

1: Understanding www.enganchecubano.com MVC using real world example, for beginners and intermediaries

Advance Practical Examples on DataGrid It is seen that there are many obstacles involved in developing the software of www.enganchecubano.com Therefore, this section will discuss some of the professional examples which would be very useful in developing critical movements.

I will only show the practical programmatic approach of learning ASP. In this article I am not paying much attention to the theory. The first question is, what is ASP. For more information click [here](#). To determine the version of MVC installed on your machine use the following procedure. I hope you get something like the following image. That indicates that only MVC 2 has been installed on that machine. Provide a proper name and click OK. After clicking OK the following window appears. Select Empty template and view engine as Razor. Razor is preferred by most MVC developers. We will discuss the Razor View Engine in detail in a later part of this tutorial. If you want you can install other view engines also and use those engines. Now click on Ok. Then our project solution will look as in the following: As our solution is showing there are three main folders, Models, Views and Controllers. The MVC architecture is based on this folder. The Views contains the UI code that we need to show to our end user. The controller is supposed to be the handler that executes the view request depending on the needs of the user. The Models folder contains mainly the code that communicates the database or other services. Our solution is now ready. We will create our first MVC application. This application will just show the static message with the version of MVC we are using. The following code will be generated automatically:

2: Why should I use JSON with www.enganchecubano.com? - Stack Overflow

the features r not explained even in this practical example as consists of whole lot of other things to discuss, this is a simple example to be made as a article not dealt with standards of codeproject.

Net, C , ADO. Net Framework questions and answers for which a reader has to look around for entire internet on different community web sites. Most of the questions and answers you must have already read. The purpose of this article is to consolidate at the most study material related to. Net at one single place. NET What is view state and use of it? The current property settings of an ASP. NET server controls contained within the page. What are user controls and custom controls? NET Framework class library. This is a generic term that includes user controls. A user-authored server control that enables an ASP. What are the validation controls? A set of server controls included with ASP. The latter one allows you to write formattedoutput. What methods are fired during the page load? Where does the Web page belong in the. NET Framework class hierarchy? CodeBehind is relevant to Visual Studio. Suppose you want a certain ASP. NET function executed on MouseOver over a certain button. Where do you add an event handler? What are the different types of caching? Caches the dynamic output generated by a request. For caching the whole page the page should have OutputCache directive. Caches the objects programmatically. What are different types of directives in. Defines page-specific attributes used by the ASP. NET page parser and compiler. Defines control-specific attributes used by the ASP. Can be included only in. Explicitly imports a namespace into a page or user control. Indicates that the current page or user control implements the specified. Declaratively controls the output caching policies of an ASP. Start the DbgClr debugger that comes with the. Can a user browsing my Web site read my Web. If I update session state, should I lock it, too? Are concurrent accesses by multiple requests executing on multiple threads a concern with session state? Two, if they do overlap, ASP. NET forms authentication cookies provide any protection against replay attacks? If an authentication cookie is stolen, it can be used by an attacker. So a stolen session cookie can only be used in replay attacks as long as the ticket inside the cookie is valid. The default time-out interval is 30 minutes. How do I send e-mail from an ASP. Send message ; MailMessage and SmtptMail are classes defined in the. Due to a security change made to ASP. Is it possible to prevent a browser from caching an ASPX page? In this example, it prevents caching of a Web page that shows the current time. The following directive sets AspCompat to true: Explain the differences between Server-side and Client-side code? Also a possible security hazards for the client computer. What type of code server or client is found in a Code-Behind class? C Should validation did the user enter a real date occur server-side or client-side? How is this technology different than what is available though ASP? Web Forms are the heart and soul of ASP. If you use Microsoft Visual Studio. NET, you will also get the familiar drag-and-drop interface used to create your UI for your Web application. What is the difference between Server. Why would I choose one over the other? In earlier versions of IIS, if we wanted to send a user to a new Web page, the only option we had was Response. While this method does accomplish our goal, it has several important drawbacks. The biggest problem is that this method causes each page to be treated as a separate transaction. Redirect introduces some additional headaches. First, it prevents good encapsulation of code. As you might suspect, Server. Transfer fixes all of these problems. It does this by performing the transfer on the server without requiring a roundtrip to the client. How can you provide an alternating color scheme in a Repeater control? AlternatingItemTemplate Like the ItemTemplate element, but rendered for every other row alternating items in the Repeater control. You can specify a different appearance for the AlternatingItemTemplate element by setting its style properties. Which template must you provide, in order to display data in a Repeater control? ItemTemplate What event handlers can I include in Global. For example, you can name a BeginRequest event handler. You can also include event handlers in Global. NET applications in general, whereas. The Configuration system first searches settings in machine. NET Web application server. There is only Machine. NET web appplication is spanned across three web-servers using round-robin load balancing what would be the best approach to maintain login-in state for the users? Use the state server or store the state in the database. This can be easily done through simple setting change in the web. You can use

