



AngularJs چیست؟

AngularJs یک فریمورک JavaScript برای ایجاد برنامه های تک صفحه ای در وب می باشد AngularJs یک فریمورک MV* (MVC+MVVM=MV*) سمت کلاینت است AngularJs. ابتدا صفحه HTML که در آن ویژگیهای تگ های سفارشی جاسازی شده است را می خواند. سپس آن ویژگیها را به دستوراتی برای مقید نمودن بخشهای ورودی و خروجی صفحه به مدلی کسسه بوسیله متغییر های جاوا اسکریپت استاندارد ارائه میشود ، تفسیر مینماید. مقادیر متغییر های جاوا اسکریپت میتواند بطور دستی درون کدها ست شود یا از منابع استاتیک یا پویای JSON اخذ گردد.

در زیر بعضی از دستوراتی که در این پرکژه آموزشی به کار برده ایم آورده ایم:

- ng-app : این دستور نقطه ورود AngularJs به صفحه است ، این نشان می دهد که برنامه از نوع AngularJs است.
- ng-controller : یک دستور برای تعریف Controller است. که یک کلاس Controller را به View متصل می کند.
- ng-repeat : یک حلقه در صفحه شما ایجاد می کند مثل حلقه foreach
- angular.module : این دستور برای ایجاد ، ثبت و بازگردانی همه ماژول های angularJs استفاده می شود.
- \$http : یک شی XMLHttpRequest برای درخواست داده های خارجی می باشد
- \$http.get : داده های Json را می خواند، این دستور دو پارامتر url و config را دریافت می کند ، اما config اختیاری است.

در این آموزش ما یک مثال ابتدایی استفاده از AngularJs در ASP.Net MVC4 به شما نشان خواهیم داد.

قدم اول:

ASP.NET MVC Web Application در Visual Studio 2012 یا Visual Studio 2015 را اجرا کنید و پروژه جدیدی از نوع ASP.NET MVC Web Application ایجاد کنید.

در این آموزش ما از روش CodeFirst استفاده می کنیم ، در این روش پس ایجاد کلاس های دامنه و Context دیتابیس در زمان اولین اجرا برنامه ایجاد خواهد شد. کلاس DataBaseContext را درون فولدر Model برای دیتابیس factory ایجاد کنید.(اضافه ، ویرایش ، حذف یا انتخاب داده های دیتابیس)

ایجاد کلاس Context

DatabaseContext.cs

```
using System;
using System.Collections.Generic;
using System.Data.Entity;
using System.Linq;
using System.Web;
namespace AngularJSWithMVC.Models
{
    public class DatabaseContext: DbContext
    {
        public DatabaseContext(): base("DefaultConnection")
        {}
        public DbSet < EmployeeModel > EmployeeModels
        {
            get;
            set;
        }
    }
}
```

```
using System;
using System.Collections.Generic;
using System.ComponentModel.DataAnnotations;
using System.Linq;
using System.Web;
namespace AngularJSWithMVC.Models
{
    public class EmployeeModel
    {
        [Key]
        public int EmployeeId
        {
            get;
            set;
        }
        public string EmployeeName
        {
            get;
            set;
        }
        public string Address
        {
            get;
            set;
        }
        public string EmailId
        {
            get;
            set;
        }
    }
}
```

افزودن Connection String در web.config برای دسترسی به دیتابیس

```
<connectionStrings>
  <add name="DefaultConnection" connectionString="Data Source=(local);DataBase=demodb;user id=sa;
password=xyz;" providerName="System.Data.SqlClient" />
</connectionStrings>
```

ایجاد یک Controller برای نمایش داده ها در View

```
using AngularJSWithMVC.Models;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
namespace AngularJSWithMVC.Controllers
{
    public class EmployeeController: Controller
    {
        public ActionResult Index()
        {
            return View();
        }
        public JsonResult GetAllEmployee()
        {
            using(DatabaseContext db = new DatabaseContext())
            {
                var employeeList = db.EmployeeModels.ToList();
                return Json(employeeList, JsonRequestBehavior.AllowGet);
            }
        }
    }
}
```

قدم دوم:

اکنون زمان نصب ماژول AngularJs ، Service و Controller می باشد. بنابراین یک فولدر با نام AngularScripts **درون** فولدر Content ایجاد کنید.

