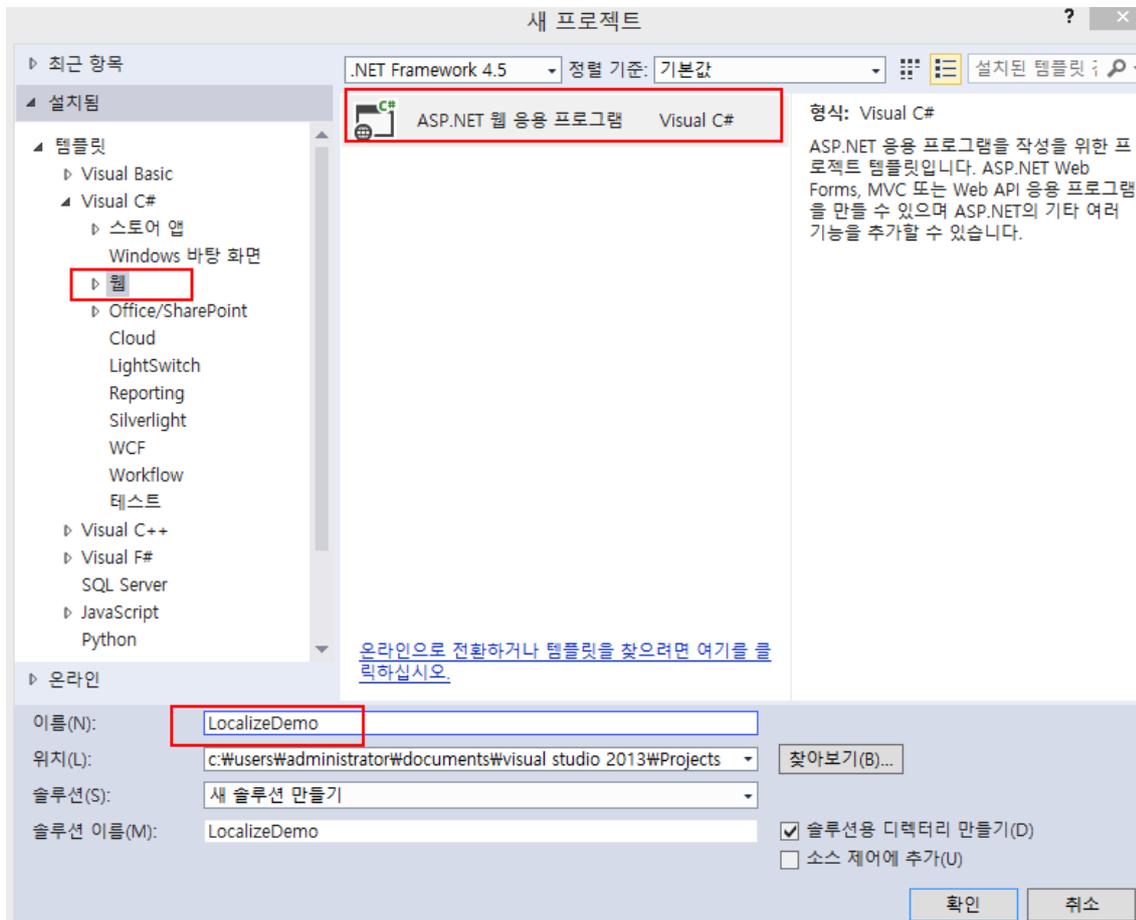
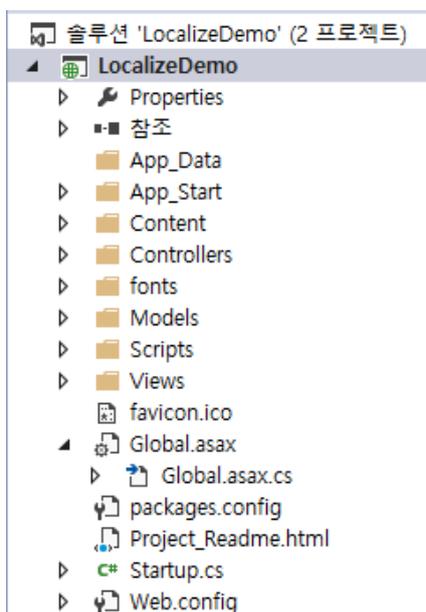


Visual studio 2013 을 열어, 프로젝트 생성을 한 다음, 웹 템플릿을 선택하고 그 중에 ASP.NET 웹 응용프로그램을 선택합니다. 이름은 LocalizeDemo 라고 기재했습니다.

