Description

This is my own preparation guide for the MCSD - Exam 70-486: Developing ASP.NET MVC 4 Web Applications, I'm sharing this document because it can be useful for someone else.
Any Tips or Feedbacks are welcome.

If you haven't taken the MCSD - Exam 70-487 - Developing Windows Azure and Web Services exam yet, check out my 70-487 guide.

Summary

Study Strategy

- Read MVC 4 Books such as: Programming ASP.NET MVC 4 by ofps.oreilly and ASP.NET MVC 4 in Action and the official Microsof press book: Exam Ref 70-486: Developing ASP.NET MVC 4 Web Applications
- Watch Pluralsight Videos such as: Building Applications with ASP.NET MVC 4, ASP.NET MVC 4 Fundamentals , ASP.NET MVC Advanced Topics,Windows Azure: The Big Picture
- Read MCPD_70_519_Web_Applications Using.NET Framework 4 Book, this book is for MCPD, however there are many topics in this book that match with the subjects of MCSD Certification.
- Study items from "What's new from 70x515 to 70x516" section of this guide, specially the azure links.
- Review items from "Already Learned on 70-480" section of this guide.
- Take a second shot token
- Set the appointment for the exam(Prometric), using the second shot voucher.
- Get the certification :)

What's New from 70x515 to 70x486?

- Design a distributed application (AZURE)
- Design and implement the Windows Azure role life cycle.
- Design and implement a Web Socket strategy.
- Compose the UI layout of an application.(MVC 4)
- Reduce network bandwidth(bundle,Azure CDN etc..)
- Test a web application (TDD,Mocks)
- Debug a Windows Azure application.
- Design and implement claims-based authentication across federated identity stores.
- Implement a secure site with ASP.NET.

Already Learned on 70-480 (HTML 5,CSS3, Javascript,Jquery, Ajax,DOM)

- Apply the user interface design for a web application.
- Design and implement UI behavior.
- Enhance application behavior and style based on browser feature detection.
- Plan an adaptive UI layout.
- Plan for search engine optimization and accessibility.

[TOP]

Skills Being Measured

Design the Application Architecture

- **Plan the application layers.**
  - This objective may include but is not limited to: plan data access; plan for separation of concerns; appropriate use of models, views, and controllers; choose between client-side and server side processing; design for scalability

- Understanding ASP.NET MVC (Model View Controller) Architecture for Beginners
- ASP.NET MVC Overview

Design a distributed application.

- This objective may include but is not limited to: design a hybrid application (on premise vs. off premise, including Windows Azure); plan for session management in a distributed environment; plan web farms
  - http://www.windowsazure.com/en-us/develop/net/fundamentals/intro-to-windows-azure/
  - https://www.microsoftvirtualacademy.com/tracks/introduction-to-windows-azure
  - .NET On-Premises/Cloud Hybrid Application Using Service Bus Relay

Design and implement the Windows Azure role life cycle.

- This objective may include but is not limited to: identify and implement Start, Run, and Stop events;identify startup tasks (IIS configuration [app pool], registry configuration, third-party tools)
  - Real World: Startup Lifecycle of a Windows Azure Role
  - MSDN – RoleEntryPoint Methods
  - Role Startup Life Cycle
  - http://brentdacodemonkey.wordpress.com/2011/09/24/leveraging-the-roleentrypoint-year-of-azure-week-12/
  - http://blogs.msdn.com/b/windowsazure/archive/2011/01/04/responding-to-role-topology-changes.aspx

Configure state management.

- This objective may include but is not limited to: 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)
  - ASP.NET Session State
  - ASP.NET State Management Overview
  - SessionStateMode Enumeration
  - Session-State Modes
  - ASP.NET View State Overview
  - ASP.NET Cookies Overview
  - ASP.NET State Management Recommendations
- *Examples*
  - **Web Storage / LocalStorage HTML5+JS**

Design a caching strategy.

