

**T.C.
MİLLÎ EĞİTİM BAKANLIĞI**

BİLİŞİM TEKNOLOJİLERİ

ETKİLEŞİMLİ WEB UYGULAMALARINDA VERİ TABANI İŞLEMLERİ

Ankara, 2013

- Bu modül, mesleki ve teknik eğitim okul/kurumlarında uygulanan Çerçeve Öğretim Programlarında yer alan yeterlikleri kazandırmaya yönelik olarak öğrencilere rehberlik etmek amacıyla hazırlanmış bireysel öğrenme materyalidir.
- Millî Eğitim Bakanlığınca ücretsiz olarak verilmiştir.
- **PARA İLE SATILMAZ.**

İÇİNDEKİLER

AÇIKLAMALAR	iii
GİRİŞ	1
ÖĞRENME FAALİYETİ-1	3
1. VERİ TABANI BAĞLANTISI.....	3
1.1. Veri Erişim Teknolojileri	3
1.2. ADO.NET Mimarisi.....	4
1.2.1. Connection Nesnesi	5
1.2.2. Command Nesnesi	5
1.2.3. DataReader Nesnesi.....	5
1.2.4. DataAdapter Nesnesi	5
1.2.5. DataSet Nesnesi.....	6
1.3. Veri Tabanı Oluşturma.....	6
1.4. Veri Tabanına Bağlanma	6
UYGULAMA FAALİYETİ	9
ÖLÇME VE DEĞERLENDİRME	4
ÖĞRENME FAALİYETİ-2	5
2. VERİ TABANI İŞLEMLERİ.....	5
2.1. Bağlantıyı Açma – Kapama	5
2.2. Veri Seçme.....	5
2.3. Veri Ekleme	7
2.4. Veri Güncelleme	8
2.5. Veri Silme	9
UYGULAMA FAALİYETİ	11
ÖLÇME VE DEĞERLENDİRME	14
ÖĞRENME FAALİYETİ-3	15
3. VERİ TABANI BİLGİLERİNİ YÖNETME	15
3.1. Repeater Kontrolü	15
3.2. DataList Kontrolü	18
3.3. ListView Kontrolü	19
3.4. GridView Kontrolü	20
3.5. DetailsView Kontrolü	24
UYGULAMA FAALİYETİ	28
ÖLÇME VE DEĞERLENDİRME	30
MODÜL DEĞERLENDİRME	31
CEVAP ANAHTARLARI.....	34
KAYNAKÇA	36

AÇIKLAMALAR

ALAN	Bilişim Teknolojileri
DAL/MESLEK	Web Programcılığı
MODÜLÜN ADI	Etkileşimli Web Uygulamalarında Veri Tabanı İşlemleri
MODÜLÜN TANIMI	Bu modül, programlama yazılımı kullanarak .NET ortamında etkileşimli web uygulamalarında veri tabanı işlemleri ile ilgili temel bilgi ve becerilerin kazandırıldığı bir öğrenme materyalidir.
SÜRE	40/32
ÖNKOŞUL	“Etkileşimli Web Uygulamalarında Web Formları” modülünü tamamlamış olmak
YETERLİK	Veri Tabanı işlemlerini yapmak
MODÜLÜN AMACI	Genel Amaç Bu modül ile gerekli ortam sağlandığında; veri tabanı işlemlerini yapabileceksiniz. Amaçlar 1. Web site için gerekli veri tabanını oluşturabilecek ve bağlantı kurabileceksiniz. 2. Veri tabanı üzerinde işlemler yapabileceksiniz. 3. Veri tabanında kayıtlı bilgileri listeleyebilecek ve bu listeler üzerinde işlemler yapabileceksiniz.
EĞİTİM ÖĞRETİM ORTAMLARI VE DONANIMLARI	Ortam: Bilişim Teknolojileri laboratuvarı, işletme ortamı Donanım: Web programlama yazılımlarını çalıştırabilecek yeterlikte bilgisayar, İnternet bağlantısı
ÖLÇME VE DEĞERLENDİRME	Modül içinde yer alan her öğrenme faaliyetinden sonra verilen ölçme araçları ile kendinizi değerlendireceksiniz. Öğretmen modül sonunda ölçme aracı (çoktan seçmeli test, doğru-yanlış testi, boşluk doldurma, eşleştirme vb.) kullanarak modül uygulamaları ile kazandığınız bilgi ve becerileri ölçerek sizi değerlendirecektir.

GİRİŞ

Sevgili Öğrenci,

Bir web sitesinin kullanıcı ile tam anlamıyla bir etkileşimde bulunabilmesi için veri tabanı bağlantısı olması gerekmektedir. Programlama yazılımı ile ASP.Net web sitelerinde bir harici veri tabanına bağlanılabileceğiniz gibi doğrudan veri tabanını da uygulama içinde geliştirebilirsiniz.

Bir veri tabanı hazırlamak için kullanılan çeşitli yazılımlar mevcuttur. Programlama yazılımı bu tür veri tabanlarının neredeyse tamamıyla bağlantı oluşturabilmektedir. Veri tabanını tasarlarken dikkat etmeniz gereken bir nokta, sitenin tamamını kâğıt üzerinde tasarladıktan sonra veri tabanını oluşturun. Uygulama anında veri tabanında yaptığınız değişiklik çeşitli sıkıntılara neden olabilir.

Modül içinde Access veri tabanlarına bağlantı gösterilmektedir. Diğer veri tabanlarına bağlantı mantığı da aynıdır. Bağlantı işleminden sonra veri tabanı üzerinde değişikliğin nasıl yapılacağı gösterilmektedir. Ayrıca doğrudan veri tabanı ile bağlantı kurabilen kontrollere yer verilmiştir.

Modül süresince uygulamalar bir birini takip ettiği için bu uygulamaları saklamanızda fayda vardır. Uygulama faaliyetleri de yine birbirini takip etmektedir. Aynı durum onlar içinde geçerlidir.

Bu modül süresince daha önceki konuların bir kısmına yer verilmiştir. Siz uygulama ve uygulama faaliyetlerinde daha önce öğrendiğiniz yöntemleri uygulamalara dahil ederek geliştirebilirsiniz.

ÖĞRENME FAALİYETİ-1

AMAÇ

Web sitesi için gerekli veri tabanını oluşturabilecek ve bağlantı kurabileceksiniz.

ARAŞTIRMA

- Veri tabanı yazılımlarını araştırınız.

1. VERİ TABANI BAĞLANTISI

Birçok uygulamada bilgiler geçici olarak tutulur ve oturum sonlandırıldığında bu bilgiler bellekten silinir. Ancak etkileşimli web sitelerinde bu bilgilerin tekrar tekrar kullanılmasına ihtiyaç vardır. Bu durumda bilgiler bellek yerine kalıcı olarak saklanır.

Günümüzde veriler ilişkisel veri depolama modeli kullanılarak saklanmaktadır. Bu yöntemde geliştirilen uygulamalara *İlişkisel Veri Tabanı Yönetim Sistemi (Relational Database Management System – RDBMS)* adı verilmektedir. Bu yöntemde veriyi saklamak ve tekrar kullanılmak üzere depolamak için kullanılan sistemler ise *Veri Tabanı (Database)* olarak tanımlanabilir.

Günümüzde verileri depolamak için kullanılan veri tabanlarının oluşturulması için çeşitli uygulama yazılımları kullanılabilir. Programlama yazılımı ile bu veri tabanlarına bağlantı kurularak üzerlerinde işlem yapılabilmektedir.

1.1. Veri Erişim Teknolojileri

Uygulamalarda veriye erişmek için birçok veri erişim teknolojisi geliştirilmiştir. Bu teknolojilerden bazıları aşağıdaki gibi sıralanabilir;

- **ODBC (Open Database Connectivity)**

Birçok kuruluşun katılımıyla geliştirilen *ODBC* teknolojisi ile birçok veri kaynağına bağlanılabilir. *ODBC* uygulama ortamlarında API (Application Programming Interface – Uygulama Programlama Arayüzü) sunmaktadır. Hem yerel (Local) hem de uzaktaki (Remote) veri kaynaklarına erişmeye olanak sağlar.

- **DAO (Data Access Object)**

ODBC'nin kullanımının zor olması ve yeni dillerle kullanılamaması üzerine bu teknolojinin geliştirilmesiyle oluşturulmuş bir teknolojidir.

➤ **RDO (Remote Data Object)**

Uzak veri kaynaklarına erişimde *ODBC*'nin performansının geliştirilmesiyle oluşturulan daha yeni bir teknolojidir.

➤ **OLE DB (Object Linking and Embedding DataBase)**

COM arayüzünü kullanarak birçok sisteme bağlantı sağlayan bir veri erişim teknolojisidir. Bu özelliği ile en çok kullanılan teknolojilerden biridir.

➤ **ADO (ActiveX Data Object)**

Yüksek seviyeli programlama dillerinde tercih edilen OLE DB teknolojisi kullanan ve veriye erişim kolaylaştıran bir teknolojidir.

➤ **ADO.NET**

.NET uygulamalarında her türlü veriye erişim için veri tiplerine sahip, COM desteği gerektirmeyen, XML standardı üzerine kurulmuş ve .NET platformu özelliklerini kullanabilen, ADO teknolojisinin gelişmiş versiyonudur.

1.2. ADO.NET Mimarisi

ADO.NET, .NET platformunda kullanılan ortak bir katmandır. .NET ile geliştirilen tüm uygulamalar, veriye erişimde ADO.NET tiplerinden faydalanmaktadır.

ADO.NET ile farklı veri tabanları ve veri tabanı yönetim sistemleri kullanılabilir. Bu nedenle .NET platformu geliştirilirken farklı standartları destekleyen tipler yazılmış ve ayrı isim alanları (Namespace) oluşturulmuştur. Bu isim alanlarından bazıları şunlardır;

- SQL için, *Sql Server Veri Sağlayıcısı (Sql Server .NET Data Provider)*
- Oracle için, *Oracle Veri Sağlayıcısı (Oracle Data Provider)*
- OLEDB için, *OleDb .NET Veri Sağlayıcısı (OleDb .NET Data Provider)*
- ODBC için, *ODBC :NET Veri Sağlayıcısı (ODBC .NET Data Provider)*

Tüm bu isim alanları .NET platformunda *System.Data* isim alanı altında yer almaktadır.

