CODE PROJECT®
For those who code

Learn MVC Project in 7 Days – Bonus day 1

Marla Sukesh, 27 Jul 2015

This article is a continuation article of the series "Learn MVC project in 7 days

Download Day7Bonus.zip

Introduction

This article is a continuation article of the series "Learn MVC project in 7 days".

- In the first 6 articles we have created a completed MVC Project from the scratch.
- In 7th day we have created a Single page version of same project.
- This one is going to be a bonus day for readers. Here we will understand importance of Web API and angular and see a small demo on both of them.

Note: In this series I am not going to create a complete project using Web API and angular. I am going to start a new series for that soon. Here we will just cover following three things.

I request you to jump to the respective section directly based on your interest.

Complete Series

- Day 1
- Day 2
- Day 3
- Day 4
- Day 5
- Day 6
- Day 7
- Bonus Day 1
- Bonus Day 2

We are pleased to announce that this article is now available as hard copy book you can get the same from www.amazon.com and www.flipkart.com

What and Why Web API?

Web API is an Asp.Net technology for creating HTTP based services. Just like Web Forms and Asp.Net MVC, Web API is a flavour of Asp.Net hence it automatically get many abilities of Asp.Net like Forms Authentication, Exception handling etc.

Let's answer couple of questions about Web API. If you are not interested in theory of Web API I recommend you to directly jump to the lab.

Web API is a third flavour of Asp.Net. Does it mean that, using Web API we create web applications?

We are very clear about the purpose of Asp.Net. It's meant for creating web applications.

Web applications are applications which can be accessed via Web. It's never been sentenced that "Web Applications must have UI". They can be just some business applications which exposes data via web.

Asp.Net Web Forms and Asp.Net MVC are the flavours of Asp.net which caters the UI part. It will be used to create Web based UI applications.

How Web API is different from Web Services and WCF Services?

Web Service and WCF Service does the same thing. It exposes data to the outside world.

Then how come Web API is different from them?

Short answer is "Web Services and WCF Services are SOAP based services. Here HTTP will be just used as a transport protocol. Web API is purely HTTP based service. Here there will not be any SOAP involvement".

Let's understand it in detail.

Why WCF and Web Services need SOAP?

In order to understand this let's talk about a scenario.

We want to send some data from Technology1 to Technology2 where Technology2 is a WCF/Web Service and Communication must happen via internet/web.

Web means HTTP. HTTP will let us make requests to a particular URL and with that request it let us pass some data as well.

Now as per the data transmission discussion we had in Day 7, Technology1 will convert its data to XML string (or JSON string mostly) and then send it to Technology2.

Note: When .Net Web services or WCF services is the point of communication, data will be always formatted as XML string.

Point's worth discuss

- When a developer design a service he/she actually creates one or more methods inside service. Each method is meant for some special functionality. We call it web methods. When it comes to communication,Technology1 will actually invoke these methods fromthe service. Now in HTTP we won't have a provision for specifying method name. The only choice left is, when client sends request via HTTP,somehow method name will be concatenated to data and will be sent as data itself.
  At service side, Service framework splits data (received from client) into actual data and method name. Then it invokes the corresponding method in the service passing actual data as an argument.
- When Technology1 will send data to the service, how come service make sure that data is sent from valid client. It may be possible that someone who is not allowed to access the service is tryingto send the data. To solve this problem just like method name solution Technology1 will concatenate Validation credentials to the data and send them as one. At service side, Service framework splitsdata (received from client) into Credentials, method name and actual data. It validates the credentials and make sure that request is an authenticated request before making web method call.

How Service extracts different parts from incoming data?

Now as per our final discussion, When Technology1 send data to service, data will actually contain three things – Actual Data, Validation credentials and method name.

At service side it will be very difficult for a service framework to understand each part independently because Service is a universal thing and it should work for everyone. Secondly client can send data in any format. Example, client can send complete data in "Data|Credential|MethodName" format or can send in "Credential|Data|MethodName" format. There is no fixed format.

Solution - Standard envelop

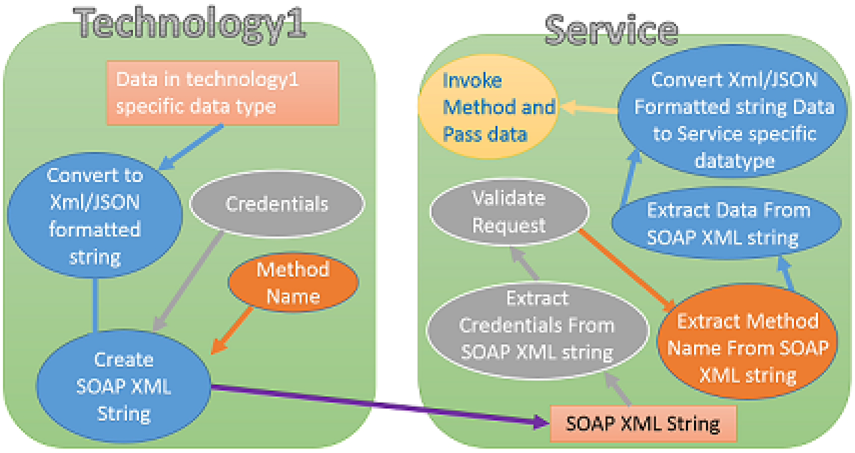
To tackle this problem industry came up with a standard envelop concept called SOAP envelop.

SOAP is nothing but a specialized XML which will encapsulate everything which need to be sent but in standard format. HTTP will send SOAP envelop from client to service and vice versa.

Below is the sample SOAP xml for the reference.

```
<?xml version="1.0"?>
<soap:Envelope

xmlns:soap="http://www.w3.org/2001/12/soap-envelope"

soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
<soap:Body xmlns:m="http://www.example.org/stock">
```

```
    <m:MyMethodName>
      <m:MyData>Data in Json or XML format</m: MyData >
    </m: MyMethodName >
  </soap:Body>
</soap:Envelope>
```


Technology1
Data in technology1 specific data type
Convert to Xml/JSON formatted string
Credentials
Method Name
Create SOAP XML String
Service
Invoke Method and Pass data
Convert Xml/JSON Formatted string Data to Service specific datatype
Validate Request
Extract Data From SOAP XML string
Extract Credentials From SOAP XML string
Extract Method Name From SOAP XML string
SOAP XML String

Note:

- As I said before when Service is Web Service or WCF service data will be represented as XML string.
- When a service returns something same steps will be followed. Service will create SOAP XML stringand client will extract response data from it.

