

xUnit - Display test names for theory memberdata (TestCase)

Ask Question

Asked 4 years, 9 months ago Modified 6 days ago Viewed 12k times

I've been using NUnit for testing and I'm really fond of test cases. In NUnit you can easily set each test name in the test case using the SetName function in a TestCaseData class.

Does xUnit have a similar function for this?

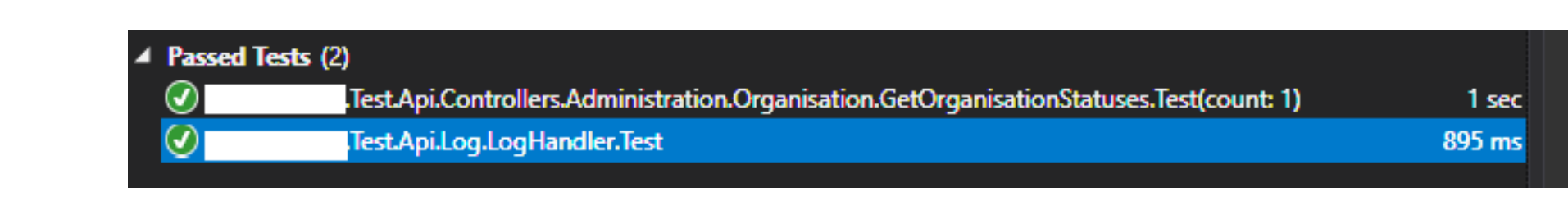
Currently I can only see one test in the test explorer even tho I have 6 tests in the test case.

```
public class LogHandler : TestBase
{
    private ILogger _logger;

    public LogHandler()
    {
        //Arrange
        LogAppSettings logAppSettings = GetAppSettings<LogAppSettings>("Log");
        IOptions<LogAppSettings> options = Options.Create(logAppSettings);
        LogService logService = new LogService(new Mock<IdentityService>().Object,
        LogProvider logProvider = new LogProvider(logService);
        _logger = logProvider.CreateLogger(null);
    }

    public static IEnumerable<object[]> TestCases => new[]
    {
        new object[] { LogLevel.Critical,
        new EventId(),
        new Exception(),
        1 },
        new object[] { LogLevel.Error,
        new EventId(),
        new Exception(),
        1 },
        new object[] { LogLevel.Warning,
        new EventId(),
        new Exception(),
        0 },
        new object[] { LogLevel.Information,
        new EventId(),
        new Exception(),
        0 },
    }
}
```

xUnit test window



Should be 6 tests here instead of one! (ignore GetOrganisationStatuses).

