**WEB PROGRAMLAMA GENEL NOTLAR**

**ASP.Net**

* Microsoft tarafından geliştirilmiş bir web uygulama gelişimi teknolojisidir.
* .Net Framework’ün bir parçasıdır.
* C# veya VBasic’den birini kullanabilir.
* Ortak dil çalışma zamanı altyapısına dayalı çalışır.
* Yani kısaca, yazılımcı tarafından yazılan kod, çalıştırılmadan önce kanal bir yazılım katmanı tarafından ortak bir dile çevrilmektedir.

**ASP.Net Web Form’un Temel Özellikleri**

* ASP.Net Code Behind denilen tasarım öğeleri ile programlama öğelerini birbirinden ayrı sayfalarda bulunduran bir özelliği destekler. Bu özellik sayesinde bir proje üzerinde tasarımcı ve programcı aynı anda çalışabilir.
* Tasarım öğeleri (Web Forms için) **.aspx** uzantılı dosyada bulunurken, program nesneleri **.aspx.cs** (C# için **.cs**, VB.Net için **.vb**) uzantılı dosyalarda bulundurulur.

**ASP.Net Formlarının Dezavantajları**

**ViewState**

* ASP.Net sayfalarında form, kullanıcı bilgileri, durumları tutması ve sunucu ile kullanıcı arasındaki postback’in kaybolmasını sağlar. Her defasında sunucu tarafında ciddi bant genişliği harcarken, kullanıcı tarafında ise sayfanın açılışı ciddi yavaşlamalara sebep oluyor.

**Page Life Cycle**

* Çok karmaşık yapıda olduğundan çıkacak hataları bulmamız oldukça zordur.

**Serseri HTML Çıktıları**

* Burada hakimiyetimiz oldukça zordur. İstediğimiz gibi bir html çıktısı alamayabiliriz. Görüntü olarak istediğimiz html dosyasını alsak dahi, kod temizliği ve W3 standartlarına uygun çıktı almak çok zordur.

**Aşırı Code-Behind Bağımlılığı**

* Sunucu ile iş katmanlarını ayırmak istediğimizde bu ASP.Net için pek mümkün değildir. Çünkü Code-Behind tarafında yazmak zorunda kalırız. Katmansal mimarisi yoktur. Projemizi daha sonra güncellemek veya eklentiler yapmak istediğimizde daha da zor olacaktır çünkü dağınık bir kod yapısı vardır.

**RAZOR**

* Yeni bir gramere sahip görünüm moturudur. Yeni gramerde bildiğimiz html vb. C# veya VBasic kullanılır. Kısacası Razor sunucu tabanlı kodu web sayfalarına eklemek için kullanılan az yer kaplayan anlamlı ve akıcı bir yapıya sahip olan görünüm motorudur.

**MVC Nedir? (Model View Controller)**

* MVC bir yazılım değil, bir mimaridir.
* MVC kavramı Microsoft’a ait bir mimari ürün değildir.

**MVC Mimarisi**

**Model**

* Veri kaynağının (veritabanı) genelde bulunduğu yerdir. Bunun yanı sıra, prosedürler ve işleyiş kuralları da bu bölümdedir. Katmanlı mimariye müthiş uyumludur. Tek katmanlı olabileceği gibi fazlaca katmana da sahip olabilir. **(.cs)**

**View**

* Arayüze ait olup, kullanıcının gördüğü şeyler bu bölümde yer alır. Html, JavaScript, css kodları bu burada yer alır. **(.cshtml)**

**Controller**

* İş akışının gerçekleştiği, arayüzden gelen kullanıcı etkileşimlerinin değerlendirildiği, işlendiği, gerekli metotların çalıştırıldığı, değişkenlerin ve nesnelerin oluşturulduğu, gerekirse Model ile View bölümleri arasındaki iletişimin sağlandığı yerdir. **(.cs)**

**Özetle**

* Client tarafından başlatılan request, controller tarafında işleme alınıyor ve neler yapılması gerekiyorsa yapıyor, sonra tekrar controller’a geliniyor ve bizim response dediğimiz sunucu cevabı Html çıktısı olarak bize geliyor.
* Her View için bir Controller vardır fakat her Controller için View şartı yoktur.

**LAMBDA İFADELERİ**

Değişkenlere değer atamak için kullanılan sadeleştirilmiş anonim fonksiyonlardır.

* Tek satır halindeki Lambda ifadelere Expression (İfade) Lambdaları denir.

