

ASP.NET CORE CRUD WITH ANGULAR2 ANIMATION ,WEB API ,EF & SQL SERVER

By : SYED SHANU

Email: syedshanumcain@gmail.com

Date : 2017/03/11



About Me

- **My Name is Syed Shanu**
- **From Madurai, Tamil Nadu, India**
- **Microsoft MVP**
- **Code Project MVP**
- **Csharp Corner MVP**
- **10+ years of Experience as Software Engineer**
- **Working as a Senior Software Engineer at Zemax Solution ,Seoul ,Korea**

Session Prerequisites

◆ Visual Studio 2015

◆ SQL Server 2014

Download Links

1. First, download and install Visual Studio 2015 with Update 3 from this [link](#).
2. Visual Studio 2015 Update 3 from this [link](#).
3. [Download](#) and install .NET Core 1.0.1
4. [Download](#) and install TypeScript 2.0
5. Node.js v4.0 or above. I have installed V6.9.1 ([Download link](#)).
6. Download and install ASP.NET Core Template Pack viz file from this [link](#).

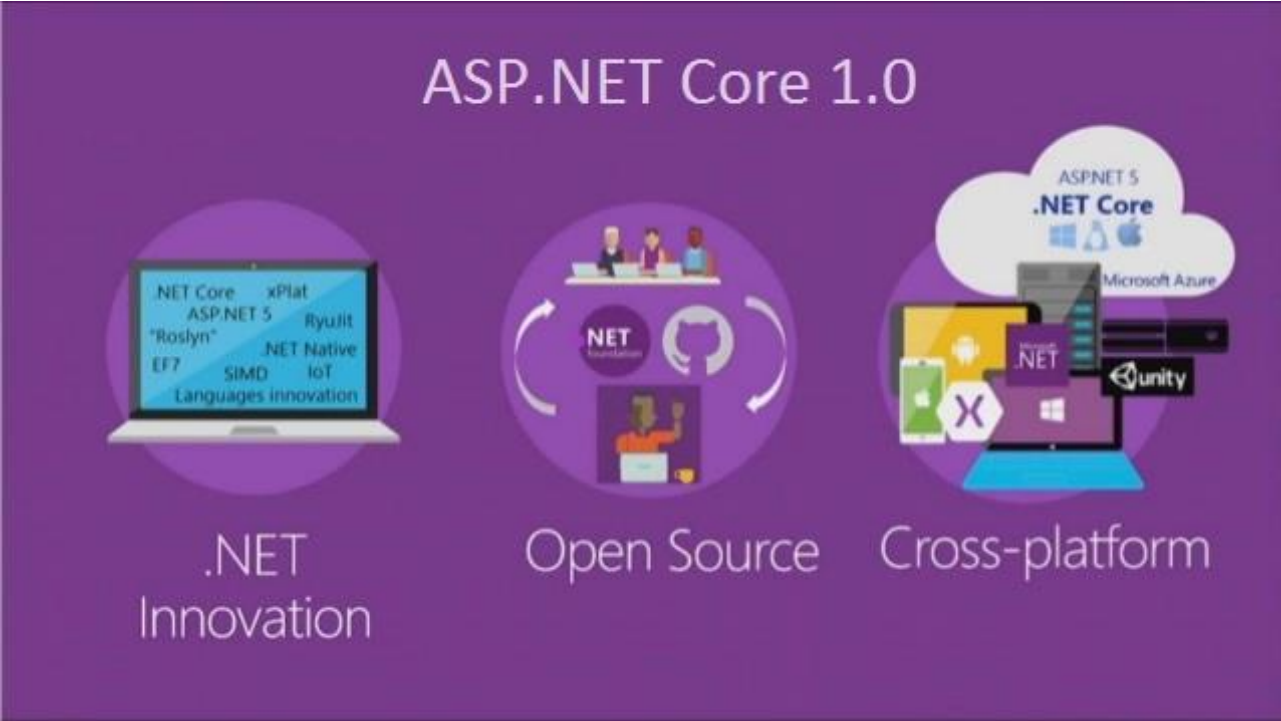
Agenda

- ◆ **Introduction to ASP.NET Core**
- ◆ **Introduction to Angular2**
- ◆ **Introduction to SQL Server**
- ◆ **Introduction to WEB API**
- ◆ **Entity Framework**
- ◆ **Demo for ASP.NET Core, Angular2 CRUD with Animation using Template Pack, WEB API and EF 1.0.1**

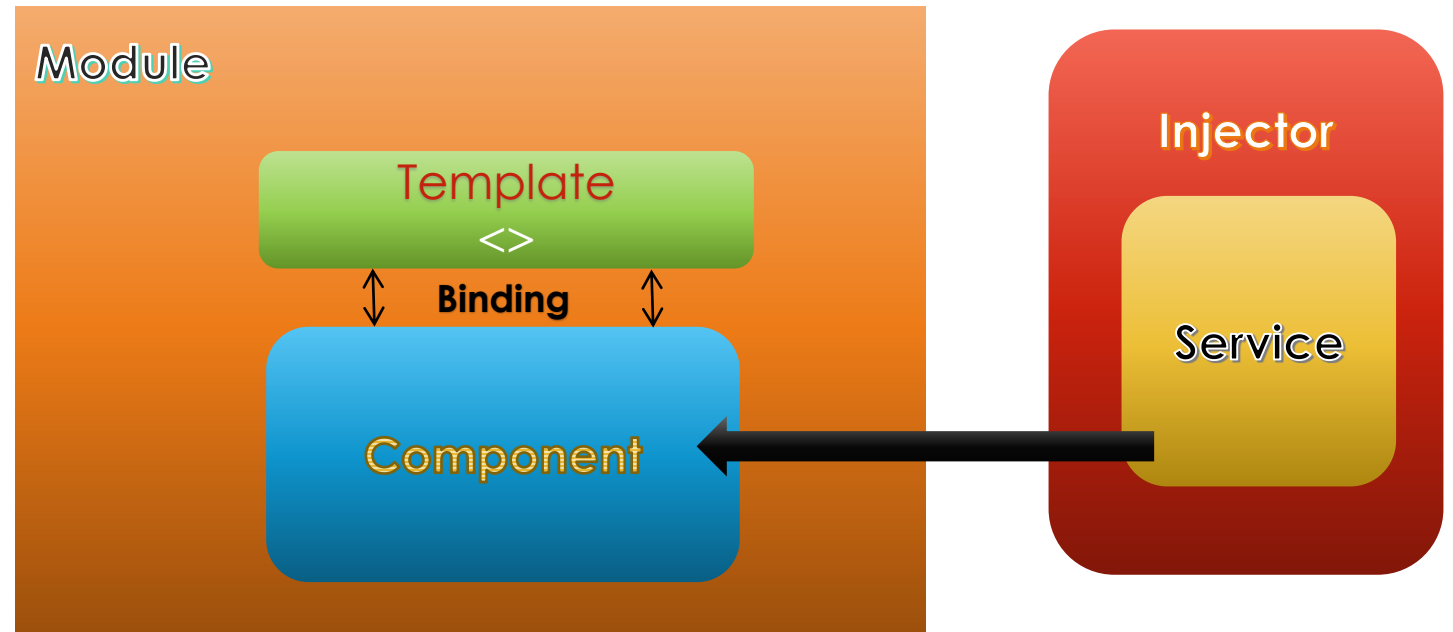
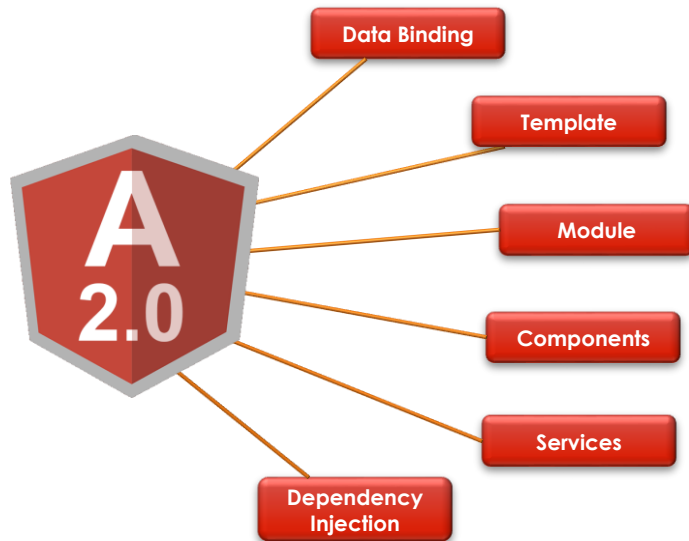
ASP.NET Core

