CRUD with SPA, ASP.NET Web API and Angular.js

mdshohelrana, 26 Oct 2014 CPOL

4.81 (42 votes)

Rate this:

CRUD with SPA, ASP.NET Web API and Angular.js

Download AngularJsCRUD.zip - 854 KB

Introduction

In this article, I am going to demonstrate a simple process of CRUD in AngularJS with lot of pictorial representations. A picture is worth more than a thousand words, that for causes I believe in an article with proper screenshots. Here I will describe how you will create CRUD in angularjs. I am going to show some successful step to below:

Background

In visual studio web application, use the Client Server architecture. Here is the two process or two application that will be communicating with each other to exchange some information. From the two processes one is acts as a client process and another process acts as a server.

In traditional web applications, the client (browser) which typically makes a request for page to the server to initiates the communication. Then the server processes the request and sends the HTML of the page to the client (browser).

In Single-Page Applications (SPAs) at first the entire page is loaded in the client by the initial request, after that the subsequent action has to update by Ajax request and no need to reload the entire page. The SPA reduces the time by to respond to user actions and result is more fluid experience.

To develop the architecture of a SPA is very hard that are not similar to traditional web applications. However, the most familiar technologies like JavaScript frameworks like AngularJS, ASP.NET Web API, and new styling features of CSS3 make it really easy to design and build SPAs.

In this article, I will show you how to create SPA and CRUD by using ASP.NET Web API, AngularJS.

Overview

Objectives

In this article, you will learn how to:

- create angular-js CRUD
- Create a responsive UI using Twitter-Bootstrap

Prerequisites

- Windows 7 or later with at least 4GB RAM
- Visual Studio Express 2013 for Web or greater
- SQL Server 2012, SQL Server Express, or LocalDB
- ASP.NET MVC 5
- AngularJS in 60-ish Minutes eBook or Video

Setup

In order to create this application, please follow the above prerequisties.

Exercises

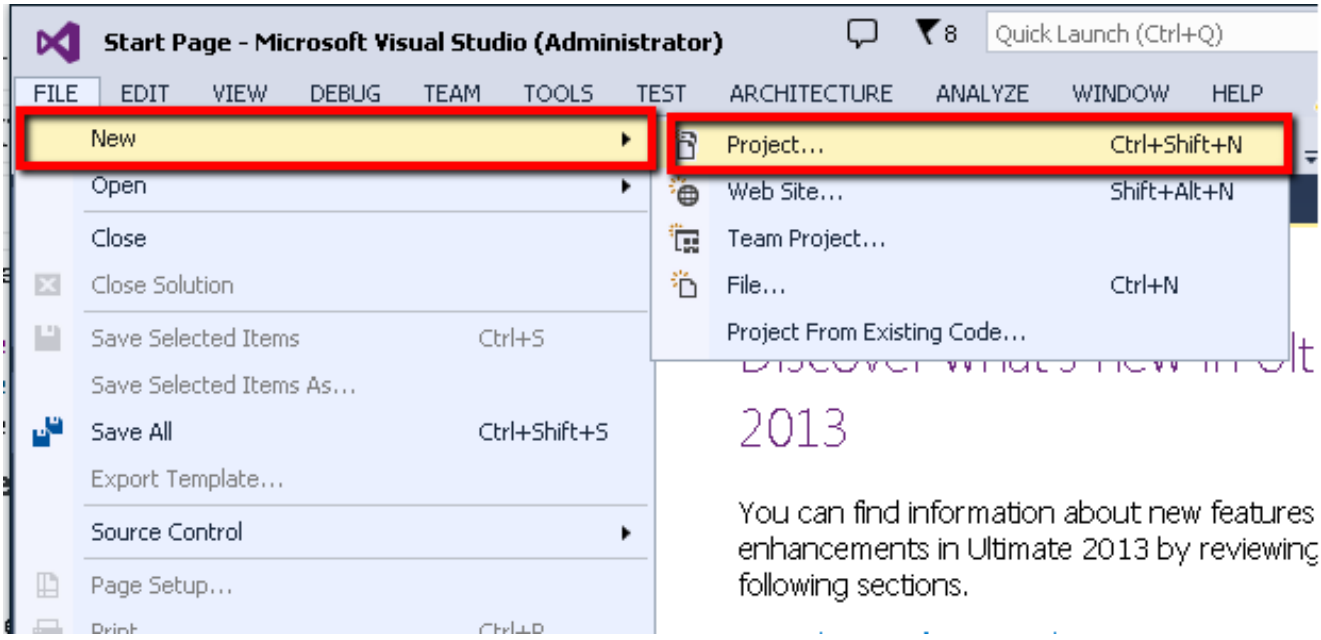
This article includes the following exercises:

1. Creating a Web API
2. Creating a SPA Interface

Exercise 1: Creating a Web API

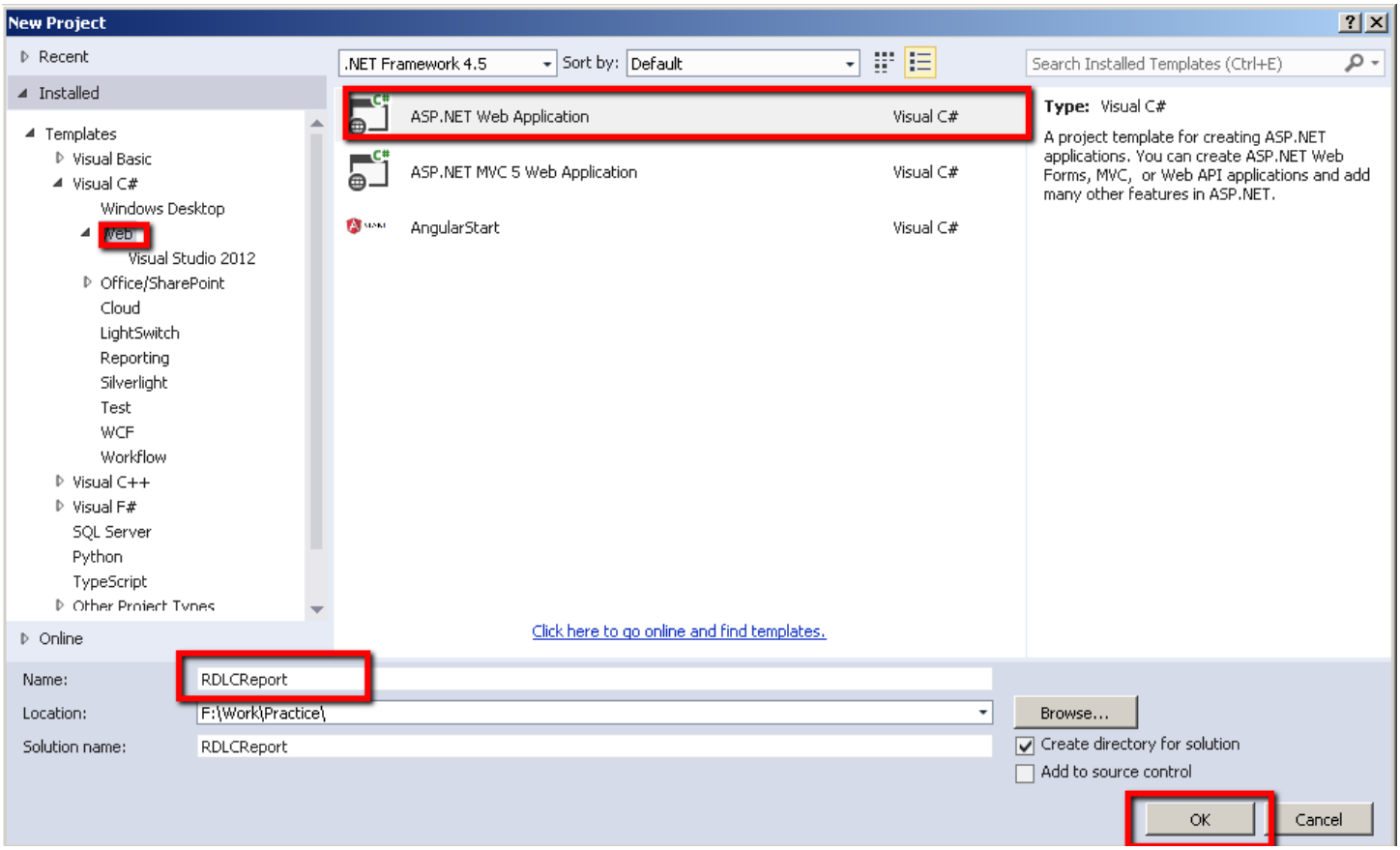
To set up a web api application

1. Open Visual **Studio Express 2013** for Web and select File | New > Project... to start a new solution.



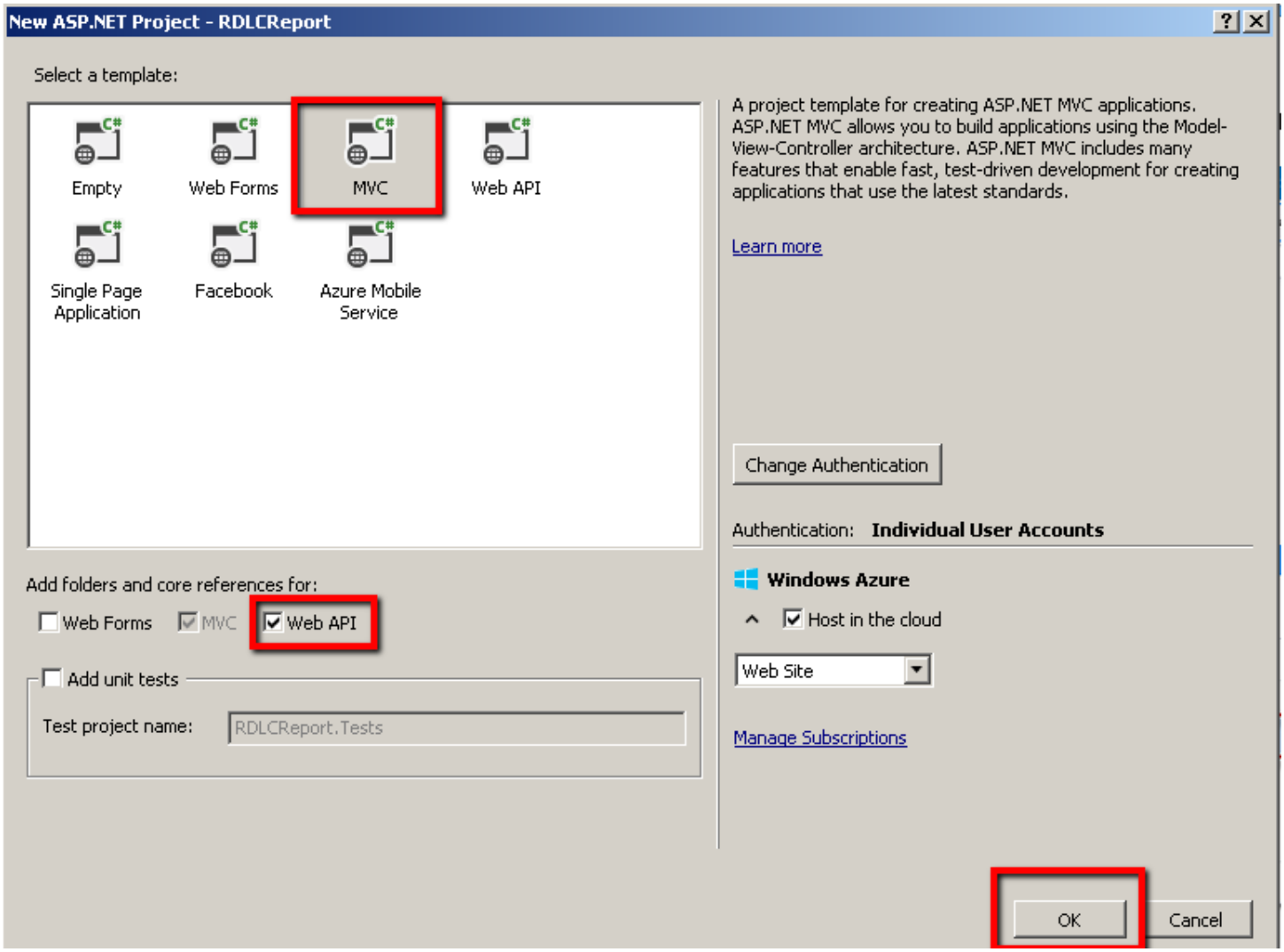
Creating a New Project

2. Create a new Wep API project in Visual Studio 2013 by selecting Project new dialog box appears and select **ASP.NET Web Application** under the Visual C# | Web tab. Make sure **.NET Framework 4.5** is selected, name it **RDLCReport**, choose a Location and click **OK.** .



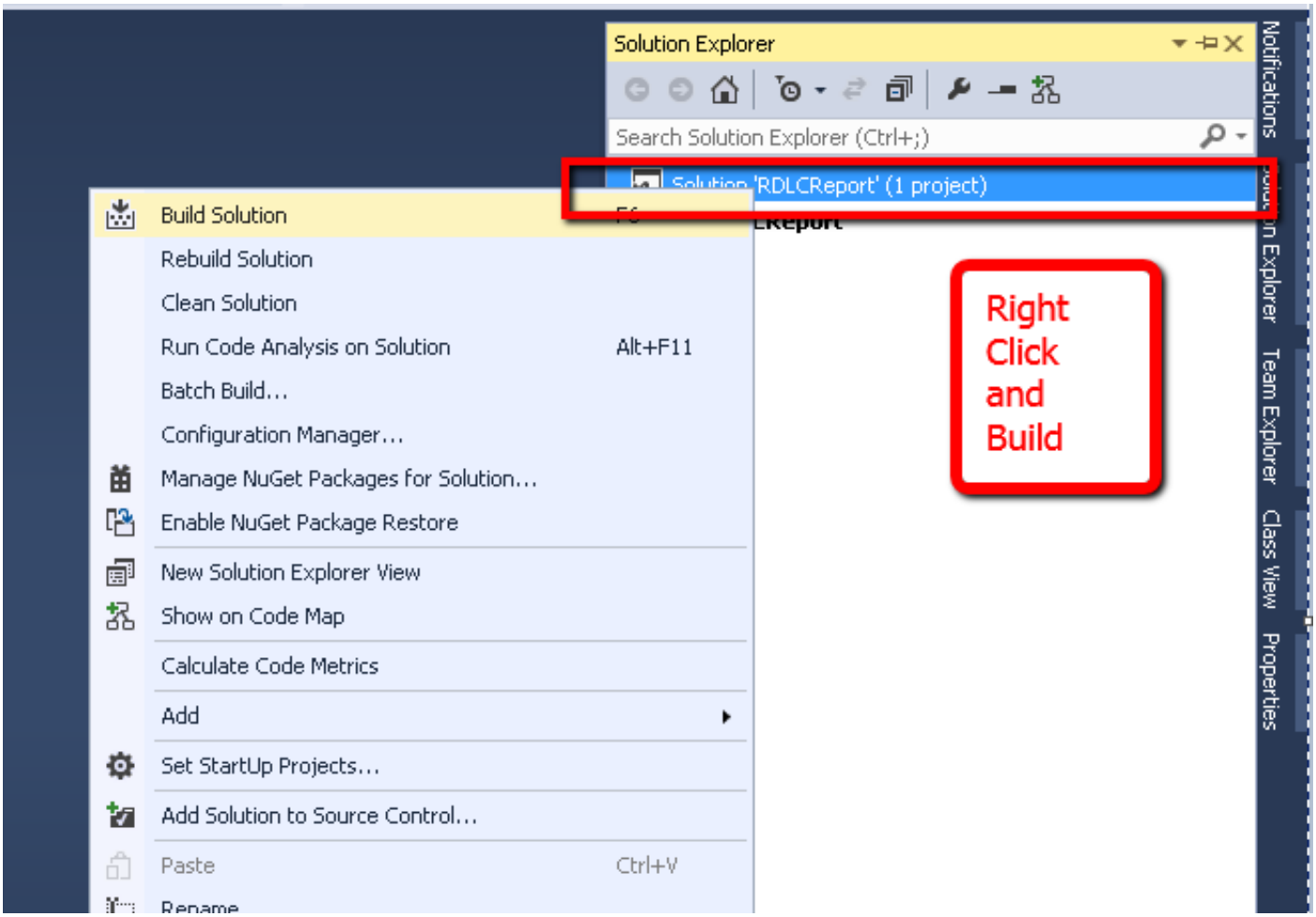
Creating a new ASP.NET Web Application project

3. In the **New ASP.NET Project** dialog box, select the **MVC** template and select the **Web API** option. Also Click **OK** to continue.