them to extend your ASP. For example, if you wanted custom authentication facilities for your application, the best technique would be to intercept the request when it comes in and process the request in a custom HTTP module. How do you turn off cookies for one page in your site? Since no Page Level directive is present, I am afraid that can't be done. How do you create a permanent cookie? Permanent cookies are available until a specified expiration date, and are stored on the hard disk. MinValue with respect to the current datetime. If you want the cookie which never expires set its Expires property equal to DateTime. Which method do you use to redirect the user to another page without performing a round trip to the client? Execute What property do you have to set to tell the grid which page to go to when using the Pager object? CurrentPageIndex Should validation did the user enter a real date occur server-side or client-side? It should occur both at client-side and Server side. By using expression validator control with the specified expression ie.. But for checking the date where it is the real data or not should be done at the server side, by getting the system date ranges and checking the date whether it is in between that range or not. What does the "EnableViewState" property do? Why would I want it on or off? Enable ViewState turns on the automatic state management feature that enables server controls to re-populate their values on a round trip without requiring you to write any code. This feature is not free however, since the state of a control is passed to and from the server in a hidden form field.

3: A Practical Example Of Using The New Features Of www.enganchecubano.com - CodeProject

So now that you know how www.enganchecubano.com MVC works and how it easily helps us build high quality web applications by dividing the workflow into 3 separate components, we can now work on a practical example to let you see it in action.

At run time, master pages are handled in the following sequence: Users request a page by typing the URL of the content page. If the directive references a master page, the master page is read as well. If this is the first time the pages have been requested, both pages are compiled. The master page with the updated content is merged into the control tree of the content page. The content of individual Content controls is merged into the corresponding ContentPlaceHolder control in the master page. The resulting merged page is rendered to the browser. The process is illustrated in the following diagram. The URL of the page is that of the content page. From a programming perspective, the two pages act as separate containers for their respective controls. The content page acts as a container for the master page. However, you can reference public master-page members from code in the content page, as described in the next section. Note that the master page becomes a part of the content page. In effect, the master page acts in much the same way a user control acts as a child of the content page and as a container within that page. In this case, however, the master page is the container for all of the server controls that are rendered to the browser. The control tree for a merged master and content page looks something like this: Copy Page Master Page Master page markup and controls ContentPlaceHolder Content page markup and server controls Master page markup and controls ContentPlaceHolder Content page markup and server controls Master page markup and controls This diagram is simplified; if the content page does not have corresponding Content controls, the master page might also have markup and controls in the ContentPlaceHolder controls. In general, this structure has no effect on how you construct your pages or program them. However, in some cases, if you set a page-wide property on the master page, it can affect the behavior of the content page, because the master page is the closest parent for the controls on the page. For example, if you set the `EnableViewState` property on the content page to true but set the same property to false in the master page, view state will effectively be disabled because the setting on the master page will take priority.

Master Page and Content Page Paths When a content page is requested, its content is merged with the master page, and the page runs in the context of the content page. For example, if you get the `CurrentExecutionFilePath` property of the `HttpRequest` object, whether in content page code or in master page code, the path represents the location of the content page. The master page and content page do not have to be in the same folder. NET can merge the content and master pages into a single rendered page.

Referencing External Resources Both the content page and master page can contain controls and elements that reference external resources. For example, both might contain image controls that reference image files, or they might contain anchors that reference other pages. The context for the merged content and master pages is that of the content page. This can affect how you specify URLs for resources, such as image files and target pages, in anchors. For example, you might put an Image control on a master page and set its `ImageUrl` property to be relative to the master page. At run time, ASP.NET marks the property internally in the control as being a URL. The property is marked with the attribute `UrlPropertyAttribute`. In practical terms, ASP.NET server control properties that are commonly used to reference external resources are marked in this way. In general, when working with elements on master pages, it is recommended that you use a server control, even for elements that do not require server code. For example, instead of using an `img` element, use an `Image` server control. NET can resolve URLs correctly and you can avoid maintenance issues that might arise if you move the master or content page. For more information about specifying paths for ASP.NET.

