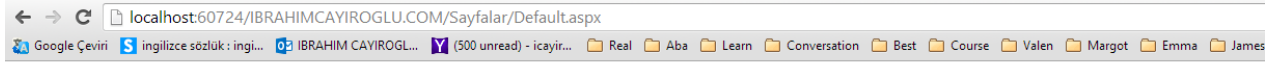


İNTERNET TABANLI PROGRAMLAMA- 13.ders

A-GRIDVIEW, DETAILSVIEW, ACCESSDATASOURCE NESNELERİ

İLE BİLGİ GÖRÜNTÜLEME



ENGLISH EXERCISE

ID	SoruMetni	IngilizceCumle	A_Sikki	B_Sikki	C_Sikki	E_Sikki	D_Sikki
1	Doğru şıkkı işaretleyiniz	Where are you from?	Ankara	cankiri	İstanbul	Afyonkarahisar	Karabük
2	Doğru şıkkı işaretleyin	What is your profession?	Engineer	Academic stuff	Worker	Student	Boss

ID	1
SoruMetni	Doğru şıkkı işaretleyiniz
IngilizceCumle	Where are you from?
A_Sikki	Ankara
B_Sikki	cankiri
C_Sikki	İstanbul
D_Sikki	Karabük
E_Sikki	Afyonkarahisar
DogrucEvap	B
Resim	
IngilizceCumleAnlami	Nerelisiniz?
A_SikkiAnlami	
B_SikkiAnlami	
C_SikkiAnlami	
D_SikkiAnlami	
E_SikkiAnlami	
Aciklama	
Düzenle Sil Yeni	

Asp.Net kodları

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs"
Inherits="Sayfalar_Default" %>
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head runat="server">
```

```
<title>Untitled Page</title>
```

```
</head>
```

```
<body>
```

```
<form id="Form1" runat="server">
```

```
<h3>
```

```
ENGLISH EXERCISE</h3>
```

```
<table cellpadding="10">
```

```
<tr>
```

```
<td>
```

```
<!-- DETAILS VIEW KONTROL NESNESİ KULLANIMI-->
```

```
<asp:GridView ID="gvIngilizce"
```

```
DataSourceID="dsSorulariGetir"
```

```
AutoGenerateColumns="False"
```

```
runat="server"
```

```
DataKeyNames="ID" >
```

```
<HeaderStyle BackColor="Blue" ForeColor="White" />
```

```
<Columns>
```

```
<asp:BoundField DataField="ID"
```

```
HeaderText="ID" SortExpression="ID" />
```

```

<asp:BoundField DataField="SoruMetni"
  HeaderText="SoruMetni" SortExpression="SoruMetni" />

<asp:BoundField DataField="IngilizceCumle"
  HeaderText="IngilizceCumle" SortExpression="IngilizceCumle" />

  <asp:BoundField DataField="A_Sikki" HeaderText="A_Sikki"
    SortExpression="A_Sikki" />
  <asp:BoundField DataField="B_Sikki" HeaderText="B_Sikki"
    SortExpression="B_Sikki" />
  <asp:BoundField DataField="C_Sikki" HeaderText="C_Sikki"
    SortExpression="C_Sikki" />
  <asp:BoundField DataField="E_Sikki" HeaderText="E_Sikki"
    SortExpression="E_Sikki" />
  <asp:BoundField DataField="D_Sikki" HeaderText="D_Sikki"
    SortExpression="D_Sikki" />

  <asp:CommandField ShowSelectButton="True" />

</Columns>
</asp:GridView>

</td>

<td valign="top">
  <%-- DETAYLARI GÖRÜNTÜLEME--%>

  <asp:DetailsView ID="dvDetaylar"
    DataSourceID="dsDetaylariGetir"

    AutoGenerateRows="false"
    AutoGenerateInsertButton="true"
    AutoGenerateEditButton="true"
    AutoGenerateDeleteButton="true"

    EmptyDataText="Kayıt bulunamadı."
    DataKeyNames="ID"
    GridLines="Both"

    OnItemInserted="CustomerDetail_ItemInserted"
    OnItemInserting="CustomerDetail_ItemInserting"
    OnItemUpdated="CustomerDetail_ItemUpdated"
    OnItemUpdating="CustomerDetail_ItemUpdating"
    OnItemDeleted="CustomerDetail_ItemDeleted"
    runat="server">

    <HeaderStyle BackColor="Navy" ForeColor="White" />
    <RowStyle BackColor="White" />
    <AlternatingRowStyle BackColor="LightGray" />
    <EditRowStyle BackColor="LightCyan" />

    <Fields>
      <asp:BoundField DataField="ID" HeaderText="ID" ReadOnly="True" />
      <asp:BoundField DataField="SoruMetni" HeaderText="SoruMetni" />
      <asp:BoundField DataField="IngilizceCumle"
HeaderText="IngilizceCumle" />
      <asp:BoundField DataField="A_Sikki" HeaderText="A_Sikki" />
      <asp:BoundField DataField="B_Sikki" HeaderText="B_Sikki" />
      <asp:BoundField DataField="C_Sikki" HeaderText="C_Sikki" />

```

```

        <asp:BoundField DataField="D_Sikki" HeaderText="D_Sikki" />
        <asp:BoundField DataField="E_Sikki" HeaderText="E_Sikki" />
        <asp:BoundField DataField="DogruCevap" HeaderText="DogruCevap" />
        <asp:BoundField DataField="Resim" HeaderText="Resim" />
        <asp:BoundField DataField="IngilizceCumleAnlami"
HeaderText="IngilizceCumleAnlami" />
        <asp:BoundField DataField="A_SikkiAnlami" HeaderText="A_SikkiAnlami"
/>
        <asp:BoundField DataField="B_SikkiAnlami" HeaderText="B_SikkiAnlami"
/>
        <asp:BoundField DataField="C_SikkiAnlami" HeaderText="C_SikkiAnlami"
/>
        <asp:BoundField DataField="D_SikkiAnlami" HeaderText="D_SikkiAnlami"
/>
        <asp:BoundField DataField="E_SikkiAnlami" HeaderText="E_SikkiAnlami"
/>
        <asp:BoundField DataField="Aciklama" HeaderText="Aciklama" />
    </Fields>

</asp:DetailsView>

</td>
</tr>
</table>

<!-- ANA TABLOYA BİLGİLERİ GETİRME -->
<asp:AccessDataSource ID="dsSorulariGetir" runat="server"

    SelectCommand="SELECT [ID], [SoruMetni], [IngilizceCumle],
[A_Sikki], [B_Sikki], [C_Sikki], [E_Sikki], [D_Sikki]
FROM [Ingilizce]"

    DataFile="~/App_Data/VERITABANI.mdb">
</asp:AccessDataSource>

<!-- DETAY TABLOSUNA BİLGİLERİ GETİRME -->

<asp:AccessDataSource ID="dsDetaylariGetir" runat="server"

    DataFile="~/App_Data/VERITABANI.mdb"

    SelectCommand="SELECT * FROM [Ingilizce] WHERE ([ID] = @ID)"

    DeleteCommand="DELETE FROM [Ingilizce] WHERE [ID] = @ID"

    InsertCommand="INSERT INTO [Ingilizce]
([Kategori],
[SoruMetni],
[IngilizceCumle],
[A_Sikki],
[B_Sikki],
[C_Sikki],
[D_Sikki],
[E_Sikki],
[DogruCevap],
[Resim],
[IngilizceCumleAnlami],
[A_SikkiAnlami],
[B_SikkiAnlami],
[C_SikkiAnlami],
[D_SikkiAnlami],
[E_SikkiAnlami],

```

```

[Aciklama])

