Step 1: First Create Web Api Service AuthenticationService

Step 2: Add a class ApplicationUser

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.Linq;

using System.Web;

namespace AuthenticationService.Models

{

public class ApplicationUser

{ [Key]

public int UserId { get; set; }

public string Username { get; set; }

public string Password { get; set; }

}

}

Step 3: Make AuthContext Class

using System;

using System.Collections.Generic;

using System.Data.Entity;

using System.Linq;

using System.Web;

namespace AuthenticationService.Models

{

public class AuthContext : DbContext

{

public DbSet<ApplicationUser> Users { get; set; }

}

}

Step 4: Add Entry in web.config file

<connectionStrings>

<add name="AuthContext" connectionString="data source=LAPTOP-53S2KQS8\SQLEXPRESS;initial catalog=AuthenticationDb;integrated security=true" providerName="System.Data.SqlClient"/>

</connectionStrings>

Step 5 : Now we need to seed data (which means we need to provide Initial Data)

For this , first add a class ApplicationUserInitiailzer which should inhrit from

DropCreateDatabaseIfModelChanges class or DropCreateDatabaseAlways

It has a seed method, which is used to provide Initial Data

using System;

using System.Collections.Generic;

using System.Data.Entity;

using System.Linq;

using System.Web;

namespace AuthenticationService.Models

{

**public class ApplicationUserInitiailzer : DropCreateDatabaseIfModelChanges<AuthContext>**

**{**

**protected override void Seed(AuthContext context)**

**{**

**var users = new List<ApplicationUser>**

**{**

**new ApplicationUser() { UserId=1, Username="user1" , Password="user1"},**

**new ApplicationUser() { UserId=2, Username="user2" , Password="user2"},**

**new ApplicationUser() { UserId=3, Username="user3" , Password="user3"},**

**new ApplicationUser() { UserId=4, Username="user4" , Password="user4"},**

**};**

**users.ForEach(s => context.Users.Add(s));**

**context.SaveChanges();**

}

}

}

Step 6: Now we need to call this method, For this , we will go to Global.asax file & call this method in App\_Start() method

using AuthenticationService.Models;

using System;

using System.Collections.Generic;

using System.Data.Entity;

using System.Linq;

using System.Web;

using System.Web.Http;

using System.Web.Mvc;

using System.Web.Optimization;

using System.Web.Routing;

namespace AuthenticationService

{

public class WebApiApplication : System.Web.HttpApplication

{

protected void Application\_Start()

{

**Database.SetInitializer<AuthContext>(new ApplicationUserInitiailzer());**

AreaRegistration.RegisterAllAreas();

GlobalConfiguration.Configure(WebApiConfig.Register);

FilterConfig.RegisterGlobalFilters(GlobalFilters.Filters);

RouteConfig.RegisterRoutes(RouteTable.Routes);

BundleConfig.RegisterBundles(BundleTable.Bundles);

}

}

}

Step 7: Build your project

Step 8 : Add Controller AuthController

Change the Route Settings , as we have to pass username from the route not id

using AuthenticationService.Models;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Net;

using System.Net.Http;

using System.Web.Http;

namespace AuthenticationService.Controllers

{

public class AuthController : ApiController

{

public IHttpActionResult Get()

{

AuthContext db = new AuthContext();

return Ok(db.Users.ToList());

}

public IHttpActionResult Post([FromUri] string Username, [FromBody] string Password)

{

AuthContext db = new AuthContext();

ApplicationUser user = db.Users.FirstOrDefault(s => s.Username == Username && s.Password == Password);

if (user != null)

return Ok();

else

return NotFound();

}

}

}

Check it in Postman

At Step No 10, create MVC Application , call this service