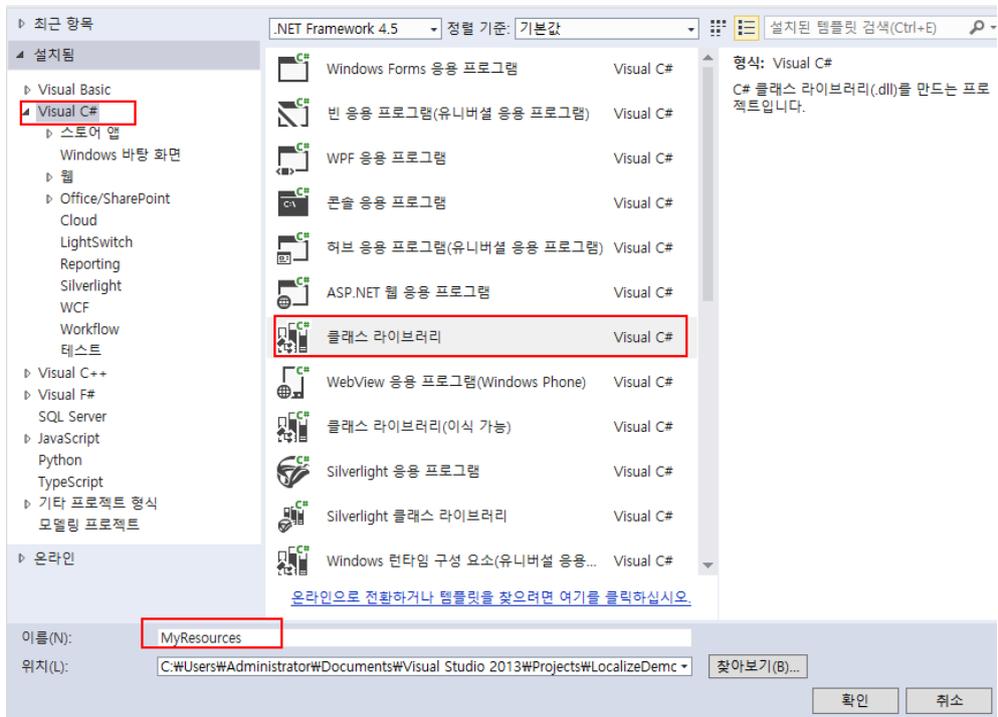


확인하고 나서, 그 다음 ASP.NET MVC 를 선택하고 이를 생성하면 아래와 같이 솔루션이 생성됩니다.

