Validating Your Web API Service Code

Benjamin Day
TRAINER | COACH | DEVELOPER

@benday   www.benday.com

Overview

Testing Web API

What is different between
testing ASP.NET MVC and Web API?

Answer:
Not much.

ASP.NET MVC Controller


```
4 references | Benjamin Day, 49 days ago | 1 author, 1 change | 0 requests
public class SearchController : Controller
{
    private IPresidentService _Service;
    private IFeatureManager _FeatureManager;

    1 reference | Benjamin Day, 49 days ago | 1 author, 1 change | 0 exceptions
    public SearchController(IPresidentService service,
        IFeatureManager featureManager)
    {
        if (service == null)
            throw new ArgumentNullException("service",
```

Base Class = Microsoft.AspNetCore.Mvc.Controller

ASP.NET Web API Controller

```csharp
[Route("api/[controller]")]
[ApiController]
1 reference | Benjamin Day, 26 days ago | 1 author, 1 change | 0 requests
public class PresidentController : ControllerBase
{
    private IPresidentService _Service;

    0 references | Benjamin Day, 26 days ago | 1 author, 1 change | 0 exceptions
    public PresidentController(IPresidentService service)
    {
        _Service = service;
    }
```

Base Class = Microsoft.AspNetCore.Mvc.ControllerBase

Base Classes

ASP.NET Web API
- Microsoft.AspNetCore.Mvc.ControllerBase

ASP.NET MVC
- Microsoft.AspNetCore.Mvc.Controller

Controller's base class is ControllerBase
- Controller knows about Views

MVC and Web API both use Dependency Injection.

From a testing perspective, ASP.NET MVC and Web API are virtually the same.

The big difference?

MVC returns HTML

Web API returns objects represented as strings

Overview

Testing Web API

Integration testing
- JSON-based responses
- XML-based responses

Unit testing JSON customization

Demo

Testing Web API

Integration testing of Accept headers
- application/json
- application/xml

Return JSON or XML

Next up:
Unit testing JSON customizations

Demo

Unit testing custom JSON formats

Summary

Testing Web API

Integration testing
- JSON-based responses
- XML-based responses

Unit testing JSON customization

Thanks!

@benday
www.benday.com