Creating a new project with the MVC template, including Web API components

4. In **Solution Explorer**, right-click the **Solution** of the **RDLCReport** project and build.



Build a new solution

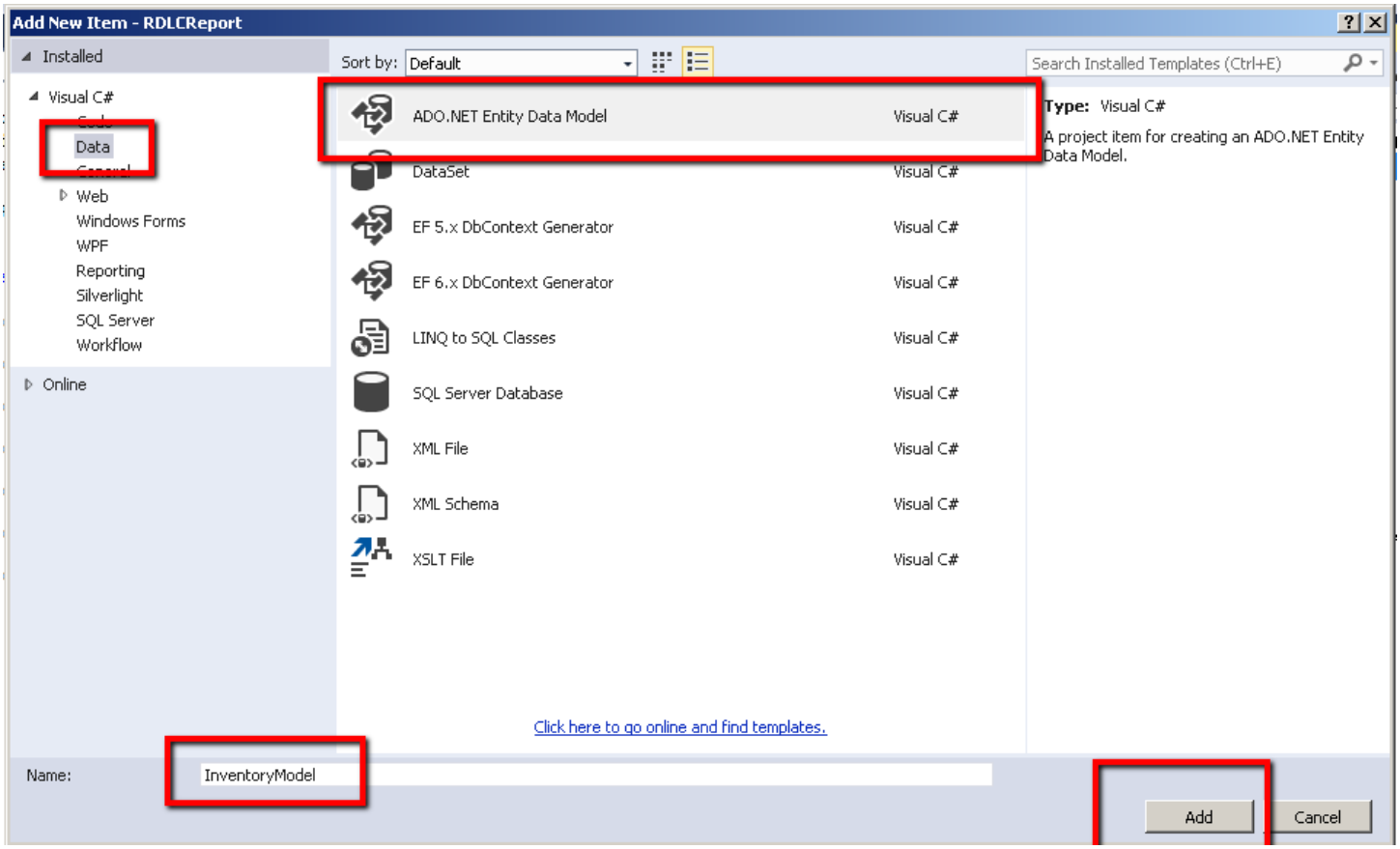
5. Now create database '**Inventory**' and table '**Customer**' and set primary key **Id,** Here is the Script below:

Hide Copy Code

```
USE [Inventory]
GO

DROP TABLE [dbo].[Customer]
GO

SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[Customer](
    [Id] [int] NOT NULL,
    [Name] [nvarchar](50) NULL,
    [Address] [nvarchar](50) NULL,
    [City] [nvarchar](50) NULL,
    [Country] [nvarchar](50) NULL,
    [DateOfBirth] [datetime] NULL,
    [Age] [int] NULL,
 CONSTRAINT [PK_Customer] PRIMARY KEY CLUSTERED
 (
    [Id] ASC
 )WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =
 ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
 ) ON [PRIMARY]
```

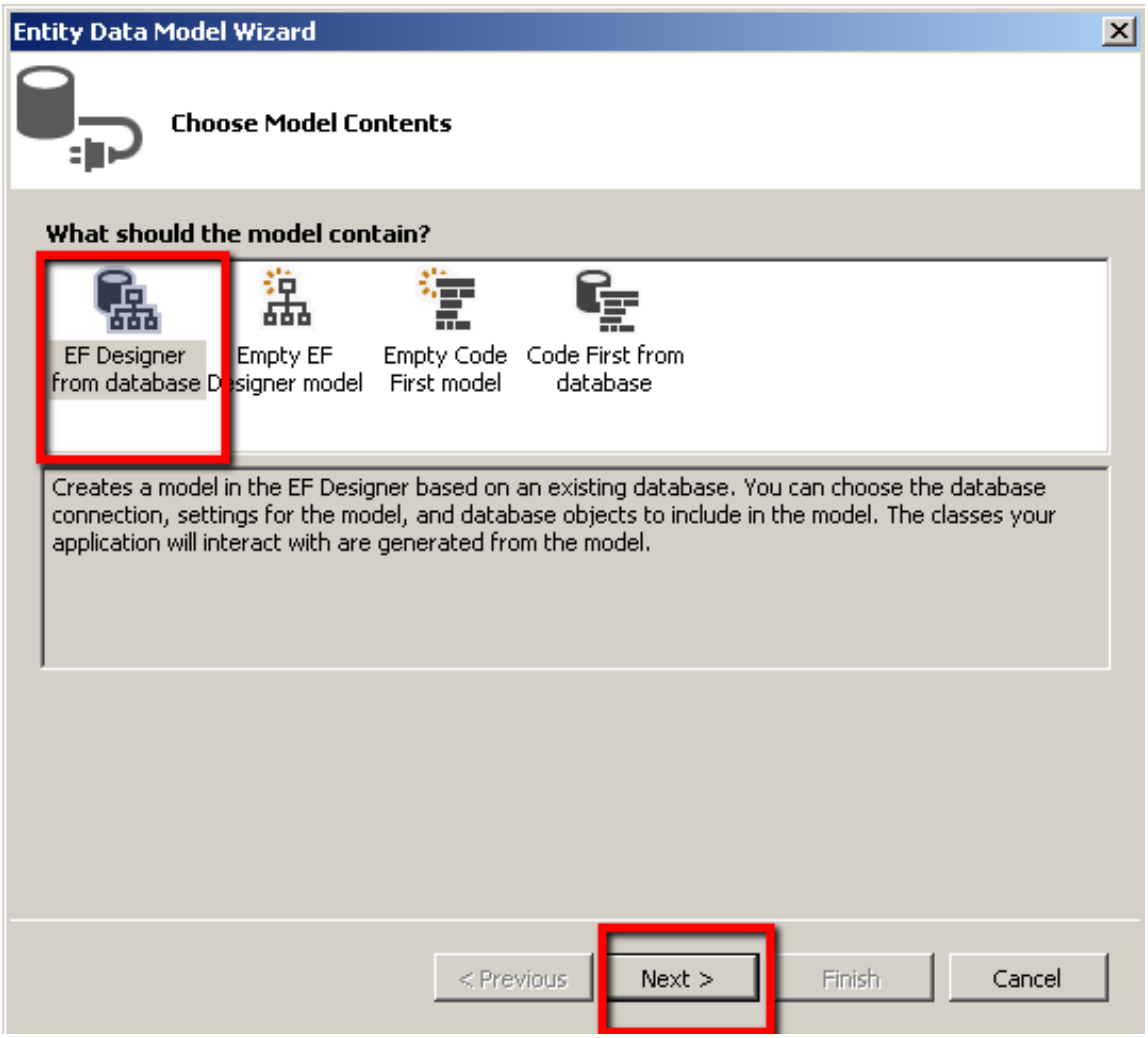
```
GO
```

6. After createing database and table, now you have to create **Entity Frame Work Model Object** by this dialog box**.**



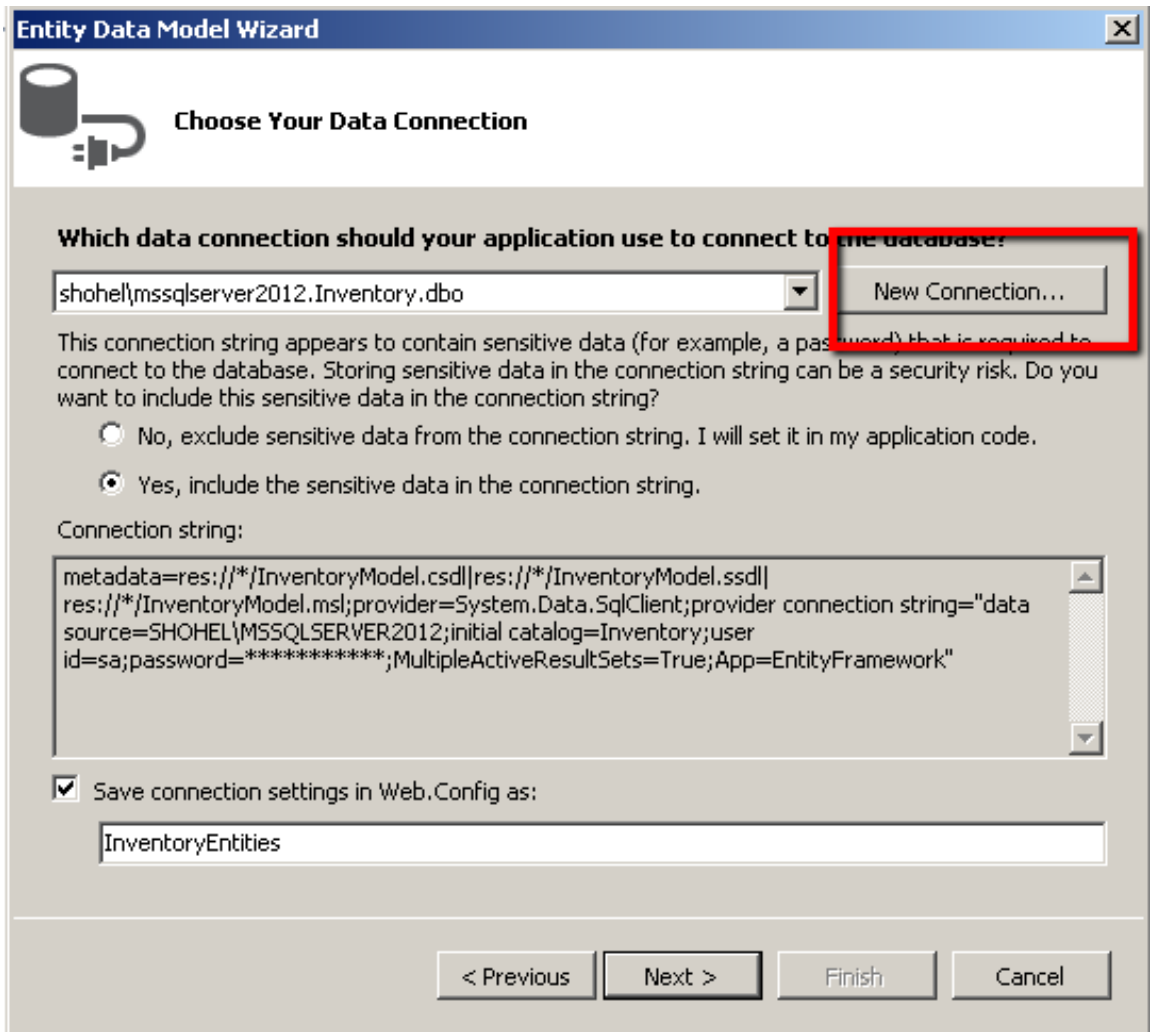
Creating Entity Framework Data Model Object