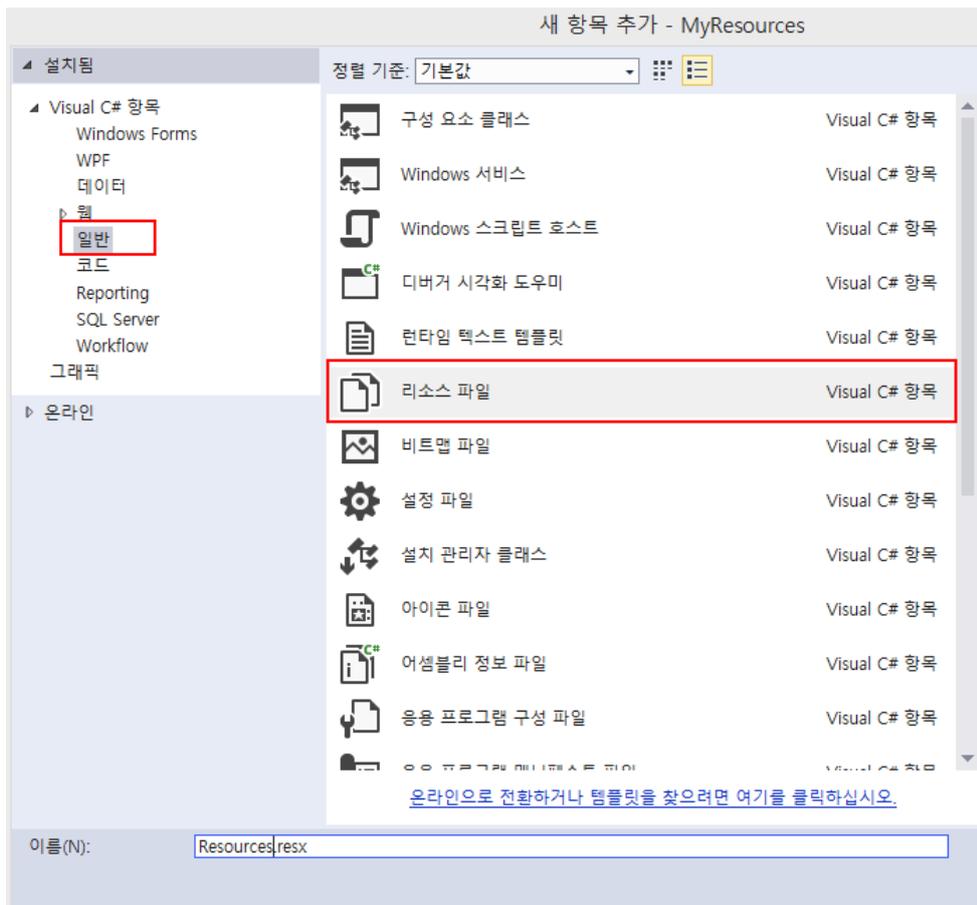


리소스를 별도로 클래스 라이브러리 생성하여 관리하면 편합니다.

제일 상단의 솔루션을 선택하고, 우클릭해서 새 프로젝트를 선택합니다. 이름은 MyResources 라고 기재하였습니다.



MyResources 프로젝트를 생성하고 나서 이제 리소스 파일을 하나 생성합니다. 이름은 MyResources.resx 라고 했습니다. 해당 파일이 구축되고 나면 다국별로 한국 경우 MyResources.ko-kr.resx 중국은 MyResources.ZH-CN.resx, 대만은 MyResources.ZH-TW.resx 라고 기재하면 됩니다. 한국어를 표기하기 위해 Resources.ko-KR.resx 을 만들었습니다.



생성한 파일은 아래의 솔루션 탐색기에서 확인이 가능합니다. 다국어 별로 파일을 만들게 되면 하나의 Designer.cs 에서 관리를 합니다.