ASP.NET 4.6 and ASP.NET Core 1.0

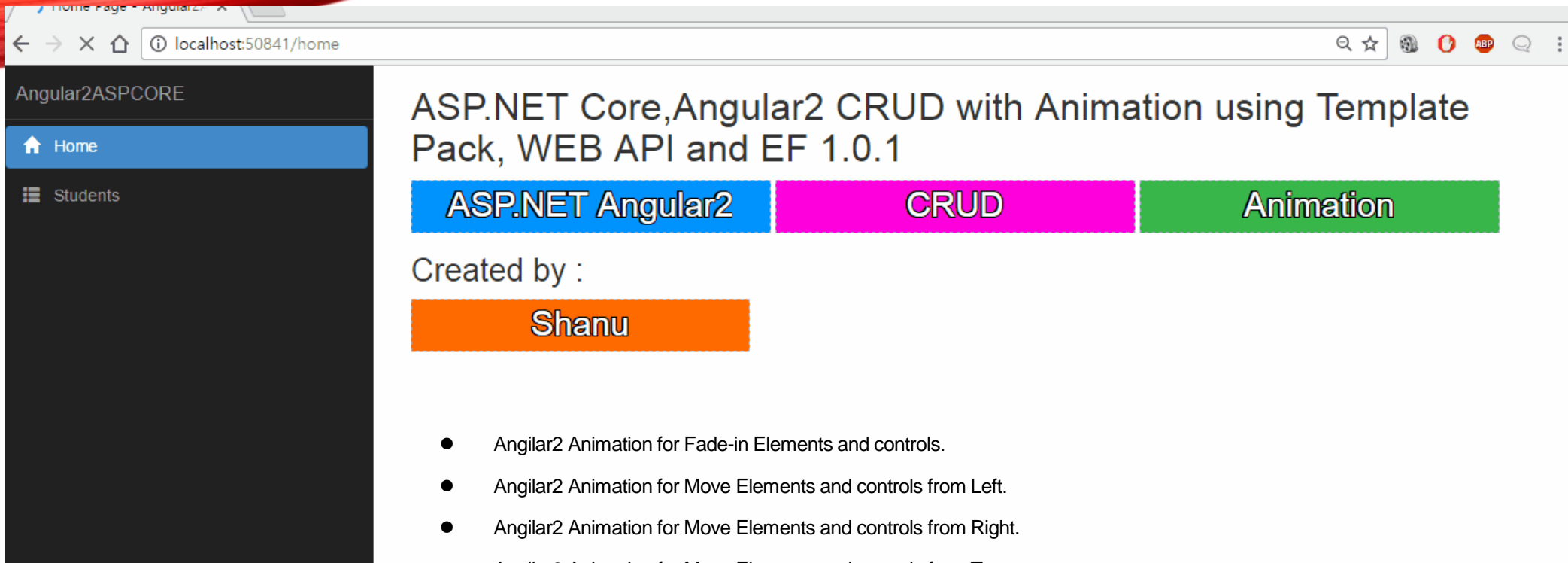
ASP.NET 4.6		ASP.NET Core 1.0	
.NET Framework 4.6		.NET Core 1.0	
.NET framework libraries		.NET core libraries	
Compilers and runtime components (.NET Compiler Platform: Roslyn, C#, VB, F# Languages, RyuJIT, SIMD)			