VALUES

([@Kategori],
[@SoruMetni],
[@IngilizceCumle],
[@A_Sikki],
[@B_Sikki],
[@C_Sikki],
[@D_Sikki],
[@E_Sikki],
[@DogruCevap],
[@Resim],
[@IngilizceCumleAnlami],
[@A_SikkiAnlami],
[@B_SikkiAnlami],
[@C_SikkiAnlami],
[@D_SikkiAnlami],
[@E_SikkiAnlami],
[@Aciklama]) "

UpdateCommand="UPDATE [Ingilizce] SET

[Kategori]=@Kategori,
[SoruMetni]=@SoruMetni,
[IngilizceCumle]=@IngilizceCumle,
[A_Sikki]=@A_Sikki,
[B_Sikki]=@B_Sikki,
[C_Sikki]=@C_Sikki,
[D_Sikki]=@D_Sikki,
[E_Sikki]=@E_Sikki,
[DogruCevap]=@DogruCevap,
[Resim]=@Resim,
[IngilizceCumleAnlami]=@IngilizceCumleAnlami,
[A_SikkiAnlami]=@A_SikkiAnlami,
[B_SikkiAnlami]=@B_SikkiAnlami,
[C_SikkiAnlami]=@C_SikkiAnlami,
[D_SikkiAnlami]=@D_SikkiAnlami,
[E_SikkiAnlami]=@E_SikkiAnlami,
[Aciklama]=@Aciklama

WHERE [ID] = @ID">
<SelectParameters>
  <asp:ControlParameter ControlID="gvIngilizce"
    Name="ID" PropertyName="SelectedValue"
    Type="Int32" />
</SelectParameters>

<DeleteParameters>
  <asp:Parameter Name="ID" Type="Int32" />
</DeleteParameters>

<UpdateParameters>
  <asp:Parameter Name="ID" Type="Int32" />
  <asp:Parameter Name="Kategori" Type="String" />
  <asp:Parameter Name="SoruMetni" Type="String" />
  <asp:Parameter Name="IngilizceCumle" Type="String" />
  <asp:Parameter Name="A_Sikki" Type="String" />
  <asp:Parameter Name="B_Sikki" Type="String" />
  <asp:Parameter Name="C_Sikki" Type="String" />

```

```
<asp:Parameter Name="D_Sikki" Type="String" />
<asp:Parameter Name="E_Sikki" Type="String" />
<asp:Parameter Name="DogruCevap" Type="String" />
<asp:Parameter Name="Resim" Type="String" />
<asp:Parameter Name="IngilizceCumleAnlami" Type="String" />
<asp:Parameter Name="A_SikkiAnlami" Type="String" />
<asp:Parameter Name="B_SikkiAnlami" Type="String" />
<asp:Parameter Name="C_SikkiAnlami" Type="String" />
<asp:Parameter Name="D_SikkiAnlami" Type="String" />
<asp:Parameter Name="E_SikkiAnlami" Type="String" />
<asp:Parameter Name="Aciklama" Type="String" />

</UpdateParameters>

<InsertParameters>
  <asp:Parameter Name="Kategori" Type="String" />
  <asp:Parameter Name="SoruMetni" Type="String" />
  <asp:Parameter Name="IngilizceCumle" Type="String" />
  <asp:Parameter Name="A_Sikki" Type="String" />
  <asp:Parameter Name="B_Sikki" Type="String" />
  <asp:Parameter Name="C_Sikki" Type="String" />
  <asp:Parameter Name="D_Sikki" Type="String" />
  <asp:Parameter Name="E_Sikki" Type="String" />
  <asp:Parameter Name="DogruCevap" Type="String" />
  <asp:Parameter Name="Resim" Type="String" />
  <asp:Parameter Name="IngilizceCumleAnlami" Type="String" />
  <asp:Parameter Name="A_SikkiAnlami" Type="String" />
  <asp:Parameter Name="B_SikkiAnlami" Type="String" />
  <asp:Parameter Name="C_SikkiAnlami" Type="String" />
  <asp:Parameter Name="D_SikkiAnlami" Type="String" />
  <asp:Parameter Name="E_SikkiAnlami" Type="String" />
  <asp:Parameter Name="Aciklama" Type="String" />
</InsertParameters>

</asp:AccessDataSource>
</form>
</body>
</html>
```

C# kodlari

```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;

public partial class Sayfalar_Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }
}
```

```
public void CustomerDetail_ItemInserted(object sender,
DetailsViewInsertedEventArgs e)
{
    // Refresh the GridView control after a new record is inserted
    // in the DetailsView control.
    gvIngilizce.DataBind();
}

public void CustomerDetail_ItemInserting(object sender,
DetailsViewInsertEventArgs e)
{
    // Iterate though the values entered by the user and HTML encode
    // the values. This helps prevent malicious values from being
    // stored in the data source.
    for (int i = 0; i < e.Values.Count; i++)
    {
        if (e.Values[i] != null)
        {
            e.Values[i] = Server.HtmlEncode(e.Values[i].ToString());
        }
    }
}

public void CustomerDetail_ItemUpdated(object sender,
DetailsViewUpdatedEventArgs e)
{
    // Refresh the GridView control after a new record is updated
    // in the DetailsView control.
    gvIngilizce.DataBind();
}

public void CustomerDetail_ItemUpdating(object sender,
DetailsViewUpdateEventArgs e)
{
    // Iterate though the values entered by the user and HTML encode
    // the values. This helps prevent malicious values from being
    // stored in the data source.
    for (int i = 0; i < e.NewValues.Count; i++)
    {
        if (e.NewValues[i] != null)
        {
            e.NewValues[i] = Server.HtmlEncode(e.NewValues[i].ToString());
        }
    }
}