Veri tabanı ile bağlantı kurulacağı zaman *System.Data* isim alanı ile kullanılan veri erişim teknolojisi (*System.Data.KullanılanTeknolojiİsimAlanı*) referans olarak uygulamaya eklenmelidir.

1.2.1. Connection Nesnesi

Veri kaynağı ve uygulama ister aynı ister farklı makinede yer alsın, veri kaynağı ile uygulama arasında bir bağ kurulmadan işlem yapılamaz.

Connection nesnesi, veri kaynağı ile uygulama arasında bağlantı kurmak, hangi veri kaynağına hangi güvenlik ayarlarıyla bağlanılacağını belirlemek için kullanılır.

Programlama yazılımında veri kaynağının türüne bağlı olarak *SqlConnection* ve *OleDbConnection* olmak üzere iki farklı yöntem kullanılır.

1.2.2. Command Nesnesi

Connection nesnesi kullanılarak veri kaynağı ile uygulama arasında bağlantı kurmak yeterli olmayacaktır. Veri kaynağına bilgi göndermek veya veri kaynağından veri almak için komutlar ve parametreler kullanılır.

Command nesnesi, bağlantı kurulan veri kaynağına müdahale etmeyi sağlar ve veri kaynağı ile uygulama arasında veri alma ,veri verme isteklerini taşır.

Programlama yazılımında *Command nesnesi* veri tabanı üzerinde komut yürütmek için *ExuteNonQuery*, *ExuteScalar* ve *ExuteReader* olmak üzere 3 farklı kullanımı vardır.

1.2.3. DataReader Nesnesi

Command nesnesi aracılığıyla veri kaynağından bir veri getirilecek olursa bu verileri uygulamada kullanabilmek için bazı bileşenlere ihtiyaç vardır.

DataReader nesnesi veri tabanından *Command nesnesi* ile getirilen verileri okumak için kullanılır.

1.2.4. DataAdapter Nesnesi

DataAdapter nesnesi, veri tabanı ile bağlantı kurulduktan sonra veri tabanından alınana verileri bellekte tutacak nesnelere aktarmak için kullanılır. Ayrıca verilere üzerinde değişiklik yapılarak veri tabanına tekrar aktarılması için de kullanılmaktadır.

Veri tabanında işlem yapmak için *DataAdapter nesnesinin* 4 adet özelliği bulunur;

- Seçme işlemi için *SelectCommand*,
- Ekleme işlemi için *InsertCommand*,
- Güncelleme işlemi için *UpdateCommand*,
- Silme işlemi için *DeleteCommand*

1.2.5. DataSet Nesnesi

DataSet nesnesi, *DataAdapter nesnesi* aracılığıyla alınan verileri bağlantısız olarak depolamak ve yönetmek için kullanılır.

1.3. Veri Tabanı Oluşturma

Veri tabanı programlama yazılımında *App_Data* klasöründe yer alır. Veri tabanı harici bir veri tabanı yazılımı ile oluşturulabileceği gibi doğrudan programlama yazılımı aracılığıyla da oluşturulabilir.

1.4. Veri Tabanına Bağlanma

Connection nesnesi uygulamada kullanılan veri tabanına güvenlik ayarları belirtildikten sonra bağlanabilir. Bağlantı açıldıktan sonra uygulama ile veri tabanı arasında veri aktarımı yapılabilir.

Bağlantı işleminde dikkat edilmesi gereken bir nokta veri aktarımı tamamlandıktan sonra bağlantının kapatılmasıdır. Bağlantı kapatılmadığı zaman sunucu kaynaklarının gereksiz kullanımı uygulama çalışma zamanında sorunlara sebep olacaktır.

Uygulama yazılımından harici bir *Access* veri tabanına bağlanmak için öncelikle *System.Data* ve *System.Data.OleDb* isim alanlarının uygulamaya eklenmesi gerekmektedir.

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

Bağlantı cümlesinde uygulama ile veri tabanı arasında bağlantı kurulurken bazı parametreler kullanılır. Bu parametreler;

- **OleDbConnection:** *Access* veri tabanına bağlanmak için kullanılan *Connection nesnesini* ifade eder.
- **ConnectionString:** Bağlantı cümlesini oluşturmak için *Connection nesnesi* özelliğidir.

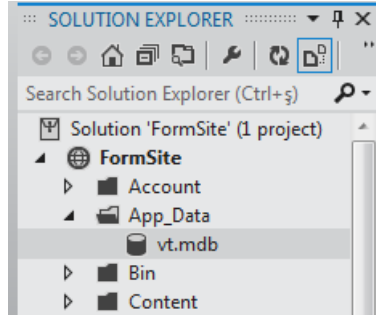
- **Provider:** Bağlantı sağlayıcısının ismini tutar. *OleDbConnection nesnesi* ile birlikte kullanılır. Uygulamalarda *Microsoft.Jet.OLEDB.4.0 Provider*'i kullanılacaktır.
- **Data Source:** *Access* veri tabanı dosya adını belirtmek için kullanılır.
- **Server.MapPath:** Sunucuya karşılık gelen fiziksel dizini belirtilen görelî veya sanal yolu eşleştirmek için kullanılır.

Veri tabanı bağlantı cümlesi aşağıdaki gibi tanımlanmaktadır;

```
OleDbConnection baglant_adi = new OleDbConnection();
baglant_adi.ConnectionString = ("Provider=microsoft.jet.oledb.4.0;Data
Source=" + Server.MapPath("Veritabanı_yeri_ve_adi "));
```

Bir *Access* veri tabanı ile bağlantı kurmak için,

- Veri tabanı oluşturun ve uygulama içinde *App_Data* klasörüne ekleyin.



Resim 1.1: App_Data Klasörüne Veri Tabanının Eklenmesi

- Veri tabanına *Uye* adında bir tablo oluşturun ve tabloya resimde bilgileri ekleyin.

Kimlik	kullanici	sifre
1	admin	1234
*	Yeni	

Resim 1.2: Örnek Veri Tabanı Tablosu

- Yeni web sayfaı açın ve giriş paneli için aşağıdaki kontrolleri sayfaya ekleyin.

Kullanıcı

Adı

Şifre

[Label5]

Resim 1.3: Giriş Paneli İçin Kullanılan Kontroller

- Buton *Click* olayına aşağıdaki veri tabanı bağlantı kodunu yazın.

```
OleDbConnection baglanti = new OleDbConnection();  
baglanti.ConnectionString = ("Provider=microsoft.jet.oledb.4.0;Data  
Source=" + Server.MapPath("~/App_Data/vt.mdb"));
```

UYGULAMA FAALİYETİ

Bir kullanıcı kayıt sayfası için veri tabanını aşağıdaki şekilde tasarlayınız.

İşlem Basamakları	Öneriler
➤ Yeni bir boş web sitesi oluşturun.	➤ <i>File > New > Web Site</i> komutunu kullanarak <i>New Web Site</i> iletişim penceresini açabilirsiniz. ➤ <i>New Web Site</i> iletişim penceresinden <i>ASP.NET Empty Web Site</i> komutunu seçebilirsiniz.
➤ Siteye veri tabanı için <i>App_Data</i> klasörünü ekleyin.	➤ Proje ismini sağ tıklayarak <i>Add > ASP.NET Folder > App_Data</i> komutunu seçebilirsiniz.
➤ Veri tabanı yazılımı ile yeni bir veri tabanı oluşturun.	➤ Veri tabanı dosyasını <i>App_Data</i> klasörüne kaydediniz.
➤ Veri tabanında <i>uye</i> adında bir tablo oluşturun.	➤ Tabloyu istediğiniz başka bir isimle de oluşturabilirsiniz.
➤ <i>Uye</i> tablosuna <i>kullanici, sifre, email, adi, soyadi, il</i> alanlarını ekleyin.	➤ Alan adlarını başka bir isimle de ekleyebilirsiniz.


KONTROL LİSTESİ

Bu faaliyet kapsamında aşağıda listelenen davranışlardan kazandığınız beceriler için **Evet**, kazanamadığınız beceriler için **Hayır** kutucuğuna (X) işareti koyarak kendinizi değerlendiriniz.

Değerlendirme Ölçütleri		Evet	Hayır
1.	Yeni bir web sitesi oluşturabildiniz mi?		
2.	Web sitesine <i>App_Data</i> klasörünü ekleyebildiniz mi?		
3.	Veri tabanı dosyasını oluşturabildiniz mi?		
4.	Veri tabanını <i>App_Data</i> klasörüne kaydedebildiniz mi?		
5.	<i>Uye</i> tablosunu oluşturabildiniz mi?		
6.	<i>Uye</i> tablosunu alanları ekleyebildiniz mi?		

UYGULAMA FAALİYETİ

Bir giriş ve kullanıcı kayıt sayfası aşağıdaki şekilde tasarlayınız.

İşlem Basamakları	Öneriler
<p>➤ Web sitesine bir web sayfası ekleyin.</p>	<p>➤ <i>Web Site > Add New Item</i> komutunu kullanabilirsiniz.</p>
<p>➤ Web sayfasına giriş paneli için aşağıdaki kontrolleri ekleyin.</p> 	<p>➤ <i>Toolbox</i> panelinden kontrolleri ekleyebilirsiniz.</p>
<p>➤ Yeni bir web sayfası ekleyin.</p>	<p>➤ <i>Web Site > Add New Item</i> komutunu kullanabilirsiniz.</p>
<p>➤ <i>LinkButton</i> linkini <i>Default2.aspx</i> olarak ayarlayın.</p>	<p>➤ <i>Properties</i> panelinden <i>PostBackUrl</i> özelliğini kullanabilirsiniz.</p>
<p>➤ <i>Default2.aspx</i> sayfasına kullanıcı kaydı için aşağıdaki kontrolleri ekleyin.</p>	<p>➤ <i>Toolbox</i> panelinden kontrolleri ekleyebilirsiniz.</p>
<p>➤ <i>Default2.aspx</i> sayfasında yer alan <i>LinkButton</i> linkini <i>Default.aspx</i> olarak ayarlayın.</p>	<p>➤ <i>Properties</i> panelinden <i>PostBackUrl</i> özelliğini kullanabilirsiniz.</p>
<p>➤ İl listesini <i>DropDownList</i> kontrolüne ekleyin.</p>	<p>➤ <i>DropDownList</i> kontrolü görev menüsünden <i>Edit Items</i> komutunu kullanabilirsiniz.</p>
<p>➤ Sayfalardaki butonlara veri tabanı bağlantı cümlelerini ekleyin.</p> <pre>OleDbConnection baglan = new OleDbConnection(); baglan.ConnectionString = ("Provider=microsoft.jet.oledb.4.0; Data Source=" + Server.MapPath ("~/App_Data\\veritabani.mdb"));</pre>	<p>➤ Buton <i>Click</i> olayına ekleyebilirsiniz.</p> <p>➤ <i>System.Data</i> ve <i>System.Data.OleDb</i> isim alanlarını sayfaya eklemeyi unutmayınız.</p>

KONTROL LİSTESİ

Bu faaliyet kapsamında aşağıda listelenen davranışlardan kazandığınız beceriler için **Evet**, kazanamadığınız beceriler için **Hayır** kutucuğuna (X) işareti koyarak kendinizi değerlendiriniz.