7. In the Entity Data Model Wizerd, Select **EF Designer From Database,** and also press next button



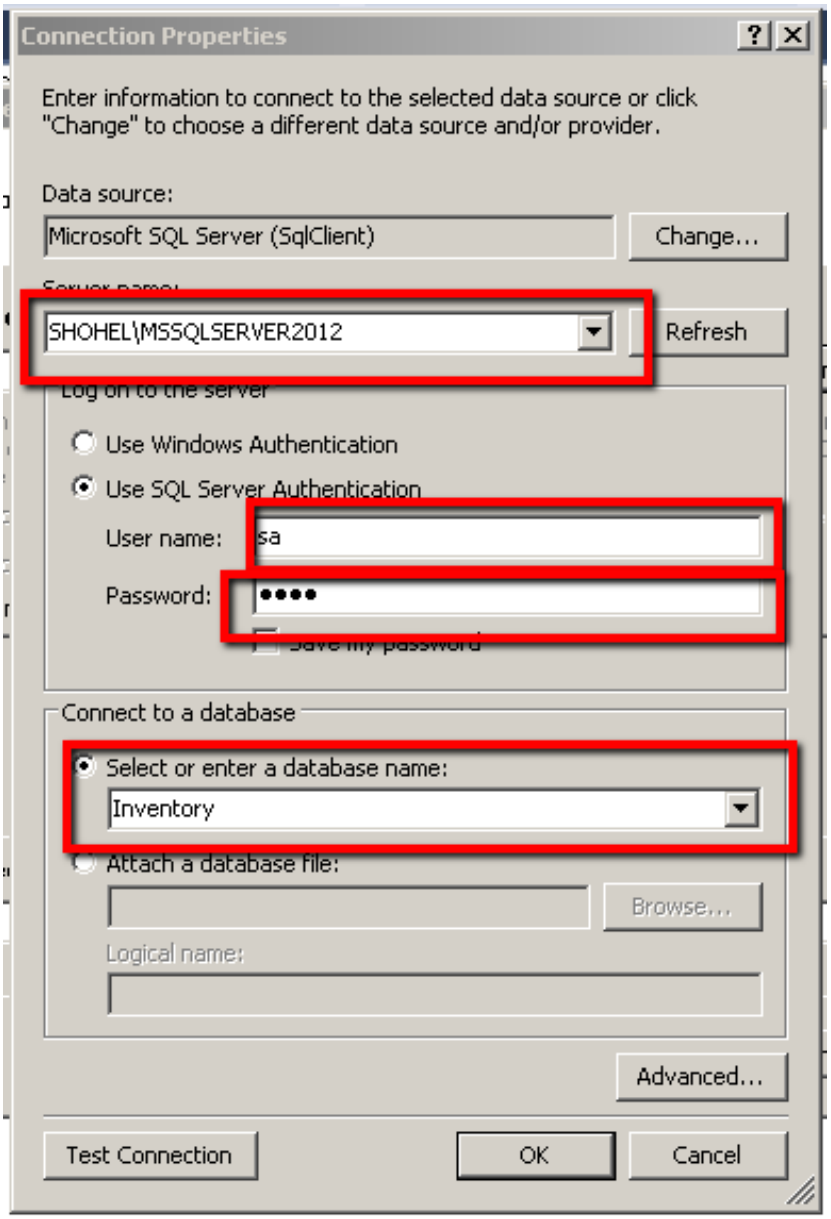
Choose Entity Framework Model Contents

8. In the **Entity Data Model Wizerd** Choose Your Data Connection bu clicking New Connection... button



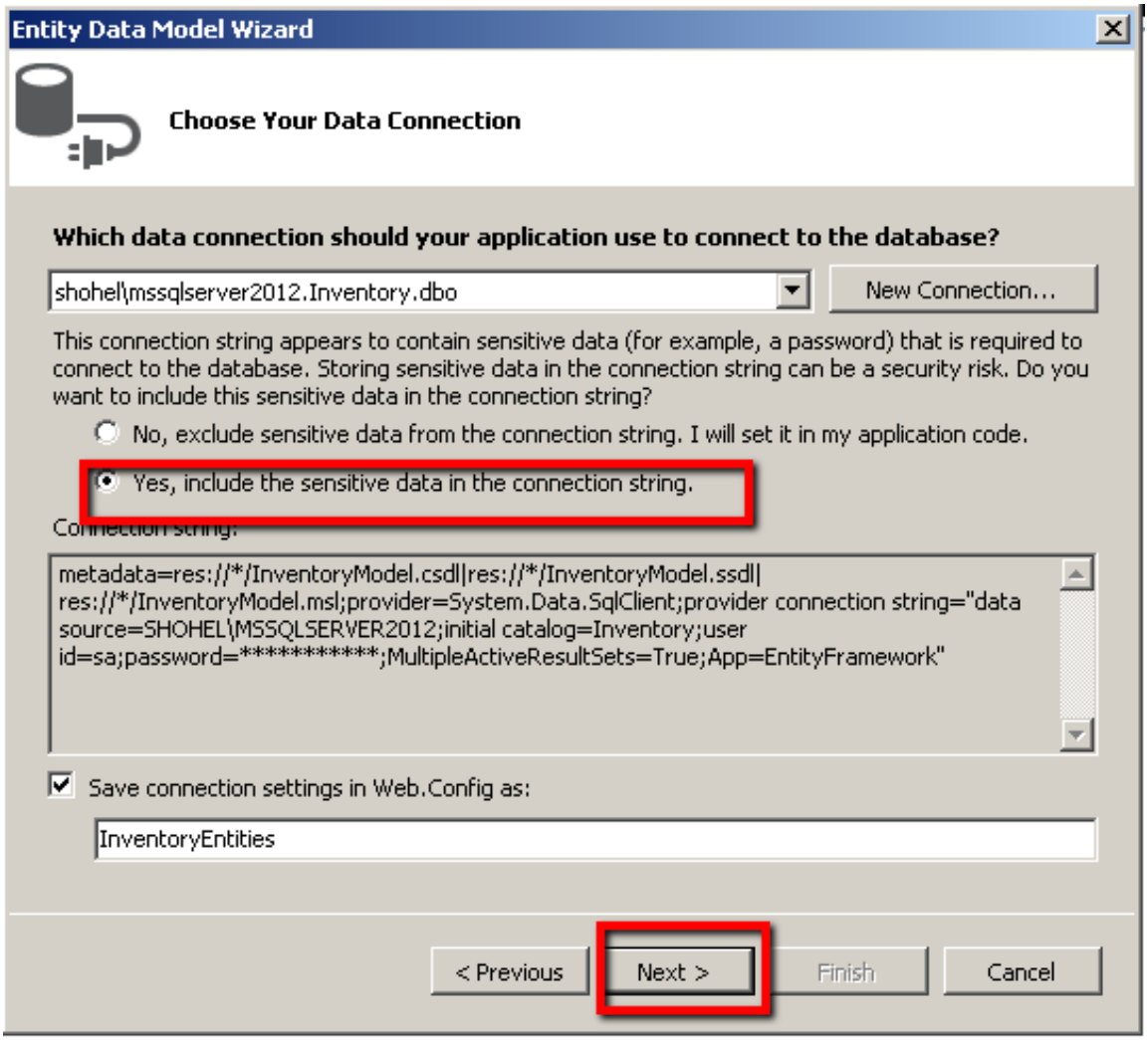
Choose Connection String From Connection Properties Dialog Box