public void CustomerDetail_ItemDeleted(object sender,
DetailsViewDeletedEventArgs e)
{
    // Refresh the GridView control after a new record is updated
    // in the DetailsView control.
    gvIngilizce.DataBind();
}
}
```

B-ASP.NET SAYFALARINDA EXCEL VERİLERİNİ GÖRÜNTÜLEME

(Öğrencimiz, Mehmet Çıkan tarafınan hazırlanmıştır)

Asp.net sayfasında dışarıdan yüklenen (upload nesnesi ile) excel bilgilerini sayfa içerisindeki gridview de görüntülemeyi bu çalışmada inceleyeceğiz.

Çalışmayı 3 başlık altında toplayacağız.

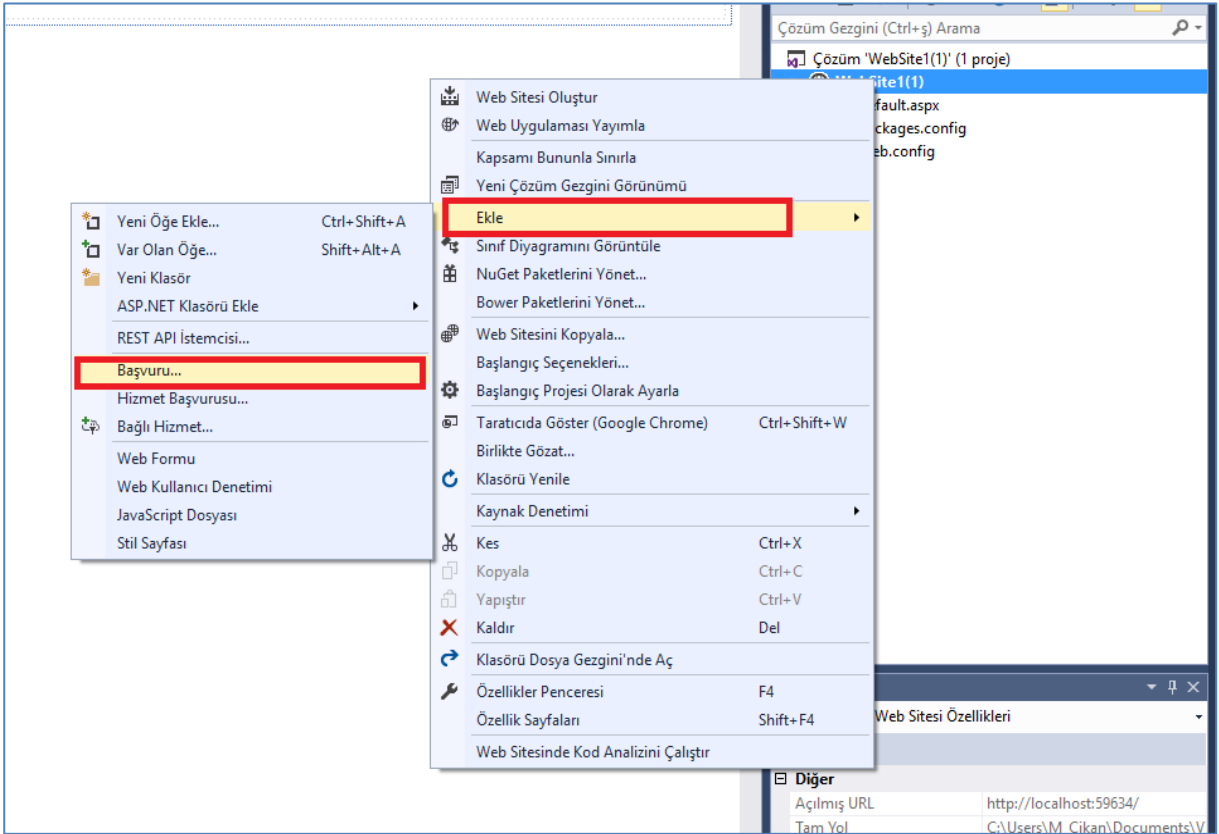
1-Asp.Net Excelden Çekilen Verinin Gridview Nesnesinde Görüntüleme

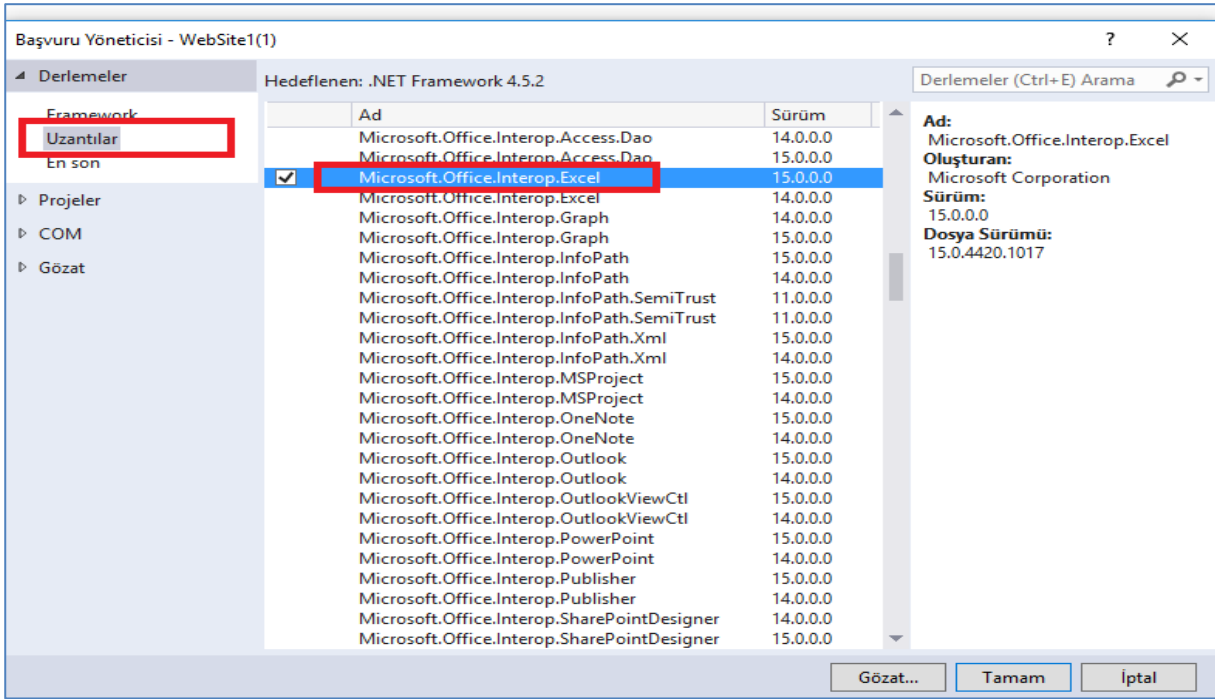
2-Excelden Çekilen Verinin Gridviewde İstenilen Başlık Altında Görüntülenmesi

3- Excelden çekilen verinin gridviewde görüntülendikten sonra veritabana kayıt edilmesi

1-Asp.Net Excelden Çekilen Verinin Gridview Nesnesinde Görüntüleme

Öncelikle Excel ile çalışabilmemiz için projemize resimlerde görüldüğü gibi gerekli referansı eklememiz gerekiyor.





Projemize eklememiz gereken toolbox nesneleri aşağıdadır.

- 1 adet Button
- 1 adet Gridview nesnesi
- 1 adet FileUpload1

Nesneleri ekledikten sonra gelelim kod kısmına

Projemizde bulunan Web.config dosyasını aşağıdaki gibi düzenliyoruz.

```
<?xml version="1.0"?>
<configuration>
  <system.web>
    <compilation debug="true" targetFramework="4.5.2">
      <assemblies>
        <add assembly="Microsoft.Office.Interop.Excel, Version=15.0.0.0, Culture=neutral,
PublicKeyToken=71E9BCE111E9429C"/>
      </assemblies>
    </compilation>
    <httpRuntime targetFramework="4.5.2"/>
  </system.web>
  <appSettings>
    <add key="ValidationSettings:UnobtrusiveValidationMode" value="none"/>
    <add key="Excelyol" value="Dosyalar"/>
  </appSettings>
  <connectionStrings>
    <!--Birinci Bağlantımız Excel 2003 ve sonrası içindir yani .xls uzantılı dosyalar -->
    <add name="Excel03baglan" connectionString="Provider=Microsoft.Jet.OLEDB.4.0;Data
Source={0}; Extended Properties='Excel 8.0;HDR={1}'"/>
    <!--İkinci Bağlantımız Excel 2007 ve sonrası içindir yani .xlsx uzantılı dosyalar -->
    <add name="Excel07baglan" connectionString="Provider=Microsoft.ACE.OLEDB.12.0;Data
Source={0}; Extended Properties='Excel 8.0;HDR={1}'"/>
  </connectionStrings>
</configuration>
```

Projemizde eklenmesi gereken kütüphaneler aşağıdaki gibidir.

```

using System;
using System.Configuration;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI; //Projenin hatasız çalışması için gerekli olan kütüphaneler.
using System.Data;
using System.IO;
using System.Data.OleDb;
using System.Data.SqlClient;
using System.Web.UI.WebControls;

```

Öncelikle özel nesne tanımlıyoruz.

```

z başvuru
public partial class _Default : System.Web.UI.Page
{
    private object dTableExcel;
    0 başvuru
    protected void Page_Load(object sender, EventArgs e)
    {
    }
}

```

Verinin Hatasız yüklenmesi için oluşturduğumuz kodlar aşağıdadır.