ANGULAR2 Architecture



ANGULAR2 Animation

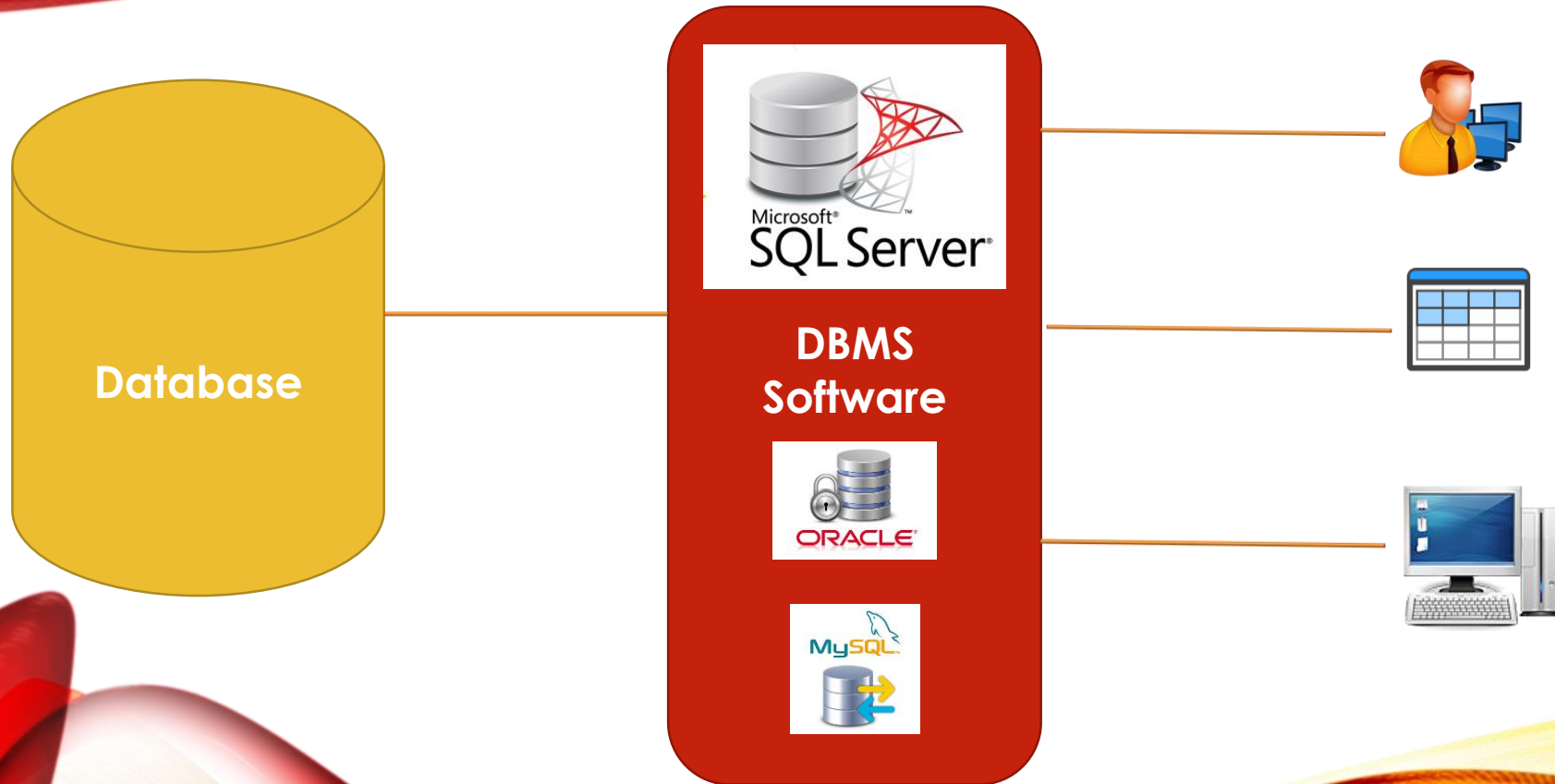


- Angular2 Animation for Fade-in Elements and controls.
- Angular2 Animation for Move Elements and controls from Left.
- Angular2 Animation for Move Elements and controls from Right.
- Angular2 Animation for Move Elements and controls from Top.
- Angular2 Animation for Move Elements and controls from Bottom.
- Angular2 Animation to enlarge Button on Click.

SQL SERVER

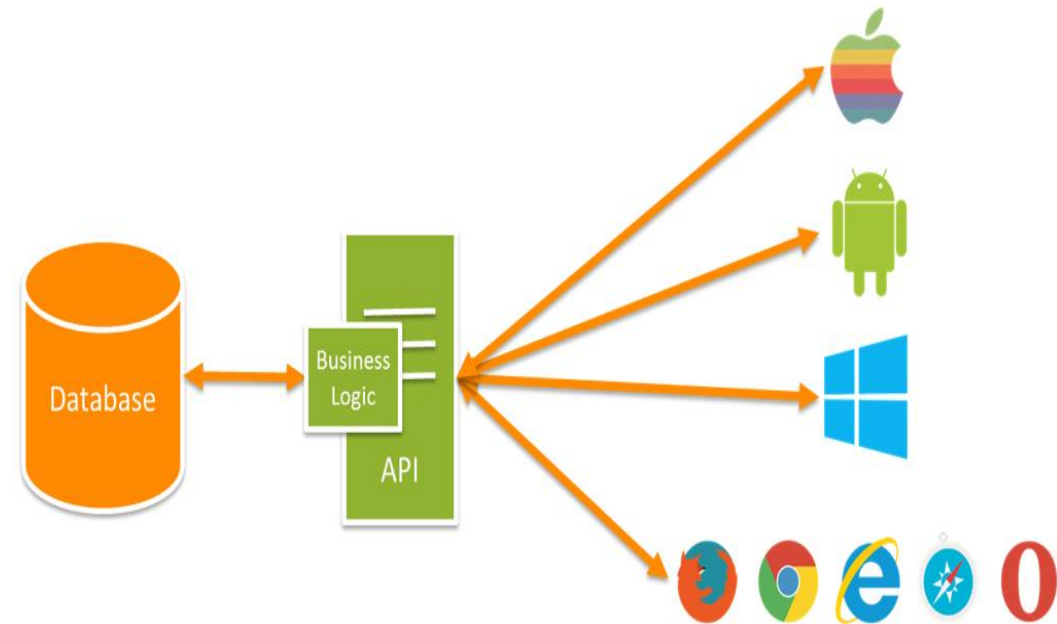


<http://www.c-sharpcorner.com/UploadFile/asmabegam/basic-sql-queries-for-beginners/>

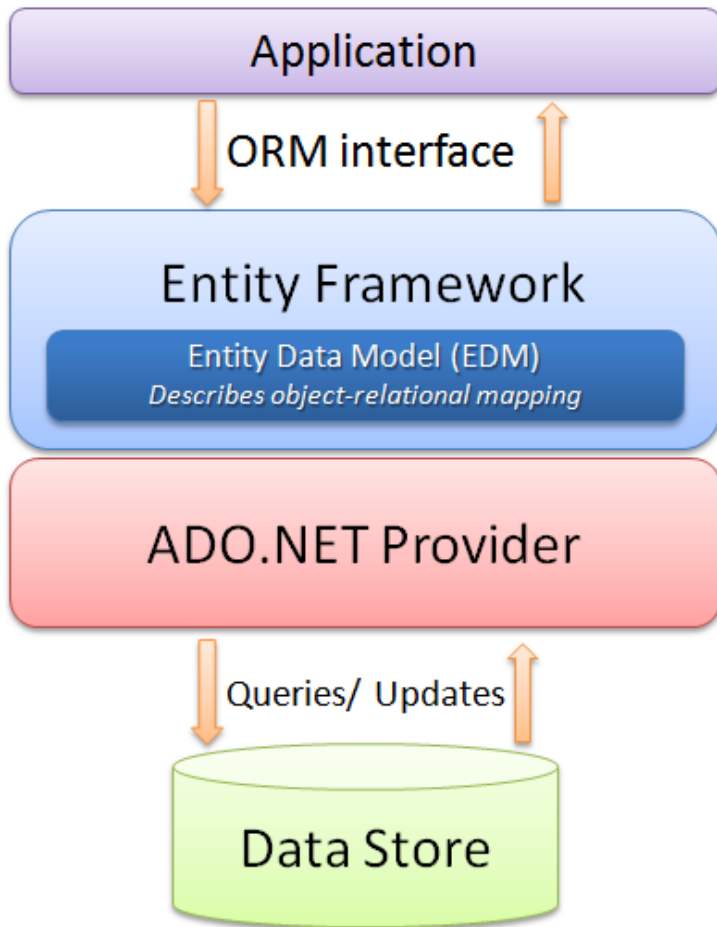


Web API

- API stands for 'Application Programming Interface'
- Web API is a simple and easy way to build HTTP Services for Browsers and Mobiles.
- Web API has the following four methods as **Get/Post/Put and Delete** where:
 - ✓ **Get** is to request for the data. (Select)
 - ✓ **Post** is to create a data. (Insert)
 - ✓ **Put** is to update the data.
 - ✓ **Delete** is to delete data.



Entity Framework



- Entity Framework is an Object Relational Mapper (ORM)
- ORM is a tool for storing data from domain objects to relational database like MS SQL Server
- Microsoft has provided an O/RM framework called "Entity Framework" to automate database related activities for your application.
- It eliminates the need for most of the data-access code that developers usually need to write.
- Performing basic CRUD (Create, Read, Update, Delete) operations.
- Easily managing "1 to 1", "1 to many", and "many to many" relationships
- We can have all data access logic written in higher level languages

ASP.NET CORE Application

Using

- ① SQL Server 2014
- ② ASP.NET Core
- ③ Entity Framework
- ④ Web API
- ⑤ Angular 2

Note: Refer this website to get SQL Script and study in depth to create this website

<http://www.c-sharpcorner.com/article/create-our-first-asp-net-core-angular-2-starter-application-net-core-using-te/>

ASP.NET CORE Application

Note: Refer this website to get SQL Script and study in depth to create this website