What are the problems with SOAP?

SOAP is XMLat the end of the day, so it causes two issues.

- It increases the overall size of the data hence affects the performance.
- We spoke about complexities of XML in Day 7. We have learned how much difficult it will be for lightweight frameworks (like JavaScript, Mobile frameworks etc)tocreate and parse XML string.

What will be the solution?

Solution will be pure HTTP based services. Services where there will not be any SOAP envelop. HTTP will directly transfer data from one location to another. This is where REST principle comes to picture.

REST says use the existing features of the web in more efficient manner and create services. It say transfer data in the format they represented instead of putting them in SOAP envelop.

HTTP based services are completely based on REST principle hence they are also called as REST based services.

What are the existing features of web?

- In web we have a concept of URL which will uniquely identifies a Resource.
- In web different actions will be identified with the help ofdifferent HTTP verbs. Example – HTTP verb GET means Retrieve data, PUT means update data, POST means create data and DELETE means Delete the data.

How industry implemented the solution?

Solution for Method Name

Now in case of SOAP based service (Web and WCF service) every service is identified by an URL. In the same way REST service will be identified with the help of an URL.

Difference will be in case of REST service, the way we defined methods is going to be different.

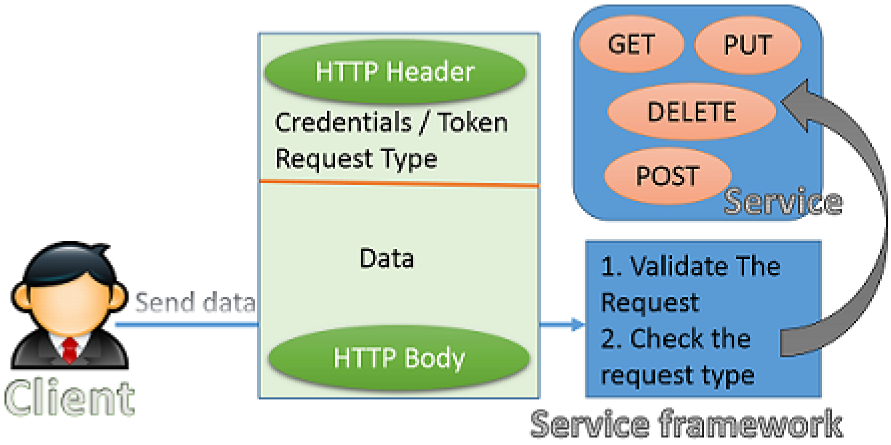
- Either it will be definedas standard methods. Standard method means method name will be fixed. It will either GET or POST or PUT or DELETE.
- Or it will define methods with any name but in that case every custom method will have a URL.

In either of the case sending method name as a parameter from client side is not required.

- In case of choice 1, client will make make a request to REST service via HTTP. Type of the request will be either GET or POST or PUT or DELETE. Now at the service end, service framework will check the type of request and correspondingly invoke the method in the service. Method name problem get solved.
- In case of choice 2, client will make a request to RESET service method directly via HTTP. This is possible because every method is going to have a unique URL.

Solution for security

When we say HTTP, we have a concept of HTTP header and HTTP body both of which have the ability to carry some information. We can always pass credentials as a part of header and data as part of body. To make sure that nobody able to see the data in header and body during transmission we can implement SSL. It make one thing clear, Security will not be an issue.



<u>What about the passing data?</u>

In this case data will passed directly by HTTP in JSON or XML format. Mostly it will be JSON.

How come WCF REST is different from Web API?

WCF Rest was an earlier Microsoft implementation for creating REST based services.

WCF was never meant for REST. Its sole purpose was to support SOA. Creating REST services using WCF involved too many steps and adding new feature in the future would have been big issue for Microsoft.

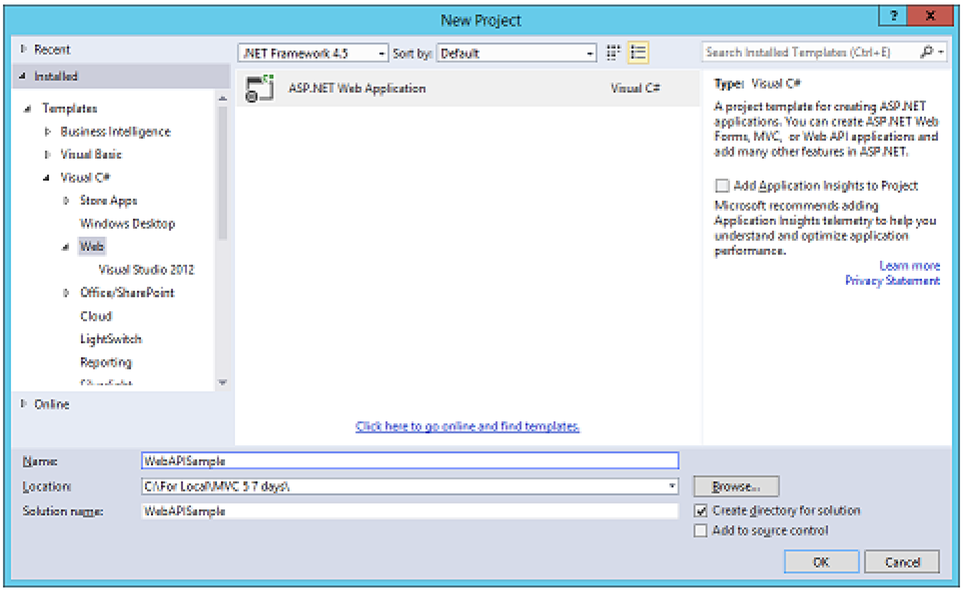
A Non WCF developer will not be able create WCF REST service.

Asp.Net Web APIwill be much simpler and even a Non WCF developer will be able to create HTTP based service using it.

Demo on Web API

Step 1 - Create a Web API project

Step 1.1 Open Visual studio and Click on File>>New>>Project. Select “Asp.Net Web Application” from the dialog box, put some name and click Ok.