```

private void Excelgrid(string DosyaYol, string Uzanti) // Oluşturduğumuz sınıf altında Excelden
çekeceğimiz verinin dosya yolu ve uzantısı tanımlıyoruz
{
    //Bu alanda excel dökümanının hazırladığımız connection stringlerler türünü
belirliyoruz.
    string baglanti = "";
    switch (Uzanti)
    {
        case ".xls": //Excel 97-03 .xls uzantılı dosyalar için
            baglanti =
ConfigurationManager.ConnectionStrings["Excel03baglan"].ConnectionString;
            break;
        case ".xlsx": //Excel 07 .xlsx uzantılı dosyalar için
            baglanti =
ConfigurationManager.ConnectionStrings["Excel07baglan"].ConnectionString;

            break;
    }
    baglanti = String.Format(baglanti, DosyaYol, "Yes"); //baglantımızı string formatına
çeviriyoruz.
    OleDbConnection Excelbaglan = new OleDbConnection(baglanti);
    OleDbCommand Excelkmt = new OleDbCommand();
    OleDbDataAdapter cagirma = new OleDbDataAdapter();
    DataTable dt = new DataTable();
    Excelkmt.Connection = Excelbaglan;
    //

    //Excel dosyamızı tablo olarak tanımlıyoruz
    Excelbaglan.Open();
    DataTable Exceltablo;
    Exceltablo = Excelbaglan.GetOleDbSchemaTable(OleDbSchemaGuid.Tables, null);
    string ExcelTabloadi = Exceltablo.Rows[0]["TABLE_NAME"].ToString();
    Excelbaglan.Close();
}

```

```

//Excel dosyasındaki verilerimizi tablo olarak okuyoruz.
Excelbaglan.Open(); // bağlantımızı açıyoruz.
Excelkmt.CommandText = "SELECT * From [" + ExcelTabloadi + "]"; // tablodan almak
istediğimiz veriyi alıyoruz. Sorgu cümlesini kendinize göre düzenleye bilirsiniz.
cagirma.SelectCommand = Excelkmt;
cagirma.Fill(dt);
Excelbaglan.Close();

//Gelen Excel Dosyasının adını başlık olarak yazdırıyoruz.
GridView1.Caption = "Görüntülenmekte olan Excel Dosyanızın Adı = " +
Path.GetFileName(DosyaYol);
//Gelen Excel dosyasındaki verileri gridview de görüntülüyoruz.
GridView1.DataSource = dt;
GridView1.DataBind();
}

```

Sınıfımız oluşturduktan sonra Projemize eklediğimiz Button altına aşağıdaki kodları yazıyoruz.

```

protected void btndosya_yukle_Click(object sender, EventArgs e)
{
    if (FileUpload1.HasFile) // Eğer upload nesnesi içinde dosya varsa aşağıdaki işlemleri
    yap
    {
        //Döküman bilgileriniz alıyoruz.
        string dosyaadi = Path.GetFileName(FileUpload1.PostedFile.FileName);
        string eklenti = Path.GetExtension(FileUpload1.PostedFile.FileName);
        string dosyayolu = ConfigurationManager.AppSettings["Excelyol"];
        string dosyakaynak = Server.MapPath(dosyayolu + dosyaadi);

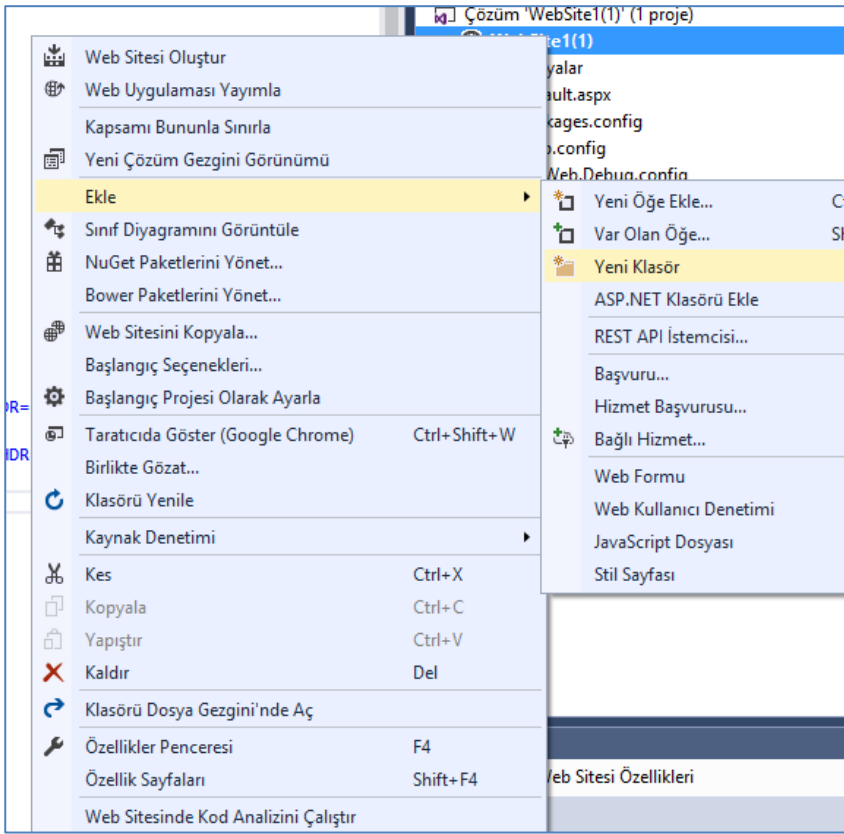
        //Dosyamızı config de belirttiğimiz yolumuza kaydediyoruz.
        FileUpload1.SaveAs(dosyakaynak);

        //Metodumuzu kullanarak Exceli Gridview içersine aktarıyoruz.
        Excelgrid(dosyakaynak, eklenti);
    }
}

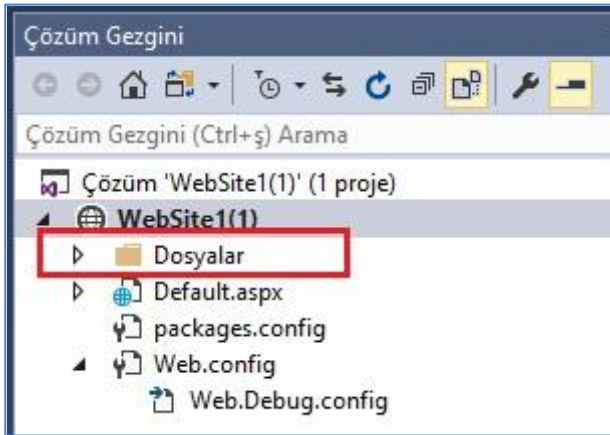
```

Burda Dikkat Edilmesi Gerekenler.

Projemize bir tane klasör ekliyoruz.



Dosyayolu olarak tanımladığımız nesnenin iki tırnak (" ") arasındaki yolu web.config dosyasında nasıl kayıt ettiyse iki tırnak arasına ordaki ismini yazacağız. Value değerine ise projemize eklediğimiz klasörün adını veriyoruz;



```

protected void btndosya_yukle_Click(object sender, EventArgs e)
{
    if (FileUpload1.HasFile) // Eğer upload nesnesi içinde dosya varsa aşağıdaki işlemleri yap
    {
        //Döküman bilgileriniz alıyoruz.
        string dosyaadi = Path.GetFileName(FileUpload1.PostedFile.FileName);
        string eklenti = Path.GetExtension(FileUpload1.PostedFile.FileName);
        string dosyayolu = ConfigurationManager.AppSettings["Excelyol"];
        string dosyakaynak = Server.MapPath(dosyayolu + dosyaadi);

        //Dosyamızı config de belirttiğimiz yolumuza kaydediyoruz.
        FileUpload1.SaveAs(dosyakaynak);

        //Metodumuzu kullanarak Exceli Gridview içersine aktarıyoruz.
        Excelgrid(dosyakaynak, eklenti);
    }
}

```