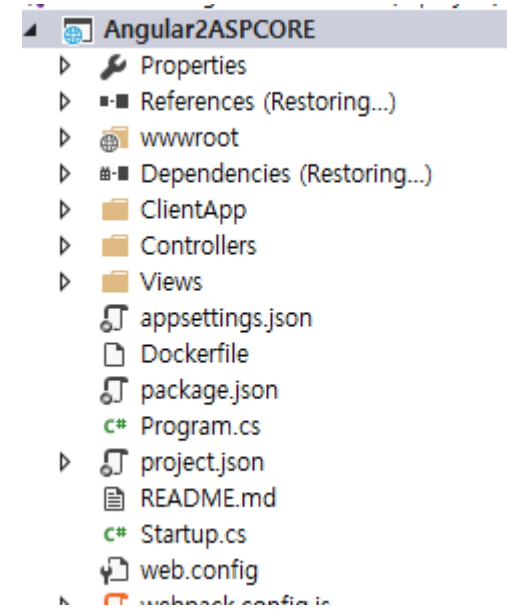
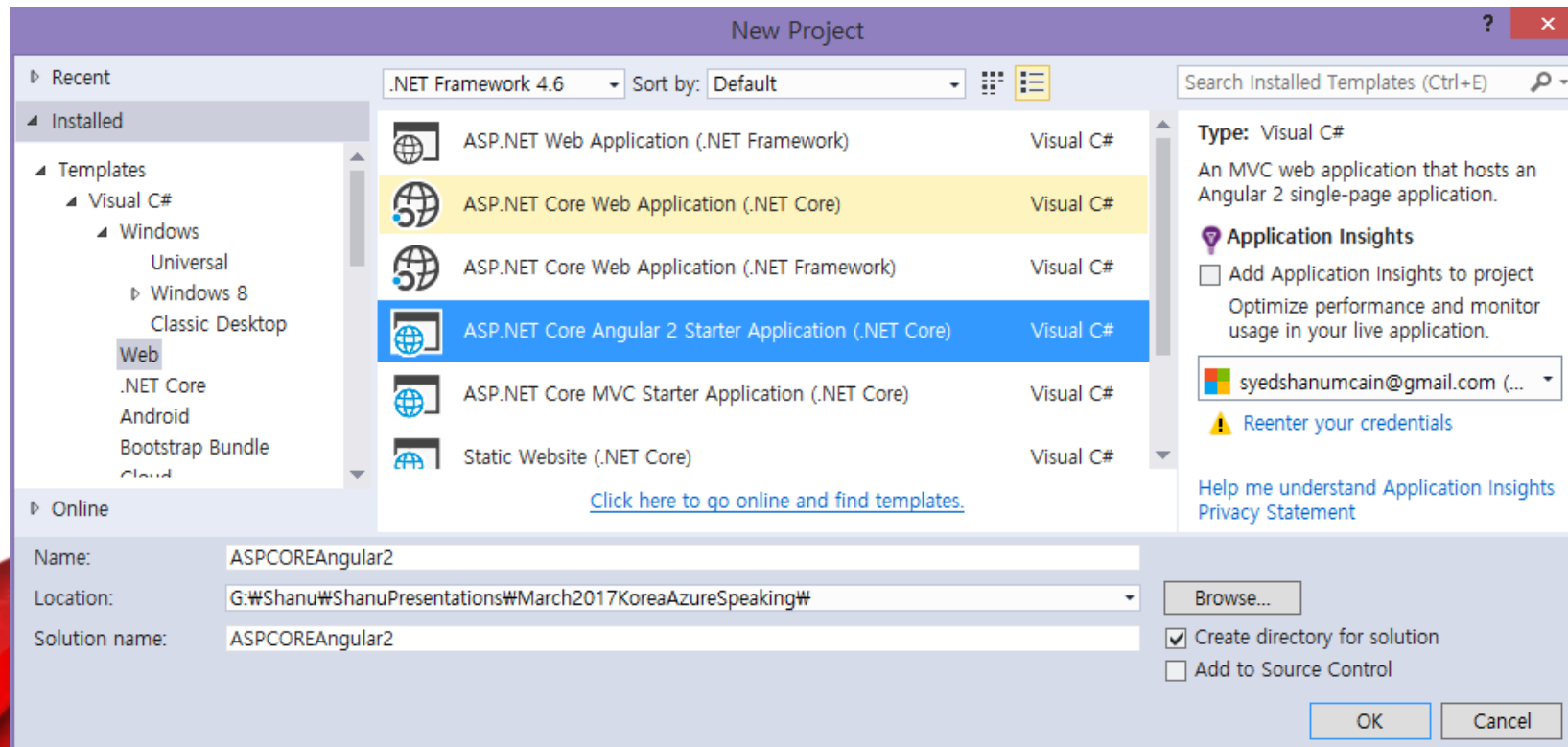
<http://www.c-sharpcorner.com/article/create-our-first-asp-net-core-angular-2-starter-application-net-core-using-te/>

① SQL Server 2014

✓ **Create Database and Table**

ASP.NET CORE Application

✓ Create ASP.NET Core Angular2 Starter Application



ASP.NET CORE Application

✓ **Create Entity framework**, we have used EF version 1.0.1

1. "Microsoft.EntityFrameworkCore.SqlServer": "1.0.1",
2. "Microsoft.EntityFrameworkCore.Tools": "1.0.0-preview2-final"

```
tp://json.schemastore.org/project
{
  "dependencies": {
    "Microsoft.NETCore.App": {
      "version": "1.0.1",
      "type": "platform"
    },
    "Microsoft.AspNetCore.AngularServices": "1.0.0-***",
    "Microsoft.AspNetCore.Diagnostics": "1.0.0",
    "Microsoft.AspNetCore.Mvc": "1.0.1",
    "Microsoft.AspNetCore.Razor.Tools": {
      "version": "1.0.0-preview2-final",
      "type": "build"
    },
    "Microsoft.AspNetCore.Server.IISIntegration": "1.0.0",
    "Microsoft.AspNetCore.Server.Kestrel": "1.0.1",
    "Microsoft.AspNetCore.StaticFiles": "1.0.0",
    "Microsoft.Extensions.Configuration.EnvironmentVariables": "1.0.0",
    "Microsoft.Extensions.Configuration.Json": "1.0.0",
    "Microsoft.Extensions.Configuration.CommandLine": "1.0.0",
    "Microsoft.Extensions.Logging": "1.0.0",
    "Microsoft.Extensions.Logging.Console": "1.0.0",
    "Microsoft.Extensions.Logging.Debug": "1.0.0",
    "Microsoft.Extensions.Options.ConfigurationExtensions": "1.0.0",
    "Microsoft.EntityFrameworkCore.SqlServer": "1.0.1",
    "Microsoft.EntityFrameworkCore.Tools": "1.0.0-preview2-final"
  },
}
```