**Örneğin**, n=>n\*n veya (int n)=>n\*n

* Lambda otomatik olarak tip belirleyebilir, bundan dolayı “int” yazmaya gerek yoktur.

**Örneğin,**

n=>

{

String s=”Merhaba”+n;

Console.WriteLine (s);

};

* İfadenin sağ tarafında süslü parantezler ve sonunda da noktalı virgül kullanılarak çok satırlı Lambda ifadesi yazılabilir. Böyle yazılan ifadelere Statement (Komut) Lambdaları denir.

**Örneğin,** ( ) => afonksiyonu ( )

* Lambda ifadelerinde parametre zorunlu değildir.

**Örneğin,** (a,b) => a>b

* Parametreler birden fazla olursa parantez içinde yazılır.

**LINQ Komutları (LINQ=Language Integrated Query)**

**Aggregate**

* Bir veri kümesindeki tüm elemanları sırasıyla belirlenen bir işleme tabi tutarak bir yığın değer elde etmek için 2 parametreli bir Lambda ifadesi ile kullanılır. 1. Parametresi yığın değişkenin son değerini, 2. Parametresi ise sıradaki elemanın referansını barındırır.

**All**

* Tüm elemanların belirli şartı sağlayıp sağlamadığını kontrol eder.

**Any**

* Kümede herhangi bir eleman olup olmadığını kontrol eder.

**Avarage**

* Sayısal değerler dizisi ortalamasını hesaplar.

**Concat**

* İki diziyi birleştirir.

**Contains**

* Bir dizinin belirtilen öğeyi içerip içermediğini denetler.

**Count**

* Bir dizideki öğelerin sayısını döndürür.

**DefaultIfEmpty**

* Boş ise varsayılanı getirir.

**Distinct**

* Aynı elemanları içermeyen bir dizi döndürür.

**Empty**

* Boş döndürür.

**Except**

* İki dizeyi kümesi farkının değerlerinin karşılaştırmak için kullanılır.

**First**

* Belirtilen bir koşula uygun bir dizideki ilk öğeyi döndürür.

**FirstOrDefault**

* Öyle bir öğe bulunursa karşılayan bir koşul veya varsayılan değeri dizinin ilk öğesini döner.

**GroupBy**

* Gruplandırma işlemi

**GroupJoin**

* İki diziyi anahtarları eşitlik bakımından temel öğelerinin belirtilirler ve sonuçları gruplandırır.

**Intersect**

* Set iki dizinin kesişimin değerleri karşılaştırarak üretir.

**Join**

* Bir anahtar veri üzerinden veri kümelerini birleştirme.

**Last**

* Son öğe belirtilen bir koşula uygun bir dizi döndürür.

**LastOrDefault**

* Öyle bir öğe bulursa karşılayan bir koşul veya varsayılan değeri son öğeye döndürür.

**LongCount**

* Sunucu long tipinde veren Count. Normal Count int’tir.

**Max**

* Her öğenin bir dizi dönüştürme işlevinin çağırır ve en büyük Single değer döner.

**Min**

* Bir dizideki en küçük Decimal değeri verir.

**OrderBy**

* Bir anahtara göre bir dizinin öğelerini artan sırada sıralar. (A-Z)

**OrderByDescending**

* Bir anahtara göre bir dizinin öğelerini azalan sırada sıralar. (Z-A)

**Range**

* Tamsayı numaraları belirtilen aralık içinde bir dizi oluşturur.

**Repeat**

* Bir yenilenen değer içeren dizi oluşturur.

**Reverse**

* Bir dizideki öğelerin sırasını ters çevirir.

**Select**

* Veri kümesi için üye seçimi.

**SelectMany**

* Veri kümesi için kolon seçimi.

**SequenceEqual**

* Kendi türü için varsayılan eşitlik karşılaştırıcıyı kullanarak öğeleri karşılaştırarak iki diziyi eşit olup olmadığını belirler.

**Single**

* Tek öğe, belirtilen bir koşula uygun ve birden fazla öğe varsa, aykırı bir dizi döndürür.

**Skip**

* Varsayılan sayı kadar elemanı atlar.

**SkipWhile**

* Belirtilen bir koşul doğrusunda geriye kalan öğeleri döndürür.

**Sum**

* Bir dizi toplamını (Decimal değerleri) hesaplar.

**Take**

* Belirtilen sayıda elemanı alır.

**TakeWhile**

* Belirtilen koşul true olduğu sürece öğeleri bir dizi döner.

**ThenBy**

* Bir sonraki öğelerin bir anahtara göre artan sırada sıralamasını gerçekleştirir.