.NET Framework 4.5

Sort by: Default

ASP.NET Web Application Visual C#

Type: Visual C#

A project template for creating ASP.NET applications. You can create ASP.NET Web Forms, MVC, or Web API applications and add many other features in ASP.NET.

□ Add Application Insights to Project

Microsoft recommends adding Application Insights telemetry to help you understand and optimize application performance.

Name: WebAPISample

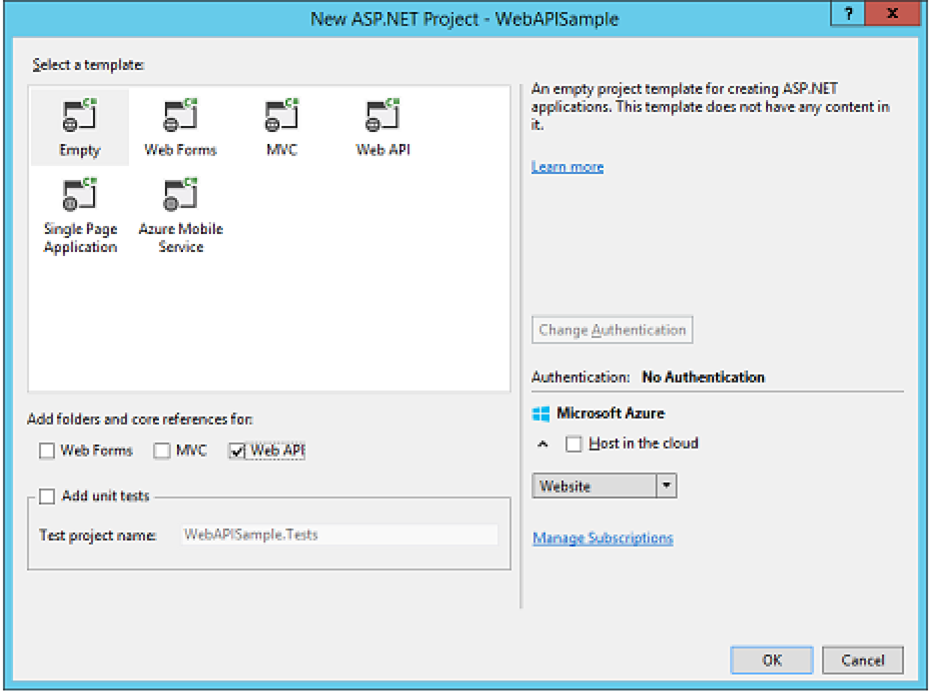
Location: C:\For Local\MVC 5 7 days\

Solution name: WebAPISample

☑ Create directory for solution

□ Add to source control

Step 1.2 Select Empty option,Web API checkbox in the next dialog box and click Ok.



New ASP.NET Project - WebAPISample

An empty project template for creating ASP.NET applications. This template does not have any content in it.

Authentication: **No Authentication**

Add folders and core references for: ☐ Web Forms ☐ MVC ☑ Web API

☐ Add unit tests

Test project name: WebAPISample.Tests

☐ Host in the cloud

Now before we proceed I would like to talk more about above this dialog box. In Day 1 where we started our MVC project I purposely skipped the discussion on this dialog box.

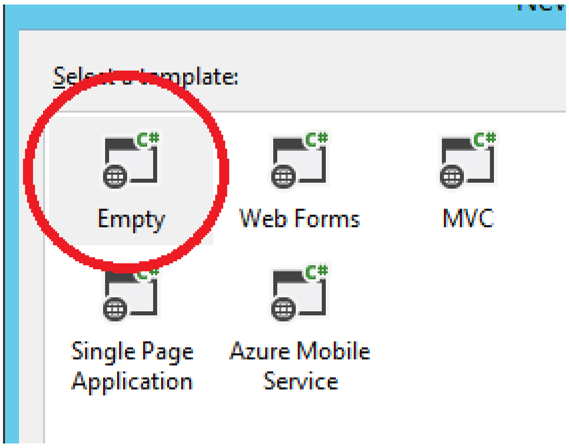
Now in future there is not going to be separate Asp.Net MVC, Web API or Web Forms Project. We have Asp.Netand we will create Asp.Net project.

In a single Asp.Net project at the same time we may have couple of options of type Web Forms and couple of options of type MVC.

There are two sections in dialog box – Template and References.

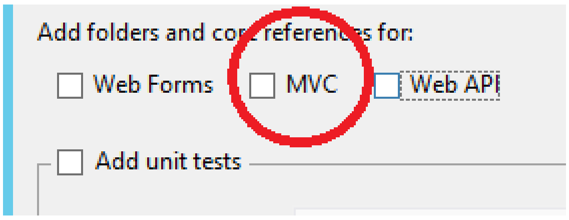
Explanation 1

Let’s say we choose Empty template in the beginning.



Now if you want to create a MVC option you have to add couple of references in the project. So manually right click the project, select add reference, choose System.Web.MVC and click Ok. Now do the same for some more references.

Instead of that simply check MVC checkbox in the dialog box.



Add folders and core references for: ☐ Web Forms ☐ MVC ☐ Web API

☐ Add unit tests

Above two options will create an empty project with all the referencesrequired for Asp.Net MVC project.

If you believe that your project is also going to contain Web Forms or Web APIsthen simply check the corresponding checkbox as well.

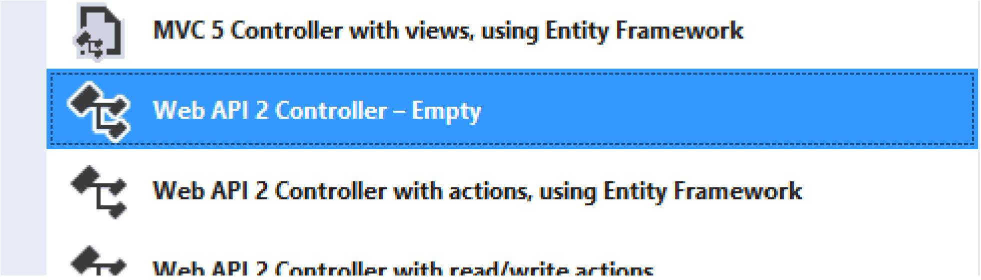
Explanation 2

Now you might be wondering what the other templates in the dialog box are?