9. In the **Connection Properties Window Dialog box,** write server name, sql authentication and select database name, then finish



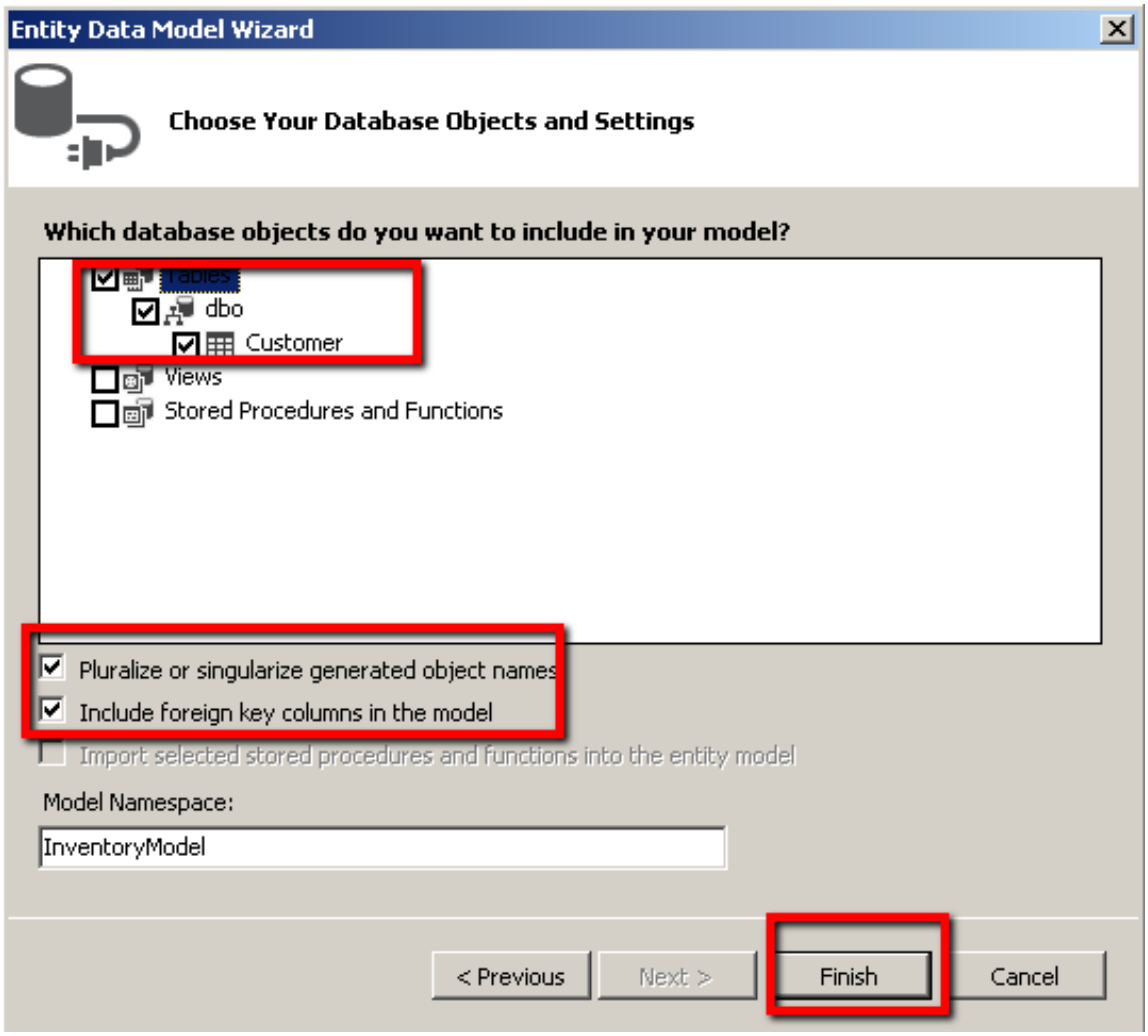
Choose Connection String From this window

10. In the **Entity Data Model Wizerd,** Select yes, Include the sensative data in the connection string, then press next button



Choose senstative data in the connection string.

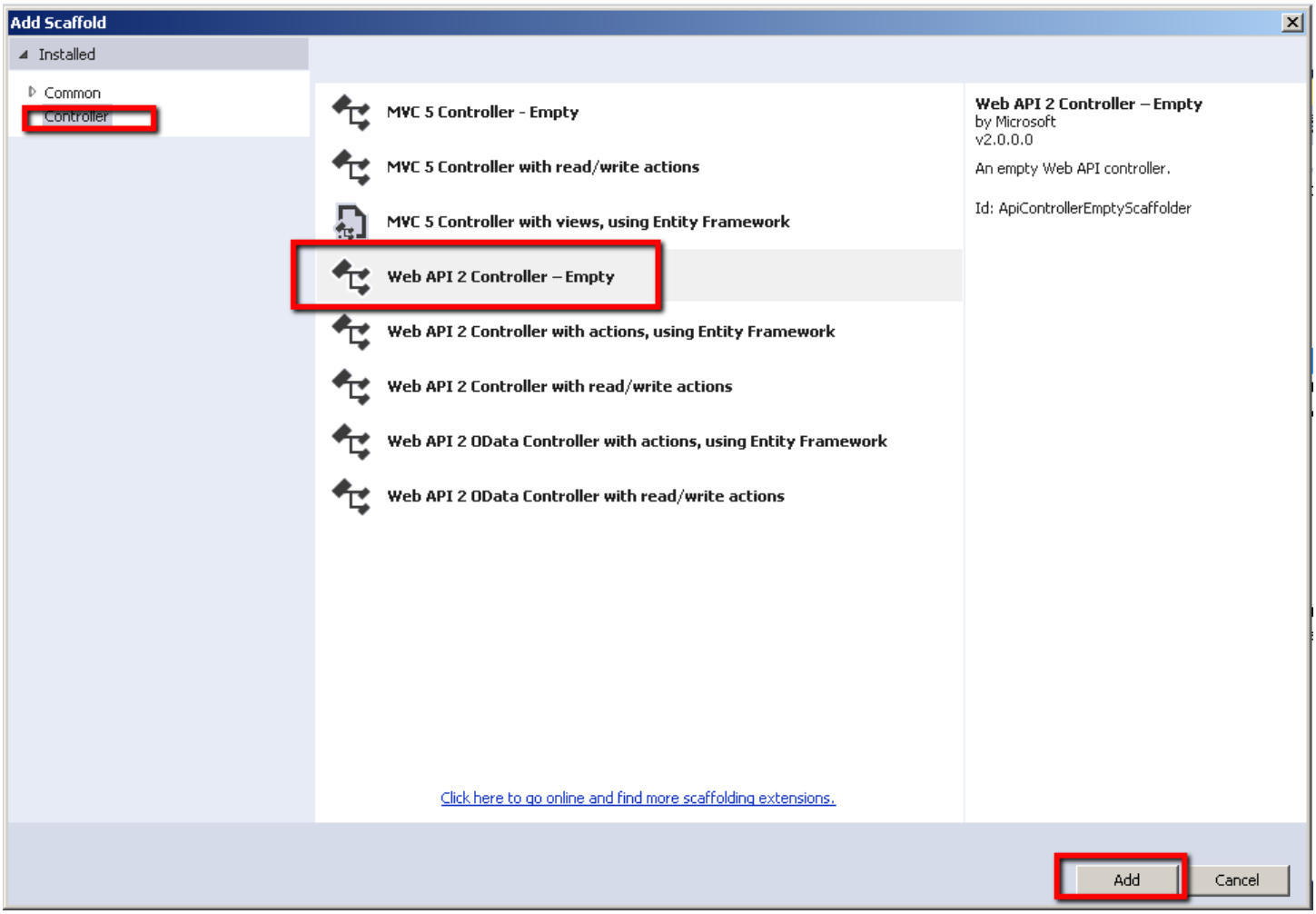
11. In the **Entity Data Model Wizerd,** select table and press finish button



Creating Customer table model object

12. **Build solution** after completing above process

13. In **Solution Explorer**, right-click the **Controllers** folder of the **RDLCReport** project and select **Add | New Controller Item...**. and Now create CustomerController by this way



choose api controller

Choose Api Controller Name

14. The **CustomerController.cs** file is then added to the **Controllers** folder of the **RDLCReport** project, containing an empty Customer**Controller** class. Add the following assembly before **RDLCReport.Controllers** namespace

Hide Copy Code

```
using System;
using System.Collections.Generic;
using System.Data.Entity;
using System.Data.Entity.Infrastructure;
using System.Linq;
using System.Net;
using System.Net.Http;
using System.Web.Http;
```

15. In the **CustomerController.cs file,** Add the following code into the CustomerController class.

Hide Copy Code

```
InventoryEntities db = new InventoryEntities();
```

Hide Copy Code

```
        //get all customer
        [HttpGet]
        public IEnumerable<Customer> Get()
        {
            return db.Customers.AsEnumerable();
        }

        //get customer by id
        public Customer Get(int id)
        {
            Customer customer = db.Customers.Find(id);
            if (customer == null)
            {
                throw new HttpResponseException(Request.CreateResponse(HttpStatusCode.NotFound));
            }
            return customer;
        }
```

