
using MySql.Data.MySqlClient;

namespace DB_Lib
{
    3 references
    public class telefon
    {
        private MySqlConnection _cn;

        1 reference
        public telefon()
        {
            _cn = new MySqlConnection(genel.cnt);
        }

        1 reference
        public DataTable Get_Telefon(string rehber_id)
        {
            try
            {
                string sql = "select telefon from telefon where rehber_id="+rehber_id;
                MySqlDataAdapter _da = new MySqlDataAdapter(sql, _cn);
                DataSet _ds = new DataSet();
                _da.Fill(_ds);

                return _ds.Tables[0];
            }
            catch
            {
                return null;
            }
        }
    }
}
```

Kalan Kısımlar

- Telefon bölümü için ekleme, silme, düzeltme gibi işlemler size bırakılmıştır. Rehberin benzeri:)
- En sade form özellikleri kullanıldı. İnternette daha güzel butonlar, resimli butonlar vs. araştırılıp daha güzel bir görüntü yapılabilir.
- OK tuşları ile TEXTBOX kutuları arasında dolaşım eklenebilir. Bulunan TEXTBOX farklı bir renge boyanabilir.
- DataGridView en temel özellikleri yapıldı. ReadOnly=true yapıp, insert, update, delete işlemleri kapatıldı.
 - DataGridView çok geniş özellikleri sahiptir. Bu özellikler probleme göre araştırılabilir.
 - Mesela DataGridView geçici silinen kayıtlar admin'in karşısına farklı bir renkte düşebilir. Bunun için DataGridView event (olay) ları incelenmelidir.

Sonuç

- Bir windows form uygulaması gördük. Veritabanı kısmını özellikle ayrı hazırladık. Bundan sonraki uygulamalarınızda web, mobil aynı veritabanlarını kullanma şansına sahipsinizdir.
- Uygulama yaparken kendi dünyanızı yarattığınızı unutmayın.
 - Örneğin A kaydı silinmiş!!! Kim sildi!!! Ne zaman Sildi!!!
 - Veya kayıt üzerine kimler hangi işlemleri yaptı.
 - Bu gibi durumları yönetmek sizin kurduğunuz dünyaya bağlıdır. İyi veritabanı tasarımları ile çok iyi sistemler kurabilirsiniz.
 - Hatta kullanıcıların programı ne kadar süre kullandıklarını, hangi sayfada ne kadar süre geçirdiklerini kısacası kullanıcı performansını elde etmek size kalmış.
 - Burası sizin dünyanız!!!