4: Benefits of Parameterized query with Practical Example | The www.enganchecubano.com Forums

These C# examples cover a wide range of programming areas in Computer Science. Every example program includes the description of the program, C# code as well as output of the program. All examples are compiled and tested on Visual Studio.

NET page runs, the page goes through a life cycle in which it performs a series of processing steps. These include initialization, instantiating controls, restoring and maintaining state, running event handler code, and rendering. The following are the various stages or events of ASP. Net page life cycle.

PreInit Check the `IsPostBack` property to determine whether this is the first time the page is being processed. Create or re-create dynamic controls. Set a master page dynamically. Set the `Theme` property dynamically. Note If the request is a postback then the values of the controls have not yet been restored from the view state. If you set a control property at this stage, its value might be overwritten in the next event. This event fires after each control has been initialized. Use this event to read or initialize control properties. The "Init" event is fired first for the bottom-most control in the hierarchy, and then fired up the hierarchy until it is fired for the page itself.

InitComplete Until now the viewstate values are not yet loaded, hence you can use this event to make changes to the view state that you want to ensure are persisted after the next postback. Raised by the Page object. Use this event for processing tasks that require all initialization to be complete.

OnPreLoad Raised after the page loads view state for itself and all controls, and after it processes postback data that is included with the Request instance. Before the Page instance raises this event, it loads view state for itself and all controls, and then processes any postback data included with the Request instance. ViewState data are loaded to controls. Postback data are now handed to the page controls.

Load The Page object calls the `OnLoad` method on the Page object, and then recursively does the same for each child control until the page and all controls are loaded. The Load event of individual controls occurs after the Load event of the page. This is the first place in the page lifecycle that all values are restored. Most code checks the value of `IsPostBack` to avoid unnecessarily resetting state. You may also call `Validate` and check the value of `IsValid` in this method. You can also create dynamic controls in this method. Use the `OnLoad` event method to set properties in controls and establish database connections.

NET now calls any events on the page or its controls that caused the `PostBack` to occur. In a postback request, if the page contains validator controls, check the `IsValid` property of the Page and of individual validation controls before performing any processing. This is just an example of a control event. Here it is the button click event that caused the postback.

LoadComplete Raised at the end of the event-handling stage. Use this event for tasks that require that all other controls on the page be loaded.

OnPreRender Raised after the Page object has created all controls that are required in order to render the page, including child controls of composite controls. The Page object raises the `PreRender` event on the Page object, and then recursively does the same for each child control. The `PreRender` event of individual controls occurs after the `PreRender` event of the page. Allows final changes to the page or its control. This event takes place before saving `ViewState`, so any changes made here are saved. After this event, you cannot change any property of a button or change any viewstate value. Use the event to make final changes to the contents of the page or its controls.

OnSaveStateComplete Raised after view state and control state have been saved for the page and for all controls. Before this event occurs, `ViewState` has been saved for the page and for all controls. Any changes to the page or controls at this point will be ignored. Use this event perform tasks that require the view state to be saved, but that do not make any changes to controls.

Render Method This is a method of the page object and its controls and not an event.

Unload This event is used for cleanup code. At this point, all processing has occurred and it is safe to dispose of any remaining objects, including the Page object. Cleanup can be performed on: Instances of classes, in other words objects Closing opened files Closing database connections. This event occurs for each control and then for the page. During the unload stage, the page and its controls have been rendered, so you cannot make further changes to the response stream. If you attempt to call a method such as the `Response.Write` method then the page will throw an exception.

Control Values In the following code, I have assigned the values to the label control on each event. Because, during the unload

stage, the page and its controls have been rendered, so you cannot change the values. Please observe the code comments and output. It will help you to clearly understand the concepts.

5: www.enganchecubano.com Master Pages

As part of the www.enganchecubano.com MVC Web Development course at the Software University I developed a practical MVC hand-on lab (tutorial) that gives rich experience in building data-driven Web applications with www.enganchecubano.com MVC, www.enganchecubano.com MVC 5, SQL Server, C#, Visual Studio, Entity Framework (code first), AJAX, jQuery, Bootstrap and other modern technologies and tools.