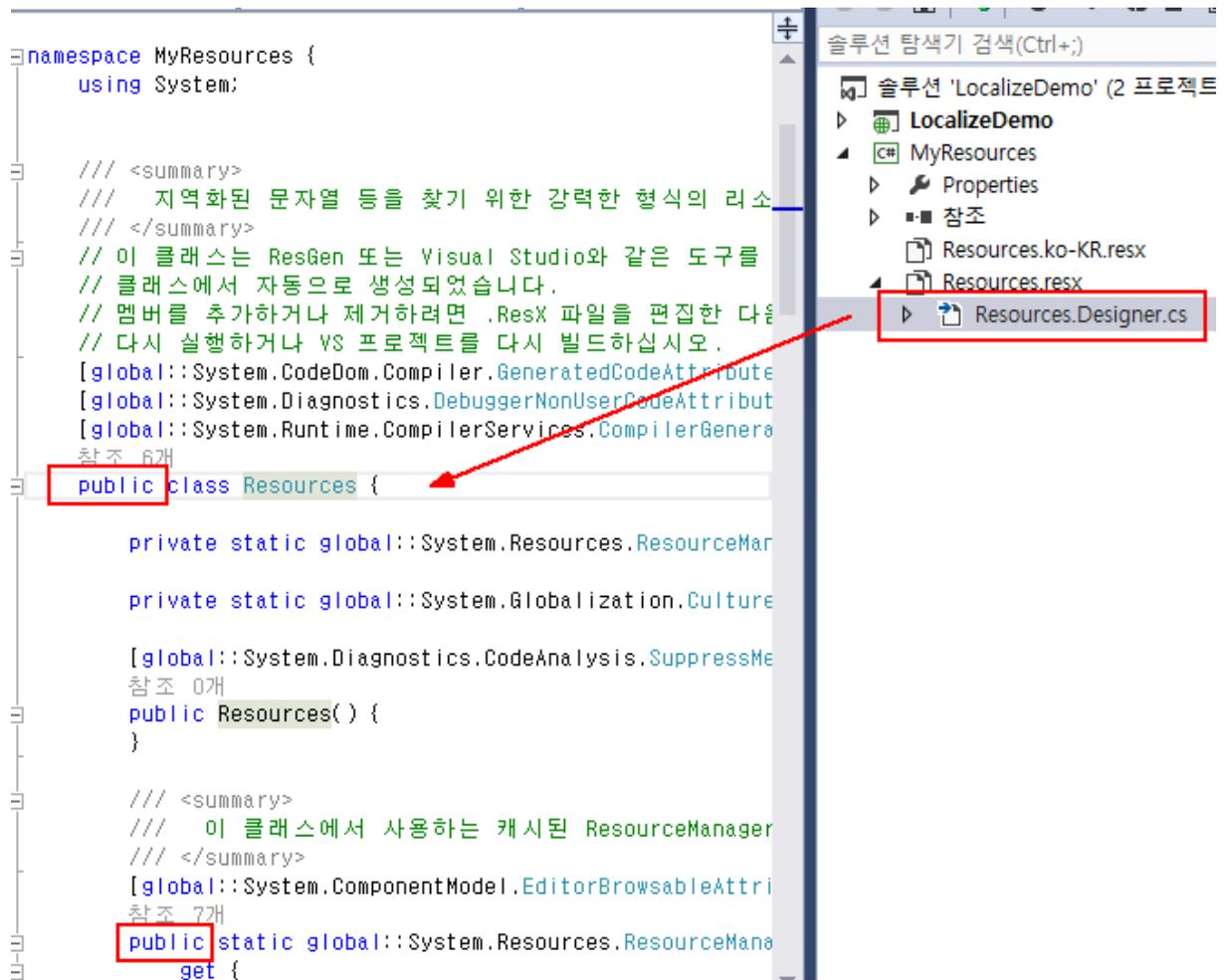
Resources.ko-kr.resx 파일을 열어서 아래와 같은 이름과 값을 입력합니다.

| | 이름 | 값 |
|---|-------------------|--------------|
| ▶ | Age | 나이 |
| | AgeRequiredError | 나이를 입력해 주세요. |
| | DisplayString | 데모 페이지 입니다. |
| | Name | 이름 |
| | NameRequiredError | 이름을 입력해 주세요. |
| | SiteHeader | 샘플 어플리케이션 |
| | Submit | 저장 |
| * | | |

Resources.resx 은 다국어 파일이 없는 국가들이 보는 리소스 입니다. 만국의 공통어인 영어로 처리하겠습니다.

| | 이름 | 값 |
|--|-------------------|---------------------|
| | Age | Age |
| | AgeRequiredError | Age is Required |
| | DisplayString | This is a demo page |
| | Name | Name |
| | NameRequiredError | Name is Required |
| | SiteHeader | Sample Application |
| | Submit | Submit |
| | | |

Resources.Designer.cs 파일을 열어보면 public 이 아닌 internal 접근자로 되어 있습니다. 이를 외부 솔루션에서 참조하여 사용할 수 있도록 internal 를 전부 public 으로 변경 합니다.