Hide Copy Code

```
        //insert customer
        public HttpResponseMessage Post(Customer customer)
        {
            if (ModelState.IsValid)
            {
                db.Customers.Add(customer);
                db.SaveChanges();
                HttpResponseMessage response = Request.CreateResponse(HttpStatusCode.Created,
customer);
                response.Headers.Location = new Uri(Url.Link("DefaultApi", new { id = customer.Id
}));
                return response;
            }
            else
            {
                return Request.CreateErrorResponse(HttpStatusCode.BadRequest, ModelState);
            }
        }
```

Hide Copy Code

```
        //update customer
        public HttpResponseMessage Put(int id, Customer customer)
        {
            if (!ModelState.IsValid)
            {
                return Request.CreateErrorResponse(HttpStatusCode.BadRequest, ModelState);
            }
            if (id != customer.Id)
            {
                return Request.CreateResponse(HttpStatusCode.BadRequest);
            }
            db.Entry(customer).State = EntityState.Modified;
            try
            {
                db.SaveChanges();
            }
            catch (DbUpdateConcurrencyException ex)
            {
                return Request.CreateErrorResponse(HttpStatusCode.NotFound, ex);
```

```
            }
            return Request.CreateResponse(HttpStatusCode.OK);
        }
```

Hide Copy Code

```
        //delete customer by id
        public HttpResponseMessage Delete(int id)
        {
            Customer customer = db.Customers.Find(id);
            if (customer == null)
            {
                return Request.CreateResponse(HttpStatusCode.NotFound);
            }
            db.Customers.Remove(customer);
            try
            {
                db.SaveChanges();
            }
            catch (DbUpdateConcurrencyException ex)
            {
                return Request.CreateErrorResponse(HttpStatusCode.NotFound, ex);
            }
            return Request.CreateResponse(HttpStatusCode.OK, customer);
        }
```

Hide Copy Code

```
        //prevent memory leak
        protected override void Dispose(bool disposing)
        {
            db.Dispose();
            base.Dispose(disposing);
        }
```

Exercise 2: Creating the SPA Interface

To do:

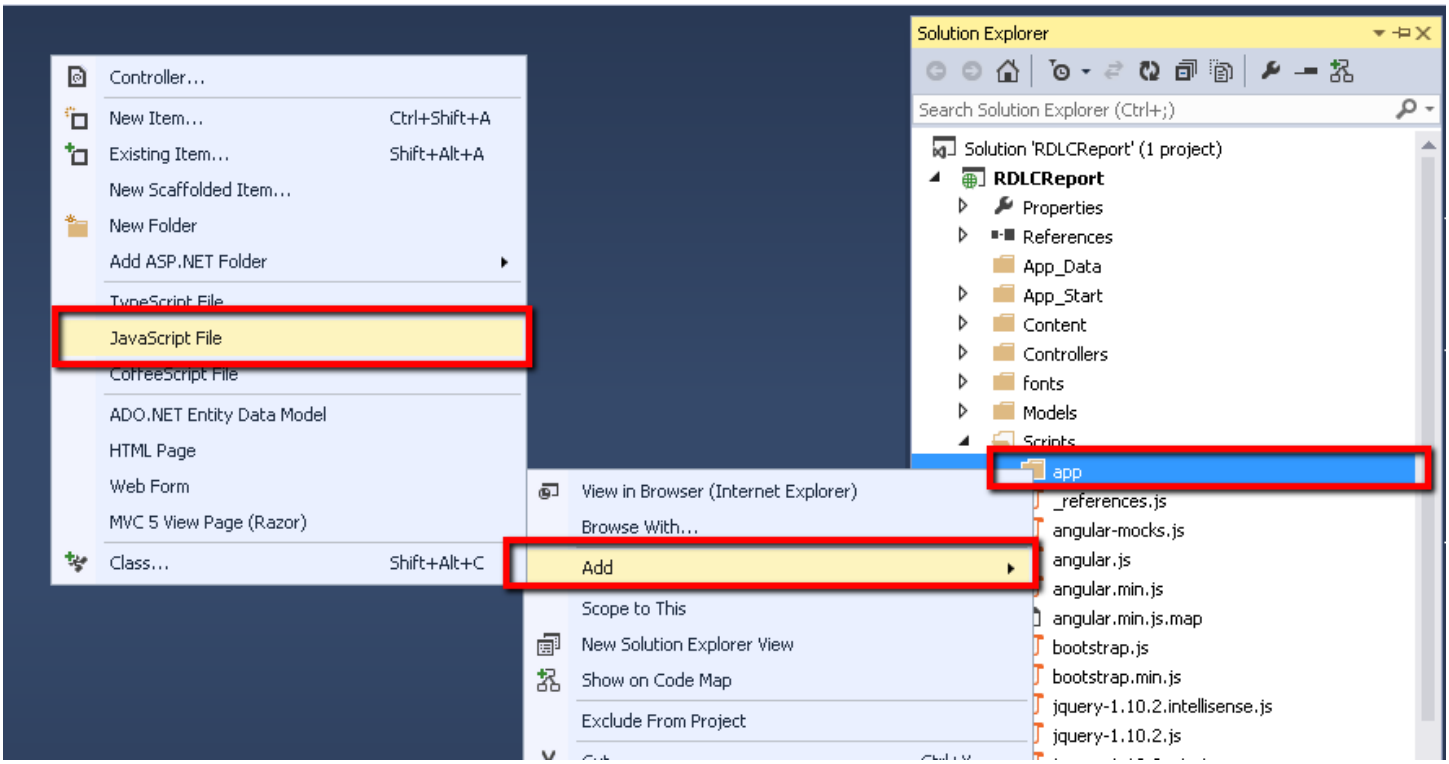
1. Open the **Package Manager Console** from **Tools** > **Library Package Manager**. Type the following command to install the **AngularJS.Core** NuGet package.

Hide Copy Code

```
Install-Package AngularJS.Core
```

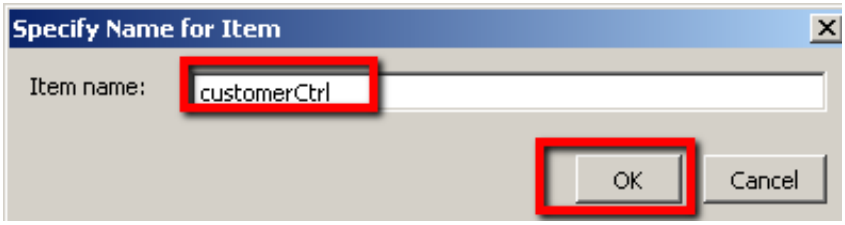
2. In **Solution Explorer**, right-click the **Scripts** folder of the **RDLCReport** project and select **Add | New Folder**. Name the folder **app** and press **Enter**.

3. Right-click the **app** folder you just created and select **Add | JavaScript File**.



Creating a new JavaScript file

4. In the **Specify Name for Item** dialog box, type *quiz-controller* in the **Item name** text box and click **OK**.



Naming the new JavaScript file

5. In the **customerCtrl.js** file, add the following code to declare and initialize the AngularJS.

Hide Shrink ▲ Copy Code

```
//create angularjs controller
var app = angular.module('app', []);//set and get the angular module
app.controller('customerController', ['$scope', '$http', customerController]);

//angularjs controller method
function customerController($scope, $http) {

    //declare variable for mainain ajax load and entry or edit mode
    $scope.loading = true;
    $scope.addMode = false;

    //get all customer information
    $http.get('/api/Customer/').success(function (data) {
        $scope.customers = data;
        $scope.loading = false;
    })
    .error(function () {
        $scope.error = "An Error has occured while loading posts!";
        $scope.loading = false;
    });
```