- This objective may include but is not limited to: implement page output caching (performance oriented); implement data caching; implement HTTP caching
  - http://ofps.oreilly.com/titles/9781449320317/ch_Caching.html
  - How To Control Page Output Caching in ASP.NET by Using Visual C# .NET
  - ASP.NET Page Output Cache
  - Caching in ASP.Net
  - ASP.NET Caching: Techniques and Best Practices
  - OutputCacheAttribute Class (System.Web.Mvc)
  - Improving Performance with Output Caching (C#)
  - https://www.windowsazure.com/en-us/develop/net/how-to-guides/cache/

- **Design and implement a Web Socket strategy.**
  - This objective may include but is not limited to: 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
    - ASP.NET 4.5 - Support for WebSockets Protocol
    - The Web Sockets API
    - Web Socket.org
    - WebSockets in .NET 4.5: a simple game
    - Web Socket Interface
    - http://www.asp.net/mvc/tutorials/mvc-4/using-asynchronous-methods-in-aspnet-mvc-4

- **Design HTTP modules and handlers.**
  - This objective may include but is not limited to: implement synchronous and asynchronous modules and handlers; choose between modules and handlers in IIS
    - Support for await and Task-Based Asynchronous Modules and Handlers
    - HTTP Handlers and HTTP Modules Overview
    - Code Guru: HTTP Handlers and HTTP Modules in ASP.NET
    - Custom HttpModule Example

[TOP]

Design the User Experience

- **Apply the user interface design for a web application.**
  - This objective may include but is not limited to: create and apply styles by using CSS; structure and layout the user interface by using HTML; implement dynamic page content based on a design
    - http://www.w3schools.com/css/default.asp
    - http://www.w3schools.com/html/default.asp
    - http://slides.html5rocks.com/#landing-slide
    - http://www.html5rocks.com/en/
    - http://html5doctor.com/
    - MSDN - HTML5

- **Design and implement UI behavior.**
  - This objective may include but is not limited to: implement client validation; use JavaScript and the DOM to control application behavior; extend objects by using prototypal inheritance; use AJAX to make partial page updates; implement the UI by using JQuery
    - Client-Side Development
    - http://ofps.oreilly.com/titles/9781449320317/ch_AJAX.html
    - http://docs.jquery.com/Plugins/Validation
    - http://www.the-art-of-web.com/html/html5-form-validation/
    - http://www.w3schools.com/html/html5_form_attributes.asp
    - http://www.w3schools.com/js/default.asp
    - http://jquery.com/
    - http://www.w3schools.com/jquery/default.asp
    - http://phrogz.net/JS/classes/OOPinJS.html
    - http://phrogz.net/JS/classes/OOPinJS2.html

  - *Examples*
    - **Jquery Ajax**

```
$.ajax(
{
    type: "GET",
    url: '@Url.Content("~/Home/GetCars")',
    //data: "{}",
    contentType: "application/json; charset=utf-8",
    dataType: "json",
    success: function (data) {
        $.each(data, function (i, theItem) {
            var combo = document.getElementById("ComboCars");
            var option = document.createElement("option");
            option.text = theItem.Name;
            option.value = theItem.Vin;
            try {
                combo.add(option, null); // Other browsers
            }
            catch (error) {
                alert('error found');
                combo.add(option); // really old browser
            }

        });
    },
    error: function (msg, url, line) {
        alert('msg = ' + msg + ', url = ' + url + ', line = ' + line);

    }
});
```

- **Jquery Get**

Example: *Request the test.php page, but ignore the return results.*

```
$.get("test.php");
```

Example: *Request the test.php page and send some additional data along (while still ignoring the return results).*

```
$.get("test.php", { name: "John", time: "2pm" } );
```

Example: *pass arrays of data to the server (while still ignoring the return results).*

```
$.get("test.php", { 'choices[]': ["Jon", "Susan"]} );
```

Example: *Alert out the results from requesting test.php (HTML or XML, depending on what was returned).*

```
$.get("test.php", function(data){
alert("Data Loaded: " + data);
});
```

Example: *Alert out the results from requesting test.cgi with an additional payload of data (HTML or XML, depending on what was returned).*

```
$.get("test.cgi", { name: "John", time: "2pm" },
   function(data){
     alert("Data Loaded: " + data);
   });
```

Example: *Gets the test.php page contents, which has been returned in json format (<?php echo json_encode(array("name"=>"John","time"=>"2pm")); ?>), and adds it to the page.*

```
$.get("test.php",
   function(data){
     $('body').append( "Name: " + data.name ) // John
              .append( "Time: " + data.time ); //  2pm
   }, "json");
```