It’s simple. Microsoft believe every project is going to ahve few things common. Like every project is going to have Login screen, master or layout page, error page etc. Microsoft says instead developer go and create them manually every time in every project let visual studio create it automatically. If you take Web Forms template all these common files will be created in Web Forms style. If you take MVC template all these common files will be create in MVC style.

Step 2 –Create API

Right click the controller folder and select Add>>Controller. This time choose “Web API 2 controller – Empty” option and click Add.


MVC 5 Controller with views, using Entity Framework
Web API 2 Controller – Empty
Web API 2 Controller with actions, using Entity Framework
Web API 2 Controller with read/write actions
Id:

Put name of the controller as EmployeeController and click Add.

You will notice that a new class get created inheriting from ApiController. It’s going to be our Web API.

Step 3 – Create Model

Create a new class called Employee inside Model folder as follows.

```
namespace WebAPISample.Models
{
    public class Employee
    {
        public string FirstName { get; set; }
        public string LastName { get; set; }
        public int Salary { get; set; }
    }
}
```

Step 4 – Create GET action

Open EmployeeController put using statement as follows.

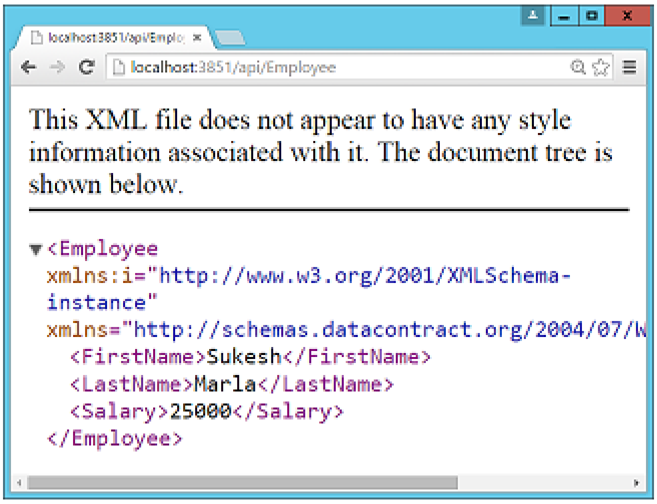
```
using WebAPISample.Models;
```

Now create GET action as follows.

```
public Employee GET()
{
    Employee e = new Employee();
    e.FirstName = "Sukesh";
    e.LastName = "Marla";
    e.Salary = 25000;
    return e;
}
```

Step 5 – Execute and Test

Press F5 and execute the application. Make request to “/api/Employee” from browser.



This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<Employee
  xmlns:i="http://www.w3.org/2001/XMLSchema-
  instance"
  xmlns="http://schemas.datacontract.org/2004/07/W
    <FirstName>Sukesh</FirstName>
    <LastName>Marla</LastName>
    <Salary>25000</Salary>
  </Employee>
```

Note: Above output will be applicable for only chrome browser. We will talk about in detail in next section.

Talk on Web API Lab

Why Api keyword is specified in the URL?

Open Global.asax file. Application_Start will be defined as follows.

```
protected void Application_Start()
{
    GlobalConfiguration.Configure(WebApiConfig.Register);
}
```

In the project, in App_Start folder you will find a file called WebApiConfig.cs. Open it.

It contains following code.

```
namespace WebAPISample
{
    public static class WebApiConfig
    {
        public static void Register(HttpConfiguration config)
        {
            // Web API configuration and services

            // Web API routes
            config.MapHttpAttributeRoutes();

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

As you can see, Web API specific route is defined inside it.

This is the reason why “api” keyword is required.

How come GET method got invoked?

In this case our HTTP service“Employee” is considered as a resource which will be identified by an URL “http://localhost:somePort/api/Employee”. When HTTP request is sent to this URL, Web API engine check the type of request. In our case type of request is “Get” hence Web API engine make Get method got invoked.

How come the request type becomes Get?

In this example we are using browser’s address bar to make the request. Request made by browsers address bar will always leads to get request.

What about other requests?

In Web API simply create four methods named GET, POST, PUT, DELETE respectively.

Each method will depict one request type.

Now from a testing point of view GET method can be tested easily via browser’s address bar but POST, PUT, DELETE cannot be directly requested via browser.Wehave to use some tool or some UI which will does that for us with the help of some coding.

We are going to have a practical lab on same soon.

What if we want to define custom methods?

In real time scenario we always want to create more than one kind of Get/post/put/delete functions. Example – GetCustomers, GetCustomerById, GetCustomerByDepartment.

In that case we have to define multiple routes or we have to change the default route.

Look at the following example.

```
config.Routes.MapHttpRoute(
                name: "SecondRoute",
                routeTemplate: "api/GetCustomerById/{id} ",
                defaults: new {controller= "Customer", action= "GetCustomerById" }
            );
            config.Routes.MapHttpRoute(
                name: "ThridRoute",
                routeTemplate: "api/{controller}/{action}"
            );//generic route like Asp.Net MVC
```

How come it’s confirmed that its REST service?

Look at the response. There is no SOAP tags involved. We got direct data.

Similarly at the time of request we didn’t sent any SOAP XML. Simply made the request using HTTP and done.

Why XML is returned as a response?

When we made a request to the service, it returned data in the form of XML.

Before I answer this, let me make you clear one thing. Yes, data is in XML format but SOAP envelop is gone☻

Now as per the XML format is concerned, data cannot be send between different technologies directly. It must be converted to either XML or JSON string and then transferred. In case of Web API we have returned pure employee data which will get converted to XML string by Web API engine and returned.

Do you think it’s an issue? Actually it’s a feature ☻

Web API have a very nice feature called Content negotiation. Client can negotiate for the output content. If client asks for XML, Web API will return XML and if client asks for JSON, Web API will return JSON.

In the last test I have used chrome browser. Check the same example in Internet explorer. You will get JSON as response.

Does it mean that my web application will have compatibility issues?

No. Basically end user will never access Web APIs directly via browser’s address bar. They will use some UI. UI will talk to Web API with the help of some programming languages like C#, JavaScript etc. It means background code which access Web API is always going to be same. When we access Web API using C# or JavaScript by default we get JSON but we can also specify response content type and get XML.