References

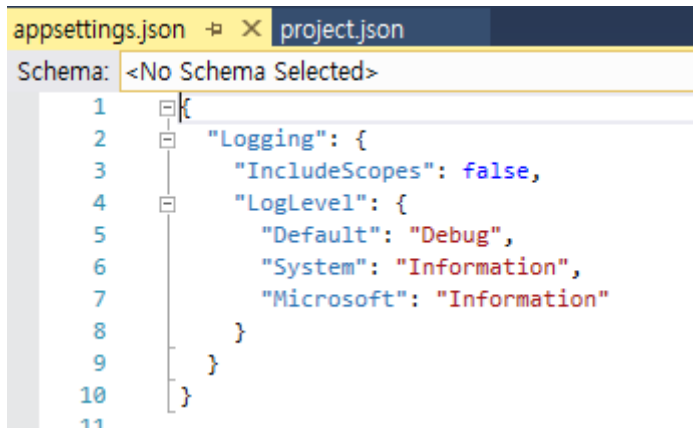
.NETCoreApp,Version=v1.0

- Microsoft.AspNetCore.AngularServices (1.0.0-beta-000021)
- Microsoft.AspNetCore.Diagnostics (1.0.0)
- Microsoft.AspNetCore.Mvc (1.0.1)
- Microsoft.AspNetCore.Razor.Tools (1.0.0-preview2-final)
- Microsoft.AspNetCore.Server.IISIntegration (1.0.0)
- Microsoft.AspNetCore.Server.Kestrel (1.0.1)
- Microsoft.AspNetCore.StaticFiles (1.0.0)
- Microsoft.EntityFrameworkCore.SqlServer (1.0.1)
- Microsoft.EntityFrameworkCore.Tools (1.0.0-preview2-final)
- Microsoft.Extensions.Configuration.CommandLine (1.0.0)

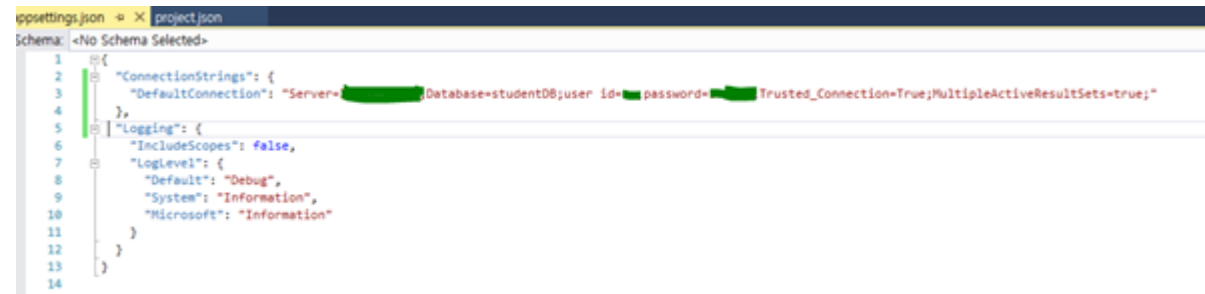
ASP.NET CORE Application

✓ Add Connection String

```
"ConnectionStrings": {  
  "DefaultConnection": "Server=YOURDBSERVER;Database=StudentsDB;user id=SQLID;password=SQLPWD;Trusted_Connection=True;MultipleActiveResultSets=true;"  
},
```



```
appsettings.json  project.json  
Schema: <No Schema Selected>  
1 {  
2   "Logging": {  
3     "IncludeScopes": false,  
4     "LogLevel": {  
5       "Default": "Debug",  
6       "System": "Information",  
7       "Microsoft": "Information"  
8     }  
9   }  
10 }  
11
```

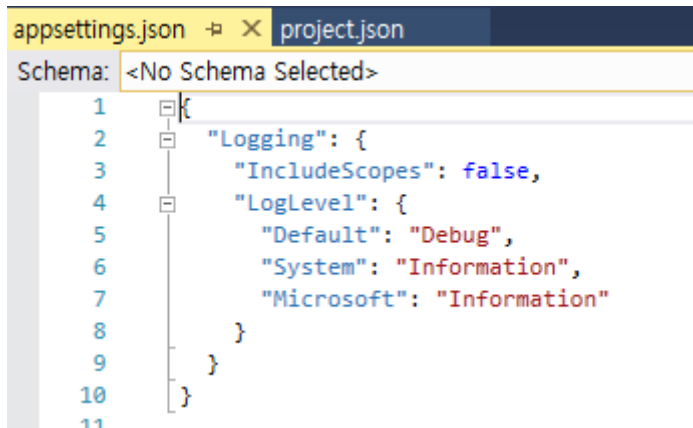


```
appsettings.json  project.json  
Schema: <No Schema Selected>  
1 {  
2   "ConnectionStrings": {  
3     "DefaultConnection": "Server=YOURDBSERVER;Database=studentDB;user id=SQLID;password=SQLPWD;Trusted_Connection=True;MultipleActiveResultSets=true;"  
4   },  
5   "Logging": {  
6     "IncludeScopes": false,  
7     "LogLevel": {  
8       "Default": "Debug",  
9       "System": "Information",  
10      "Microsoft": "Information"  
11    }  
12  }  
13 }  
14
```

ASP.NET CORE Application

✓ Add Connection String

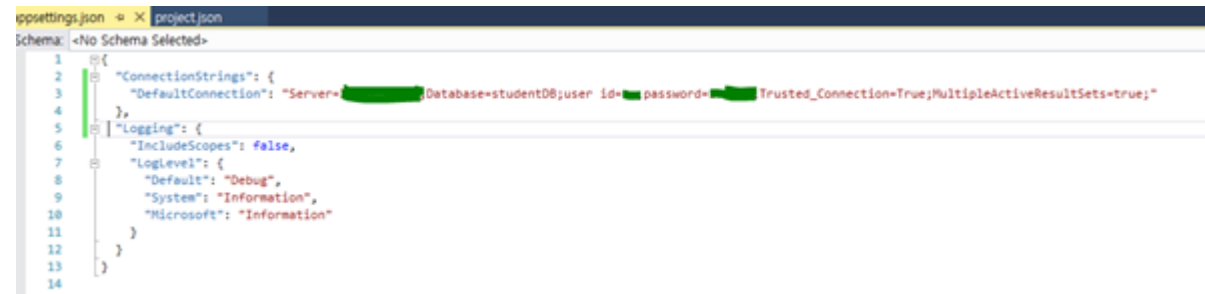
```
"ConnectionStrings": {  
  "DefaultConnection": "Server=YOURDBSERVER;Database=StudentsDB;user id=SQLID;password=SQLPWD;Trusted_Connection=True;MultipleActiveResultSets=true;"  
},
```



appsettings.json project.json

Schema: <No Schema Selected>

```
1 {  
2   "Logging": {  
3     "IncludeScopes": false,  
4     "LogLevel": {  
5       "Default": "Debug",  
6       "System": "Information",  
7       "Microsoft": "Information"  
8     }  
9   }  
10 }  
11
```



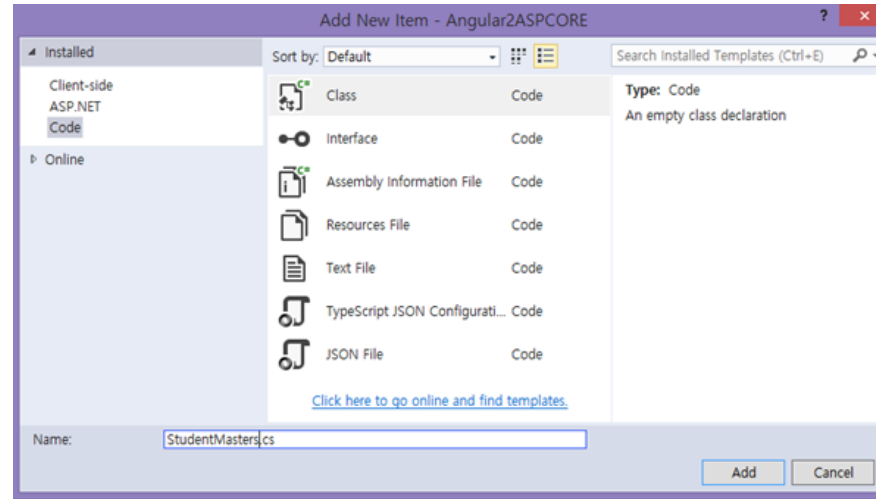
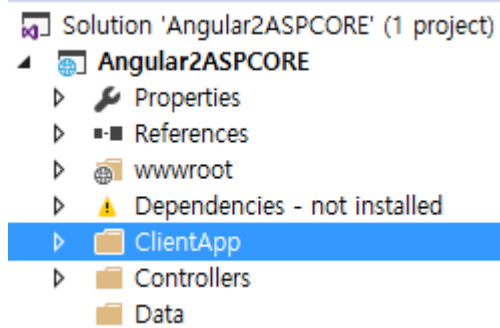
appsettings.json project.json