- **Jquery Post**

Example: *Post a form using ajax and put results in a div*

```
<!DOCTYPE html>
<html>
<head>
  <script src="http://code.jquery.com/jquery-latest.js"></script>
</head>
<body>
  <form action="/" id="searchForm">
   <input type="text" name="s" placeholder="Search..." />
   <input type="submit" value="Search" />
  </form>
  <!-- the result of the search will be rendered inside this div -->
  <div id="result"></div>

<script>
  /* attach a submit handler to the form */
  $("#searchForm").submit(function(event) {

    /* stop form from submitting normally */
    event.preventDefault();

    /* get some values from elements on the page: */
    var $form = $( this ),
        term = $form.find( 'input[name="s"]' ).val(),
        url = $form.attr( 'action' );

    /* Send the data using post and put the results in a div */
    $.post( url, { s: term },
      function( data ) {
          var content = $( data ).find( '#content' );
          $( "#result" ).empty().append( content );
      }
    );
  });
</script>

</body>
</html>
```

- **Extend objects by using prototypal inheritance**

```javascript
//Define a functional object to hold persons in javascript
var Person = function (name) {
    this.name = name;
};

//Add dynamically to the already defined object a new getter
Person.prototype.getName = function () {
    return this.name;
};

//Create a new object of type Person
var john = new Person("John");

//Try the getter
alert(john.getName());

//If now I modify person, also John gets the updates
Person.prototype.sayMyName = function () {
    alert('Hello, my name is ' + this.getName());
};

//Call the new method on john
john.sayMyName();
```

Until now I've been extending the base object, now I create another object and then inheriting from Person.

```
//Create a new object of type Customer by defining its constructor. It's not
//related to Person for now.
var Customer = function (name) {
    this.name = name;
};

//Now I link the objects and to do so, we link the prototype of Customer to
//a new instance of Person. The protype is the base that will be used to
//construct all new instances and also, will modify dinamically all already
//constructed objects because in Javascript objects retain a pointer to the
//prototype
Customer.prototype = new Person();

//Now I can call the methods of Person on the Customer, let's try, first
//I need to create a Customer.
var myCustomer = new Customer('Dream Inc.');
myCustomer.sayMyName();

//If I add new methods to Person, they will be added to Customer, but if I
//add new methods to Customer they won't be added to Person. Example:
Customer.prototype.setAmountDue = function (amountDue) {
    this.amountDue = amountDue;
};
Customer.prototype.getAmountDue = function () {
    return this.amountDue;
};

//Let's try:
myCustomer.setAmountDue(2000);
alert(myCustomer.getAmountDue());
```

While as said I can't call setAmountDue(), getAmountDue() on a Person.

```
//The following statement generates an error.
john.setAmountDue(1000);
```

- **Parse a JSON String**

Parse a JSON string.

```
var obj = jQuery.parseJSON('{"name":"John"}');
alert( obj.name === "John" );
```

- **Compose the UI layout of an application.**
  - This objective may include but is not limited to: implement partials for reuse in different areas of the application; design and implement pages by using Razor templates (Razor view engine); design layouts to provide visual structure; implement master/application pages
    - The Razor View Engine Basics
    - Oreilly - Chapter 15. Reusable UI Components
    - Templates with razor - HTML HELPERS
    - Master page in MVC 4 web application using VS11

- Render Action X Render Partial
- Html Partial vs Html RenderPartial

- *Examples*
  - **Master Page/Layout.cshtml**

```
<!DOCTYPE html>

<html lang="en">
    <head>
        <meta charset="utf-8" />
        <title>@View.Title</title>
    </head>
    <body>
        <div class="header">
            @RenderSection("Header")
        </div>

        @RenderBody()

        <div class="footer">
            @RenderSection("Footer")
        </div>
    </body>
</html>
```

- **Enhance application behavior and style based on browser feature detection.**
  - This objective may include but is not limited to: detect browser features and capabilities; create a web application that runs across multiple browsers and mobile devices; enhance application behavior and style by using vendor-specific extensions, for example, CSS
    - Detect HTML5 & CSS 3 features in your ASP.NET Web Forms, MVC, or Razor Pages with Modernizr
    - Browser and feature detection: Make your website look great everywhere
    - http://msdn.microsoft.com/en-us/magazine/hh475813.aspx
    - Detecting devices and their features
    - http://modernizr.com/
    - https://developer.mozilla.org/en-US/docs/CSS/Media_queries
    - http://css-tricks.com/css-media-queries/