```

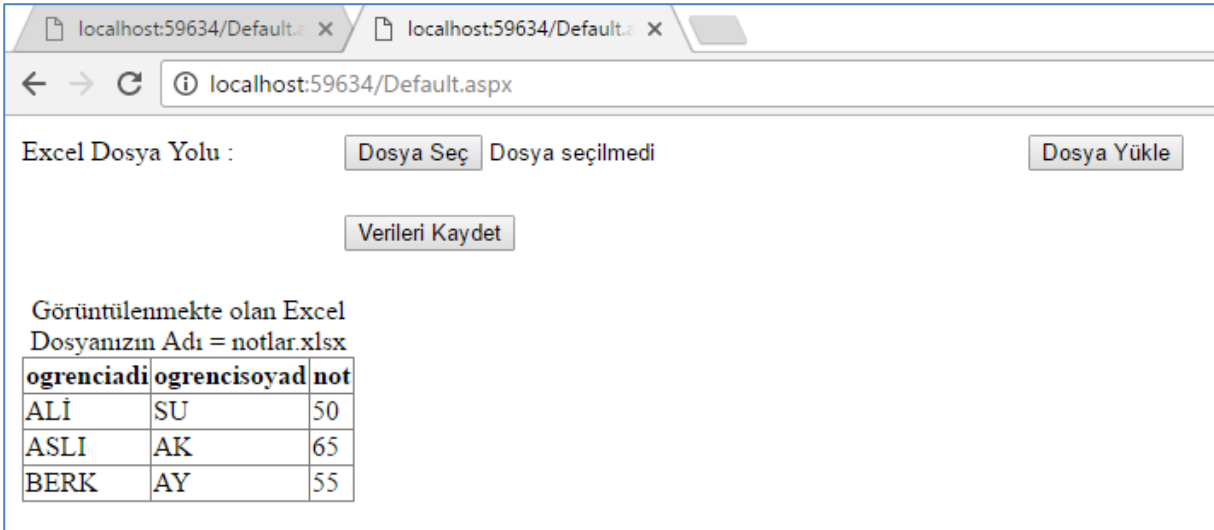
Default.aspx | Default.aspx.cs | Web.config
1 <?xml version="1.0"?>
2 <!--
3 ASP.NET uygulamanızı nasıl yapılandıracağınız hakkında daha fazla bilgi için lütfen şu adresi ziyaret edin:
4 https://go.microsoft.com/fwlink/?LinkId=169433
5 -->
6 <configuration>
7 <system.web>
8 <compilation debug="true" targetFramework="4.5.2">
9 <assemblies>
10 <add assembly="Microsoft.Office.Interop.Excel, Version=15.0.0.0, Culture=neutral, PublicKeyToken=71E9BCE111E9429C"/>
11 </assemblies>
12 </compilation>
13 <httpRuntime targetFramework="4.5.2"/>
14 </system.web>
15 <appSettings>
16 <add key="ValidationSettings:UnobtrusiveValidationMode" value="none"/>
17 <add key="Excelyol" value="Dosyalar"/>
18 </appSettings>
19 <connectionStrings>
20 <!--Birinci Bağlantımız Excel 2007 ve sonrası içindir yani .xls uzantılı dosyalar -->
21 <add name="Excel03baglan" connectionString="Provider=Microsoft.Jet.OLEDB.4.0;Data Source={0}; Extended Properties='Excel 8.0;HDR={1}'/">