What is Angular?

We use JavaScript to make static HTML dynamic. With JavaScript we change appearance, value or structure of the existing HTML dynamically.

Angular is one of the most demanding JavaScript framework in industry now a days. So I believe at least you should know the basics about it.

Angular is a JavaScript framework introduced by Google mainly focusing at Single Page Applications.

Note: in this series I am just trying to make all of you familiar with AngularJS. It won’t contain detailed angular topics.

Is AngularJS a library?

No it’s a framework.

What’s the difference between library and Framework?

Library provides the set of reusable APIs whereas framework does some special automation along with providing reusable APIs.

Example –

JQuery is a reusable library which provides set of functions and make dom manipulation and dom traversing easier.

.Net is Framework which provides reusable functionalities but it also handles compilation, execution, it makes garbage collection run in the background and make sure unused memory get cleared.

So Framework is something which does many things by its own.

Why angular is required?

In order to understand this we have to understand the problems of traditional JavaScript programming.

(Read the following explanation again after completion of all the labs in this article)

I think you will be able to answer it well after looking at following sample.

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title></title>
<script>
        var SomeValue;
        function SetValue() {
            SomeValue= "Initial Value";
            document.getElementById('MyTextbox').value = SomeValue;
        }
        function UpdateValue() {
            SomeValue = "New Value";
        }
</script>
</head>
<body >
<input type="text" id="MyTextbox" value="" disabled />
<input type="button" id="BtnClick" value="Click Me" />
</body>
</html>
```

What we are trying to achieve in above demo?

- There is variable called SomeValue and we are loading the value of that variable in textbox in page load.
- When button is clicked value of the variable got updated to some new value.

Now let's understand the problems with above sample code or in simple words traditional JavaScript programming.

1. Let's sometime in the future we have been asked to change the UI appearance. Let's say client is demanding a label or a span instead of textbox. It looks like a very easy change but don't you think it will also affect your JavaScript. So problem is every time we change the UI design there is high possibility of JavaScript getting affected
2. Button click do update the value of variable but will it update the value in the textbox? NO. So the problem is every time JavaScript value is updated an explicit update is required in UI.
3. JavaScript code we wrote is very much cluttered. You will not find a fix separation between various parts. For an instance our JavaScript will hold both,

    a. Logic part

        I. Communication with server to get the new value
        II. Validating inputs
        III. Combine FirstName and LastName before using it.

    b. Presentation part

        I. Display salary in TxtSalary textbox
        II. Textbox should be red in colour if salary is greater than 20000 or else in green colour

Solution - angular

With Angular we get an ability where we will write our HTML and write JavaScript independent of HTML. At runtime angular will connect JavaScript to HTML dynamically

Angular creates a two way binding between UI and JavaScript at runtime. Thus any change in UI value will automatically JavaScript variables and any JavaScript change automatically reflect in UI.

Binding is completely runtime hence we get a clear separation between presentation (HTML) and logic (JavaScript). Angular brought MVW architecture in JavaScript programming world.

How angular works

For working with angular we need two thingsJavaScriptand HTML

Here is the step by step explanation on working of angular. Don't worry if you won'tunderstand it 100%,complete the demo and read it again.

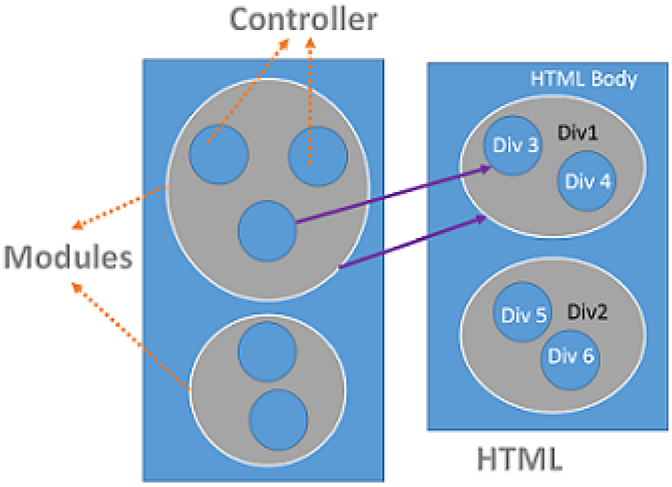
1. **JavaScript -** In angular key concept is Module. Module acts like a container for different parts of the application. Module mainly encapsulates following two things.

    a. Model – which is the data to be displayed in UI. Model can also be something which will decide how UI should be displayed.
    b. Controller – which connects model to the UI

2. In angular there is a concept called Directives. They are special attributes which will be attached to the HTML elements and they will change the default behaviour of those elements.
3. **HTML -** View will be designed using HTML. A single HTML may be broken down into multiple independent sections with the help of HTML container elements like 'table', 'div' etc.
   Each section will be treated as individual applications with the help of Angular directive "ng-App". "ng-App" attribute will be assigned to the module created in step 1.
   In some cases entire HTML page is treated as one application. In that case "ng-App" directive will be applied to body or HTML tag.
4. **JavaScript –** As I said in step1, every module will create one or more controllers.
5. **HTML -** Every application may contain one or more sub sections defined inside it with the help of HTML container elements like 'table', 'div' etc. These sub sections will be dependent on each other. Based on logical operation they will be divided. Each sub section will get its data from different controllers. Which section will take data from which controller will be decided with the help of special directive called "ng-controller"
6. **HTML –** Inside every sub section some HTML will be written each of which may get its data from controller in the form of model.

7. **Execution**

   a. When application is executed, at runtime angular parser will parse all theapplications in the HTML.To identify applications it will search for ng-App directive.
   b. Once the parsable element is found, angular will find ng-controller directive inside parsable element and accordingly it creates the object of controller.
   c. New child scope will be created and made available as an injectable parameter to the Controller's constructor function as $scope. (Don’t worry about this point. You will get in some time)
   d. After that angular parses the HTML using data (model) exposed by controller.



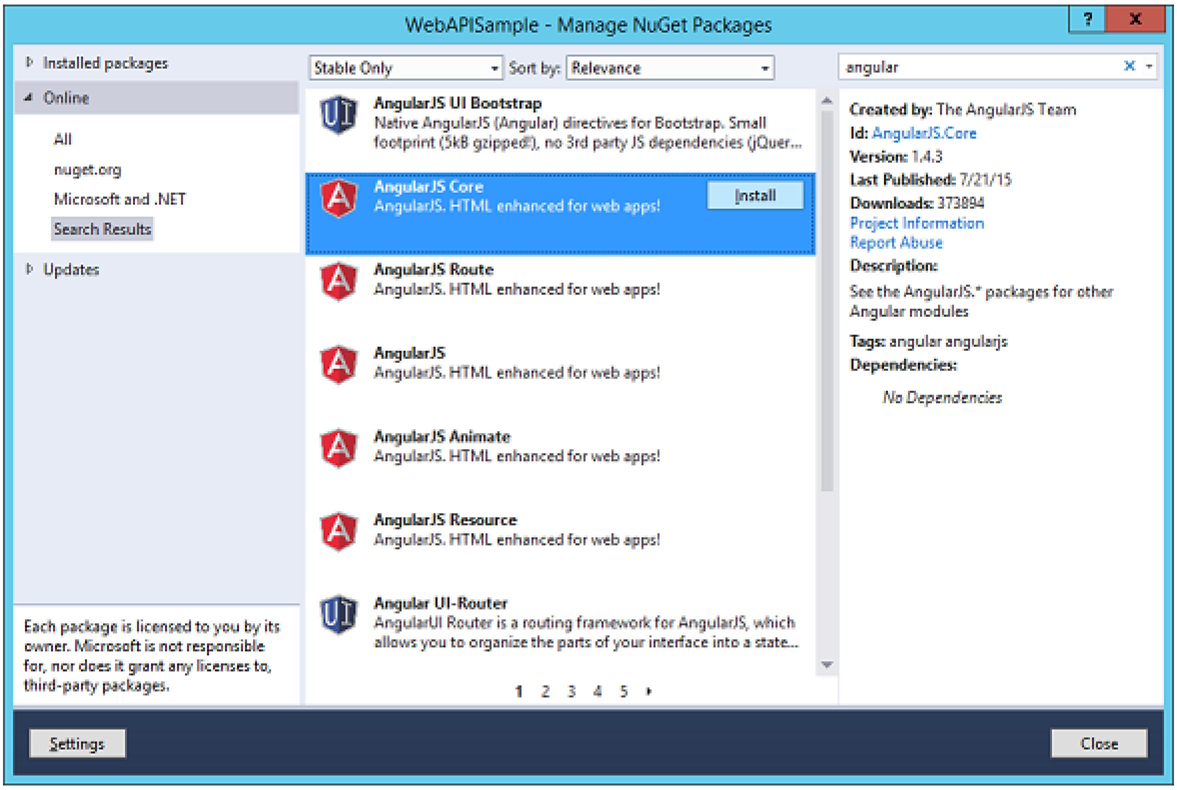
Demo on Angular

Lab1– Simple Angular Demo

Step 1 – Download a Angular

We will use the same Web API project created in last labfor this demo.

Right click the project and select “Manage Nuget Packages”. Search for angular in online section and install “AngularJs core”.



AngularJS UI Bootstrap
Native AngularJS (Angular) directives for Bootstrap. Small footprint (5kB gzipped!), no 3rd party JS dependencies (jQuer...

AngularJS Core
AngularJS. HTML enhanced for web apps!

AngularJS Route
AngularJS. HTML enhanced for web apps!

AngularJS
AngularJS. HTML enhanced for web apps!

AngularJS Animate
AngularJS. HTML enhanced for web apps!

AngularJS Resource
AngularJS. HTML enhanced for web apps!

Angular UI-Router
AngularUI Router is a routing framework for AngularJS, which allows you to organize the parts of your interface into a state...

Created by: The AngularJS Team
Id: AngularJS.Core
Version: 1.4.3
Last Published: 7/21/15
Downloads: 373894
Project Information
Report Abuse
Description:
See the AngularJS.* packages for other Angular modules
Tags: angular angularjs
Dependencies:
No Dependencies

Each package is licensed to you by its owner. Microsoft is not responsible for, nor does it grant any licenses to, third-party packages.

Step 2 –Create HTML

Create a new folder called HTML in the project and create a new HTML file called SimpleAngular.html as follows.

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title></title>
</head>
<body>
```

```
</body>
</html>
```