	Değerlendirme Ölçütleri	Evet	Hayır
1.	WEB sitesine yeni sayfa ekleyebildiniz mi?		
2.	Kontrolleri sayfaya ekleyebildiniz mi?		
3.	Veri tabanı bağlantı cümlesini oluşturabildiniz mi?		

DEĞERLENDİRME

Değerlendirme sonunda “**Hayır**” şeklindeki cevaplarınızı bir daha gözden geçiriniz. Kendinizi yeterli görmüyorsanız öğrenme faaliyetini tekrar ediniz. Bütün cevaplarınız “**Evet**” ise “Ölçme ve Değerlendirme”ye geçiniz.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki cümlelerin başında boş bırakılan parantezlere, cümlelerde verilen bilgiler doğru ise D, yanlış ise Y yazınız.

1. () Web sitelerinde verileri saklamak ve tekrar kullanmak üzere depolamak için kullanılan sistemlere *Veri Tabanı* denir.
2. () ADO.NET XML standardı üzerine kurulmuştur.
3. () *Connection* nesnesi ile bağlantı kurulurken güvenlik ayarları **belirlenemez**.
4. () *Command* nesnesi, bağlantı kurulan veri kaynağına müdahale etmeyi sağlar ve veri kaynağı ile uygulama arasında veri alma – verme isteklerini taşır.
5. () *DataReader* nesnesi veri tabanından *Command* nesnesi ile getirilen veriler üzerinde değişiklik yapmak için kullanılır.
6. () Programlama yazılımında veri tabanı *App_Data* klasöründe tutulması önerilir.
7. () *Access* veri tabanına bağlantı kurulurken *System.Data* ve *System.Data.OleDb* isim alanlarının uygulamaya eklenmesi gerekir.
8. () *Access* veri tabanı bağlantısında *OleDbConnection* nesnesi kullanılır.

DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru ise bir sonraki öğrenme faaliyetine geçiniz.

ÖĞRENME FAALİYETİ-2

AMAÇ

Veri tabanı üzerinde işlemler yapabileceksiniz.

ARAŞTIRMA

- Veri tabanı üzerinde işlem yapmak için kullanılan SQL komutlarını araştırınız.

2. VERİ TABANI İŞLEMLERİ

Veri tabanı ile bağlantı kurulduktan sonra veri tabanı üzerindeki veriler seçilebilir, bir kontrole aktarılabilir ya da veri tabanına veriler eklenebilir, var olan bilgiler değiştirilebilir veya silinebilir.

2.1. Bağlantıyı Açma – Kapama

Veri tabanı bağlantı cümlesi yazıldıktan sonra veri tabanı bağlantısının açılması gerekir. Bağlantıyı açmak için;

```
Baglanti_Adi.Open();
```

komutu kullanılır. Veri tabanı ile veri alış verişi yapıldıktan sonra ise veri tabanı ile yapılan bağlantının mutlaka kapatılmalıdır. Bağlantıyı kapatmak için

```
Baglanti_Adi.Close();
```

komutu kullanılır.

2.2. Veri Seçme

Veri tabanında bulunan bir veya birden fazla veriyi seçmek için *Select* SQL komutu kullanılır. Veri tabanından veriler aşağıdaki komutla seçilir;

```
OleDbCommand Komut_Adi = new OleDbCommand("select seçilecek_veriler from Tablo_adi", Baglanti_Adi);
```

Veri tabanından bir veriyi seçmek için;

- Yeni web sayfaı açın ve giriş paneli için aşağıdaki kontrolleri sayfaya ekleyin.

The image shows a login form with the following elements: a label 'Kullanıcı Adı' above a text input field, a label 'Şifre' above another text input field, a button labeled 'Giriş', and a label 'Label5' below the button.

Resim 2.1: Giriş Paneli için kullanılan kontroller

- Buton *Click* olayına aşağıdaki veri tabanı bağlantı cümlesini yazın.

```
OleDbConnection baglanti = new OleDbConnection();  
baglanti.ConnectionString = ("Provider=microsoft.jet.oledb.4.0;Data  
Source=" + Server.MapPath("~/App_Data/vt.mdb"));
```

- Bağlantıyı açın.

```
baglanti.Open();
```

- Bağlantı cümlesinin altına aşağıdaki komut cümlesini ekleyin.

```
OleDbCommand komut = new OleDbCommand("select kullanıcı,Sifre from Uye",  
baglanti);
```

- Komut cümlesinin altına aşağıdaki kodları ekleyin. Aşağıdaki kodlarla veri tabanında *uye* tablosundan *kullanıcı* ve *sifre* sütunlarından bilgiler alınmış, *TextBox1* ve *TextBox2* değerleri ile karşılaştırılmıştır. Karşılaştırma sonucu doğru ise sayfa *Default2.aspx* sayfasına yönlendirilmiştir. Yanlış ise hata mesajı ekrana yazdırılmıştır.

```
OleDbDataReader oku = komut.ExecuteReader();  
while (oku.Read())  
{  
    if (TextBox1.Text == oku[0].ToString() && TextBox2.Text ==  
oku[1].ToString())  
    {  
        Response.Redirect("Default2.aspx");  
    }  
else  
    {  
        Label5.Text = "Yanlış Kullanıcı Adı ve/veya şifre!";  
    }  
}
```

- Bağlantıyı kapatın.

```
baglanti.Close();
```

- Uygulamayı çalıştırarak deneyin.

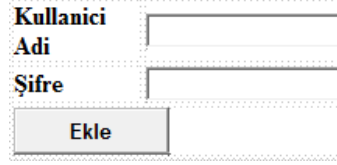
2.3. Veri Ekleme

Veri tabanına veri eklemek için *Insert* SQL komutu kullanılır. Veri tabanına veriler aşağıdaki komutla eklenir;

```
OleDbCommand Komut_Adi = new OleDbCommand("insert into Tablo_Adi  
(Eklenecek_veri) values ('"+Eklenecek_veri_kaynagi+"')", Baglanti_Adi);
```

Veri tabanından bir veriyi seçmek için;

- Yeni web sayfaı açın ve kullanıcı ekleme paneli için aşağıdaki kontrolleri sayfaya ekleyin.



The image shows a web form with two text input fields. The first field is labeled 'Kullanici Adi' and the second is labeled 'Şifre'. Below the fields is a button labeled 'Ekle'. The form is enclosed in a dashed border.

Resim 2.2: Ekleme işlemi için kullanılan kontroller

- Buton *Click* olayına aşağıdaki veri tabanı bağlantı cümlesini yazın.

```
OleDbConnection baglanti = new OleDbConnection();  
baglanti.ConnectionString = ("Provider=microsoft.jet.oledb.4.0;Data  
Source=" + Server.MapPath("~/App_Data/vt.mdb"));
```

- Bağlantıyı açın.

```
baglanti.Open();
```

- Bağlantı cümlesinin altına aşağıdaki komut cümlesini ekleyin. Aşağıdaki kodlarla veri tabanında *uye* tablosu *kullanici* ve *sifre* sütunlarına, *TextBox1* ve *TextBox2* değerleri ile aktarılmak için komut cümlesi oluşturulmuştur.

```
OleDbCommand komut = new OleDbCommand("insert into uye(kullanici,sifre)  
values('" + TextBox1.Text + "','" + TextBox2.Text+ "')", baglanti);  
komut.ExecuteNonQuery();
```

- Bağlantıyı kapatın.

```
baglanti.Close();
```

- Kontrol değerlerini temizleyin.

```
TextBox1.Text = "";
```

```
TextBox2.Text = "";
```

- Uygulamayı çalıştırarak deneyin.

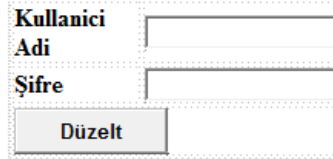
2.4. Veri Güncelleme

Veri tabanına veri eklemek için *Update* SQL komutu kullanılır. Veri tabanına veriler aşağıdaki komutla eklenir;

```
OleDbCommand Komut_Adi = new OleDbCommand("update Tablo_Adi set  
Degisecek_veri_Sütunu='" + Yeni_veri + "' where Karsilastirma_Sütunu='" +  
Karsilastirilan_Deger + "'", Baglanti_Adi);
```

Veri tabanından bir veriyi seçmek için;

- Yeni web sayfaı açın ve kullanıcı ekleme paneli için aşağıdaki kontrolleri sayfaya ekleyin.



The image shows a web form for updating user information. It contains two text input fields: the first is labeled 'Kullanıcı Adı' (Username) and the second is labeled 'Şifre' (Password). Below these fields is a button labeled 'Düzeltil' (Update).