22 <!--İkinci Bağlantımız Excel 2007 ve sonrası içindir yani .xlsx uzantılı dosyalar -->
23 <add name="Excel07baglan" connectionString="Provider=Microsoft.ACE.OLEDB.12.0;Data Source={0}; Extended Properties='Excel 8.0;HDR={1}'/">
24 </connectionStrings>
25 </configuration>

```

Excel dosyamız bu şekildedir.

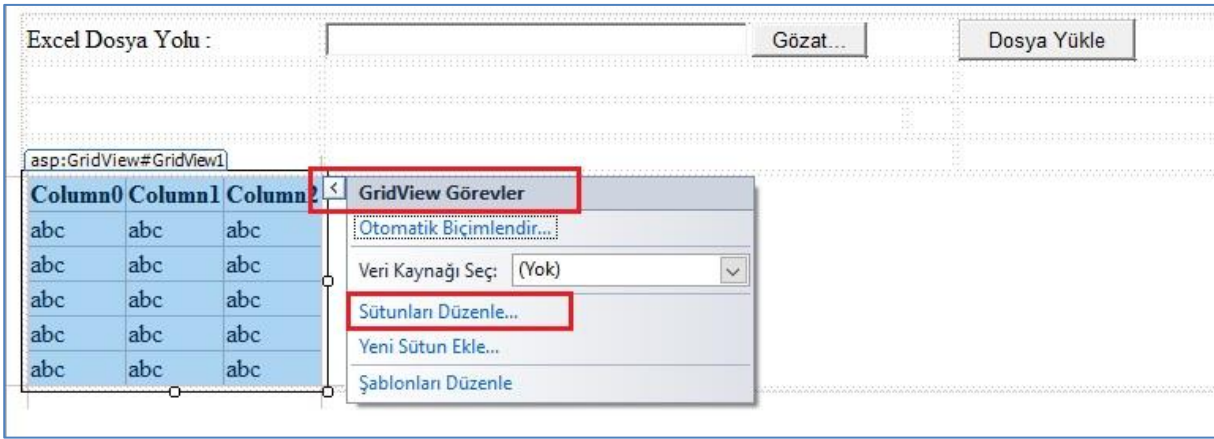
	A	B	C	D	E	F
1	ogrenciad	ogrenciso	not			
2	ALİ	SU	50			
3	ASLI	AK	65			
4	BERK	AY	55			
5						
6						
7						

Sitede Görüldüğü şekil ise;

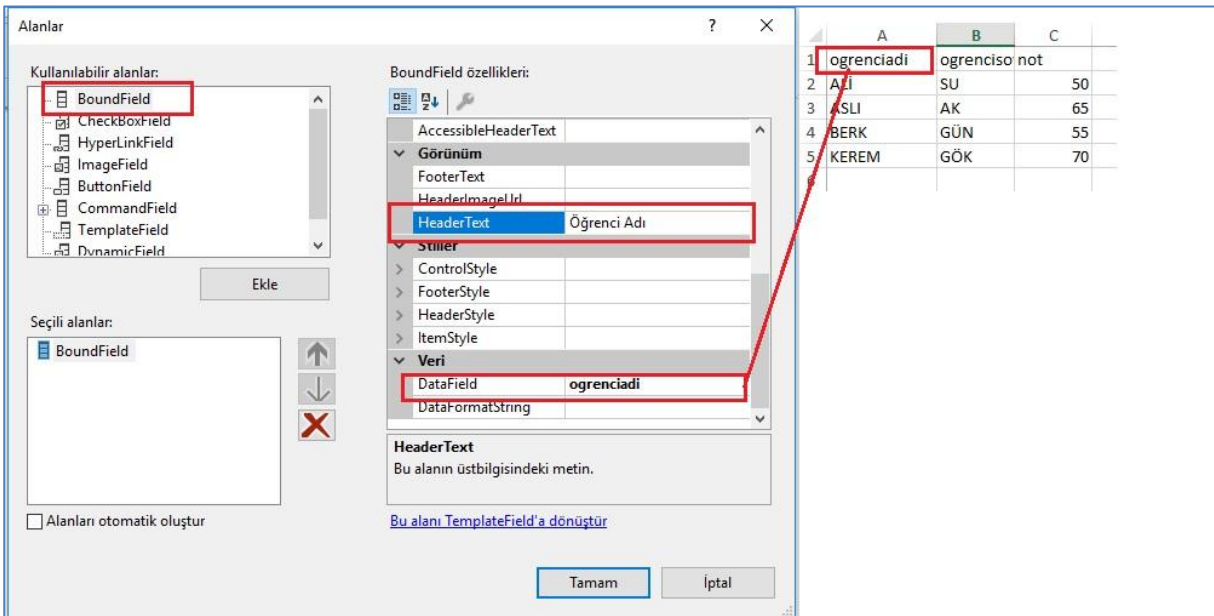


2-Excelden Çekilen Verinin Gridviewde İstenilen Başlık Altında Görüntülenmesi

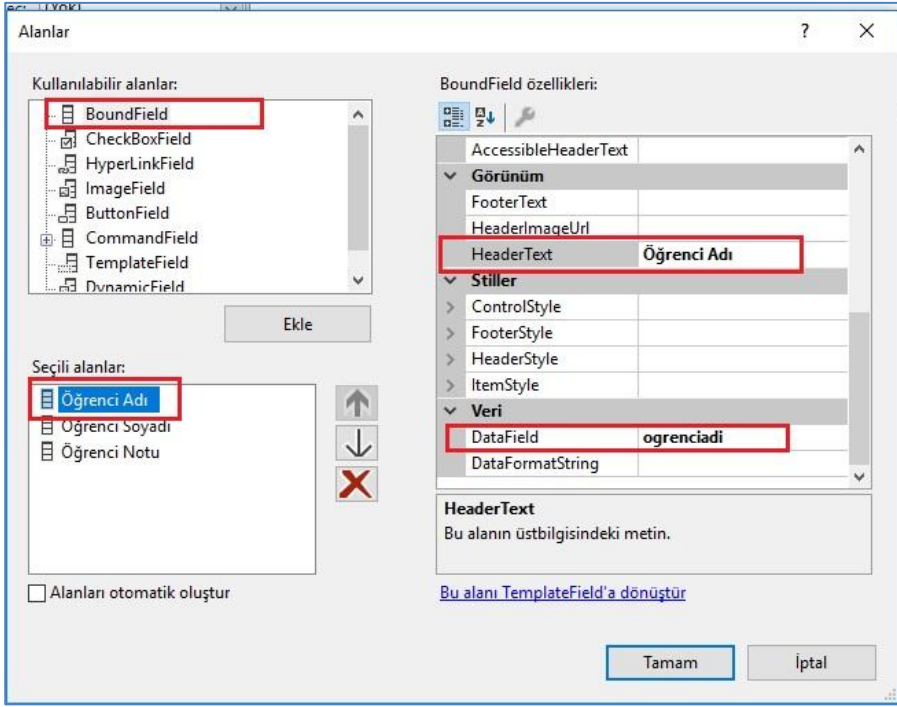
Öncelikle Gridview nesnemizde bazı düzenlemeler yapmamız gerekiyor.



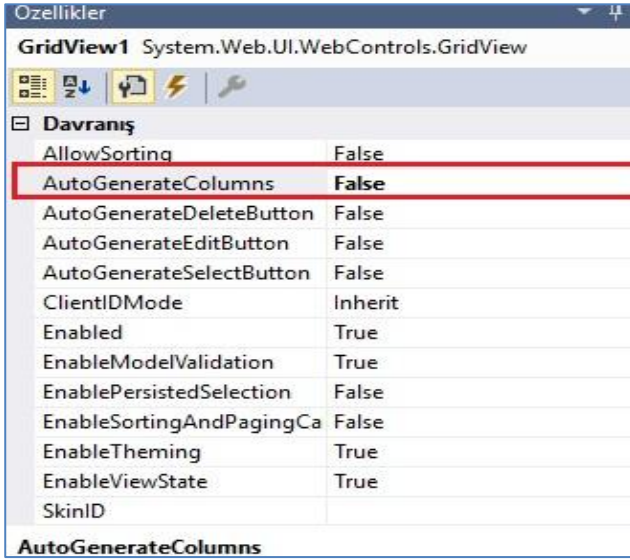
Sütunları düzenle ye bastıktan sonra karşımıza aşağıdaki gibi bir pencere gelecektir.



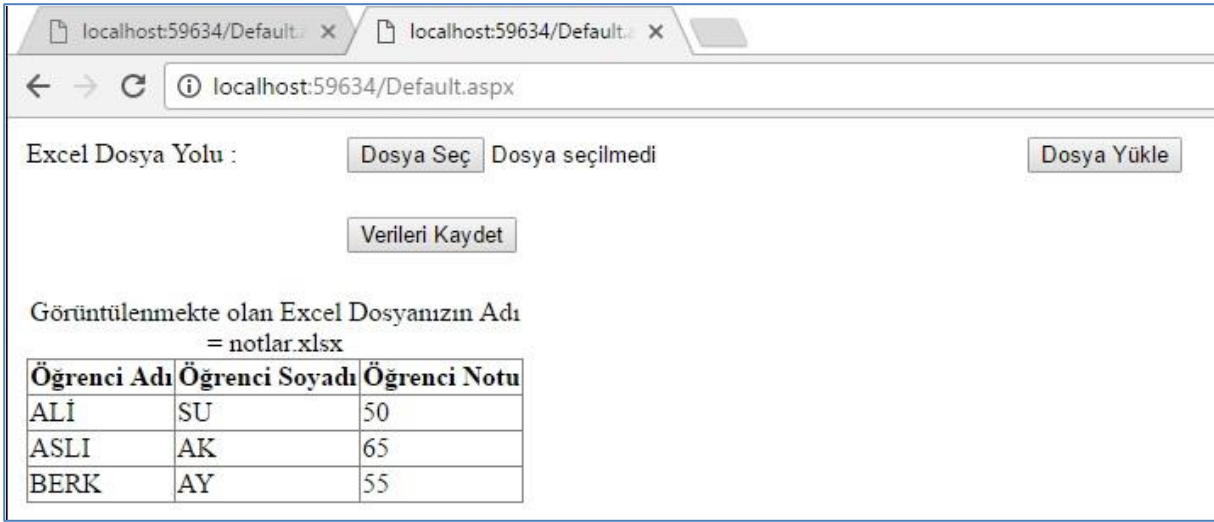