- **Plan an adaptive UI layout.**
  - This objective may include but is not limited to: plan for running applications in browsers on multiple devices (screen resolution, CSS, HTML); plan for mobile web applications
    - http://www.asp.net/mvc/tutorials/mvc-4/aspnet-mvc-4-mobile-features
    - http://ofps.oreilly.com/titles/9781449320317/ch_Mobile.html
    - http://www.codeproject.com/Articles/472005/Techniques-for-Mobile-friendly-ASP-NET-MVC-4-0-web
    - Develop Hybrid Native and Mobile Web Apps

[TOP]

Develop the User Experience

- **Plan for search engine optimization and accessibility.**
  - This objective may include but is not limited to: 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
    - Introduction to WAI ARIA
    - Using HTML5's New Semantic Tags Today

- http://www.deliveron.com/blog/post/SEO-Friendly-Routes-with-ASPnet-MVC.aspx

- **Plan and implement globalization and localization.**
  - This objective may include but is not limited to: plan a localization strategy; create and apply resources to UI including JavaScript resources; set cultures; create satellite resource assemblies
    - http://www.hanselman.com/blog/GlobalizationInternationalizationAndLocalizationInASPNETMVC3JavaScriptAndJQueryPart1.aspx
    - http://stackoverflow.com/questions/1560796/set-culture-in-an-asp-net-mvc-app
    - http://afana.me/post/aspnet-mvc-internationalization.aspx
    - http://afana.me/post/aspnet-mvc-internationalization-part-2.aspx
    - http://odetocode.com/blogs/scott/archive/2009/07/16/resource-files-and-asp-net-mvc-projects.aspx

- **Design and implement MVC controllers and actions.**
  - This objective may include but is not limited to: apply authorization attributes and global filters; implement action behaviors; implement action results; implement model binding
    - Filtering in ASP.NET MVC
    - The Features and Foibles of ASP.NET MVC Model Binding
    - ASP.NET MVC Custom Model Binding
    - ASP.NET MVC ActionResults explained
    - http://ofps.oreilly.com/titles/9781449320317/ch_Intro.html
    - Action Filtering in ASP.NET MVC Applications
    - ActionResult Class (System.Web.Mvc)
    - http://brendan.enrick.com/post/Types-of-ASPNET-MVC-3-Action-Results.aspx
    - http://www.codeproject.com/Articles/161963/ASP-NET-MVC-Model-Binding-Part-2
    - Use ViewModels to manage data & organize code in ASP.NET MVC applications

- **Design and implement routes.**
  - This objective may include but is not limited to: define a route to handle a URL pattern; apply route constraints; ignore URL patterns; add custom route parameters; define areas
    - ASP.NET Routing
    - Understanding Routing in ASP.NET MVC
    - ASP.NET MVC Areas–A Better Way To Structure The Application
    - Using Areas in ASP.NET MVC Application
    - Custom Controller Factory in ASP.NET MVC
    - http://ofps.oreilly.com/titles/9781449320317/ch_Routing.html
    - Creating a Route Constraint (C#)

- **Control application behavior by using MVC extensibility points.**
  - This objective may include but is not limited to: implement MVC filters and controller factories; control application behavior by using action results, viewengines, model binders, and route handlers
    - http://haacked.com/archive/2011/01/06/razor-syntax-quick-reference.aspx
    - http://weblogs.asp.net/scottgu/archive/2010/07/02/introducing-razor.aspx
    - http://stackoverflow.com/questions/6569809/mvc-3-aspx-vs-razor-view-engine
    - See: Design and implement MVC controllers and actions
    - MvcRouteHandler Class (System.Web.Mvc)
    - Custom Controller Factory in ASP.NET MVC develoq.net
- **Reduce network bandwidth.**
  - This objective may include but is not limited to: 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
    - Bundling and Minification
    - New Bundling and Minification Support (ASP.NET 4.5 Series)
    - Using CDN for Windows Azure
    - Introducing the Windows Azure Content Delivery Network
    - http://ofps.oreilly.com/titles/9781449320317/ch_ClientOptimization.html
    - http://www.yetanotherchris.me/home/2009/7/13/gzip-and-deflate-page-compression-in-aspnet.html
    - Visual Studio 2012 RC is released – The Big Web Rollup (Web Optimization)

[TOP]

