* **CRUD Açılımı:** Create 🡪 Oluştur / Read 🡪 Oku / Update 🡪 Güncelle / Delete 🡪 Sil
* Bir veritabanı bağlamında (DBContext) yapılmış olan ve bellekte duran değişikliklerin veritabanına uygulaması (kaydedilmesi) için çağırılan DBContext sınıfı metodu **db.SaveChanges();**
* Bir modelin verilerinde hata olup olmadığını, model durumunun (model state) geçerli olup olmadığını, bir aksiyon metodu içerisinde denetlemek için gereken ifade ModelState kullanılarak **if(ModelState.isValid)** şeklinde yazılır.
* LINQ kısaltmasının açılımı **Language Integrated Query (Dil İlişkili Sorgu)** idir.
* Çoklu dil için kullanılan dosyaların uzantısı **.resx** idir.
* Bir projede “Yoneticiler” isminde bir rol mevcuttur. Bir görünümü (view), yalnızca “Yoneticiler” rolüne sahip kullanıcılarınn erişimine açmak için, ilgili aksiyon metodunun üst tarafına yazılması gereken veri notu (data annotation) **[Authorize(Roles = “Yoneticiler”)]** idir.
* Select \* FROM Personel Order by DogumTarihi, AdSoyad sorgusu LINQ kullanılarak bu şekilde yazılır.

**db.Personeller.OrderBy(x => x.DogumTarihi).ThenBy(x => x.AdSoyad);**

* Bölüm tablosunun ilk 5 sorgusunu elde eden tablo **db.Bolum.Take(5);**
* Personelleri doğum tarihlerine göre büyükten küçüğe olacak şekilde sıralayacak sorgu db.Personeller.OrderByDescending(x => x.DogumTarihi);
* Select \* FROM Bolum WHERE ID NOT IN (1,5,6) sorgusu LINQ kullanılarak bu şekilde yazılır.

**db.Bolum.Where(x => x.ID != 1 && x.ID != 5 && x.ID != 6);**

* App\_Data : Proje içerisinde ki veritabanı vb. veri kaynaklarının, dosyalar halinde eklendiği klasördür.
* App\_Start : Projenin ilk açılışı esnasında (sunucunun ilk başlatılışı ya da yeniden başlatılması), işlem yapması istenilen sınıfların, filtrelerin, metodların bulunduğu klasördür. Örneğin Route, yani sayfa yönlendirme dosyasının ya da projeye eklediğimiz JavaScript, Css gibi dosyaların sıkıştırılarak istemciye gönderilmesi mimarisi olan BundleConfig gibi dosyaları burada barındırırız.
* Content : Bu klasör, zorunlu olmamakla birlikte genelde css dosyaları bu klasörde barındırılabilir.
* Controllers : Controller dosyalarının barındırıldığı klasördür.  
  fonts : [Asp.Net](https://l.facebook.com/l.php?u=http%3A%2F%2FAsp.Net%2F&h=JAQEAwbev) Mvc 5 sürümü ile birlikte, örnek projeye eklenen (bu klasörü kullanmak zorunda değilsiniz), font dosyalarını barındırabileceğiniz alandır.
* Models : Model dosyalarının barındırıldığı klasördür.
* Scripts : Javascript tarzı script dosyalarını barındırabileceğiniz alandır.
* Views : View dosyalarının barındırıldığı klasördür.
* Görünüm (view) dosyalarına yazılan Razor Görünüm Motoru (Razor View Engine) komutlarının başına **@** gelir.
* Bir kontrollerdan ( controller) , görünüm (view) dosyalarına bilgi aktarmak için kullanılan dinamik nesneye **Viewbag** denir.
* ASP.NET MVC projeleri için Visual Studio 2013 tarafından sunulan kimlik doğrulama metodları şu şekildedir.
* Bireysel Kullanıcı Hesapları (Individual User Accounts)
* Organizasyon Hesapları (Organizational Accounts)
* Windows Kimlik Doğrulama (Windows Authentication)
* Kimlik Doğrulama Seçeneği Olmayan (No Authentication)
* Projeye boothstrap vb. kütüphaneleri eklemek ve yönetmek için kullanılan paket yöneticisi **nuget** idir.
* Lambda ifadesinin sembolü **=>** şeklindedir.
* MVC projelerinde veritabanı ayarları (bağlantı karakter dizini) **Web.config** yapılandırma dosyasında gerçekleşir.
* MVC projelerinde yönlendirmeleri düzenlemek için **RouteConfig.cs** dosyası kullanılır.
* Visual Studio 2013 ile varsayılan olarak kurulan SQL Server Express Veritabanı motorunun hafif sürümüne **LocalDB** adı verilir.
* El ile yazılan model ve veritabanı baglam (dbcontext) dosyaları yardımı ile veritabanı , tablolar ve arasındaki tüm ilişkilerin entity fremework tarafından otomatik olarak oluşturulmasını saglayan geliştirme yaklaşımına **Önce Kod (Code First)** adı verilir.
* Asp.Net Mvc'de veri doğrulama için kullanılan ve köşeli parantezler içerisine yazılan ifadelere (örn, [Required]) **Veri Açıklama Notu (Data Annotation)** adı verilir.
* ASP:NET MVC çatısında görünümlerin (view) kaynak kodların saklandıgı dosyaların uzantısı **.cshtml** idir.
* ASP:NET MVC çatısında bir aksiyon (action) metoduna bağlı <a href= “”> oluşturmak için kullanılan Razor komutu **Html.ActionLink**
* ASP.NET MVC çatısında html formları oluşturmaya başlamak için kullanılan razor komutu **Html.BeginForm**
* Bir nesnenin modelin özellikleri için hmtl biçimlendirmeye yapılarını döndüren (Şablon ID veri) razor komutu **Html.DisplayFor**
* Çapraz site istek sahteciliği (Cross site saldırısı) ataklarından korunmak için controller içerisinde ilgili aksiyom metodlarının başında **ValidateAntiForgeryToken** ifadesi yer alır.
* Controller içinde veritabanı ile işlem sonlandığında veritabanı bağlantısının kapatılması için kullanılan metod **dispose**
* Güçlü tipli görünüm (Strong Type View) dosyasındaki kaynak kodlar **Model** anahtar kelimesi ile başlar.
* Yorum satırı yazmak için kullanılan razor sembolleri **@\* \*@** şeklindedir.
* Yerel veritabanı kullanılırken oluşturulan .mdf veritabanı dosyası Asp.Net MVC projesinde **App\_Data** içerisinde yer alır.
* .Net MVC uygulamasında web tarayıcıları adres çubuğuna adres girildiğinde çağırılan controller action metodlarına verilen genel isim **Aksiyon metodu**
* Çoğu SQL komutlarına benzer ya da aynı olan C# anahtar kelimelerinden oluşan ifadeler ile .NET koleksiyonlarına LINQ sorguları yazılır. Bu anahtar kelimeler, **LINQ Standart Sorgu Operatörleri (LINQ Standart Query Operators)** olarak anılır. Bu operatörler **System.Linq** isim alanı altında yer alan **Enumerable** sınıf tarafından tanımlanır.