**ToArray**

* Sonuç diziye aktarılır.

**ToDictionary**

* Sonuçları anahtar değer çifti şeklinde döndürür.

**ToList**

* Sonuçları listeler.

**ToLookUp**

* Belirli ortak özelliğe sahip veya isteğimize göre gruplandırdığımız nesnelere bir tek nesne kullanarak erişmeyi sağlar.

**Union**

* İki diziyi kümesinin birleşimini üretir.

**Where**

* Kısıtlama operatörü.

**Zip**

* Her iki kümenin elemanlarını belirlenen fonksiyona iletir.

**LINQ Sorguları**

Select \* from dbo.tbl\_Dept (Listeleme)

* db.tbl\_Dept.ToList();

Select top(2) \* from tbl\_Dept (Listeden 2 değer alma)

* db.tbl\_Dept.Take(2).ToList();

Select \* from tbl\_Dept order by Did (Did’e göre sıralama)

* db.tbl\_Dept.OrderBy(x=> x.Did).ToList();

Select \* from tbl\_Dept where Did=4 or Did=7 (Did’in 4 veya 7’ye göre sıralanması)

* db.tbl\_Dept.Where(x=> x.Did==4 || x.Did==7).ToList();

İçinde 1, 5 ve 6 numaralı ID’leri olmayan bölümleri listeleme

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

**Kimlik Doğrulama Çeşitleri**

* Bireysel Kullanıcı Hesapları
* Şirket Hesabı
* Windows Authentication
* Kimlik Doğrulama Yok

**CRUD**

* Create-Read-Update-Delete

**Bellekte duran değişikliğin veritabanına kaydedilmesi**

* SaveChanges()

**Yöneticilere izin vermek için gereken veri notu**

* [Authorize(Roles=’Yoneticiler’)]

**ModelState hata kontrolü**

* If (ModelState.IsValid)

**HTML form oluşturmaya başlamak için kullanılan Razor komutu**

* Html.BeginForm

**HTML biçimlendirme yapılarını döndüren Razor komutu**

* Html.DisplayFor

**HTML aksiyon metoduna bağlantı sağlayan Razor komutu**

* Html.ActionLink

**Cross site saldırısı önleyen veri açıklama notu**

* ValidateAntiForgeryToken

**Çoklu dil için kullanılan dosyaların uzantısı nedir?**

* .resx

**Veritabanı ayarları dosyası**

* Webconfig

**WEB PROGRAMLAMA TEST SORU-CEVAP**

Görünüm (view) dosyalarına yazılan Razor Görünüm Motoru (Razor View Engine) komutlarının başına ne konur?

**@**

Bir kontrolcüden (controller), görünüm (view) dosyalarına bilgi aktarmak için kullanılan dinamik nesneye ne ad verilir?

**ViewBag**

ViewBag hangi tipte bir nesnedir?

**Dinamik**

Aşağıdakilerden hangisi ASP.NET MVC projeleri için Visual Studio 2013 tarafından sunulan kimlik doğrulama metotlarından bir tanesi değildir?

**Tekli kimlik doğrulama (single authentication)**

Projenizde bootstrap vb. kütüphaneleri eklemek ve yönetmek için kullandığımız Visual Studio ile entegre gelen paket yöneticinin adı?

**nuget**

Lambda ifadesinin (lambda expression) sembolü̈ aşağıdakilerden hangisidir?

**=>**

ASP.NET MVC projelerinde yönlendirmeleri düzenlemek için hangi dosya kullanılır?

**RouteConfig.cs**

Visual Studio 2013 ile varsayılan olarak kurulan SQL Server Express Veritabanı motorunun hafif sürümüne ne ad verilir?

**LocalDB**

El ile yazılan model ve veritabanı bağlam (dbcontext) dosyaları yardımı ile veritabanı, tablolar ve arasındaki tüm ilişkilerin entity framework tarafından otomatik olarak oluşturulmasını sağlayan geliştirme yaklaşımına ne ad verilir?

**Önce Kod (Code First)**

Asp.Ne​t​ MVC'de veri doğrulama için kullanılan ve köşeli parantezler içerisine yazılan ifadelere (örn, [Required]) ne ad verilir?

**Veri açıklama notu (Data annotation)**

ASP:NET MVC çatısında görünümlerin (view) kaynak kodların saklandığı dosyaların l nedir?

**cshtml**

HTML’de yorum satırı nasıl oluşturulur?