Troubleshoot and Debug Web Applications

- **Prevent and troubleshoot runtime issues.**
  - This objective may include but is not limited to: 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)
    - MSDN - ASP.NET Tracing Overview
    - MSDN - ASP.NET Health Monitoring Overview
    - .NET Diagnostics and Logging In ASP.NET - Scott Allen - OdeToCode
    - http://knucklebones.blogspot.com/2010/11/enable-tracing-in-aspnet-mvc.html
    - Exercise 1: Logging Actions
    - NuGet Package of the Week #5 - Debugging ASP.NET MVC applications with Glimpse
    - Code Contracts
    - Code Contracts in C#
    - Code Contracts and Controllers
    - Chapter 10 - Performance Tuning and Monitoring
- **Design an exception handling strategy.**
  - This objective may include but is not limited to: 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
    - http://www.codeproject.com/Articles/422572/Exception-Handling-in-ASP-NET-MVC
    - http://www.prideparrot.com/blog/archive/2012/5/exception_handling_in_asp_net_mvc
    - http://perspectivespace.com/error-handling-in-aspnet-mvc-3-part-3-handlee
- **Test a web application.**
  - This objective may include but is not limited to: create and run unit tests, for example, use the Assert class, create mocks; create and run web tests
    - Ofps.oreilly - Automated Testing
    - Building Testable ASP.NET MVC Applications
    - MSDN - Walkthrough: Using TDD with ASP.NET MVC
    - http://www.codemerlin.com/2011/07/mocking-httpcontext-httpresponse-httprequest-httpsessionstate-etc-in-asp-net/
    - http://code.google.com/p/moq/
    - http://nunit.org/
    - http://stackoverflow.com/questions/494083/making-sure-a-view-exists
    - http://haacked.com/archive/2007/12/16/testing-routes-in-asp.net-mvc.aspx
    - http://stackoverflow.com/questions/366388/how-can-i-unit-test-my-asp-net-mvc-controller-that-uses-formsauthentication
    - http://stackoverflow.com/questions/1057776/how-to-test-action-filters-in-asp-net-mvc
    - http://martinfowler.com/articles/mocksArentStubs.html
    - C# Unit Testing Basics – C# Tutorials | Dream.In.Code
- **Debug a Windows Azure application.**
  - This objective may include but is not limited to: 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 IntelliTrace and Remote Desktop Protocol (RDP)
    - MSDN - Collecting Logging Data by Using Windows Azure Diagnostics
    - http://www.codeproject.com/Articles/303686/Windows-Azure-Diagnostics-Performance-Counters-In
    - http://www.windowsazure.com/en-us/manage/windows/best-practices/troubleshooting/
    - MSDN - Debug Your App by Recording Code Execution with IntelliTrace
    - Using IntelliTrace to debug Windows Azure Cloud Services
    - http://blog.elastacloud.com/2011/06/29/tips-and-tools-for-a-better-azure-deployment-lifecycle-2

[TOP]

Design and Implement Security

- **Configure authentication.**
  - This objective may include but is not limited to: authenticate users; enforce authentication settings; choose between Windows, Forms, and custom authentication; manage user session by using cookies; configure membership providers; create custom membership providers
    - MSDN - Membership Providers