Schema: <No Schema Selected>

```
1 {  
2   "ConnectionStrings": {  
3     "DefaultConnection": "Server=[REDACTED];Database=studentDB;user id=[REDACTED];password=[REDACTED];Trusted_Connection=True;MultipleActiveResultSets=true;"  
4   },  
5   "Logging": {  
6     "IncludeScopes": false,  
7     "LogLevel": {  
8       "Default": "Debug",  
9       "System": "Information",  
10      "Microsoft": "Information"  
11    }  
12  }  
13 }  
14
```

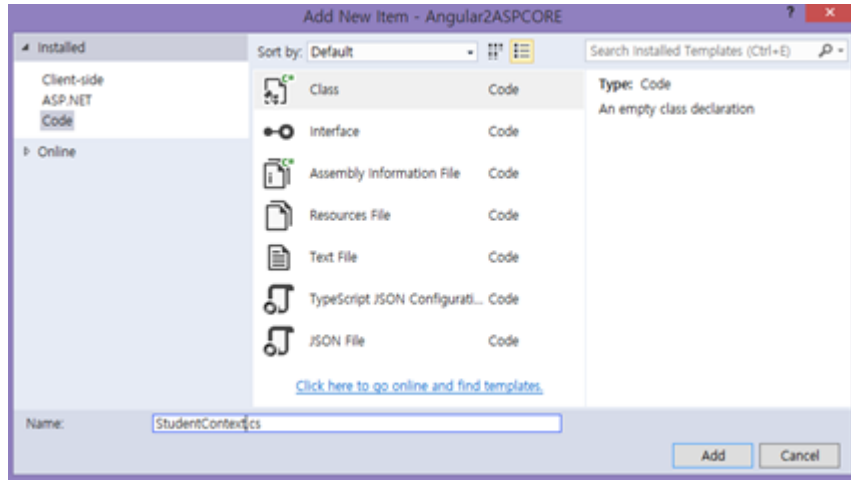
ASP.NET CORE Application

✓ Creating Model Class



ASP.NET CORE Application

✓ Creating Database Context

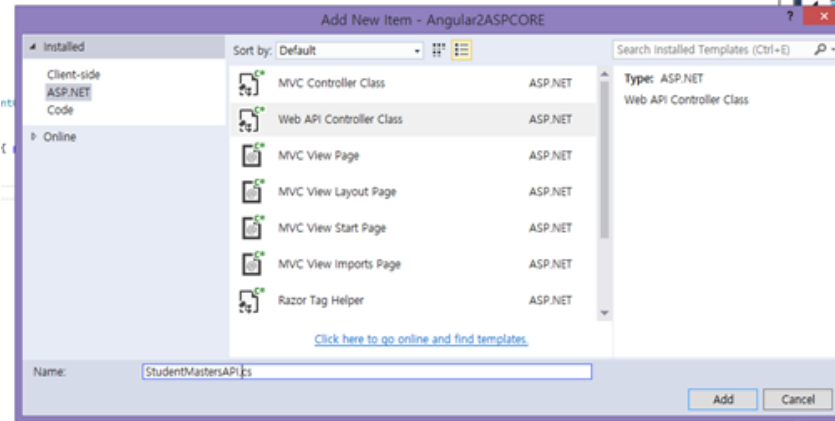
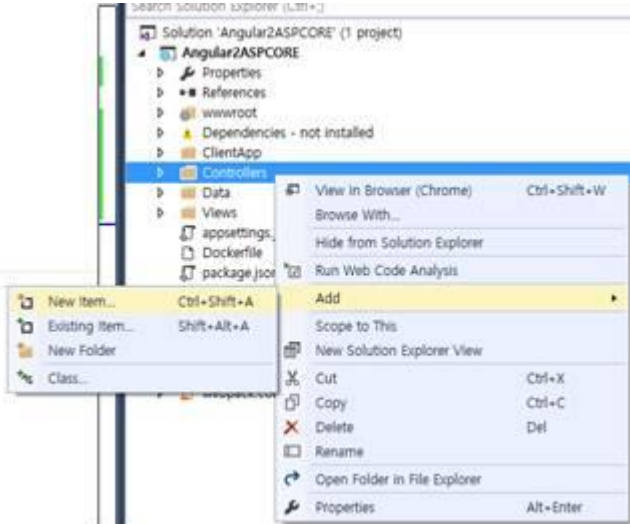


✓ Startup.CS

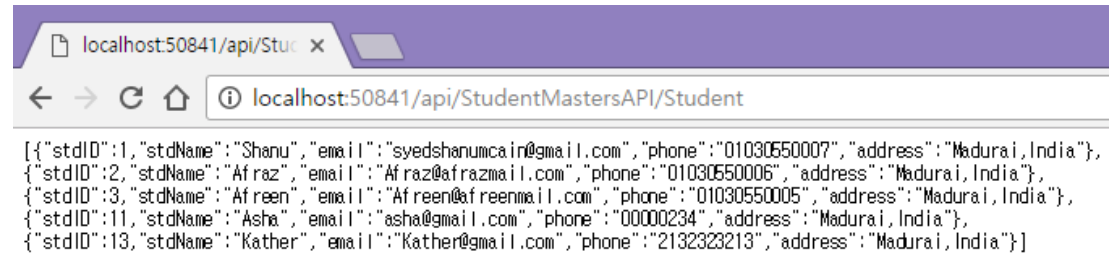
```
services.AddDbContext < studentContext > (options =>  
    options.UseSqlServer(Configuration.GetConnectionString("DefaultC  
onnection")));
```