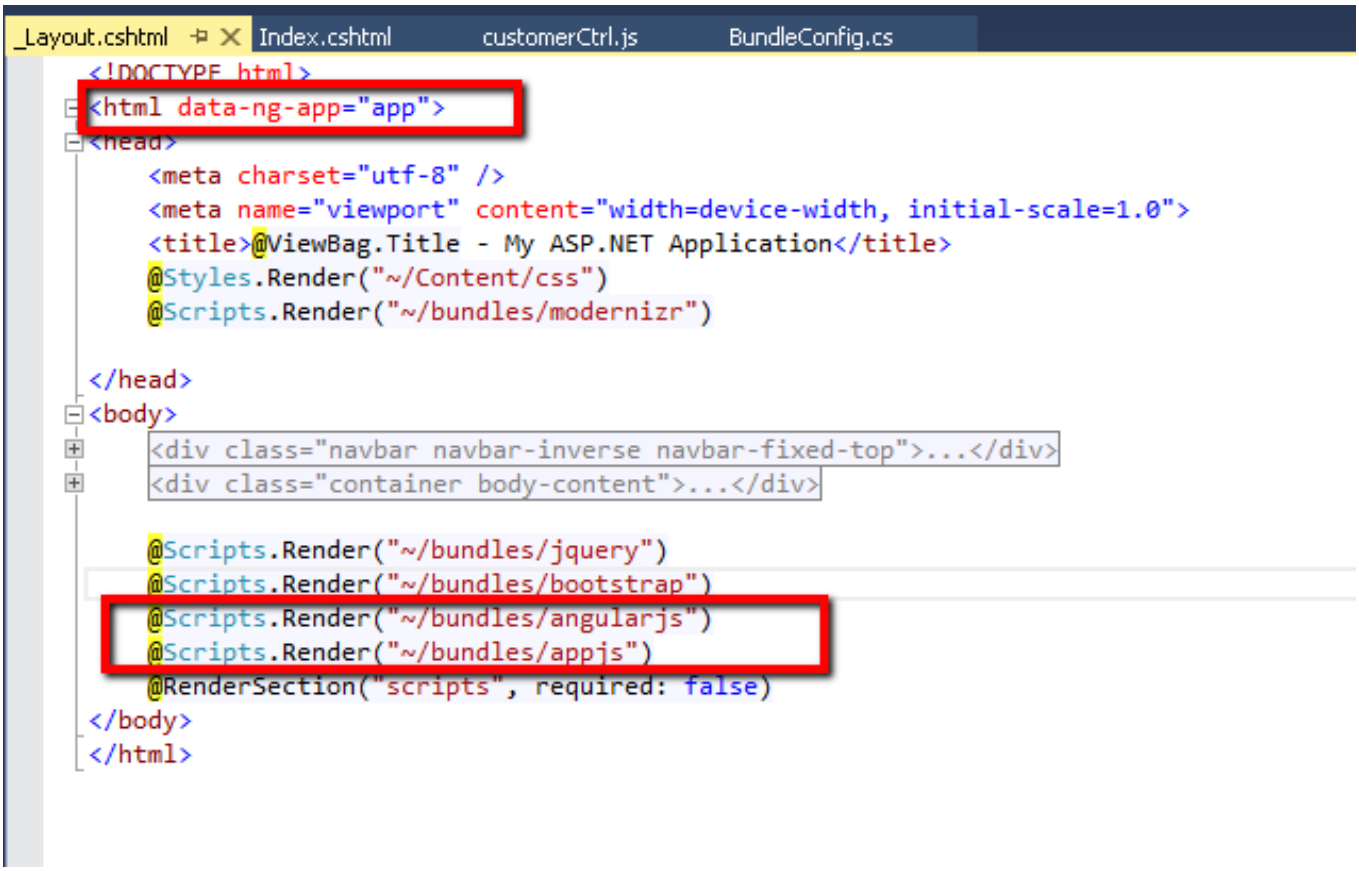
```
        //by pressing toggleEdit button ng-click in html, this method will be hit
        $scope.toggleEdit = function () {
            this.customer.editMode = !this.customer.editMode;
        };

        //by pressing toggleAdd button ng-click in html, this method will be hit
        $scope.toggleAdd = function () {
            $scope.addMode = !$scope.addMode;
        };

        //Inser Customer
        $scope.add = function () {
            $scope.loading = true;
            $http.post('/api/Customer/', this.newcustomer).success(function (data) {
                alert("Added Successfully!!");
                $scope.addMode = false;
                $scope.customers.push(data);
                $scope.loading = false;
            }).error(function (data) {
                $scope.error = "An Error has occured while Adding Customer! " + data;
                $scope.loading = false;
            });
        };

        //Edit Customer
        $scope.save = function () {
            alert("Edit");
            $scope.loading = true;
            var frien = this.customer;
            alert(frien);
            $http.put('/api/Customer/' + frien.Id, frien).success(function (data) {
                alert("Saved Successfully!!");
                frien.editMode = false;
                $scope.loading = false;
            }).error(function (data) {
                $scope.error = "An Error has occured while Saving customer! " + data;
                $scope.loading = false;
            });
        };

       //Delete Customer
        $scope.deletecustomer = function () {
            $scope.loading = true;
            var Id = this.customer.Id;
            $http.delete('/api/Customer/' + Id).success(function (data) {
                alert("Deleted Successfully!!");
                $.each($scope.customers, function (i) {
                    if ($scope.customers[i].Id === Id) {
                        $scope.customers.splice(i, 1);
                        return false;
                    }
                });
                $scope.loading = false;
            }).error(function (data) {
                $scope.error = "An Error has occured while Saving Customer! " + data;
                $scope.loading = false;
            });
        };

    }
```

6. In the **customerCtrl.js,** All of the above code, you can put into this javascript function.

Hide Copy Code

```
(function () {
      'use strict';

   /*Write above code here*/

})();
```

7. In _Layout.cshtml page which path is **Views> Shared> _Layout.cshtml,** add the following line



Creating angular bootstraping

8. In **BundleConfig.cs** file, add two line, which path is **App_Start> BundleConfig.cs**

```
public class BundleConfig
{
    // For more information on bundling, visit http://go.microsoft.com/fwlink/?LinkId=301862
    1 reference
    public static void RegisterBundles(BundleCollection bundles)
    {
        bundles.Add(new ScriptBundle("~/bundles/jquery").Include(
                    "~/Scripts/jquery-{version}.js"));

        bundles.Add(new ScriptBundle("~/bundles/jqueryval").Include(
                    "~/Scripts/jquery.validate*"));

        // Use the development version of Modernizr to develop with and learn from. Then, when you're
        // ready for production, use the build tool at http://modernizr.com to pick only the tests you need.
        bundles.Add(new ScriptBundle("~/bundles/modernizr").Include(
                    "~/Scripts/modernizr-*"));

        bundles.Add(new ScriptBundle("~/bundles/bootstrap").Include(
                  "~/Scripts/bootstrap.js",
                  "~/Scripts/respond.js"));

        bundles.Add(new StyleBundle("~/Content/css").Include(
                  "~/Content/bootstrap.css",
                  "~/Content/site.css"));


        bundles.Add(new ScriptBundle("~/bundles/angularjs").Include(
                  "~/Scripts/angular.min.js"));

        bundles.Add(new ScriptBundle("~/bundles/appjs").Include(
                 "~/Scripts/app/customerCtrl.js"));
    }
```