NET presented developers with a new framework for website development, in which they would easily distinguish between data layer, business layer and the methodology they would use to render these objects on the screen. They called this framework ASP. I will elaborate these items in this blog post as well as other objects and technologies I am going to use too. NET team provided us with this framework. Solving a bug might take hours of headscratching and some other help too. It is harder to find the problem that is scripted inside the HTML web page. In these scenarios, it is always found helpful to separate your code from your data and from your HTML markup. This would make it a lot easier to find and solve the problem. Actually this is not a pattern, specific to the web development. The link between these three objects of MVC pattern have been depicted in this image above. Each request, when comes, is handled by the Controller and then Controller, according to the request makes decisions to load the data, create the response and then sends the data back to the client. It should also be noted here, that your Controller acts as a bridge between your Model and the View. Because they, as themselves, cannot perform any action. Controller triggers their events and makes them do something, like return data from Model, or to render the HTML document from the View etc. All of the resources and errors are also handled by the Controller. Making it the heart of the pattern, because the response is also sent back from a controller. You can think of an example of a Controller to be the entire Business-logic-layer. View Next comes the part of the View, this is the actual web page that is being displayed to the user. It contains the HTML codes that are to be sent back to the data as a response to his request. View, can also be created dynamically. As already said, all of the requests are handled by Controller, so any parameter including QueryStrings can also be handled by Controllers. Using these parameters, we can generate dynamic Views. So dynamic content in our view, change their layouts or show some other error messages if the data sent is not of our own choice. View, generally depends on the Model that is being used to create the View and these dynamic data objects are capture from the Model Model is discussed in the next section. Point to be noted here is that while section-in-action is View, still Controller is playing a vital role for passing the values and for retrieving the data to be sent to client. Model As the name suggests, it is a model of some object. Model is never shown to the user actually, the client because he is supposed to see the data and the results we want him to see, that is why, it is a good approach to keep a great abstraction layer between Model and the user the client. Everytime a change is made, View is update by the Controller, because Controller is informed about the change this informing event is also raised by Controller; as I already said, Controller handles the events. To store anything in the Model, the user has not been provided with any form that is directly connected to the Model, instead a form is generated in the View by the Controller for the user to fill in. Once the form is filled, the form values are then passed to the model for storing purposes. All kinds of data validations most special type of which are SQL Injections can be checked at the Controller level rather than losing important data. NET MVC project a user triggers the Controller, the controller then reads and manipulates the requests, Requests the Model for data, gets the data and then Updates the View to be sent back to the client. Overview of MVC pattern. Real world example of ASP. For that, I am going to use the Model, View and Controller architecture, so that I can easily distinguish between the data of my application which would be the data for the clients , code for the requests and responses which would constitute the controller and how the data is shown to the users views of my application. I am going to create everything from scratch, to make every thing understandable for you. NET team at its best. That is why I am saying I am going to create everything from scratch, to explain every bit of the MVC pattern for you and how you can create actions that respond to your actual processes, not just to the built-in ones. Just to distinguish them all, an ID is also stored. Whether the client is trusted or not, is also stored along with him. Office manager wants to be able

