# Owin Tutorials : File Server

1. Nuget

install-package Microsoft.Owin.StaticFiles

install-package Microsoft.Owin.SelfHost

2. FIRST VERSION

using System;

using Microsoft.Owin.Hosting;

namespace Owin.Tutorials.FileServer

{

    class Program

    {

        static void Main(string[] args)

        {

            var url = "http://localhost:8080";

            WebApp.Start(url, builder => builder.UseFileServer(enableDirectoryBrowsing: true));

            Console.WriteLine("Listening at " + url);

            Console.ReadLine();

        }

    }

}

3. 2ND VERSION : with root folder  
namespace Owin.Tutorials.FileServer

{

    class Program

    {

        static void Main(string[] args)

        {

            var url = "http://localhost:8080";

            var root = @"C:\Work";

            var fileSystem = new PhysicalFileSystem(root);

            var options = new FileServerOptions

            {

                EnableDirectoryBrowsing = true,

                FileSystem = fileSystem

            };

            WebApp.Start(url, builder => builder.UseFileServer(options));

            Console.WriteLine("Listening at " + url);

            Console.ReadLine();

        }

    }

}

# Owin Tutorials : File Server

1. Nuget

Microsoft.AspNet.WebApi.OwinSelfHost

2. Add a Startup class

namespace Owin.Tutorials.WebApi

{

    class Startup

    {

        // This code configures Web API. The Startup class is specified as a type

        // parameter in the WebApp.Start method.

        public void Configuration(IAppBuilder appBuilder)

        {

            // Configure Web API for self-host.

            HttpConfiguration config = new HttpConfiguration();

            config.Routes.MapHttpRoute(

                name: "DefaultApi",

                routeTemplate: "api/{controller}/{id}",

                defaults: new { id = RouteParameter.Optional }

            );

appBuilder.UseWebApi(config);

        }

    }

}

3. Add a controller class

public class ValuesController : ApiController

    {

        // GET api/values

        public IEnumerable<string> Get()

        {

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

        }

        // GET api/values/5

        public string Get(int id)

        {

            return "value";

        }

        // POST api/values

        public void Post([FromBody]string value)

        {

        }

        // PUT api/values/5

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

        {

        }

        // DELETE api/values/5

        public void Delete(int id)

        {

        }

    }

4. Start server and client

class Program

    {

        static void Main()

        {

            string baseAddress = "http://localhost:8080/";

            // Start OWIN host

            using (WebApp.Start<Startup>(url: baseAddress))

            {

                // Create HttpCient and make a request to api/values

                HttpClient client = new HttpClient();

                var response = client.GetAsync(baseAddress + "api/values").Result;

                Console.WriteLine(response);

                Console.WriteLine(response.Content.ReadAsStringAsync().Result);

            }

            Console.ReadLine();

        }

    }