Add java script file

9. In Views> Home> Index.cshtml file add following code

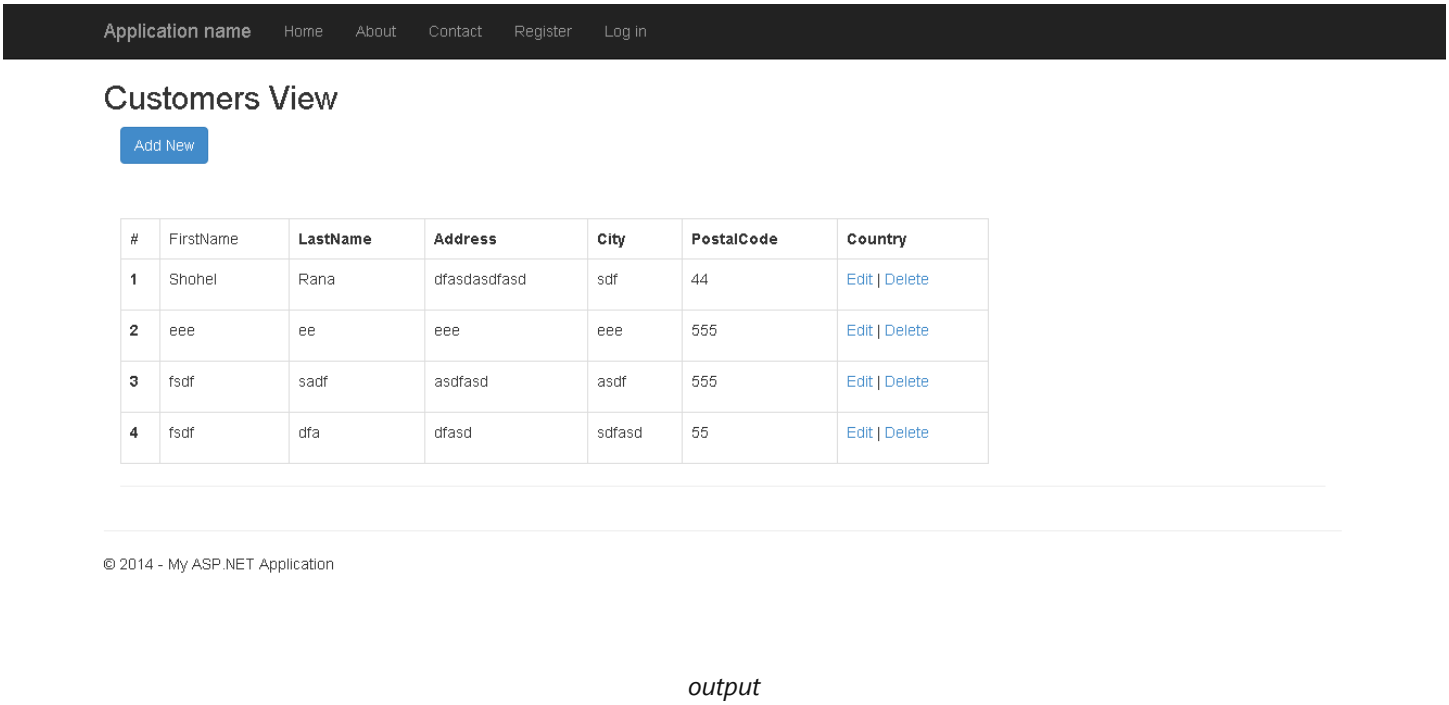
Hide Shrink ▲ Copy Code

```
<div data-ng-controller="customerController" class="container">
    <div class="row">
        <div class="col-md-12">
            <strong class="error">{{ error }}</strong>
            <p data-ng-hide="addMode"><a data-ng-click="toggleAdd()" href="javascript:;" class="btn
btn-primary">Add New</a></p>
            <form name="addCustomer" data-ng-show="addMode" style="width:600px;margin:0px auto;">
                <div class="form-group">
                    <label for="cid" class="col-sm-2 control-label">ID:</label>
                    <div class="col-sm-10">
                        <input type="text" class="form-control" id="cid" placeholder="please enter
id" data-ng-model="newcustomer.Id" required />
                    </div>
                </div>
                <div class="form-group">
                    <label for="cname" class="col-sm-2 control-label">Name:</label>
                    <div class="col-sm-10">
                        <input type="text" class="form-control" id="cname" placeholder="please enter
your name" data-ng-model="newcustomer.Name" required />
                    </div>
                </div>
                <div class="form-group">
                    <label for="address" class="col-sm-2 control-label">Address:</label>
                    <div class="col-sm-10">
                        <input type="text" class="form-control" id="address" placeholder="please
enter your address" data-ng-model="newcustomer.Address" required />
                    </div>
                </div>
                <div class="form-group">
```

```
                    <label for="city" class="col-sm-2 control-label">City:</label>
                    <div class="col-sm-10">
                        <input type="text" class="form-control" id="city" placeholder="please enter
your city" data-ng-model="newcustomer.City" required />
                    </div>
                </div>
                <div class="form-group">
                    <label for="country" class="col-sm-2 control-label">Country:</label>
                    <div class="col-sm-10">
                        <input type="text" class="form-control" id="country" placeholder="please
enter your country" data-ng-model="newcustomer.Country" required />
                    </div>
                </div>
                <div class="form-group">
                    <label for="age" class="col-sm-2 control-label">Age:</label>
                    <div class="col-sm-10">
                        <input type="text" class="form-control" id="age" placeholder="please enter
your age" data-ng-model="newcustomer.Age" required />
                    </div>
                </div>
                <br />
                <div class="form-group">
                    <div class="col-sm-offset-2 col-sm-10">
                        <input type="submit" value="Add" data-ng-click="add()" data-ng-
disabled="!addCustomer.$valid" class="btn btn-primary" />
                        <input type="button" value="Cancel" data-ng-click="toggleAdd()" class="btn
btn-primary" />
                    </div>
                </div>
                <br />
            </form>
        </div>
    </div>
    <div class="row">
        <div class="col-md-12">
            <br />
            <br />
        </div>
    </div>
    <div class="row">
        <div class="col-md-12">
            <div class="table-responsive">
                <table class="table table-bordered table-hover" style="width:800px">
                    <tr>
                        <th>#</th>
                        <td>FirstName</td>
                        <th>LastName</th>
                        <th>Address</th>
                        <th>City</th>
                        <th>PostalCode</th>
                        <th>Country</th>
                    </tr>
                    <tr data-ng-repeat="customer in customers">
                        <td><strong data-ng-hide="customer.editMode">{{ customer.Id }}</strong></td>
                        <td>
                            <p data-ng-hide="customer.editMode">{{ customer.Name }}</p>
                            <input data-ng-show="customer.editMode" type="text" data-ng-
model="customer.Name" />
                        </td>
                        <td>
                            <p data-ng-hide="customer.editMode">{{ customer.Address }}</p>
                            <input data-ng-show="customer.editMode" type="text" data-ng-
model="customer.Address" />
                        </td>
                        <td>
```

```
                            <p data-ng-hide="customer.editMode">{{ customer.City }}</p>
                            <input data-ng-show="customer.editMode" type="text" data-ng-
model="customer.City" />
                        </td>
                        <td>
                            <p data-ng-hide="customer.editMode">{{ customer.Country }}</p>
                            <input data-ng-show="customer.editMode" type="text" data-ng-
model="customer.Country" />
                        </td>
                        <td>
                            <p data-ng-hide="customer.editMode">{{ customer.Age }}</p>
                            <input data-ng-show="customer.editMode" type="text" data-ng-
model="customer.Age" />
                        </td>
                        <td>
                            <p data-ng-hide="customer.editMode"><a data-ng-
click="toggleEdit(customer)" href="javascript:;">Edit</a> | <a data-ng-
click="deletecustomer(customer)" href="javascript:;">Delete</a></p>
                            <p data-ng-show="customer.editMode"><a data-ng-click="save(customer)"
href="javascript:;">Save</a> | <a data-ng-click="toggleEdit(customer)"
href="javascript:;">Cancel</a></p>
                        </td>
                    </tr>
                </table>
                <hr />
            </div>
        </div>
    </div>
    <div id="mydiv" data-ng-show="loading">
        <img src="Images/ajax-loader.gif" class="ajax-loader" />
    </div>
</div>
```

10. Run and enjoy the CRUD in angularjs, the out put is:



output

Conclusion

I think this article will be very helpful for the beginners who are want to learn AngularJS, WebApi, Twitter-Bootstrap, SPA and

SQL-SERVER Data First. I am not good at English, If I mistake anything please forgive me. Thanks for having patience.

License

Share

About the Author

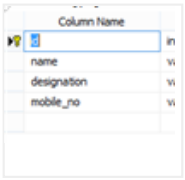
mdshohelrana

Software Developer (Senior)

Bangladesh

I am expert in Asp.Net (WebForm, MVC), Html5, Javascript, jQuery, JSON, CSS3, AngularJs, KnockoutJs, Breezejs, WCF, Restful service, EF, C# .NET, XAML, XML, UML, SQL-SERVER

You may also be interested in...

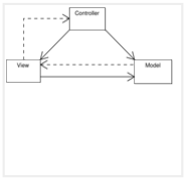


CRUD in ASP.MVC 4 with Angular.js

Developer Tips for Scanning on the Web

CRUD in Angular.js

Add HTML5 Document Viewer to ASP.NET MVC 5 Project



Angular Tutorial - Part 1: Introduction to Angular.js

SAPrefs - Netscape-like Preferences Dialog

Comments and Discussions

You must Sign In to use this message board.

Search Comments Go

First Prev Next

regarding data-ng-model value
Arjun Mourya 13hrs 34mins ago

Just thanks
richard.holguing 10-Sep-15 14:20

CRUD MVC AngularJS sample
Sami L 2-Aug-15 14:01

CRUD in AngularJS project
Roger Rudolph 4-Jul-15 9:12

Customers vs Customer
Member 11757203 10-Jun-15 15:07

Re: Customers vs Customer
mdshohelrana 13-Jun-15 19:23

Re: Customers vs Customer
qwerty@mailinator.com 16-Aug-15 21:06

Re: Customers vs Customer
mdshohelrana 18-Aug-15 16:06

Angular getting started
msman88 10-Jun-15 0:25

Great Article
alexander ocampo hidalgo 7-Jun-15 19:11

Re: Great Article
mdshohelrana 8-Jun-15 20:49

Re: Great Article

alexander ocampo hidalgo 9-Jun-15 17:27

Re: Great Article
mdshohelrana 13-Jun-15 19:49

Re: Great Article
mdshohelrana 13-Jun-15 22:03

CRUD with SPA, ASP.NET Web API and Angular
latif127 3-Jun-15 18:37

Excellent!
Zenu 25-May-15 10:44

Re: Excellent!
mdshohelrana 26-May-15 16:24

Another Newbie Question
Member 11676328 9-May-15 6:23

My vote of 5
M Rayhan 3-Feb-15 20:07

Crud using AngularJS and MVC Web API
Member 10527748 9-Jan-15 18:52

Re: Crud using AngularJS and MVC Web API
mdshohelrana 11-Jan-15 2:08

Tried fitting to my own schema... struggling w/ edit function
Member 10983977 7-Jan-15 8:56

Re: Tried fitting to my own schema... struggling w/ edit function
mdshohelrana 7-Jan-15 19:37

Good one
zala gopal 23-Dec-14 23:13

Worked the first time!
Member 11067451 12-Nov-14 13:48

Re: Worked the first time!
mdshohelrana 15-Nov-14 23:09

My vote of 5
Mahsa Hassankashi 9-Nov-14 2:09

Some issues (Ajax Loader, 405 on Put Verb)
Rinaldo J. Sassaroli 29-Oct-14 9:19

Re: Some issues (Ajax Loader, 405 on Put Verb)
Rinaldo J. Sassaroli 29-Oct-14 9:40

Re: Some issues (Ajax Loader, 405 on Put Verb)
mdshohelrana 29-Oct-14 16:26

Re: Some issues (Ajax Loader, 405 on Put Verb)
mdshohelrana 29-Oct-14 16:21

newbie question
benny856694 **27-Oct-14 21:23**

Re: newbie question
mdshohelrana 27-Oct-14 22:26

Re: newbie question
benny856694 27-Oct-14 22:28

My vote of 5
Humayun Kabir Mamun **26-Oct-14 19:36**

Refresh 1

General News Suggestion Question Bug Answer Joke Rant Admin

Use Ctrl+Left/Right to switch messages, Ctrl+Up/Down to switch threads, Ctrl+Shift+Left/Right to switch pages.