ASP.NET CORE Application

✓ Creating Web API



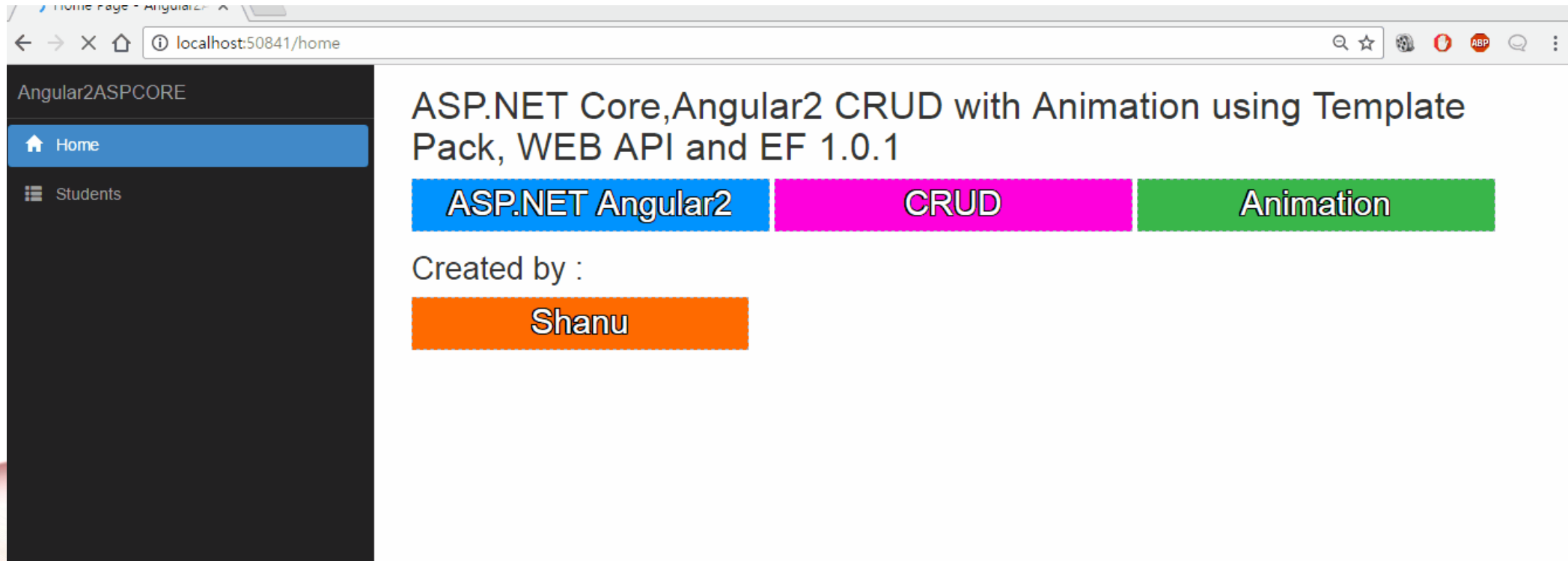
/api/StudentMastersAPI/Student



ASP.NET CORE Application

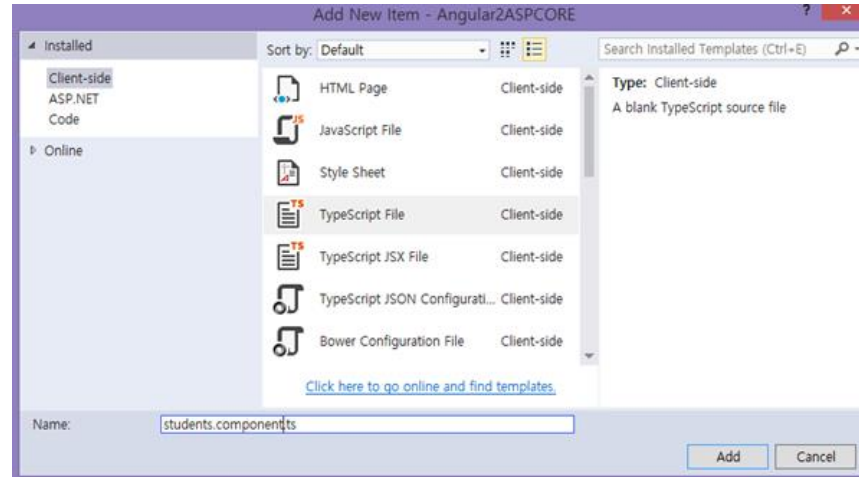
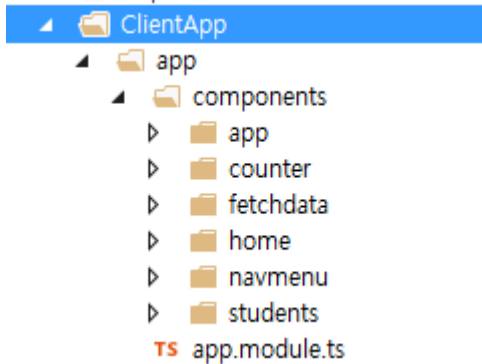
✓ Open Home Component and HTML

- Add Animation Code.

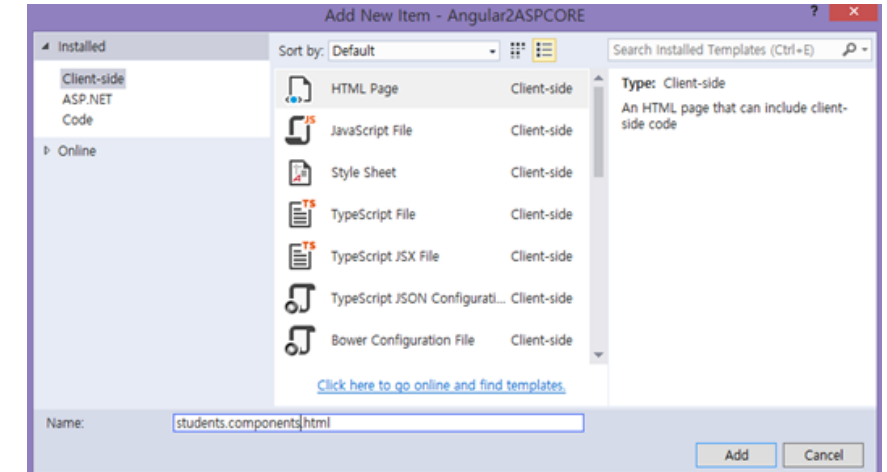


Working with Angular2

✓ First create Component TypeScript



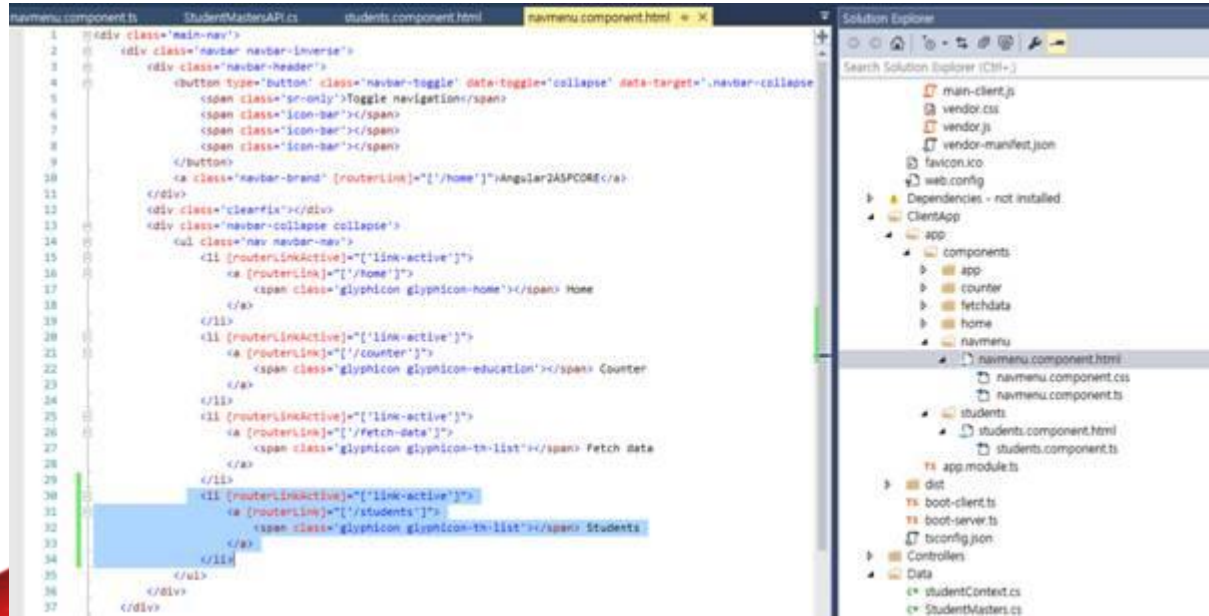
✓ Create HTML file to view our result



- import part
- component part
- class for writing our business logics.