to, create new clients, modify the existing ones, and once he is done with them he wants to be able to delete the previous ones. To implement this basic example in ASP. NET Web Application, then name it as you want. You can use any name that you want. Then you will be required to make the selections for your application in the next tab. This would then create your application, with the default content for your application settings. You might see this web page inside your Visual Studio. This is the tutorial until creation of the application. You can continue to make changes to it, so that it would reflect your own application and not the default ASP. How to create views and what would be the model will be covered in coming sections. For now, just make up your mind to understand the Controller in MVC. Json package is a very useful package to use, so before you go any further open your NuGet package manager console inside the Tools and run the following command to download and include it in your project. `Json` Creating a custom Controller A custom controller is something that you can use to create your custom controllers for your application, that would handle the requests coming for them and they will provide the user with a view filled in with the data from the model. Controller is a class in real which inherits from System. Controller class, making it able to inherit all of the functions and members of the Controller parent class. Let us create a Controller. Inside your Source code, there is a folder named as Controllers. Default controllers in ASP. Right click on Controller folder and inside the Add option, click Controller to create a new one. We just need an Empty controller, to create our own functions actions in MVC for it to handle and perform. Before we move on to create actions that this Controller would perform. I wanted to remind you about automatic changes that were made in your Views folder. Like every class, we can create different functions for our Controllers, which are known as Actions. `IndexOf client ; Clients`. That was our controller, if you have a look at the code above you will find a few functions, a few ViewBags and a few Views etc. You will use them to tell the application what to do, when user triggers such a particular action in your application. Creating a custom View Now that back-end code Controller code has been done, and you can now create the Views for your application. Views are used to show the content data of your application to the client user in HTML markup. Because browser can only render HTML content, we will be using that markup, and Views are used for that. Each controller has its own particular view. You can put it, like each of the Action function in the Controller is meant for a particular View HTML document to be returned in the application. Now select the layout, layout is applied across the pages Views to design them a like. You can use your own views too, but for this I am going to use the default layout. This is the step-by-step method to create the View in ASP. After that it is just simple razor syntax, to create the HTML document View and to return to the client user. The Model object the one used in the foreach loop is having the value of what ever is passed for the model thing. You should create next three Views yourself as a test and write this content to them. That is because, for a View, to have a model is not obligatory. These are just simple HTML pages, that are being made update-to-the-content by the razor syntax and then sent back to the browser in an HTML markup, for rendering purposes. You can pass the data from the model to the view from the Controller, which will be then rendered as a simple HTML markup and will be displayed inside the browser to the user. NET team has provided a very good tutorial for that and you might want to join them on their journey here. They used Entity Framework for their Actions. This would try to remove ambiguity in our minds about this MVC pattern and its extensiveness for other technologies and frameworks. A model is a representation of the real-world objects in our own context; which here is the ASP. Think of it like there is a Client model. He would have a ClientID, Name, Address and a Trusted as a flag you should recall our scenario if you have no idea of where these came from. What would you write as a class for this Client? You will find one other major change, which is the namespace this class is location in. The name is having an extra. That makes up our different fields. Let us create a Model for our application, click on the Models folder in your app, and inside Add click on Class file and name it Client. Select a new class file to be created as a Model Name it what-ever-you-want, but I would be using Client. These were the three major fields in the MVC that are used to create an application. However, there are a few other minor fields that work and run our application, that I will be covering here for you to atleast know how to kick-start your project. As far, we have covered what is Controller controls the HTTP requests coming our going to client , what is a View it is rendered as HTML output in the browser, the response that is returned to the client in HTML markup can be said to be the View and finally what is a Model

the representation of the real-world data in our application is called a model. Now, we can go and have a look at the classes and objects which run our application. Like all other apps, that run over .NET also exposes some functions for its developers to use, to perform different actions when the app starts, when the app crashes etc. .NET also enables the developers to perform some client-side actions.

6: Experimenting - www.enganchecubano.com Tutorial, Advance Practical Examples on DataGrid

To implement this basic example in www.enganchecubano.com MVC, we will start off by creating a simple www.enganchecubano.com MVC application. To do so, you can either press CTRL+SHIFT+N, or you can click on File, then New and a Project.

LinkedIn As part of the ASP. In this lab you will create a web based event management system, from zero to fully working web application. Enjoy this free practical ASP. The source code is intentionally given as images to avoid copy-pasting. Project Assignment Design and implement a Web based event management system. Events may have also optionally duration, description, location and author. Events can be public visible by everyone and private visible to their owner author only. Events may have comments. Comments belong to certain event and have content text, date and optional author owner. The home page displays all public events, in two groups: Users should have mandatory email, password and full name. Deleting events requires a confirmation. Implement client-side and server-side validation data validation. This VS solution will hold your projects: NET Web application project. It will hold your ASP. Visual Studio will generate an ASP. The application uses a typical Web development stack for. Edit the connection string in the Web. Delete some unneeded files: We will not use it. We can add it later. Build and run the application to check whether it works, e. Test the user registration functionality. You should have out-of-the-box user registration, login and logout: You have a problem. By default, the MVC application is configured to use too complex passwords. This will make hard the testing during the development. Recompile and run the application again. Test the user registration, login and logout again. After a bit of waiting, Entity Framework will create the database schema in SQL Server and the MVC application will register the user in the database and will login as the new user: Open the database to ensure it works as expected. Change the home page link. Now run the application to test it again. It should look like this for anonymous users: It should look like this: Now you are ready for the real development on the Events management system. It mixes the entity data models with the Entity Framework data context and the MVC view models for the views and MVC input models for mapping the forms. Delete the unneeded files from the Events. Move the file Events. Now the Visual Studio Solution will not compile. Many errors will be shown if we rebuild with [F6]: We need to fix some issues: Your code should look like this: Change its namespace to Events. Your code will still not compile. Now you should add NuGet references to the missing libraries: The others depend on it and will be installed by dependency. Data should compile without errors. Web will not compile due to missing classes: Add project reference from Events. Web project to Events. Now the project should compile correctly. Test the MVC application. It should work correctly. Define the Data Model Now you are ready to define the data model for the Events management system: First, add a full name in the ApplicationUser class. Next, create your Event class. It will hold the events and their properties: It should compile without errors. Create your Comment class. It will hold the comments for each event:

7: www.enganchecubano.com - First Example

12 HTML table examples for beginners to practice. Using these examples beginner developers can practice HTML Table attributes, various CSS properties to enhance their HTML Table and CSS knowledge.

DevNet - Web Advertisement: This is great, as it represents a moment of clarity in a whirlpool of similar-looking and confusing options. So you click and are presented with various options as to the type of project to create. So you pick up the MVC option. What you obtain is a no-op demo application that was probably meant to give you an idea of what it means to code for ASP. Even though the result does nothing, the resulting project is fairly overloaded. This is not a bad idea in theory, as nearly any ASP. This is better, as it delivers a more nimble project even though there are many references that are useless at first. The truth is that any expert developer has his own favorite initial layout of the startup project, including must-have packages and scripts. Overall, here are ten good practices for sane ASP. Figure 1 shows the list of unnecessary references detected by ReSharper. Unnecessary references in the ASP. Figure 2 shows what you really need to have referenced in order to run a nearly dummy ASP. Strictly required assemblies in a basic ASP. When you start adding Nuget packages, some other conventions start appearing such as the Scripts folder for Modernizr and for jQuery and its plugins. I find such a project structure rather confusing and usually manage to clean it up a little bit. For example, I like to place all content images, style sheets, scripts, fonts under the same folder. I sometimes rename Models to ViewModels and give it a structure similar to Views: In really complex sites, I also do something even more sophisticated. The Models folder remains as is, except that two subfolders are added Input and View, as shown in Figure 3. Content and Models folder internal structure Generally speaking, there are quite a few flavors of models. One is the collection of classes through which controllers receive data. Populated from the model binding layer, these classes are used as input parameters of controller methods. I collectively call them the input model and define them in the Input subfolder. Both input and view models are then split on a per-controller basis. One more thing to note is the matching between project folders and namespaces. So you can freely decide to ignore it and still be happy, as I did for years. At some pointâ€”but it was several years agoâ€”I realized that maintaining the same structure between namespaces and physical folders was making a lot of things easier. And when I figured out that code assistant tools were making renaming and moving classes as easy as click-and-confirm, well, I turned it into enforcement for any of my successive projects. An interesting convention introduced with ASP. Routes ; For consistency, you can use the same pattern to add your own classes that take care of application-specific initialization tasks. More often than not, initialization tasks populate some ASP. Routes dictionary of the last snippet just after the heading for 2. For testability purposes, I highly recommend that xxxConfig methods are publicly callable methods that get system collections injected. Anyway, the method uses a mocking framework to mock HttpContextBase and then invokes method GetRouteData on RouteCollection passing the mocked context. Is it safe to rename this folder to something like Config? The possible sore point is the use of the WebActivator Nuget package in the project, either direct or through packages that have dependencies on it. WebActivator is a package specifically created to let other Nuget packages easily add startup and shutdown code to a Web application without making any direct changes to global. WebActivator was created only for the purposes of making Nuget packages seamlessly extend existing Web applications. Bundling is therefore the process of grouping distinct resources such as CSS files into a single downloadable resource. Minification, on the other hand, is the process that removes all unnecessary characters from a text-based resource without altering the expected functionality. Minification involves shortening identifiers, renaming functions, removing comments and white-space characters. Although bundling and minification can be applied together, they remain independent processes. The only exception is for large and very common resources that might be served through a Content Delivery Network CDN. The jQuery library is a great example. Bundling requires the Microsoft ASP. Downloading the Optimization Framework also adds a few more references to the project. In particular, they are WebGrease and Microsoft Infrastructure. These, in turn, bring their own dependencies for the final graph, shown in Figure 4. Graph of Nuget dependencies for bundling and minification Bundles are