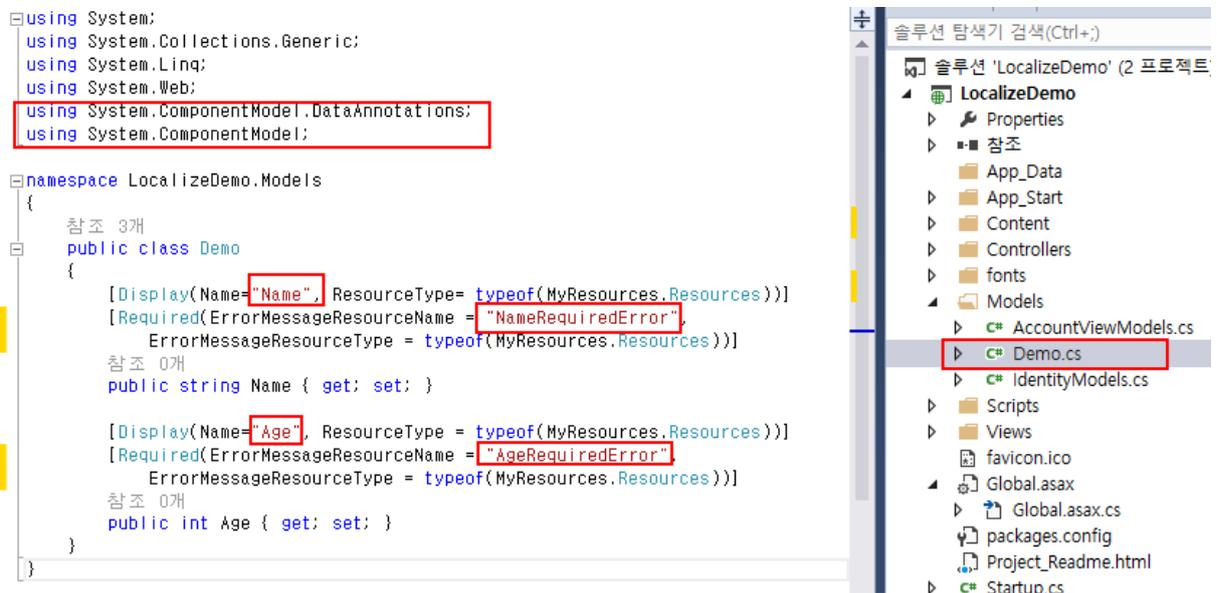
이제 리소스 프로젝트를 전부 작업 완료하였으니, 이젠 asp.net mvc 웹 프로젝트에서 이를 어떻게 가져올지 개발해 봅니다.

Global.asax.cs 파일 열어서, 브라우저의 다국어 값을 읽기 위해 Application_BeginRequest () 함수를 통해 기술합니다. Request.UserLanguages[0] 이 부분에서 한국어 브라우저이면 ko-KR 이라는 값을 얻게 되며, 이를 스레드 문화권 리소스 속성에 값을 할당하게 됩니다.



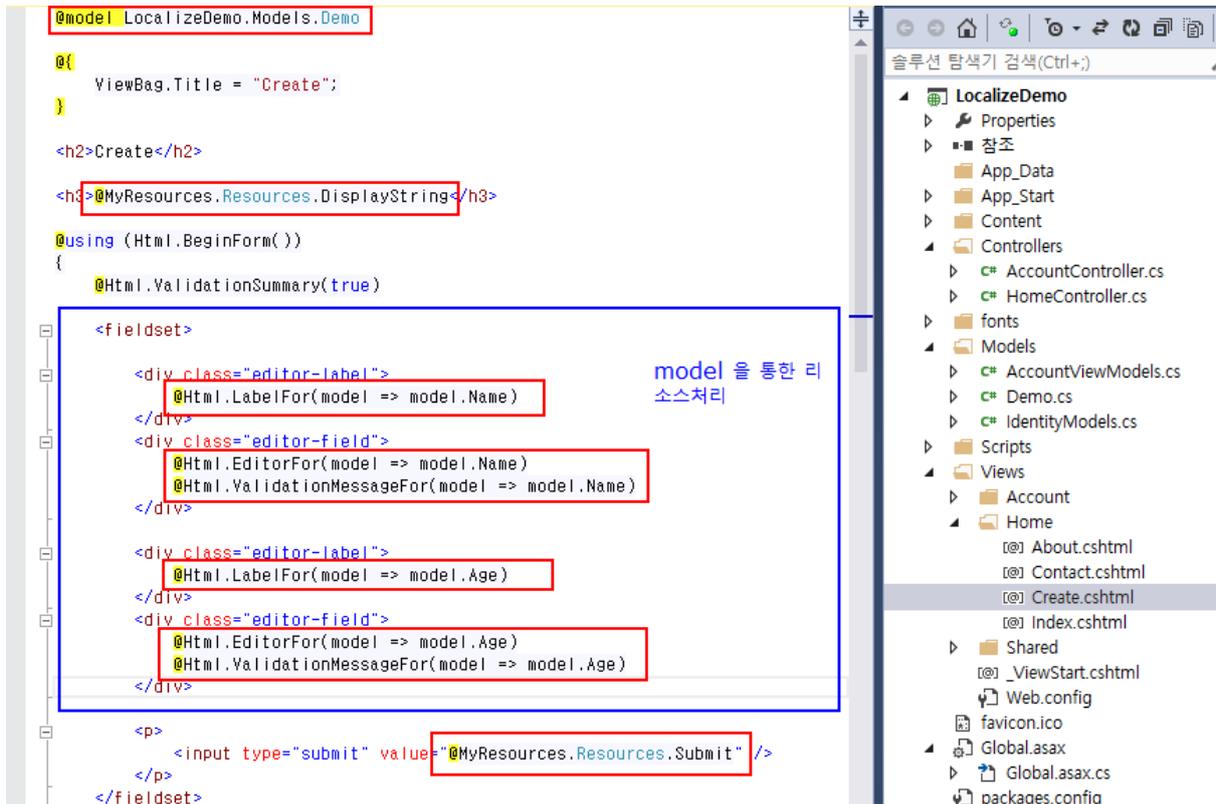
Model 속성에 리소스를 적용하기 위해서 System.ComponentModel.DataAnnotations; 를 사용하며, 이전에 만든 리소스의 이름을 Name="" 에 기재를 합니다.