Step 3 – Include Angular

Include Angular.js file in Simple.html file as follows

```
<script src="../Scripts/angular.js"></script>
```

Step 4 - Create Angular module

```
<script>
    var MyModule = angular.module("MyAngularModule", []);
</script>
```

As you can see angular.module function takes two parameter. First parameter is the name of the module and second parameter will be used to specify other dependent modules.

We will not get into the second parameter in this series.

Note: “MyAngularModule” is the name of the angular module where as “MyModule” is just a JavaScript reference object,

Step 5–Create HTML appliaction

Attach ng-App directive to body tag as follows.

```
</script>
</head>
<body ng-app="MyAngularModule">

</body>
```

As you can see this time entire body is considered as oneapplication.

Step 6–Create Controller

Create a controller inside module as follows.

```
var MyModule = angular.module("MyAngularModule", []);
    MyModule.controller("MyController", function ($scope) {

    });
</script>
```

For now just take $scope as a variable which will encapsulate all the models required for view. We will talk about it in detail soon.

Step 7 – Attach Controller to View

Simply define a sub section inside body with the help of ‘Div’ tag and attach controller to it.

```
<body ng-app="MyAngularModule">

    <div ng-controller="MyController">


    </div>

</body>
```

Step 8 – Define Model data

In the controller function defined model data as follows.

```
<script>
    var MyModule = angular.module("MyAngularModule", []);
    MyModule.controller("MyController", function ($scope) {
        $scope.CustomerName = "Sukesh Marla";
        $scope.Designation = "Corporate Trainer";
    });
</script>
```

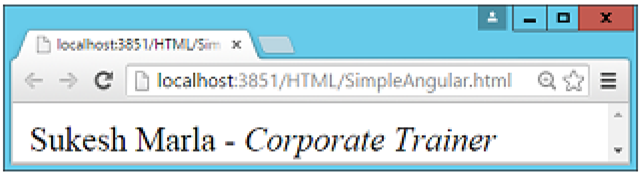
Step 9 – Display Model data in View

Use following HTML in view to display model data

```
<body ng-app="MyAngularModule">

    <div ng-controller="MyController">

        <span>{{CustomerName}}</span>

        <span> - </span>

        <i>{{Designation}}</i>

    </div>

</body>
```

Step 10 – Execute and Test

Press F5 and execute the application. Put the physical URL of the html file and check the output.