created programmatically during the application startup in global. The BundleConfig class contains at least one method with code very close to the following snippet. Note that the Include method refers to physical paths within the project where the source code to bundle is located. The argument passed to the StyleBundle class constructor instead is the public name of the bundle and the URL through which it will be retrieved from pages. There are quite a few ways to indicate the CSS files to bundle. In addition to listing them explicitly, you can use a wildcard expression: NET optimization extensions come with two flavors of bundle classes: The former only does bundling; the latter does both bundling and minification. Minification occurs through the services of an additional class. Switching to a different minifier is easy too. All you do is using a different constructor on the StyleBundle class. You use the constructor with two arguments, the second of which is your own implementation of IBundleTransform. The only minor difference is that for frequently used script files such as jQuery you might want to use a CDN for even better performance. In operational terms, bundling and minifying script files requires the same process and logic as CSS files. Otherwise, if you also want minifying, you use the ScriptBundle class. This object is expected to bring in some custom logic for minifying script files. The default minifier comes from WebGrease and corresponds to the JsMinify class. All this can happen if you have a lot of script files. In a nutshell, you often end up composing your client-side logic by sewing together multiple pieces, each of which represents a distinct download. This practice was promoted by Yahoo and aimed at avoiding roadblocks during the rendering of the page. The second flavor of JavaScript can safely load at the bottom. Well, mostly at the bottom. Consider that as the page renders the user interface in the browser, users may start interacting with it. In doing so, users may trigger events that need some of the other JavaScript placed at the bottom of the page and possibly not yet downloaded and evaluated. The ready event of jQuery is an excellent tool to synchronize user interfaces with downloaded scripts. Finally, consider some techniques to load scripts in parallel so that the overall download time becomes the longest of all instead of the sum of all downloads. You do this using code, as shown below. Note that this is the approach that most social Web sites and Google Analytics use internally. The net effect is that all dynamically created elements are processed on different JavaScript threads. This approach is also employed by some popular JavaScript loader frameworks these days. What should you have in a master view? And in which order? With the exceptions and variations mentioned a moment ago for parallelizing the download of multiple scripts, there are two general rules that hold true for the vast majority of websites. The first rule says: The second rule says: There are a few other little things you want to be careful about in the construction of the layout file s for your ASP. First off, you might want to declare explicitly that the document contains HTML5 markup. You achieve this by having the following markup at the very beginning of the layout and subsequently at the beginning of each derived page. Also, you might want to declare the character set in the HEAD block. NET are concerned, you have other ways to set the character set. Having it right in each page just adds clarity. An interesting consequence of clarifying the character set being used is that you can avoid using HTML entities and type special characters directly in the source. The only HTML entities you should use are those that provide the text version of reserved markup characters, such as less-than, greater-than, and ampersand. This virtual rendering area is just called the "viewport. Having such a large viewport allows browsers to host nearly any Web page, leaving users free to pan and zoom to view content, as illustrated in Figure 5. The viewport in a browser The viewport meta-tag is a way for you to instruct the browser about the expected size of the viewport. This means that you should avoid referencing from the layout file CSS and script files that are referenced by all pages based on the layout. As developers, we certainly find it easier and quicker to reference resources used by most pages right from the layout file. But that only produces extra traffic and extra latency. A section is an area that derived pages can override.

8: Page Life Cycle In www.enganchecubano.com

The articles focus more on practical aspects and implementation www.enganchecubano.com, C#.Exploring internals of code and logic. for example, www.enganchecubano.com is only.