[Display(Name="Name", ResourceType= typeof(MyResources.Resources))] 이는 label 에 기재된 model 에 보여줄 일반 용도의 리소스 이며, 경고 메시지는 [Required(ErrorMessageResourceName = "NameRequiredError", ErrorMessageResourceType = typeof(MyResources.Resources))] 처럼 Required 어트리뷰트를 통해 기술합니다.

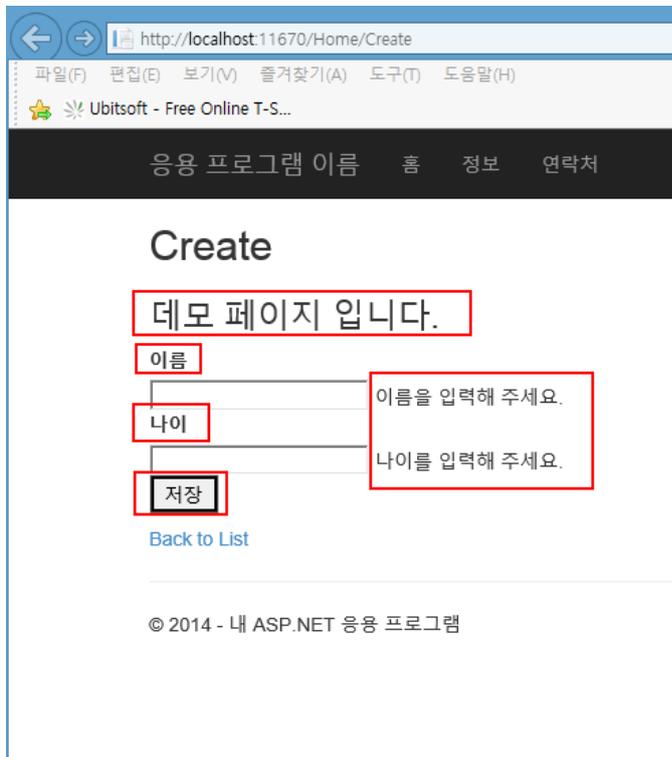


이제 Model 을 만들어서 다국어 지정하였고, 이를 보여줄 View 화면에서 기술해 봅니다.

Create.cshtml 의 뷰 페이지를 생성하고, 리소스의 이름 중에 DisplayString 값을 직접 @MyResources.Resources.DisplayString 처럼 호출이 가능하며, Model 에 기술한 부분은 그냥 글자 정도 뿌려주는 LabelFor 을 통해 속성 중 Name 을 @Html.LabelFor(model => model.Name) 식으로 호출 가능합니다. Validation 부분은 @Html.ValidationMessageFor(model => model.Name) 처럼 ValidationMessageFor 함수를 통해 model 을 기재 가능하고요.



페이지를 호출하게 되면 아래 화면 처럼 붉은 부분에 기재된 부분이 리소스에 호출되어 화면에 출력한 부분입니다. 저장 버튼을 그냥 선택하면 입력 값이 없어서 Require 로 인해 우측 옆에 경고 메시지가 호출되어 보여진 부분입니다.



감사합니다. shimpark@gmail.com 심재운