AppModule.js

درون فولدر AngularScripts فایل AppModule.js را ایجاد کنید ، هنگامی که شما می خواهید تنظیمات ماژول را مدیریت می کنید و نام آن را AngularApp بگذارید.

```
var app = angular.module("AngularApp", []);
```

Service.js

یک فایل جاوا اسکریپت دیگر با نام Service.js برای واکنشی داده ها از دیتابیس و معرفی به Controller ایجاد کنید. شما باید یک نام برای سرویس مانند **EmployeeService** معرفی کنید. در اینجا ما داده های json را با استفاده از GetAllEmployee گرفته ایم که JsonResult از EmployeeController برگردانده می شود.

```
app.service("EmployeeService", function($http)
{
  this.getEmployee = function()
  {
    debugger;
    return $http.get("/employee/getallemployee");
  };
});
```

Controller.js

یک فایل جاوااسکریپت دیگر با نام Controller.js که داده را در View نمایش می دهد ایجاد کنید. شما همچنین باید یک نام یکتا برای Controller انتخاب کنید ، که ما در اینجا از **EmpCtrl** استفاده کرده ایم.

```
app.controller("EmpCtrl", function($scope, EmployeeService)
{
  GetAllEmployee();

  function GetAllEmployee()
  {
    debugger;
    var getAllEmployee = EmployeeService.getEmployee();
    getAllEmployee.then(function(emp)
    {
      $scope.employees = emp.data;
    }, function()
    {
      alert('Data not found');
    });
  }
});
```

قدم سوم:

اکنون زمان آن است که داده ها را در View نمایش دهیم ، اما قبل از آن ما باید نصب کامپوننت AngularJS و تنظیمات بر روی فایل Layout را انجام دهیم. مثل اضافه کردن همه کامپوننت های جاوااسکریپت که قبل از Service.js و Controller.js اضافه کردیم.

در زیر می توانید ببینید که ما از ng-app استفاده کرده ایم ، که به این معنی است که این یک برنامه Angular است و شما باید نامه برنامه ای که در AngularApp.js تعریف کرده اید معرفی کنید. ما همچنین از فایل CDN برای استفاده از ویژگی های Angular استفاده کرده ایم.

```
<!DOCTYPE html>
<html ng-app="AngularApp">

<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width" />
  <title>@ViewBag.Title</title>
  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.13/angular.min.js"></script>
  <script src="~/Content/AngularScripts/AppModule.js"></script>
  <script src="~/Content/AngularScripts/Service.js"></script>
  <script src="~/Content/AngularScripts/Controller.js"></script> @Styles.Render("~/Content/css")
</head>

<body> @RenderBody() @Scripts.Render("~/bundles/jquery") @RenderSection("scripts", required: false)
</body>

</html>
```

```
@{
    ViewBag.Title = "Index";
    Layout = "~/Views/Shared/_Layout.cshtml";
}<div ng-controller="EmpCtrl">
    <div>
        <h2 align="center">Angular JS Basic Example</h2>
        <h5 align="center">Employee List</h5>
        <table border="1" cellpadding="10" align="center">
            <tr>
                <td>
                    Employee Id
                </td>
                <td>
                    Employee Name
                </td>
                <td>
                    Address
                </td>
                <td>
                    Email Id
                </td>
            </tr>
            <tr ng-repeat="emp in employees">
                <td>
                    {{emp.EmployeeId}}
                </td>
                <td>
                    {{emp.EmployeeName}}
                </td>
                <td>
                    {{emp.Address}}
                </td>
                <td>
                    {{emp.EmailId}}
                </td>
            </tr>
        </table>
    </div>
</div>
```