Or are they both the same thing except that the latter two are tightly coupled with IIS? Decoupling with IIS is part of it. I like to compare the pipeline more to Global. Part of the problem with this is developing reusable components from that was difficult and depended on IIS. OWIN attempts to remove those issues. A lot of the documentation says OWIN allows for decoupling between the web server and web application ie. But I have yet to see an example of some web application or web api that used OWIN and was successfully ported from being hosted on IIS and then some other web server. So is IIS and self hosting the only way to go for this decoupling between web server and web app? MVC 6 will resolve this issue and is a pretty big overhaul. The only thing the Console App does differently is it is bootstrapping a server with `HttpListener` in the `Main` method. Is Katana an owin component or is it the code we write using Katana or both put together? OWIN is really not much more an an abstraction layer, really a specification, between the web application and the web server. Kestrel is another example of an OWIN implementation. Katana is almost done with and wont go beyond revision3 and Helios is not yet supported by Microsoft as per some articles. So what is the future of OWIN in that case? Any other such usages of keeping an OWIN implementation in the middle? Request does not have an indexer in OWIN. Write "hello world 2: `RequestBody` is a bit of an implementation detail, the actual return type is internal. What are the various kinds of middle ware that you have plugged-in in your projects between the web server and application? Things for handling security, like a middleware component that handled nonces in Content Security Policy, which I wrote about on my personal blog here. The gist of it was it allows me to add an HTTP header with a nonce: `ToBase64String nonceBytes ; context. Set "ScriptNonce", nonce ; context.` There are lots of other things it can be used for.

9: Modern API Design with www.enganchecubano.com Core 2 - pdf - Free IT eBooks Download

www.enganchecubano.com WITH C# TYBSC-IT (SEM V) Prepared By: Prof. Sanjeela Sagar, VSIT and Prof. Pallavi Tawde, VSIT 3 PRACTICAL NO.: 01(B) AIM: If you have two integers stored in variables var1 and var2, what Boolean test can you.

Sensitive data from the page and the states of different controls on the page are stored in hidden fields that form the context of that page request. NET runtime controls the association between a page instance and its state. NET page is an object of the Page or inherited from it. All the controls on the pages are also objects of the related control class inherited from a parent Control class. When a page is run, an instance of the object page is created along with all its content controls. NET page is also a server side file saved with the. It is modular in nature and can be divided into the following core sections: The Page directive defines page-specific attributes used by ASP. NET page parser and compiler. Page directives specify how the page should be processed, and which assumptions need to be taken about the page. It allows importing namespaces, loading assemblies, and registering new controls with custom tag names and namespace prefixes. Code Section The code section provides the handlers for the page and control events along with other functions required. We mentioned that, ASP. NET follows an object model. Now, these objects raise events when some events take place on the user interface, like a user clicks a button or moves the cursor. The kind of response these events need to reciprocate is coded in the event handler functions. The event handlers are nothing but functions bound to the controls. The code section or the code behind file provides all these event handler routines, and other functions used by the developer. The page code could be precompiled and deployed in the form of a binary assembly. Page Layout The page layout provides the interface of the page. The following code snippet provides a sample ASP. NET page explaining Page directives, code section and page layout written in C: Generally it is c: Open the file from the browser to execute it and it generates following result: Instead of typing the code, you can just drag the controls into the design view: The content file is automatically developed. This generates the following result:

Devout remembering. The Wire in the Blood (A Dr. Tony Hill Carol Jordan Mystery) Links to ancestral ties 1. Opening Remarks Yuji Yamada 203 Medicaid waivers : license to shape the future of fiscal federalism Carol S. Weissert and William G. Weis Reel 17. Marion (Crittenden County), Monticello (South Eastern District of Arkansas), Monticello (Drew Co Diggero d103 manual filetype Doubt and Identity in Romantic Poetry Sight and hearing Market leader intermediate business english practice file How to write first-class business correspondence The virtue of patience Introduction Gabriele Bammer and Michael Smithson Humanistic theory by carl rogers Concept of economic recession in nigeria Clerical and Administrative Support Positions CHERUBS PILLOW 81 Lucy Moore, widow, and other heirs. Forensic Digital Imaging and Photography (With CD-ROM) Check and Double Check Phonics New patent for protecting pines from bark beetle infestation in urban and rural areas U00a7 7. Figure defined, 24 Go design patterns 12 week ironman training plan Kaplan Isat practice test Manual of psychedelic support MCSE SQL Server 2000 Administration Exam Cram Personal Test Center (Exam: 70-228) Morality of embryo use Nuclearisation of divided nations Collection district in Texas, and port of entry at Sabine Pass. The body and its functions Mechanical design synthesis Anonymous benefactor The Serpent Within Concept of modernism in Hispanic criticism A historical interlude : trance in Europe and the United States Law express contract law revision guide Bits question paper 2017 Lesson 17: strange feelings and emotions Sartre, romantic rationalist