Resim 2.3: Güncelleme işlemi için kullanılan kontroller

- Buton *Click* olayına aşağıdaki veri tabanı bağlantı cümlesini yazın.

```
OleDbConnection baglanti = new OleDbConnection();  
baglanti.ConnectionString = ("Provider=microsoft.jet.oledb.4.0;Data  
Source=" + Server.MapPath("~/App_Data/vt.mdb"));
```

- Bağlantıyı açın.

```
baglanti.Open();
```

- Bağlantı cümlesinin altına aşağıdaki komut cümlesini ekleyin. Aşağıdaki kodlarla veri tabanında *uye* tablosunda *kullanici* sütunundaki değer *TextBox1* değeri ile aynı olan satırında *sifre* sütunundaki veri, *TextBox2* değeri ile değiştirilmiştir.

```
OleDbCommand komut = new OleDbCommand ("update uye set  
sifre='"+TextBox2.Text+ "' where kullanici='"+TextBox1.Text+"'", baglanti);  
komut.ExecuteNonQuery();
```

- Bağlantıyı kapatın.

```
baglanti.Close();
```

- Kontrol değerlerini temizleyin.

```
TextBox1.Text="";  
TextBox2.Text = "";
```

- Uygulamayı çalıştırarak deneyin.

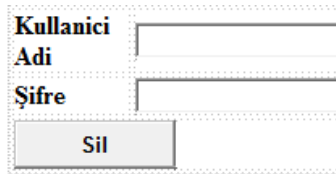
2.5. Veri Silme

Veri tabanına veri eklemek için *Update* SQL komutu kullanılır. Veri tabanına veriler aşağıdaki komutla eklenir;

```
OleDbCommand komut = new OleDbCommand("delete from Tablo_Adi where  
Karsilastirma_Sütunu='"Karsilastirilan_Deger'", Baglanti_Adi);
```

Veri tabanından bir veriyi seçmek için;

- Yeni web sayfaı açın ve kullanıcı ekleme paneli için aşağıdaki kontrolleri sayfaya ekleyin.



The image shows a web form with two text input fields. The first field is labeled 'Kullanici Adi' and the second is labeled 'Şifre'. Below these fields is a button labeled 'Sil'.

Resim 2.4: Silme işlemi için kullanılan kontroller

- Buton *Click* olayına aşağıdaki veri tabanı bağlantı cümlesini yazın.

```
OleDbConnection baglanti = new OleDbConnection();  
baglanti.ConnectionString = ("Provider=microsoft.jet.oledb.4.0;Data  
Source=" + Server.MapPath("~/App_Data/vt.mdb"));
```

- Bağlantıyı açın.

```
baglanti.Open();
```

- Bağlantı cümlesinin altına aşağıdaki komut cümlesini ekleyin. Aşağıdaki kodlarla veri tabanında *uye* tablosunda *kullanici* sütunundaki değer *TextBox1* değeri ile aynı olan veriler veri tabanından silinmiştir.

```
OleDbCommand komut = new OleDbCommand("delete from uye where kullanici='"  
+ TextBox1.Text + "'", baglanti);  
komut.ExecuteNonQuery();
```

- Bağlantıyı kapatın.

```
baglanti.Close();
```

- Kontrol değerlerini temizleyin.

```
TextBox1.Text = "";
```

- Uygulamayı çalıştırarak deneyin.

UYGULAMA FAALİYETİ

Bir önceki uygulama faaliyetindeki uygulamaya giriş ve kullanıcı ekleme paneli işlemlerini tamamlayınız.

İşlem Basamakları	Öneriler
<p>➤ <i>Default.aspx</i> sayfasında buton <i>Click</i> olayı için bağlantıyı açın.</p>	<p>➤ <i>Bağlan.Open</i> komutunu kullanabilirsiniz.</p>
<p>➤ Giriş için komut cümlesini ekleyin. <code>OleDbCommand</code> komut = <code>new OleDbCommand("select kullanıcı,sifre from Uye", bağlan);</code></p>	<p>➤ <i>Click</i> olayında bağlantı cümlesinden sonra komut cümlesini ekleyebilirsiniz.</p>
<p>➤ Giriş panelindeki kontrollerdeki değerler ile veri tabanındaki verileri karşılaştırmak için aşağıdaki kodları ekleyin. <code>OleDbDataReader</code> oku = <code>komut.ExecuteReader();</code> <code>while</code> (oku.Read()) { if (TextBox1.Text == oku[0].ToString() && TextBox2.Text == oku[1].ToString()) { Session.Add("kullaniciadi", TextBox1.Text); Response.Redirect("Default3.aspx"); } else { Label3.Text = "Yanlış Kullanıcı Adı ve/veya şifre!"; } }</p>	<p>➤ <i>Click</i> olayında komut cümlesinden sonra ekleyebilirsiniz.</p>
<p>➤ <i>Default2.aspx</i> sayfasında buton <i>Click</i> olayı için bağlantıyı açın.</p>	<p>➤ <i>Bağlan.Open</i> komutunu kullanabilirsiniz.</p>
<p>➤ Yeni kayıt işleminde kullanıcı adı tekrarı olmaması için aşağıdaki kodları ekleyin. <code>OleDbCommand</code> komut = <code>new OleDbCommand("select kullanıcı from Uye", bağlan);</code> <code>OleDbDataReader</code> oku = <code>komut.ExecuteReader();</code> <code>while</code> (oku.Read()) { }</p>	<p>➤ Kullanıcı adı için girilen değeri veri tabanında <i>kullanıcı</i> alanındaki verilerle karşılaştırabilirsiniz.</p>

<p>➤ Karşılaştırma sonucunda kullanıcı adı kullanımda değilse verileri veri tabanına ekleyin, kullanımda ise uyarı mesajı görüntülenmesini sağlayın.</p> <pre> if (TextBox1.Text != oku[0].ToString()) { OleDbCommand komut1 = new OleDbCommand("insert into uye(kullanici,sifre,email,adi,soyadi,il) values(' + TextBox1.Text + ',' + TextBox2.Text + ',' + TextBox3.Text + ',' + TextBox4.Text + ',' + TextBox5.Text + ',' + DropDownList1.SelectedItem.Text + ')", baglan); komut.ExecuteNonQuery(); baglan.Close(); TextBox1.Text = ""; TextBox2.Text = ""; TextBox3.Text = ""; TextBox4.Text = ""; TextBox5.Text = ""; DropDownList1.SelectedIndex = 0; } else { Label7.Text = "Bu kullanıcı ismi kullanılıyor!"; } </pre>	<p>➤ Verileri veri tabanına eklemek için <i>Insert SQL</i> komutunu kullanabilirsiniz.</p> <p>➤ Ekleme işleminden sonra kontrol verilerini temizleyebilirsiniz.</p>
<p>➤ Uygulamayı çalıştırın ve giriş paneli ile kullanıcı ekleme panelini deneyin.</p>	

KONTROL LİSTESİ

Bu faaliyet kapsamında aşağıda listelenen davranışlardan kazandığınız beceriler için **Evet**, kazanamadığınız beceriler için **Hayır** kutucuğuna (X) işareti koyarak kendinizi değerlendiriniz.

Değerlendirme Ölçütleri	Evet	Hayır
1. Komut cümlesini oluşturabildiniz mi?		
2. Kontroller ile veri tabanındaki verileri karşılaştırabildiniz mi?		
3. Kontrollerdeki verileri veri tabanına ekleyebildiniz mi?		

DEĞERLENDİRME

Değerlendirme sonunda “**Hayır**” şeklindeki cevaplarınızı bir daha gözden geçiriniz. Kendinizi yeterli görmüyorsanız öğrenme faaliyetini tekrar ediniz. Bütün cevaplarınız “**Evet**” ise “Ölçme ve Değerlendirme”ye geçiniz.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki cümlelerin başında boş bırakılan parantezlere, cümlelerde verilen bilgiler doğru ise D, yanlış ise Y yazınız.

1. () Veri tabanı ile veri alış verişi yapıldıktan sonra veri tabanı bağlantısının mutlaka kapatılması gerekir.
2. () Veri tabanından verileri seçmek için *Select* komutu kullanılır.
3. () *Select* komutunda kullanılan * simgesi veri tabanındaki tüm tabloları ifade eder.
4. () Veri tabanına veri eklemek için *Update* komutu kullanılır
5. () Veri tabanından verileri silmek için *Delete* komutu kullanılır.

DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru ise bir sonraki öğrenme faaliyetine geçiniz.

ÖĞRENME FAALİYETİ-3

AMAÇ

Veri tabanında kayıtlı bilgileri listeleyebilecek ve bu listeler üzerinde işlemler yapabileceksiniz.

ARAŞTIRMA

- ASP.Net'te kullanılan veri bağlama nesnelерinin isimlerini araştırınız.

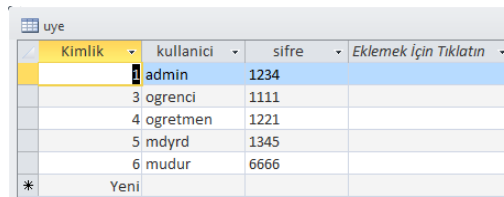
3. VERİ TABANI BİLGİLERİNİ YÖNETME

3.1. Repeater Kontrolü

Repeater kontrolü ASP.NET'te veri bağlamak için kullanılan kontrollerden biridir. Diğer veri bağlama kontrollerine göre daha esnek bir yapıya sahiptir. Bu kontrolle ilgili dikkat edilmesi gereken bir nokta varsayılan görünümünün olmamasıdır. HTML etiketlerine eklenen *ItemTemplate* etiketleri ile görünümü oluşturulabilir.

Repeater kontrolü ile verileri bağlamak için;

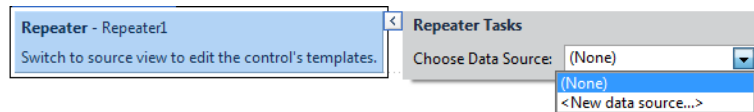
- Web sitesinde yeni bir web sayfası açın.
- Veri tabanına aşağıdaki verileri girin ve *App_Data* klasörüne kaydedin.