Akabinde BoundField ekledikten sonra kırmızı kutu içerisindeki veriyi görüldüğü gibi excel dosyanıza uygun şekilde düzenleyebilirsiniz. Düzenledikten sonra aşağıdaki gibi görünecektir.



Düzenleme bittikten sonra GridView tıklayıp sağ altta özellikler panosundan Ekranındaki görünen özelliği kapatın

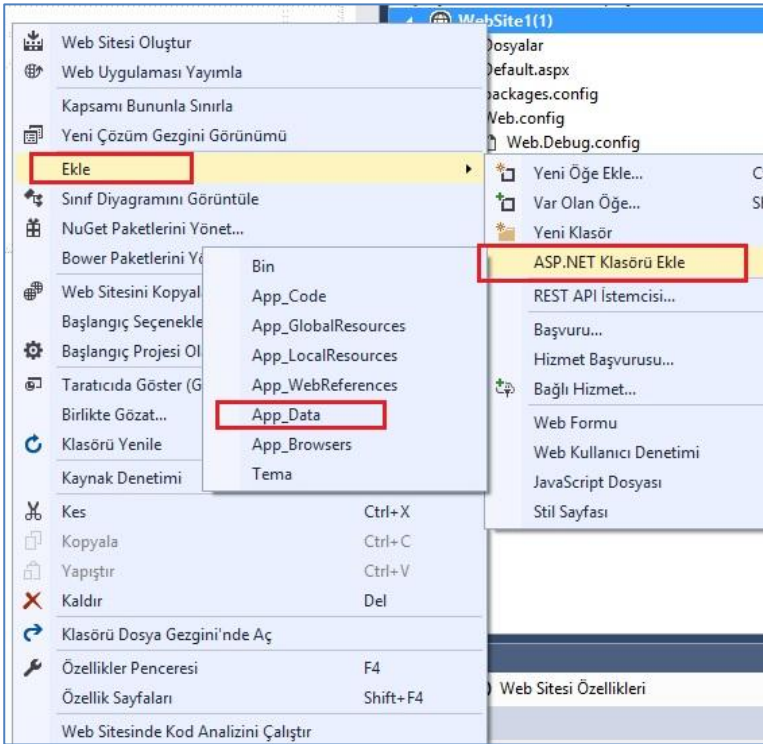


Bu işlemler bittikten sonra yukarıda verilen kod kısmını olduğu gibi uygulayın. Sitenizi çalıştırdıktan sonra istenilen başlık altında görüntülediğini göreceksiniz.

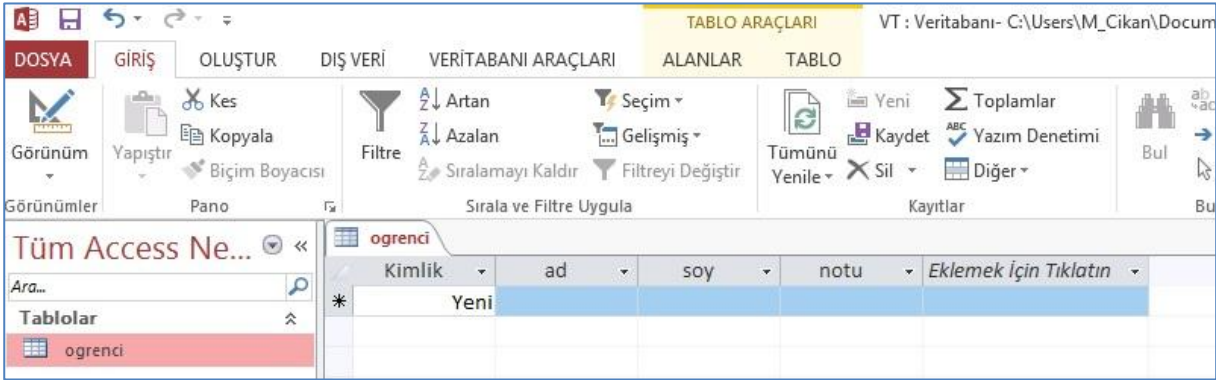
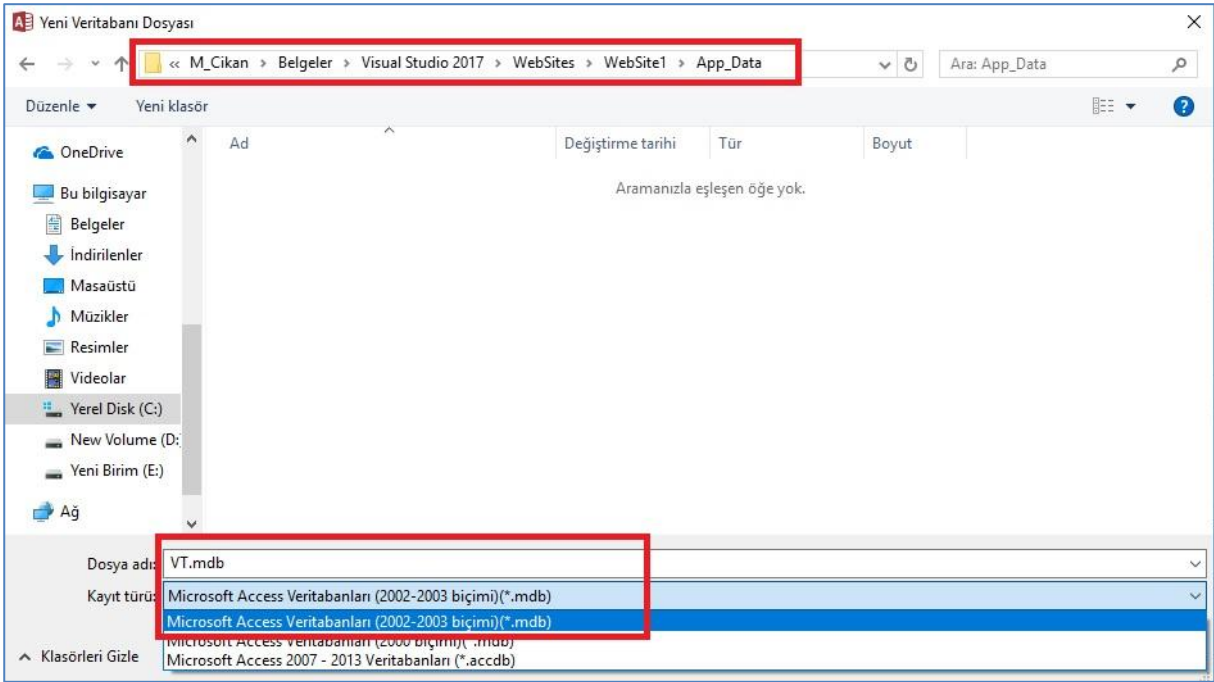


3- Excelden Çekilen Verinin Gridviewde Görüntüledikten Sonra Veritabana Kayıt Edilmesi

Öncelikle projemize App_Data klasörünü ekliyoruz.

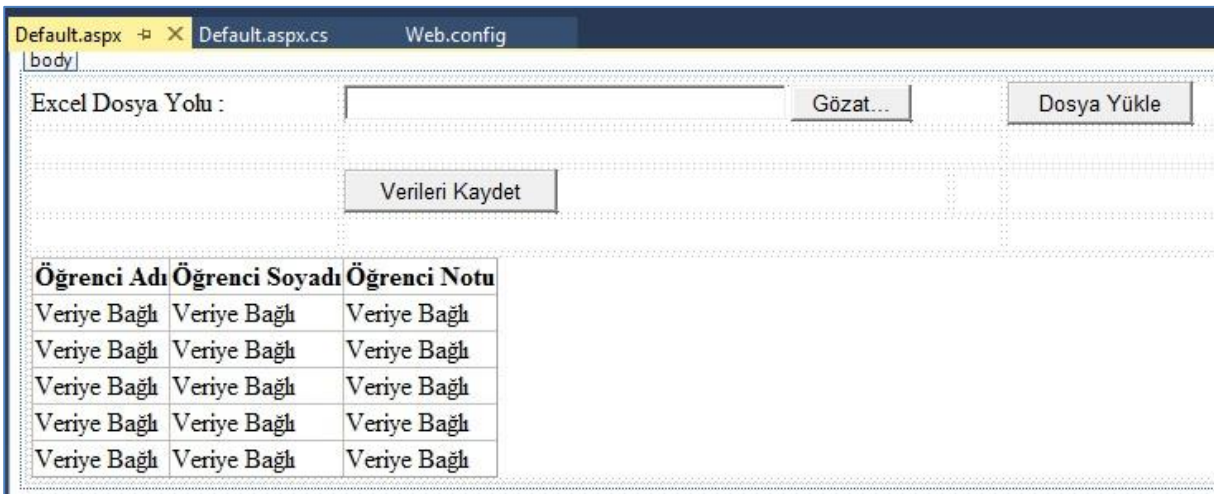


Eklemeden sonra veritabanımızı oluşturuyoruz.



Veritabanımızı oluşturduk.

Projemize bir adet button daha ekliyoruz.



Eklediğimiz buttonun altına aşağıdaki kodları yazıyoruz. Burada dikkat edilmesi gereken konu gridview nesnesinde kaçtane sütun varsa ;