Sukesh Marla - *Corporate Trainer*

Talk on Lab 1

What is $scope

It's a child scope. When angular parser find ng-Controller it creates two things.

- new instance of Controller
- new child scope

This new child scope is made available as an injectable parameter to the controllers constructor function as $scope.

We define everything which is required for UI (view) as a member of this $scope.

What was {{}} signifies?

It's called expression in angular. Angular parser will get the value of variable specified between {{ and }} from current child scope and displays it.

What is MVW?

We will talk about this at end of the lab3 on angular

Lab 2- Change UI design

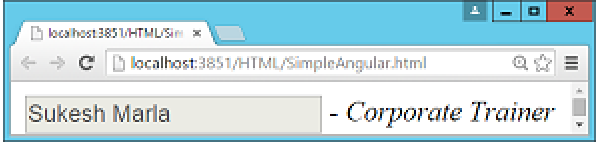
In this lab we have a small change requirement. Our requirement says "Display CustomerName in disabled textbox".

Step 1 – Change UI design to follows.

```
<body ng-app="MyAngularModule">

    <div ng-controller="MyController">

        <input type="text"  value="{{CustomerName}}" disabled />

        <span> - </span>

        <i>{{Designation}}</i>

    </div>

</body>
```

Step 2 – Execute and Test

Press F5 and execute the application. Put the physical URL of the html file and check the output.



Sukesh Marla - Corporate Trainer

As you can see, change in the UI no more affects JavaScript code.

Lab 3 – Understand Two way binding

In this lab we will just take the above lab into next level.

Requirement

- Designation will not be displayed in UI anymore
- CustomerName textbox will be enabled
- CustomerName will also be displayed in span beside textbox
- Update button will be added to UI
- On Update click textbox value will be updated to "New Updated Value"

Step 1 –Change the UI part

Change the body content to following.

```
<div ng-controller="MyController">

    <input type="text" ng-model="CustomerName" />

    <span>{{CustomerName}}</span><br />

    <input type="button" value="Update"



            ng-click="UpdateValue();" />

</div>
```

As you can see two new directives are introduced.

- ng-model
  As I said before, directives will change the default behaviour of an HTML UI element. ng-model will change the default behaviour of input control. Usually input control won't do anything when its value get updated but When ng-bind is attached it will create a two binding between model and control. Henceas soon as model value get changed, UI value will be refreshed and every time UI value get updated automatically model get updated.
- ng-click
  Just like onclick,it will attach click event handler to the control but this time event handler is going to be a part of current child scope. Hence inside event handler we get access to all the current child scope elements including model.

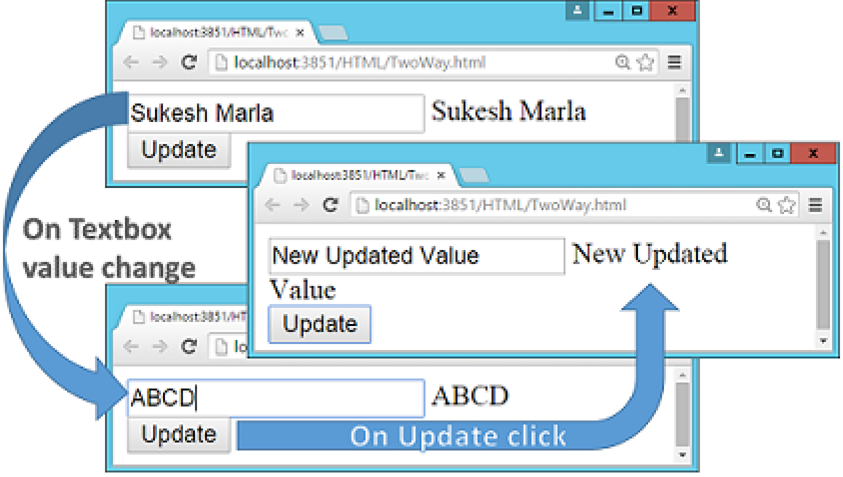
Step 2 – Redefine controller

Redefine controller code to following

```
<script>
    var MyModule = angular.module("MyAngularModule", []);
    MyModule.controller("MyController", function ($scope) {
        $scope.CustomerName = "Sukesh Marla";
        $scope.Designation = "Corporate Trainer";
        $scope.UpdateValue = function () {
            $scope.CustomerName = "New Updated Value";
        }
    });
</script>
```

Step 3 – Execute and Test

Press F5 and execute the application. Put the physical URL of the html file and check the output.



Talk on Lab 3

What is MVW?

In angular View means UI with which user will interact and Model means data required for View.

Now let’s talk a little about Controller.

- Controller will controls the communication between View and Data (Model). Hence we can call it as controller in MVC architecture.
- Controller not only encapsulate all the data required for view inside $scope but also it encapsulate events in the view. Hence we can call controller as ViewModel in MVVM architecture.
- • It also encapsulate the presentation logic. For instance let say we want to display some value and then format in based on some condition. For example – display salary in red colour if it’s greater than 2000 or else display it in blue colour. In Asp.Net MVC we handled such situation with the help of ViewModel but when we speak about MVP architecture it’s the feature of Presenter. Hence we can call it Presenter.

This is why we call it Model-View-Whatever. Whatever that can connect view and model.

Demo on Angular with Web API

Again this lab is going to be an upgrade for previous lab. In this lab,

- There will be two textboxes, one for CustomerName and one for Desingation
- We will get the initial value for textbox from Web API
- On button click current textbox value will be passed to Web API. Web API will upgrade the value and return the new value back to angular.

Step 1 – Create Web API model

Create a new class called Customer in Model folder as follows.

```
namespace WebAPISample.Models
{
    public class Customer
    {
        public string CustomerName { get; set; }
        public string Designation { get; set; }
    }
}
```

Step 2 – Create Web API Controller

Create a new Web API called Customer in Controller folder.