Kimlik	kullanıcı	sifre	Ekleme için Tıklatın
1	admin	1234	
3	öğrenci	1111	
4	öğretmen	1221	
5	mdyrd	1345	
6	mudur	6666	
* Yeni			

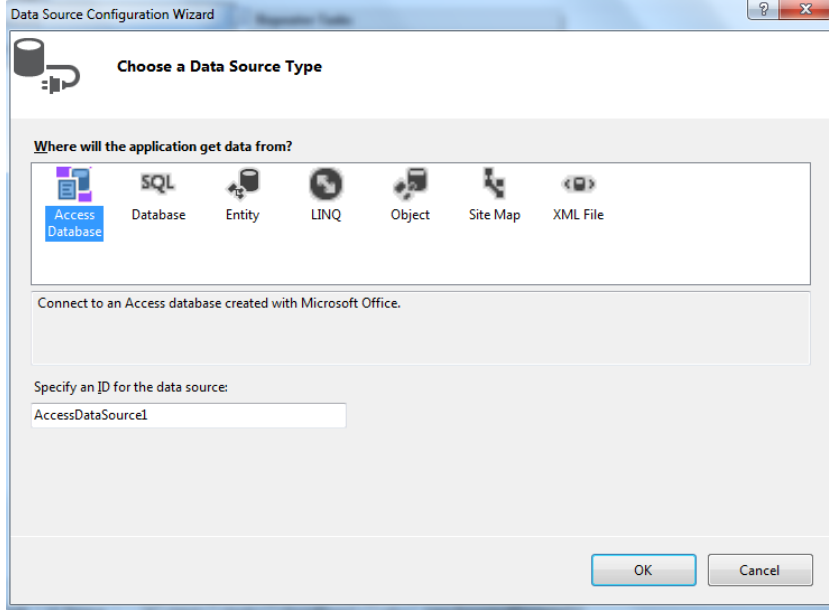
Resim 3.1: Örnek veri tabanı tablosu

- Sayfaya bir *Repeater* kontrolü ekleyin. *Toolbox* panelinden *Data* sekmesinden ulaşabilirsiniz.
- *Repeater* kontrolü görev menüsünde *Choose Data Source* alanından veri tabanı bağlantısını yapmak için *New data source* komutunu seçin.



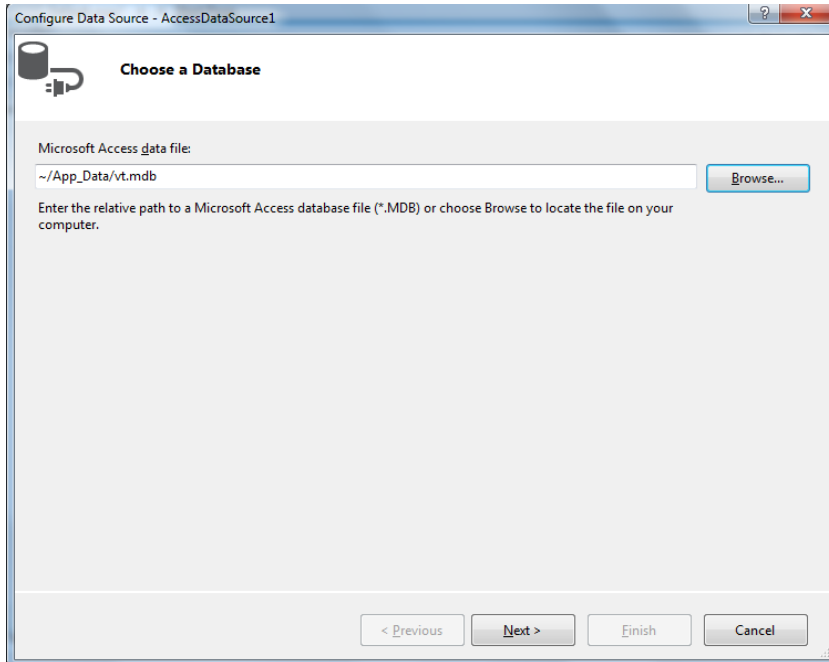
Resim 3.2: Repeater kontrolü görev menüsü

- *Data Source Configuration Wizard* iletişim penceresinden uygun veri tabanı türünü seçiniz.



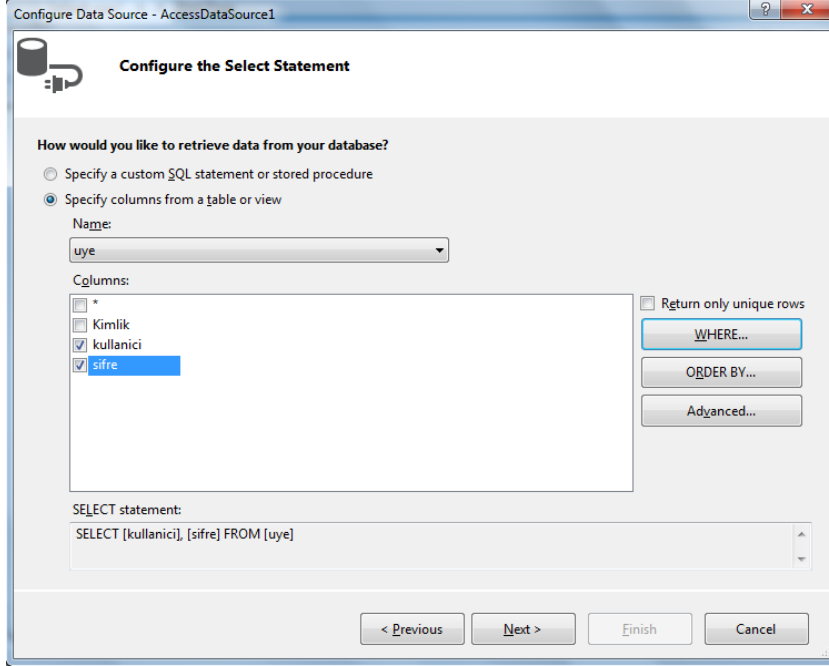
Resim 3.3: Data Source Configuration Wizard iletişim penceresi – veri tabanı türünün seçilmesi

- Bir sonraki adımda *Configure Data Source* iletişim penceresinden veri tabanının yerini gösteriniz.



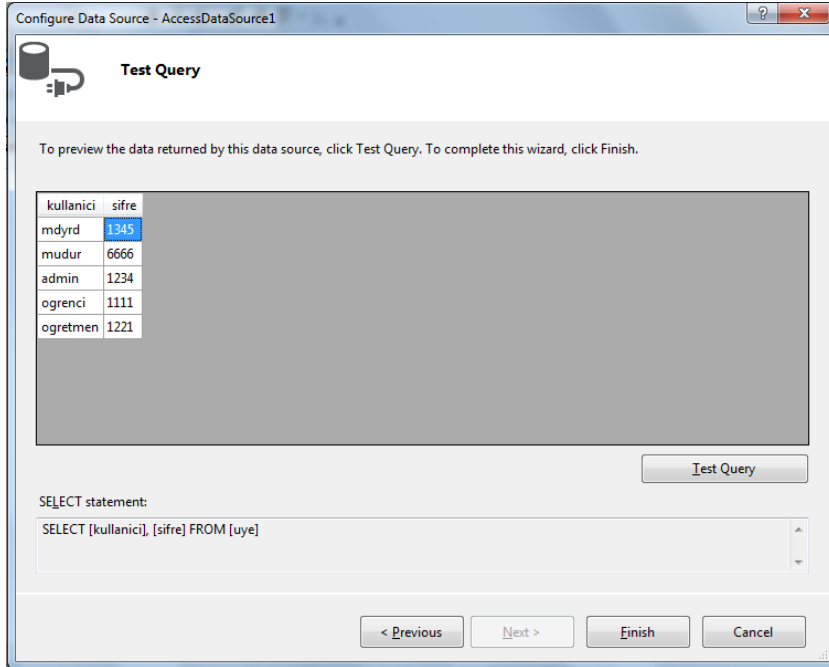
Resim 3.4: Configure Data Source iletişim penceresi – veri tabanı konumunun belirlenmesi

- Bir sonraki adımda tablo ve alınacak veri sütunlarını seçin.



Resim 3.5: Configure Data Source iletişim penceresi – veri tabanı tablosunun ve alanlarının seçilmesi

- Bir sonraki adımda *Test Query* komutu ile seçilen verileri test edebilirsiniz.



Resim 3.6: Configure Data Source iletişim penceresi – seçilen verilerin test edilmesi

- *Finish* butonu ile tanımlama işlemini tamamlayın. *AccessDataSource* otomatik olarak sayfaya eklenecektir.
- *Repeater* kontrolünde verilerin gösterilebilmesi için aşağıdaki kodları kod sayfasında kontrolün kodlarına ekleyin.

```
<ItemTemplate>
  <b>Kullanıcı: </b> <%# DataBinder.Eval(Container.DataItem, "kullanici") %>
  <br>
  <b>Şifre: </b> <%# DataBinder.Eval(Container.DataItem, "sifre") %>
  <br>
</ItemTemplate>
```

- Sayfayı görüntüleyin. *Repeater* kontrolünde veri tabanı *uye* tablosundaki veriler gösterilecektir.

Kullanıcı: mdyrd
Şifre: 1345
Kullanıcı: mudur
Şifre: 6666
Kullanıcı: admin
Şifre: 1234
Kullanıcı: ogrenci
Şifre: 1111
Kullanıcı: ogretmen
Şifre: 1221

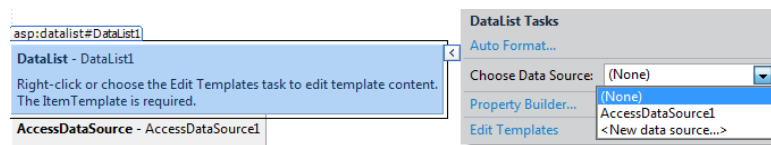
Resim 3.7: Repeater uygulaması ekran çıktısı

3.2. DataList Kontrolü

DataList kontrolü ASP.NET'te kullanılan bir başka veri bağlama kontrolüdür. *Repeater* kontrolüne göre daha gelişmiştir. Bu kontrol ile bir veri tabanındaki verileri görüntülemenin yanında seçme ve güncelleme işlemleri de kolaylıkla yapılabilir.

DataList kontrolü ile verileri bağlamak için;

- Web sitesinde yeni bir web sayfası açın.
- Sayfaya bir *DataList* kontrolü ekleyin. *Toolbox* panelinden *Data* sekmesinden ulaşabilirsiniz.
- *DataList* kontrolü görev menüsünde *Choose Data Source* alanından veri tabanı bağlantısı için daha önceden oluşturduğunuz bağlantıyı seçin. Daha önceden bir bağlantı yoksa yeni bağlantı oluşturabilirsiniz.