ASP.NET CORE Application

✓ Adding Navigation in menu



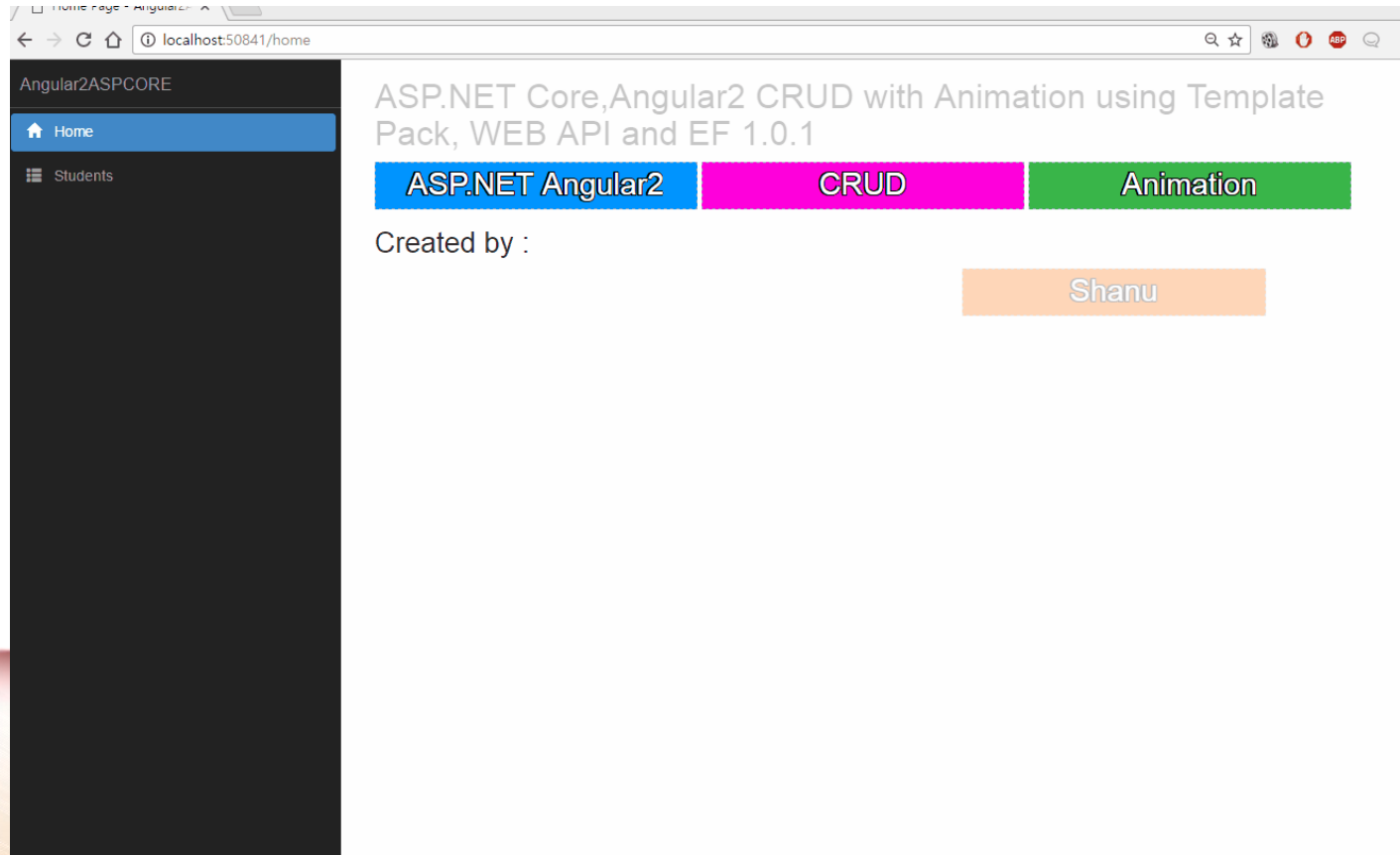
```
1 <div class='main-nav'>
2   <div class='nav navbar-inverse'>
3     <div class='navbar-header'>
4       <button type='button' class='navbar-toggle' data-toggle='collapse' data-target='.navbar-collapse'
5         <span class='sr-only'>Toggle navigation</span>
6         <span class='icon-bar'></span>
7         <span class='icon-bar'></span>
8         <span class='icon-bar'></span>
9       </button>
10      <a class='navbar-brand' [routerLink]='["/home"]'>Angular2ASP.NET</a>
11    </div>
12    <div class='collapse navbar-collapse'>
13      <ul class='nav navbar-nav'>
14        <li [routerLinkActive]='["link-active"]'>
15          <a [routerLink]='["/home"]'>
16            <span class='glyphicon glyphicon-home'></span> Home
17          </a>
18        </li>
19        <li [routerLinkActive]='["link-active"]'>
20          <a [routerLink]='["/counter"]'>
21            <span class='glyphicon glyphicon-education'></span> Counter
22          </a>
23        </li>
24        <li [routerLinkActive]='["link-active"]'>
25          <a [routerLink]='["/fetch-data"]'>
26            <span class='glyphicon glyphicon-th-list'></span> Fetch data
27          </a>
28        </li>
29        <li [routerLinkActive]='["link-active"]'>
30          <a [routerLink]='["/students"]'>
31            <span class='glyphicon glyphicon-th-list'></span> Students
32          </a>
33        </li>
34      </ul>
35    </div>
36  </div>
37 </div>
```

```
<li [routerLinkActive]='["link-active"]'>
  <a [routerLink]='["/students"]'>
    <span class='glyphicon glyphicon-th-list'></span>
  </a>
</li>
```

ASP.NET CORE Application

✓ App Module

- Import Students Component
- Add Student Module



ASP.NET CORE Application

✓ Build and run the application

Angular2ASPCORE

Home

Students

ASP.NET Core, Angular2 CRUD with Animation using Template Pack, WEB API and EF 1.0.1

Created by :
Shanu

Add New Students Information

Add New Student Information

Insert/Edit Student Details :

Students ID: 3 Students Name: Afreen
Email: Afreen@afreenmail.com Phone: 01030550005
Address: Madurai, India

Save Close

Edit	Delete	Student ID	Student Name	Email	Phone	Address
		1	Shanu	syedshanumcain@gmail.com	01030550007	Madurai, India
		2	Afraz	Afraz@afrazmail.com	01030550006	Madurai, India
		3	Afreen	Afreen@afreenmail.com	01030550005	Madurai, India
		11	Asha	asha@gmail.com	00000234	Madurai, India
		13	Kather	Kather@gmail.com	2132323213	Madurai, India

Download



Download

<https://github.com/syedshanu/ASPNETCOREAngular2>

More Resources



<https://mvp.microsoft.com/en-us/PublicProfile/5001980?fullName=Syed%20%20Shanu>



C#Corner

<http://www.c-sharpcorner.com/members/syed-shanu>



<http://www.codeproject.com/Articles/syed-shanu>

Questions ?



Contact Me



syedshanumcain@gmail.com



@syedshanu3



<https://www.facebook.com/syed.shanu.9>



Microsoft®
Most Valuable
Professional