Step 3 – Create GET action

Open CustomerController.cs file and put using statement as below.

```
using WebAPISample.Models;
```

Now create a new action method called GET inCustomerController. Overall picture looks like this.

```
namespace WebAPISample.Controllers
{
    public class CustomerController : ApiController
    {
        public Customer GET()
        {
            Customer c = new Customer();
            c.CustomerName = "Sukesh Marla";
            c.Designation = "Corporate Trainer";
            return c;
        }
    }
}
```

Step 4 -Change Controller code in AngularJs side

```
MyModule.controller("MyController", function ($scope,$http) {
    $http.get("/api/Customer").then
    (
        function (r) {
            $scope.Customer = r.data;
        }
    );
    $scope.UpdateValue = function () {

    }
});
```

What is $http?

To make life easier angular provides several useful services. Services are just wrapper over some reusable functionalities.

Example $http service will let us make call to Server APIs.

When does the instance of $http get created?

Angular will create an instance of controller and new child scope when it finds the ng-controller directive. Although services are injected into controller like $scope, services won't get instantiated in the beginning. It will get instantiated when it is first used.

Note: Instance of service will be created only once.After that same object will be reused across all references. It strictly follows singleton pattern.

What is $http.get?

$http.get will make a get request to server APIs. It is going to be an asynchronous call. In simple words next line after $http.get will execute even before server execution completes.

What is the purpose of "then" function used with $http.get?

As I said before $http.get makes service call asynchronously. So instead of returning an actual value it will return a special object called Promise object.

Promise object exposes a function called "then" which will expect a parameter of type function.

Function passed to the "then" function will be executed automatically when execution of Web API completes.

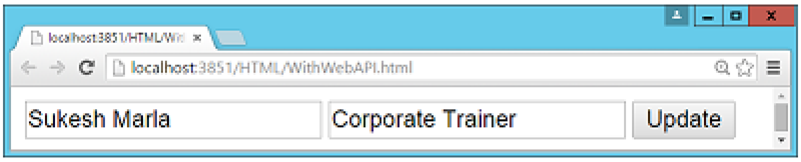
Step 5 – Change UI

Change the contents in the body section to following markup.

```
<div ng-controller="MyController">

    <input type="text" ng-model="Customer.CustomerName" />

    <input type="text" ng-model="Customer.Designation" />

    <input type="button" value="Update"



            ng-click="UpdateValue();" />

</div>
```

Step 6 – Execute and Test

Press F5 and execute the application. Put the physical URL of the html file and check the output.



localhost:3851/HTML/WithWebAPI.html

Sukesh Marla

Corporate Trainer

Note: You may get a little delay to textbox to get populate with values

Step 7 – Create POST action

Create a new action method called POST in CustomerController as follows.

```
public Customer POST(Customer c)
{
    c.CustomerName += " New";
    c.Designation += " New";
    return c;
}
```

Step 8 – Change UpdateValue function

Change the UpdateValue function in angular side as follows.

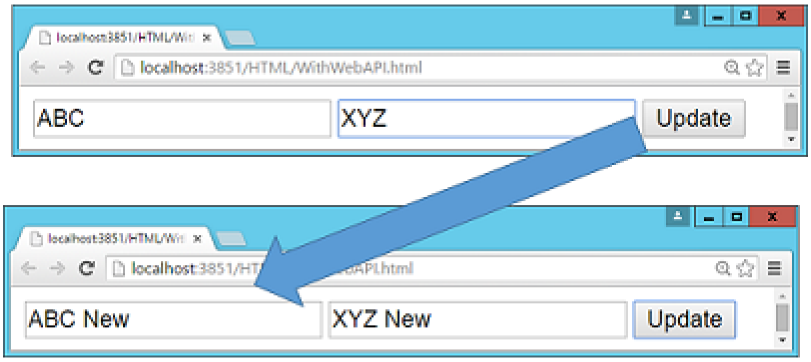
```
<script>
    var MyModule = angular.module("MyAngularModule", []);
    MyModule.controller("MyController", function ($scope,$http) {
        $http.get("/api/Customer").then
        (
            function (r) {
                $scope.Customer = r.data;
            }
        );
        $scope.UpdateValue = function () {
            $http.post("/api/Customer", $scope.Customer).then
            (
                function (r) {
                    $scope.Customer = r.data;
                }
            );
        }
    });
</script>
```

Just like GET, POST will return promise object and everything else is same

$http.post will send post request to Web API and at server side Web API engine will make POST action get executed.

Step 8 – Execute and Test

Press F5 and execute the application. Put the physical URL of the html file and check the output.



ABC

XYZ

ABC New

XYZ New

Complete magic of two way binding.

Start with MVC 5

In case you want to start with MVC 5 start with the below video Learn MVC 5 in 2 days.

Learn ASP.NET MVC 5 (Model view controller) step by step in 2 days (16 hours)
Learn MVC
in
2 Days

Conclusion

Here will complete our Bonus day article - Introduction to Web API and angular.

Your comments, Mails always motivates us to do more. Put your thoughts and comments below or send mail to SukeshMarla@Gmail.com

Connect us on Facebook, LinkedIn or twitter to stay updated about new releases.

For Offline Technical trainings in Mumbai visit StepByStepSchools.Net

For Online Trainings visit JustCompile.com or www.Sukesh-Marla.com

License

About the Author

Marla Sukesh

Instructor / Trainer Train IT
India

Learning is fun but teaching is awesome.

Code re-usability is my passion ,Teaching and learning is my hobby, Becoming an successful entrepreneur is my goal.

By profession I am a Corporate Trainer.
I do trainings on WCF, MVC, Business Intelligence, Design Patterns, HTML 5, jQuery, JSON and many more Microsoft and non-Micrsoft technologiees.

Find my profile here

My sites

- **justcompile.com**
- **www.sukesh-marla.com**

@Twitter
@Facebook

Comments and Discussions

7 messages have been posted for this article Visit **https://www.codeproject.com/Articles/1012532/Learn-MVC-Project-in-Days-Bonus-day** to post and view comments on this article, or click **here** to get a print view with messages.