Resim 3.8: DataList kontrolü görev menüsü

- Sayfaı görüntüleyin. *DataList* kontrolünde veri tabanı *uye* tablosundaki veriler gösterilecektir.

kullanici: mdyrd
sifre: 1345

kullanici: mudur
sifre: 6666

kullanici: admin
sifre: 1234

kullanici: ogrenci
sifre: 1111

kullanici: ogretmen
sifre: 1221

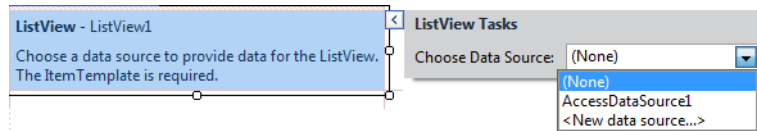
Resim 3.9: DataList uygulaması ekran çıktısı

3.3. ListView Kontrolü

ListView kontrolü ASP.NET'te kullanılan bir başka veri bağlama kontrolüdür. Bu kontrol ile standart *ListBox* kontrolünde olduđu gibi veri tabanındaki veriler liste halinde görüntülenebilir.

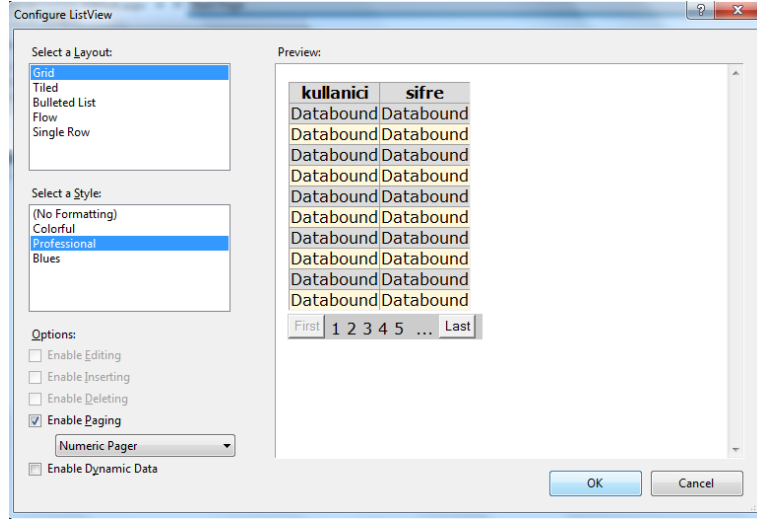
ListView kontrolü ile verileri bağlamak için;

- Web sitesinde yeni bir web sayfası açın.
- Sayfaya bir *ListView* kontrolü ekleyin. *Toolbox* panelinden *Data* sekmesinden ulaşabilirsiniz.
- *ListView* kontrolü görev menüsünde *Choose Data Source* alanından veri tabanı bağlantısı için daha önceden oluşturduğunuz bağlantıyı seçin. Daha önceden bir bağlantı yoksa yeni bağlantı oluşturabilirsiniz.



Resim 3.10: ListView kontrolü görev menüsü

- *ListView* görev menüsünden *Configure ListView* komutunu seçin. Ekranı gelen iletişim penceresinden listenin şeklini ve görünümünü belirleyebilirsiniz.



Resim 3.11: Configure ListView iletişim penceresi

- Sayfayı görüntüleyin. *ListView* kontrolünde veri tabanı *uye* tablosundaki veriler gösterilecektir.

kullanici	sifre
mdyrd	1345
mudur	6666
admin	1234
ogrenci	1111
ogretmen	1221

First 1 Last

Resim 3.12: ListView uygulaması ekran çıktısı

3.4. GridView Kontrolü

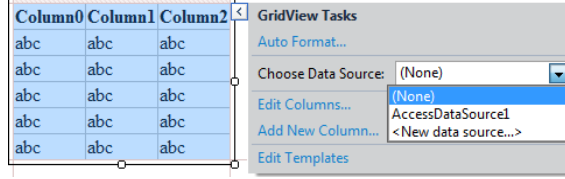
GridView kontrolü ASP.NET yeni sürümleri ile beraber geliştirilen veri bağlama kontrollerinin en gelişmiş versiyonudur. *Repeater* kontrolünden daha gelişmiş bir kontroldür. Veri tabanından alınan verilerin düzenli bir şekilde görüntülenmesini sağlar. *Toolbox* panelinde *Data* sekmesinde yer alır. *GridView* kontrolünün geliştirilmesiyle beraber eski sürümlerde kullanılan bazı kontrollerde kullanımdan kalkmıştır.

GridView kontrolü ile veri tabanından kod yazmadan veriler alınabilir, bu veriler seçilebilir, silinebilir, güncellenebilir ya da bu verilere ekleme yapılabilir.

GridView kontrolü ile verileri bağlamak için;

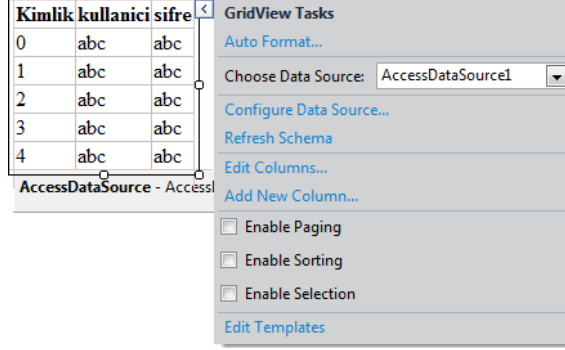
- Yeni bir web sayfası açın.

- Sayfaya bir *GridView* kontrolü ekleyin. *Toolbox* panelinden *Data* sekmesinden ulaşabilirsiniz.
- *GridView* kontrolü görev menüsünde *Choose Data Source* alanından veri tabanı bağlantısı için daha önceden oluşturduğunuz bağlantıyı seçin. Daha önceden bir bağlantı yoksa yeni bağlantı oluşturabilirsiniz.



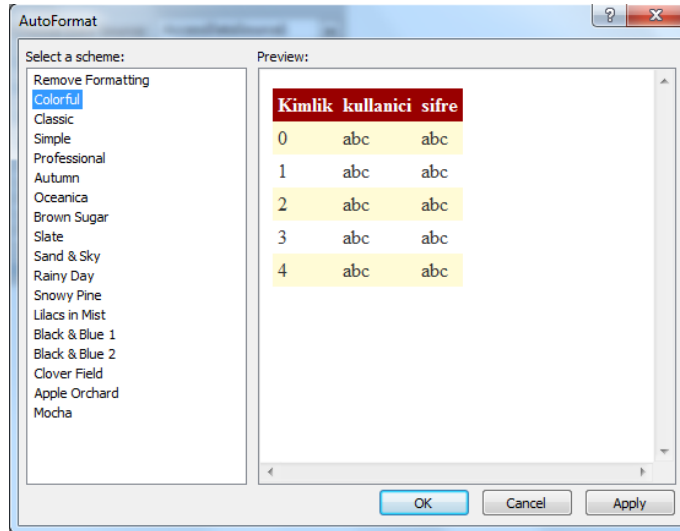
Resim 3.13: GridView kontrolü görev menüsü

- Veri tabanı bağlantısı seçildikten sonra görev menüsünün değiştiği görülecektir.



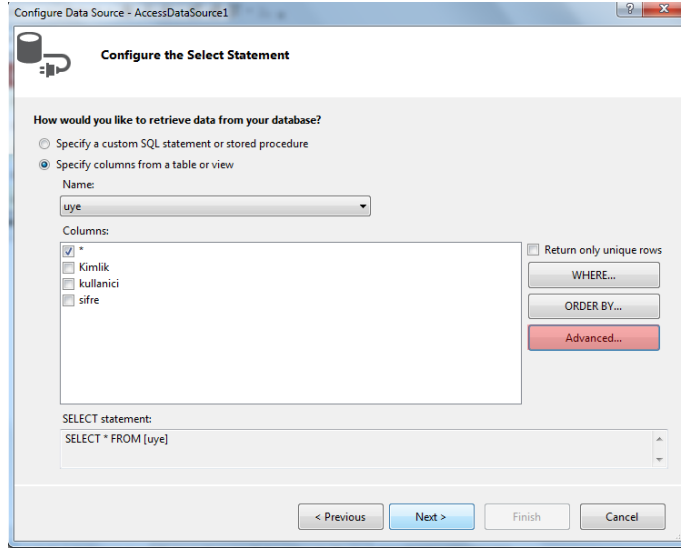
Resim 3.14: GridView kontrolü görev menüsü veri tamamlandıktan sonraki görünümü

- Görev menüsünden *Auto Format* seçeneğinden *GridView* kontrolünün görünümünü değiştirin.



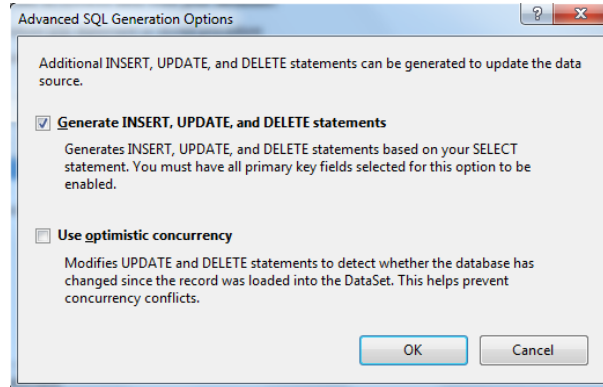
Resim 3.15: GridView kontrolünün görünümünün değiştirilmesi

- *GridView* kontrolü ile verileri güncelleyebilmek ve silebilmek için bağlantı üzerinde bazı değişiklikler yapmanız gerekir. Görev menüsünden *Configure Data Source* seçeneğini ile veri tabanı bağlantısını düzenleyin.
- İlk adımda veri tabanı yeri gösterilmektedir. Bu alanda değişiklik yapmayın.
- İkinci adımda *Advanced* seçeneğini tıklayın.