**<!-- -->**

Controller içerisinde veri tabanı ile işlem sonlandığında veri tabanı bağlantısının kapatılması gerekmektedir bunun için hangi metot kullanılır?

**Dispose**

Güçlü tipli görünüm (strong type view) dosyasındaki kaynak kodlar aşağıdaki anahtar kelimelerden hangisi ile başlar?

**Model​**

Bir görünüm içerisinde yorum satırı oluşturmak için Razor sembolleri aşağıdakilerden hangi?

**@\* \*@**

Yerel veri tabanı ile çalışırken kullanılan .mdf veritabanı dosyası Asp.Net MVC projesinde hangi klasör içerisinde yer alır?

**App\_Data**

.Net MVC uygulamasında web tarayıcıları adres çubuğuna adres girildiğinde çağırılan controller action metotlarına verilen genel isim nedir?

**Aksiyon Metodu**

Aşağıdakilerden hangisi doğru bir helper kullanımı değildir?

**@Html.CommentFor(m => m.Content)**

Aşağıdakilerden hangisi “\_Footer” isimli Partial View çıktısını ekrana basar?   
**@Partial(“\_Footer”)**

Versiyon kontrol sistemi (git, svn) kullanmanın amacı aşağıdakilerden hangisi değildir?

**Birden fazla versiyon aynı anda yönetilebilir.**

LINQ için aşağıdakilerden hangisi yanlıştır?

**Web Api’de kullanılmaz.**

MVC’de Razor’un kullanım amacı nedir?

**View’de Sunucu taraflı kod yazabilmek**

Aşağıdaki dosyalardan hangisi ile routing tanımlamaları yapılır?

**/App\_Start/RouteConfig.cs**

ASP.Net MVC de varsayılan route sıralaması aşağıdakilerden hangisidir?

**Controller / Action / Parameter**

Bir web projesinde stiller hangi dosyalarda saklanır?

**.js (javascript)**

**.cs (controller ve model dosyaları)**

**.css (stil dosyaları)**

**.cshtml (view dosyaları)**

**.resx (çoklu dil dosyaları)**

Bir Controller içindeki Action için sadece “moderator” rolündekilerin erişebilmesi için aşağıdaki filtrelerden hangisi eklenmelidir?

**[Authorize(Roles=”moderator”)]**

Aşağıdakilerden hangisi ASP.Net MVC için yanlıştır?

**“AllowAnonymous” sadece giriş yapmayan kullanıcıların ilgili Action’ı çalıştırabilmesini sağlar.**

ModelState.IsValid ifadesi için aşağıdakilerden hangisi doğrudur?

**Gerekli doğrulamalardan geçtiği takdirde (boş geçilemez, doğru format vb.) true olacak ve içerisinde yazacağımız işlemleri yapacak. Doğrulama olmadığı takdirde aynı sayfa ekrana tekrar yazdırılacak.**

Aşağıdakilerden hangisi form güvenliğinin sağlanması için gizli bir anahtar oluşturur?   
**@Html.AntiForgeryKey()**

Bir MVC uygulamasında istemciden gelen ilk istek nereye düşer?   
**Controller**

Aşağıdakilerden hangisi Controller da yazılmış̧ bir metodun adresten doğrudan çağrılmasını engelleyen bir Attribute’dur?

**ValidateAreForgeryTokenAttribute**

ASP.Net MVC’de varsayılan çoklu dil desteği için aşağıdakilerden hangisi doğrudur?

**Web.Config üzerinde ayarlamalar yapılmalıdır.**

ASP.Net Web Api için aşağıdaki hangi ifadeler yanlıştır?

**View kullanılır.**

Bir ASP.Net MVC sayfası için aşağıdakilerden hangisi yanlış bir ifadedir?

**Model zorunludur.**

Aşağıdakilerden hangileri ASP.NET MVC Partial View için doğrudur?

1. **Bakımı kolaydır.**
2. **Yeniden kullanılabilirliği arttırır.**
3. **Sayfa boyutunu azaltır.**

Aşağıdaki ifadelerden hangisi yanlıştır? (CF: Code First, DF: Database First)

**CF ve DF yaklaşımlarının her ikisinde de bir DbContext vardır.**

Bir layout sayfasının tüm sayfalarda ön tanımlı olması için aşağıdaki dosyaların hangisinde tanımlama yapılmalıdır?

**/Views/Shared/\_Layout.cshtml**

Aşağıdakilerden hangisi form güvenliğinin sağlanması için gizli bir anahtar oluşturur?

**@Html.AntiForgeryToken()**