```
string ad = row.Cells[0].Text.ToString();
```

cümlesindeki [0] kapalı parantezdeki sayıyı ona göre arttırmanızdır.

```
protected void btnverikayit_Click(object sender, EventArgs e)
{
    foreach (GridViewRow row in GridView1.Rows)
    {
        string ad = row.Cells[0].Text.ToString();
        string soy = row.Cells[1].Text.ToString();
        string notu = row.Cells[2].Text.ToString();
        String BaglantiYolu = "Provider=Microsoft.Jet.OLEDB.4.0; data source
=|DataDirectory|VT.mdb";
        OleDbConnection baglan = new OleDbConnection(BaglantiYolu);
        baglan.Open();
        string sorgu = "INSERT INTO ogrenci(ad,soy,notu) VALUES (@ad,@soy,@notu)";

        OleDbCommand komut = new OleDbCommand(sorgu, baglan);
        komut.Parameters.AddWithValue("@ad", ad);
        komut.Parameters.AddWithValue("@soy", soy);
        komut.Parameters.AddWithValue("@notu", notu);
        komut.ExecuteNonQuery();
        baglan.Close();
    }
    Label1.Text = "Veriler Kayıt Edildi.";
}
}
```

localhost:59634/Default.aspx

Excel Dosya Yolu : Dosya seçilmedi

Veriler Kayıt Edildi.

Görüntülenmekte olan Excel Dosyanızın Adı = notlar.xlsx

Öğrenci Adı	Öğrenci Soyadı	Öğrenci Notu
ALİ	SU	50
ASLI	AK	65
BERK	AY	55

DOSYA GİRİŞ OLUŞTUR DIŞ VERİ VERİTABANI ARAÇLARI ALANLAR TABLO

Görünüm Yapıştır Kopyala Biçim Boyacısı Filtre Artan Azalan Sıralamayı Kaldır Filtreyi Değiştir Seçim Gelişmiş Tümü Yenile Yeni Kaydet Sil Toplamlar Yazım Denetimi Diğer Bul

Sürünümler Pano Sırala ve Filtre Uygula Kayıtlar

Tüm Access Ne... ogrenci

Kimlik	ad	soy	notu	Ekleme için Tıklatın
5	ALİ	SU	50	
6	ASLI	AK	65	
7	BERK	AY	55	
*	Yeni			

B-GRIDVIEW SÜTÜNLERİNİ PROGRAMLANABİLİR OLUŞTURMA

localhost:7044/DENEME2.aspx

chkSecim2chkSecim3

Seç	İsim	Şehir	Git
<input type="checkbox"/>	Ahmet	Ankara	Detay
<input type="checkbox"/>	Mehmet	Karabük	Detay
<input checked="" type="checkbox"/>	Hasan	Çankırı	Detay
<input checked="" type="checkbox"/>	Mustafa	Bartın	Detay

localhost:7044 web sitesinin mesajı:

ID: 2, AD= Mehmet, SEHIR: Karabük. Bu bilgilerden ID bilgisi ile kullanıcının tüm detayı ayrı bir sayfada gösterilebilir.

Tamam

Deneme.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="DENEME.aspx.cs"
Inherits="DENEME" %>
```

```
<!DOCTYPE html>
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head runat="server">
```

```
<title></title>
```

```
</head>
```

```
<body>
```

```
<form id="form1" runat="server">
```

```
<asp:GridView ID="GridView1" HeaderStyle-BackColor="#3AC0F2" HeaderStyle-
ForeColor="White"
```

```
runat="server" OnRowDataBound="OnRowDataBound">
```

```
</asp:GridView>
```

```
</form>
```

```
</body>
```

```
</html>
```

Deneme.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
```

```
public partial class DENEME2 : System.Web.UI.Page
```

```
{
```

```
protected void Page_Load(object sender, EventArgs e)
```

```
{
```

```
if (!this.IsPostBack)
```

```

{
    //GridView içerisinde 4 tane sütun ekliyor.
    TemplateField Sutun1Template = new TemplateField(); //Sutun tipi Template
    Sutun1Template.HeaderText = "Seç";
    GridView1.Columns.Add(Sutun1Template);

    BoundField Sutun2Bound = new BoundField(); //Sutun tipi Bound
    Sutun2Bound.HeaderText = "İsim";
    Sutun2Bound.DataField = "Name";
    GridView1.Columns.Add(Sutun2Bound);

    TemplateField Sutun3Template = new TemplateField(); //Sutun tipi Template
    Sutun3Template.HeaderText = "Şehir";
    GridView1.Columns.Add(Sutun3Template);

    TemplateField Sutun4Template = new TemplateField(); //Sutun tipi Template
    Sutun4Template.HeaderText = "Git";
    GridView1.Columns.Add(Sutun4Template);

    this.GridView1.AutoGenerateColumns = false;
}
this.BindGrid();
}

private void BindGrid()
{
    DataTable DataTable1 = new DataTable();
    DataTable1.Columns.AddRange(new DataColumn[3]
    {
        new DataColumn("Id", typeof(int)),
        new DataColumn("Name", typeof(string)),
        new DataColumn("City",typeof(string))
    }
    );

    DataTable1.Rows.Add(1, "Ahmet", "Ankara");
    DataTable1.Rows.Add(2, "Mehmet", "Karabük");
    DataTable1.Rows.Add(3, "Hasan", "Çankırı");
    DataTable1.Rows.Add(4, "Mustafa", "Bartın");

    GridView1.DataSource = DataTable1;
    GridView1.DataBind();
}

protected void OnRowDataBound(object sender, GridViewRowEventArgs e)
{
    if (e.Row.RowType == DataControlRowType.DataRow)
    {
        CheckBox CheckBox1 = new CheckBox();
        CheckBox1.ID = "chkSecim" + e.Row.RowIndex.ToString();
    }
}

```

```

CheckBox1.CheckedChanged += CheckedChaged;
CheckBox1.AutoPostBack = true;
e.Row.Cells[0].Controls.Add(CheckBox1);

```

//Dikkat: Bound tipinde oluşturulan Ad sütunu buraya konulmamıştır.

```

TextBox TextBox1 = new TextBox();
TextBox1.ID = "txtSehir";
TextBox1.Text = (e.Row.DataItem as DataRowView).Row["City"].ToString();
e.Row.Cells[2].Controls.Add(TextBox1);

```

```

LinkButton LinkButton1 = new LinkButton();
LinkButton1.ID = "linkDetay";
LinkButton1.Text = "Detay";
LinkButton1.Click += Detaylandir;
LinkButton1.CommandArgument = (e.Row.DataItem as
DataRowView).Row["Id"].ToString();
e.Row.Cells[3].Controls.Add(LinkButton1);
}
}

```

```

protected void Detaylandir(object sender, EventArgs e)
{
    LinkButton LinkButton2 = (sender as LinkButton);
    GridViewRow row2 = (LinkButton2.NamingContainer as GridViewRow);
    string id2 = LinkButton2.CommandArgument;
    string name2 = row2.Cells[1].Text;
    string City2 = (row2.FindControl("txtSehir") as TextBox).Text;
    ClientScript.RegisterStartupScript(this.GetType(), "alert", "alert('ID: " + id2 + ", AD= " +
name2 + ", SEHIR: " + City2 + ". Bu bilgilerden ID bilgisi ile kullanıcının tüm detayı ayrı bir
sayfada gösterilebilir. ')", true);
}

```

```

protected void CheckedChaged(object sender, EventArgs e)
{
    GridViewRow row = ((GridViewRow)((CheckBox)sender).NamingContainer);
    int SiraNo = row.RowIndex;

    string chkAdi = ((CheckBox)GridView1.Rows[SiraNo].FindControl("chkSecim" +
SiraNo.ToString())).ID;
    Response.Write(chkAdi);
    //this.BindGrid();
}

```