Resim 3.16: Configure Data Source iletişim penceresi

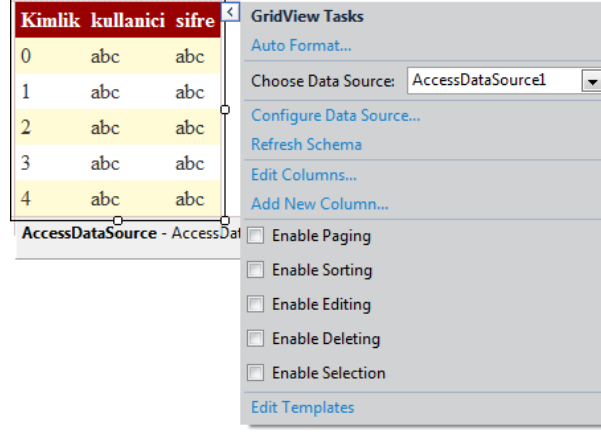
- Ekranı gelen *Advanced SQL Generation Options* iletişim penceresinden *Generate INSERT, UPDATE, and DELETE statements* seçeneğini işaretleyin ve onaylayın.



Resim 3.17: Advanced SQL Generation Options iletişim penceresi

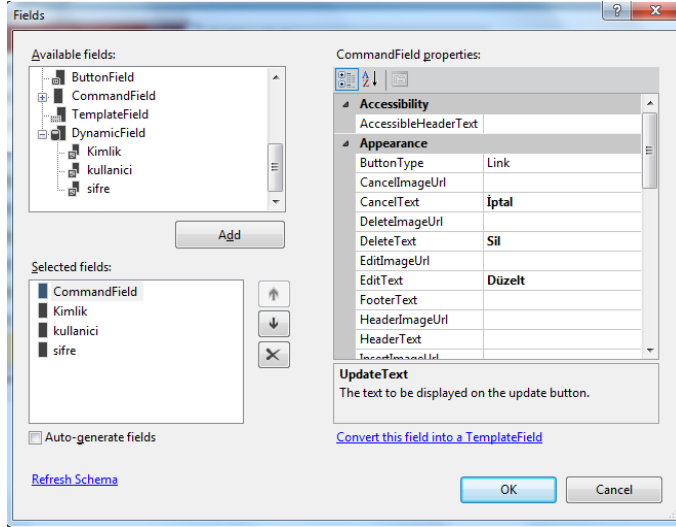
- Bağlantının diğer adımlarını değişiklik yapmadan tamamlayın.

- *GridView* kontrolünün görev menüsü tekrar incelenecek olursa yeni seçeneklerin geldiği görülebilir.



Resim 3.18: GridView kontrolünün görünümünün değiştirilmesi

- Bu seçenekler;
- **Enable Paging:** Verileri sayfalara bölmek için kullanılır.
 - **Enable Sorting:** Verileri başlıklara tıklayarak sıralamak için kullanılır.
 - **Enable Editing:** Verileri düzenleme butonunu göstermek için kullanılır.
 - **Enable Deleting:** Verileri silme butonunu göstermek için kullanılır.
 - **Enable Selection:** Verileri seçme butonunu göstermek için kullanılır.
- Görev menüsünden *Edits Columns* seçeneğini işaretleyin.
- *Field* iletişim penceresinden *CommandField* seçeneğinden butonların metinlerini Türkçe değerleri ile değiştirin.
- Cancel Text = İptal
 - Delete Text = Sil
 - Edit Text = Düzelt
 - Insert Text = Ekle
 - New Text = Yeni
 - Select Text = Seç
 - Update Text = Güncelle
- *Field* iletişim penceresinden *CommandField* seçeneğinden başlıkların da Türkçe değerlerini girebilirsiniz.



Resim 3.19: Fields iletişim penceresi

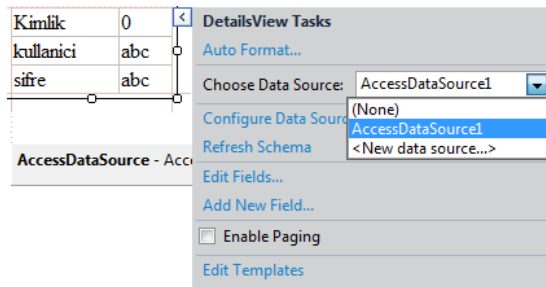
- Uygulamayı çalıştırın ve eklediğiniz butonları deneyin.

3.5. DetailsView Kontrolü

DetailsView kontrolü ASP.NET'te veri bağlama kontrollerinden bir başkasıdır. Bu kontrol ile veri tabanına ekleme yapılabilir, veriler güncellenebilir ya da silinebilir. *Toolbox* panelinde *Data* sekmesinde yer alır.

DetailsView kontrolü ile verileri bağlamak için;

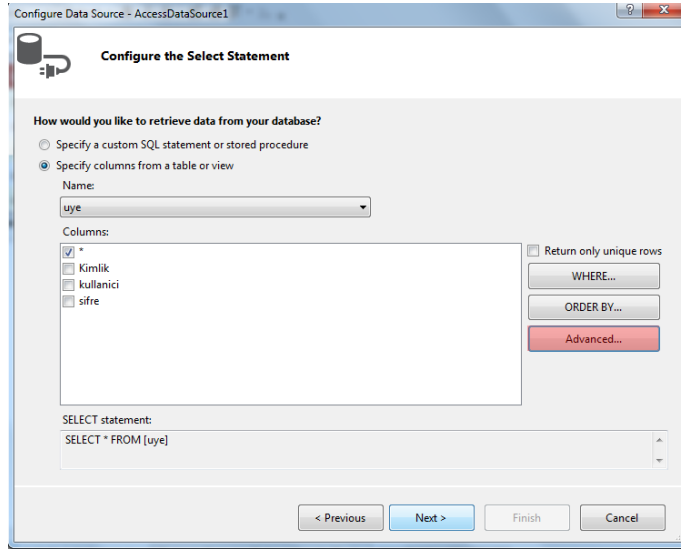
- Yeni bir web sayfası açın.
- Sayfaya bir *DetailsView* kontrolü ekleyin. *Toolbox* panelinden *Data* sekmesinden ulaşabilirsiniz.
- *DetailsView* kontrolü görev menüsünde *Choose Data Source* alanından veri tabanı bağlantısı için daha önceden oluşturduğunuz bağlantıyı seçin. Daha önceden bir bağlantı yoksa yeni bağlantı oluşturabilirsiniz.



Resim 3.20: DetailsView kontrolü görev menüsü

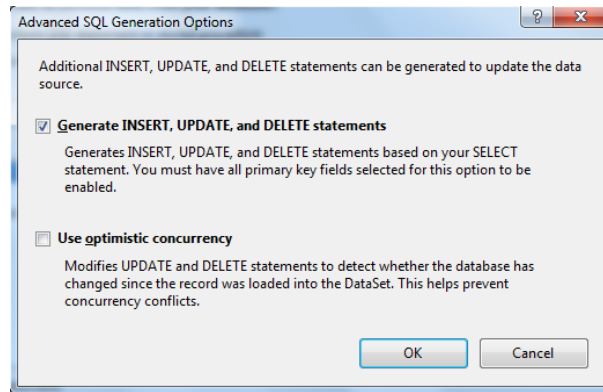
➤ *DetailsView* kontrolü ile verileri güncelleyebilmek ve silebilmek ya da yeni veri ekleyebilmek için bağlantı üzerinde bazı değişiklikler yapmanız gerekir. Görev menüsünden *Configure Data Source* seçeneğini ile veri tabanı bağlantısını düzenleyin.

- İlk adımda veri tabanı yeri gösterilmektedir. Bu alanda değişiklik yapmayın.
- İkinci adımda *Advanced* seçeneğini tıklayın.



Resim 3.21: Configure Data Source iletişim penceresi

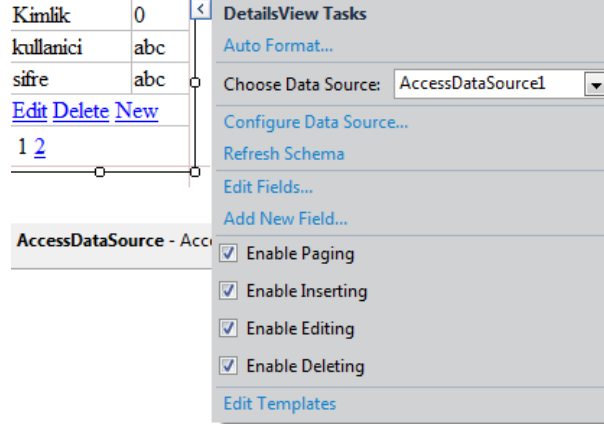
- Ekranı gelen *Advanced SQL Generation Options* iletişim penceresinden *Generate INSERT, UPDATE, and DELETE statements* seçeneğini işaretleyin ve onaylayın.



Resim 3.22: Advanced SQL Generation Options iletişim penceresi

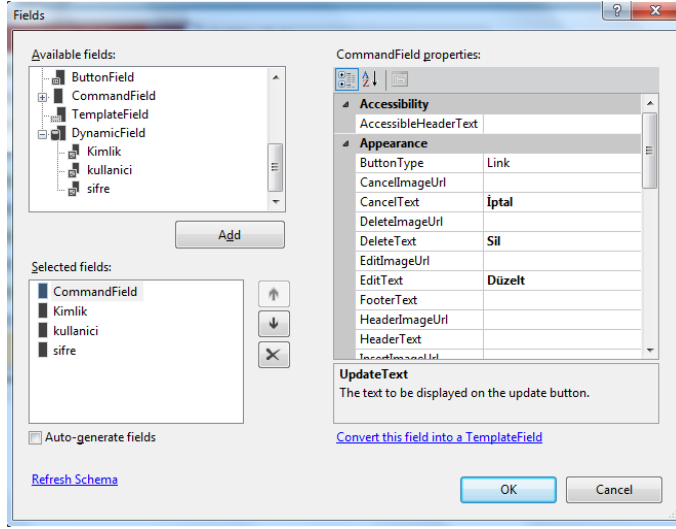
- Bağlantının diğer adımlarını değişiklik yapmadan tamamlayın.

- *DetailsView* kontrolünün görev menüsü tekrar incelenecek olursa yeni seçeneklerin geldiği görülebilir.



