Olabilicek en basit şekilde yazılıcak.

İlk olarak Visual Studio 2015’i açın, eğer bilgisayarınızda kurulu değil ise şu adresten\* indirip kurun.

Ardından .Net Core’un kurulu olduğunu kontrol edin. Yüklü değil ise bu adresten\* indirip kurun.

VS dan new project’e tıklayıp açılan diyalogtan Templetes> Visual C#> Web altından ASP.Net Core Web Application templete’ini seçin. Name alanına proje adını (Ex. AspNetCoreSampleApp) yazdıktan sonra OK butonuna tıklayın. Açılan bir sonraki dialogtan WebApi seçeneğini seçiyoruz. Host in the Cloud tikini kaldırıyoruz. Çünkü suanlık Azure üzerinde uygulamamızı sunmıcaz. Konumuz Identity olmadığından Herhangi bir Authotentication tipide seçmiyoruz. Bunları yaptıktan sonra OK butonuna tıklıyouz.

Ve Uygulamamız oluştu.

Database ten işlem yapacağımız için Ilk önce Entity fameworkun eklı oldugunu kontrol edelim.

"Microsoft.EntityFrameworkCore": "1.0.1",

"Microsoft.EntityFrameworkCore.Sqlite": "1.0.1",

"Microsoft.EntityFrameworkCore.Design": "1.0.0-preview2-final" "Microsoft.AspNetCore.Mvc.WebApiCompatShim": "1.0.1"

"tools": {

"Microsoft.AspNetCore.Server.IISIntegration.Tools": "1.0.0-preview2-final",

"Microsoft.EntityFrameworkCore.Tools": "1.0.0-preview2-final"

Project json’a Ekliyerek yüklüyoruz.

[Route("api/[controller]")]

public class ValuesController : Controller

{

// GET api/values

[HttpGet]

public IEnumerable<string> Get()

{

return new string[] { "value1", "value2" };

}

// GET api/values/5

[HttpGet("{id}")]

public string Get(int id)

{

return "value";

}

// POST api/values

[HttpPost]

public void Post([FromBody]string value)

{

}

// PUT api/values/5

[HttpPut("{id}")]

public void Put(int id, [FromBody]string value)

{

}

// DELETE api/values/5

[HttpDelete("{id}")]

public void Delete(int id)

{

}

}

Mevcut kontrollerımızı silip notes diye kontroller olusturuyoruz db den okuma yazma gıbı ıslemler yapmak ıcın duzenlıyoruz.

Model diye klasor oluşturuyoruz. Note Class

using System.ComponentModel.DataAnnotations;

ekliyoruz

classımızı hazırlıyoruz

dotnet restore