


Question1

Friday, August 30, 2024 10:34 AM

Question :

Create a new c# web api in dot net core. Add two controllers named Patient and Provider. Then add a common static class named Address, with an exposed static property named HomeAddress. Provider an endpoint to update and get the HomeAddress from Patient controller. Also provided an endpoint to get and update the homeaddress from Provider controller.


Create a new c# web api in dot net core. Add two controllers named Patient and Provider. Then add a common static class named Address, with an exposed static property named HomeAddress. Provider an endpoint to update and get the HomeAddress from Patient controller. Also provided an endpoint to get and update the homeaddress from Provider controller.

 GitHub Copilot

Here's how you can create a new C# Web API in .NET Core with the specified requirements:

1. **Create a new Web API project:** Open Visual Studio and create a new project. Select "ASP.NET Core Web API" and follow the prompts to create the project.
2. **Add the Address static class:** Create a new file named `Address.cs` in the `Models` folder (you may need to create this folder).

```
namespace YourNamespace.Models
{
    public static class Address
    {
        public static string HomeAddress { get; set; }
    }
}
```

 Insert in new file Preview

3. **Add the PatientController:** Create a new file named `PatientController.cs` in the `Controllers` folder.

```
using Microsoft.AspNetCore.Mvc;
using YourNamespace.Models;

namespace YourNamespace.Controllers
{
    [ApiController]
    [Route("api/[controller]")]
    public class PatientController : ControllerBase
    {
        [HttpGet("homeaddress")]
        public ActionResult<string> GetHomeAddress()
        {
            return Ok(Address.HomeAddress);
        }
    }
}
```

```
        [HttpPost("homeaddress")]
        public ActionResult UpdateHomeAddress([FromBody] string homeAddress)
        {
            Address.HomeAddress = homeAddress;
            return Ok();
        }
    }
}
```

Insert in new file Preview

4. **Add the ProviderController:** Create a new file named `ProviderController.cs` in the `Controllers` folder.

```
using Microsoft.AspNetCore.Mvc;
using YourNamespace.Models;

namespace YourNamespace.Controllers
{
    [ApiController]
    [Route("api/[controller]")]
    public class ProviderController : ControllerBase
    {
        [HttpGet("homeaddress")]
        public ActionResult<string> GetHomeAddress()
        {
            return Ok(Address.HomeAddress);
        }

        [HttpPost("homeaddress")]
        public ActionResult UpdateHomeAddress([FromBody] string homeAddress)
        {
            Address.HomeAddress = homeAddress;
            return Ok();
        }
    }
}
```

```
        return Ok();
    }
}
```

Insert in new file Preview

5. **Run the application:** Press `F5` to run the application. You can test the endpoints using tools like Postman or cURL.

Endpoints:

- GET `/api/patient/homeaddress` - Get the home address from the `PatientController`.
- POST `/api/patient/homeaddress` - Update the home address from the `PatientController`.
- GET `/api/provider/homeaddress` - Get the home address from the `ProviderController`.
- POST `/api/provider/homeaddress` - Update the home address from the `ProviderController`.

Make sure to replace `YourNamespace` with the actual namespace of your project.

[🔗 How can I test the endpoints in my C# Web API?](#)

References

Was this helpful?