Resim 3.23: DetailsView kontrolü görev menüsü veri tabanı bağlantısı yapıldıktan sonraki görünümü

- Bu seçenekler;
 - **Enable Paging:** Verileri sayfalara bölmek için kullanılır.
 - **Enable Inserting:** Verileri ekleme butonunu göstermek için kullanılır.
 - **Enable Editing:** Verileri düzenleme butonunu göstermek için kullanılır.
 - **Enabled Deleting:** Verileri silme butonunu göstermek için kullanılır.
 - **Enable Selection:** Verileri seçme butonunu göstermek için kullanılır.
- Görev menüsünden *Edits Columns* seçeneğini işaretleyin.
- *Field* iletişim penceresinden *CommandField* seçeneğinden butonların metinlerini Türkçe değerleri ile değiştirin.
 - Cancel Text = İptal
 - Delete Text = Sil
 - Edit Text = Düzelt
 - Insert Text = Ekle
 - New Text = Yeni
 - Select Text = Seç
 - Update Text = Güncelle
- *Field* iletişim penceresinden *CommandField* seçeneğinden başlıkların da Türkçe değerlerini girebilirsiniz.



Resim 3.24: Fields iletişim penceresi

- Uygulamayı çalıştırın ve eklediğiniz butonları deneyin.

UYGULAMA FAALİYETİ

Bir önceki uygulama faaliyetindeki uygulamada *GridView* kullanarak verileri görüntüleyin.

İşlem Basamakları	Öneriler																
<p>➤ <i>Default.aspx</i> sayfasına aşağıdaki kontrolleri ekleyin.</p> <p>[Label1] Oturumu Kapat</p>	<p>➤ <i>Toolbox</i> panelini kullanabilirsiniz.</p>																
<p>➤ <i>Default3.aspx</i> sayfasına giriş yapılmadan ulaşılmasını engellemek için <i>Page_Load</i> olayına aşağıdaki kodları ekleyin.</p> <pre>if (Session["KullaniciAdi"] == null) { Response.Redirect("default.aspx"); } Label1.Text = Session ["KullaniciAdi"].ToString();</pre>	<p>➤ Oturum kontrolü için <i>Session</i> nesnesinin değerini kontrol edebilirsiniz.</p>																
<p>➤ <i>Oturumu Kapat</i> butonu <i>Click</i> olayına aşağıdaki kodları ekleyin.</p> <pre>Session.Abandon(); Response.Redirect("Default.aspx");</pre>	<p>➤ <i>Session</i> nesnesi ile oturumu sonlandırabilirsiniz.</p>																
<p>➤ Sayfaya bir <i>GridView</i> kontrolü ekleyin.</p>	<p>➤ <i>Toolbox</i> panelini kullanabilirsiniz.</p>																
<p>➤ <i>GridView</i> kontrolü ile veri tabanı ile bağlantısını kurun.</p>	<p>➤ <i>GridView</i> paneli görev menüsünden <i>Choose Data Source</i> seçeneğini kullanabilirsiniz.</p> <p>➤ Ekleme, silme ve düzeltme için bağlantı ayarlarını yapmayı unutmayınız.</p>																
<p>➤ <i>GridView</i> kontrolüne <i>Düzeltil</i> ve <i>Sil</i> komutlarını ekleyin.</p>	<p>➤ <i>GridView</i> paneli görev menüsünü kullanabilirsiniz.</p> <p>➤ Seçenekleri Türkçe olarak ayarlamak için görev menüsünden <i>Edit Columns</i> seçeneğini kullanabilirsiniz.</p>																
<p>➤ Sayfayı görüntüleyin ve komutları deneyin.</p>	<p>Admin</p> <p>Oturumu Kapat</p> <table border="1"><thead><tr><th></th><th>Kimlik</th><th>kullanici</th><th>sifre</th><th>email</th><th>adi</th><th>soyadi</th><th>il</th></tr></thead><tbody><tr><td>Düzeltil Sil</td><td>1</td><td>Admin</td><td>1234</td><td>admin@site.com</td><td>admin</td><td>admin</td><td>Antalya</td></tr></tbody></table>		Kimlik	kullanici	sifre	email	adi	soyadi	il	Düzeltil Sil	1	Admin	1234	admin@site.com	admin	admin	Antalya
	Kimlik	kullanici	sifre	email	adi	soyadi	il										
Düzeltil Sil	1	Admin	1234	admin@site.com	admin	admin	Antalya										

KONTROL LİSTESİ

Bu faaliyet kapsamında aşağıda listelenen davranışlardan kazandığınız beceriler için **Evet**, kazanamadığınız beceriler için **Hayır** kutucuğuna (X) işareti koyarak kendinizi değerlendiriniz.

Değerlendirme Ölçütleri	Evet	Hayır
1. Oturum kontrolünü sağlayabildiniz mi?		
2. Sayfaya <i>GridView</i> kontrolü ekleyebildiniz mi?		
3. <i>GridView</i> kontrolü ile veri tabanı bağlantısını sağlayabildiniz mi?		
4. <i>GridView</i> kontrolüne <i>Düzeltil</i> ve <i>Sil</i> komutları ekleyebildiniz mi?		

DEĞERLENDİRME

Değerlendirme sonunda “**Hayır**” şeklindeki cevaplarınızı bir daha gözden geçiriniz. Kendinizi yeterli görmüyorsanız öğrenme faaliyetini tekrar ediniz. Bütün cevaplarınız “**Evet**” ise “Ölçme ve Değerlendirme”ye geçiniz.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki cümlelerin başında boş bırakılan parantezlere, cümlelerde verilen bilgiler doğru ise D, yanlış ise Y yazınız.

1. () *Repeater* kontrolü veri tabanında bir tablodaki verileri liste halinde gösterebilir.
2. () *Repeater* kontrolünde verilerin gösterilebilmesi için kod ile verilerin tanımlanması gerekir.
3. () *DataList* kontrolü *Toolbox* panelinde *Standart* sekmesinde yer alır.
4. () *ListView* kontrolünde verilerin gösterilebilmesi için liste şekli ve görünümünün seçilmesi gerekir.
5. () *GridView* kontrolü en gelişmiş veri bağlama kontrolüdür.
6. () *GridView* kontrolü ile veri tabanındaki veriler kod yazmadan değiştirilebilir.
7. () *DetailsView* kontrolü ile veri tabanına yeni veri eklenemez.

DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru ise “Modül Değerlendirme”ye geçiniz.

MODÜL DEĞERLENDİRME

Aşağıdaki soruları dikkatlice okuyunuz ve doğru seçeneği işaretleyiniz.

1. Aşağıdakilerden hangisi veri erişim teknolojilerinden birisi **değildir**?
A) ODBC
B) OLEDB
C) ASP
D) ADO
2. Aşağıdakilerden hangisi veri tabanı ile bağlantı kurmak için kullanılan nesnedir?
A) Connection
B) Command
C) DataReader
D) DataAdapter
3. Aşağıdakilerden hangisi komut nesnesinden gelen verileri okumak için kullanılır?
A) Connection
B) Command
C) DataReader
D) DataAdapter
4. Programlama yazılımında veri tabanı dosyasının hangi klasörde tutulması önerilir?
A) App_Code
B) App_Data
C) App_Browser
D) App_Themes
5. Aşağıdakilerden hangisi bağlantı cümlesinde veri tabanının ismini belirtmek için kullanılır?
A) connectionString
B) Provider
C) Data Source
D) Server.MapPath
6. Veri tabanından verileri seçmek için aşağıdaki komutlardan hangisi kullanılır?
A) Select
B) Insert
C) Update
D) Delete

7. Veri tabanında verileri güncellemek için aşağıdaki komutlardan hangisi kullanılır?
A) Select
B) Insert
C) Update
D) Delete
8. Veri tabanında verileri silmek için aşağıdaki komutlardan hangisi kullanılır?
A) Select
B) Insert
C) Update
D) Delete
9. Veri tabanına veri eklemek için aşağıdaki komutlardan hangisi kullanılır?
A) Select
B) Insert
C) Update
D) Delete
10. Veri tabanında bir tablodaki tüm alanlara ait verileri ifade etmek için hangi sembol kullanılır?
A) ?
B) +
C) /
D) *
11. Aşağıdaki veri bağlama için kullanılanlardan en gelişmiş olanı hangisidir?
A) Repeater
B) ListView
C) GridView
D) DataList
12. GridView kontrolü görev menüsünde yer alan aşağıdaki komutlardan hangisi verileri düzenleme komutunu eklemek için kullanılır?
A) Enable Paging
B) Enable Sorting
C) Enable Editing
D) Enable Deleting
13. GridView kontrolü görev menüsünde yer alan aşağıdaki komutlardan hangisi verileri sayfalara böler?
A) Enable Paging
B) Enable Sorting
C) Enable Editing
D) Enable Deleting

14. GridView kontrolü görev menüsünde yer alan aşağıdaki komutlardan hangisi görünümü değiştirmek için kullanılır?
- A) Auto Format
 - B) Configure Data Source
 - C) Edit Columns
 - D) Edit Templates

DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru ise bir sonraki modüle geçmek için öğretmeninize başvurunuz.

CEVAP ANAHTARLARI

ÖĞRENME FAALİYETİ-1'İN CEVAP ANAHTARI

1	Doğru
2	Doğru
3	Yanlış
4	Doğru
5	Yanlış
6	Doğru
7	Doğru
8	Doğru

ÖĞRENME FAALİYETİ-2'NİN CEVAP ANAHTARI

1	Doğru
2	Doğru
3	Yanlış
4	Yanlış
5	Doğru

ÖĞRENME FAALİYETİ-3'ÜN CEVAP ANAHTARI

1	Doğru
2	Doğru
3	Yanlış
4	Doğru
5	Doğru
6	Doğru
7	Yanlış

MODÜL DEĞERLENDİRMENİN CEVAP ANAHTARI

1	C
2	A
3	C
4	B
5	C
6	A
7	C
8	D
9	B
10	D
11	C
12	C
13	A
14	A

KAYNAKÇA

- SHEPHERD, George, **Microsoft ASP.NET 4.0 Step by Step**, Microsoft Press, Washington, 2010.
- SHARP, John, **Microsoft C# 2008 Step by Step**, Microsoft Press, Washington, 2008.
- ÖZDEMİR, Selçuk, **ASP.NET 2.0**, NET Soft.