        - Introduction to Membership
        - Examining ASP.NET's Membership, Roles, and Profile
        - Membership and Role Providers in ASP.NET 2.0 Part I
        - Membership and Role Providers in ASP.NET 2.0 Part II
        - http://www.hanselman.com/blog/IntroducingSystemWebProvidersASPNETUniversalProvidersForSessionMembershipRolesAndUserProfileOnSQLCompactAndSQLAzure.aspx
        - http://www.codeproject.com/Articles/165159/Custom-Membership-Providers
        - http://www.danharman.net/2011/06/23/asp-net-mvc-3-custom-membership-provider-with-repository-injection/
        - http://www.codeproject.com/Articles/98950/ASP-NET-authentication-and-authorization
        - ASP.NET Cookies Overview
        - Configuring an ASP.NET Application to Use Membership
- **Configure and apply authorization.**
    - This objective may include but is not limited to: create roles; authorize roles by using configuration; authorize roles programmatically ; create custom role providers; implement WCF service authorization
        - http://www.davidhayden.me/blog/asp.net-mvc-4-allowanonymous-attribute-and-authorize-attribute
        - ASP.NET MVC Authentication - Global Authentication and Allow Anonymous
        - Authorization In WCF-Based Services
        - Implementing a Role Provider
        - http://www.asp.net/web-forms/tutorials/security/roles/creating-and-managing-roles-cs
        - Authentication and Authorization in WCF Services - Part 1
- **Design and implement claims-based authentication across federated identity stores.**
    - This objective may include but is not limited to: 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, LiveID, and Facebook) for SAML and SWT tokens
        - MSDN - Federated Identity for Web Applications
        - MSDN - Federated Identity with Windows Azure Access Control Service
        - Write a custom security token and handler in Windows Identity Foundation
        - OAuth/OpenID Support for WebForms, MVC and WebPages
        - Access Control Tutorial
        - Implementing federated security with Azure Access Control Service
        - How to: Add a Custom Token Handler
- **Manage data integrity.**
    - This objective may include but is not limited to: apply encryption to application data; apply encryption to the configuration sections of an application; sign application data to prevent tampering
        - MSDN - How To: Encrypt Configuration Sections in ASP.NET 2.0 Using DPAPI
        - Encrypting Data in .NET Applications
        - Encrypting and Decrypting Configuration Sections
        - Setting up SSL correctly with IIS and ASP.NET
- **Implement a secure site with ASP.NET.**
    - This objective may include but is not limited to: secure communication by applying SSL certificates; salt and hash passwords for storage; 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)
        - http://www.codeproject.com/Articles/358993/Examining-Request-Validation-with-ASP-NET-4-5
        - http://ofps.oreilly.com/titles/9781449320317/ch_Security.html
        - http://haacked.com/archive/2010/04/06/using-antixss-as-the-default-encoder-for-asp-net.aspx
        - MSDN - Microsoft Anti-Cross Site Scripting Library V1.5: Protecting the Contoso Bookmark Page
        - http://stackoverflow.com/questions/2138429/hash-and-salt-passwords-in-c-sharp
        - http://haacked.com/archive/2010/04/28/replacing-html-encode.aspx
        - http://www.dotnetperls.com/sqlparameter
        - https://www.owasp.org/index.php/Cross-Site_Request_Forgery_(CSRF)_Prevention_Cheat_Sheet
        - http://www.codinghorror.com/blog/2008/10/preventing-csrf-and-xsrf-attacks.html

[TOP]

Sources/References

- **Must READ**

- MSDN - ASP.NET MVC 4
- Programming ASP.NET MVC 4 - EBOOK
- https://github.com/ProgrammingAspNetMvcBook/CodeExamples
- What's New in ASP.NET 4.5 and Visual Web Developer 11 Developer Preview

Guides from Others

- http://www.bloggedbychris.com/2012/11/06/microsoft-exam-70-486-study-guide/
- go.mellbourn.net/70_486
- http://www.delicious.com/mellbourn/70_-_486

ASP.NET blogs - MVC

- Scott Guthrie on ASP.NET MVC
- Phil Haack on ASP.NET MVC
- Steve Sanderson on ASP.NET MVC
- Scott Hanselman on ASP.NET MVC
- Rachel Appel on ASP.NET MVC
- Rick Anderson on ASP.NET MVC
- Stuart Leeks on ASP.NET MVC
- David Hayden (David Hayden)
- Imran Baloch on ASP.NET MVC
- Andrei Ignat on ASP.NET MVC
- Radu Enuca on ASP.NET MVC
- Kazi Manzur Rashid on ASP.NET MVC
- Simon Ince's Blog on MVC

Others

- Web Camps Training Kit
- ASP.NET 4.5 and Visual Studio 2012
- Intro to ASP.NET MVC 4
- http://www.asp.net/mvc/tutorials/mvc-4/aspnet-mvc-4-mobile-features
- http://www.asp.net/mvc/mvc4
- http://www.w3schools.com/aspnet/mvc_intro.asp
- http://www.microsoft.com/learning/en/us/Exam.aspx?ID=70-486
- http://www.windowsazure.com
- http://www.windowsazure.com/en-us/develop/net/tutorials/web-site-with-sql-database/
- http://stackoverflow.com/questions/1560796/set-culture-in-an-asp-net-mvc-app
- http://www.windowsazure.com/en-us/develop/net/fundamentals/intro-to-windows-azure/
- https://www.microsoftvirtualacademy.com/tracks/introduction-to-windows-azure
- http://www.meetwindowsazure.com/Azure101
- http://channel9.msdn.com/shows/Cloud+Cover/
- Deploying an ASP.NET Web Application to a Windows Azure Web Site and SQL Database

Document Created By Victor Hugo do V C Mello

- LinkedIn
- Twitter
- Facebook
- E-Mail

[TOP]