NUnit test case

```
public static IEnumerable TestDatabaseCases
{
    get
    {
        yield return new TestCaseData(LogLevel.Critical,
        new EventId(),
        new Exception("Exception"),
        0,
        1).SetName("InsertException_Should_Insert_When_LogLevel_Critical");

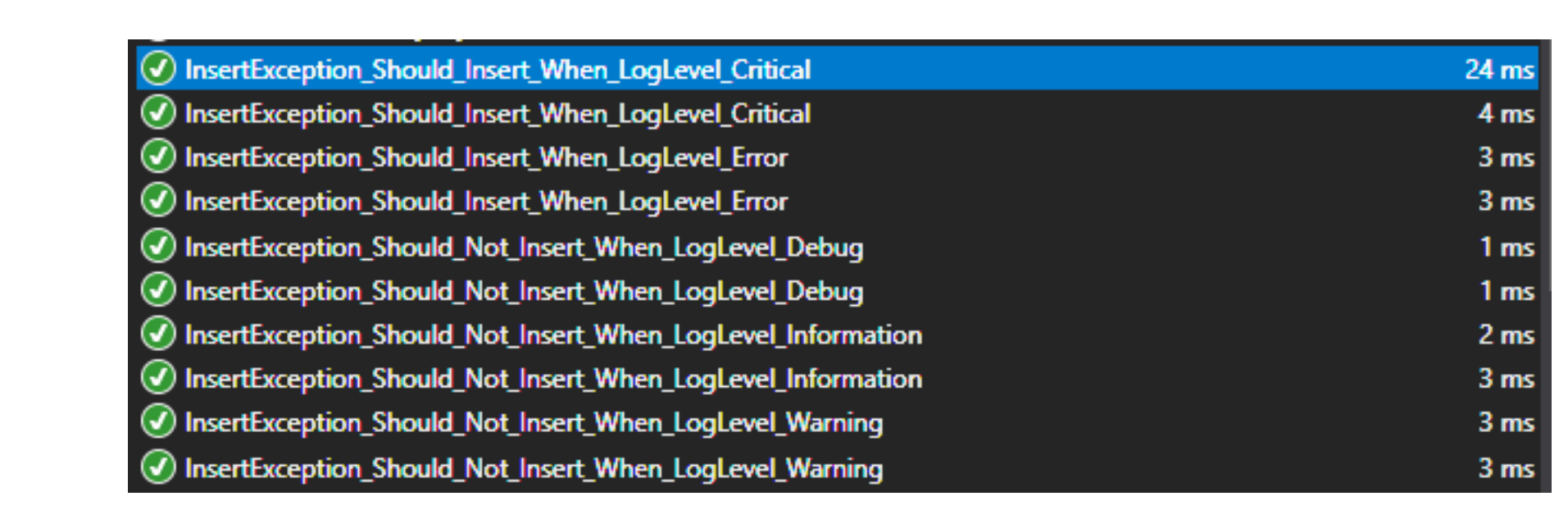
        yield return new TestCaseData(LogLevel.Error,
        new EventId(),
        new Exception("Exception"),
        0,
        1).SetName("InsertException_Should_Insert_When_LogLevel_Error");

        yield return new TestCaseData(LogLevel.Warning,
        new EventId(),
        new Exception("Exception"),
        0,
        0).SetName("InsertException_Should_Not_Insert_When_LogLevel_Warning");

        yield return new TestCaseData(LogLevel.Information,
        new EventId(),
        new Exception("Exception"),
        0,
        0).SetName("InsertException_Should_Not_Insert_When_LogLevel_Information");

        yield return new TestCaseData(LogLevel.Debug,
        new EventId(),
        new Exception("Exception"),
        0,
        0).SetName("InsertException_Should_Not_Insert_When_LogLevel_Debug");
    }
}
```

NUnit test window



This is what I want in xUnit!

How do I in xUnit set a name for each test in the test case?

Share Improve this question Follow

edited Oct 26, 2017 at 13:52

asked Oct 14, 2017 at 20:43

Ref 2,203 4 30 64

I had the same problem before. You class will need to inherit from IXunitSerializable. It's quite cumbersome to implement. Not sure if there is a better method. – Seb Oct 15, 2017 at 2:24

There is a ready to use implementation which bypasses a need to change code under test and test code. Simply include new custom xUnit TheoryAttribute, XunitDiscoverer class and two test classes - see second answer for details and MIT licensed working code. – Jacek Blaszczyński Oct 21, 2017 at 14:07

Add a comment

4 Answers Sorted by: Trending sort available Highest score (default)

Currently I can only see one test in the test explorer even tho I have 6 tests in the test case.

This is because your test data is not considered serializable by xUnit.net. See this issue: <https://github.com/xunit/xunit/issues/1473>

The salient details are:

The short answer

If some of your theory data can't be "serialized" by xUnit.net, then it cannot be encapsulated into the serialization of a test case which we're required to do for the Visual Studio test runner.

The long answer

In the Visual Studio test runner, test cases are discovered in one process, and executed in another. Therefore, test cases must be able to be turned into an unqualified string representation (aka, "serialized") in order to be run. We can also serialize at the test method level, because that just involves knowing the type and method name (both strings). When you start putting data into the mix, we need to ensure we know how to serialize that data; if we can't serialize **all** of the data for a theory, then we have to fall back to just a single method (which we know we can serialize).

The full answer contains more detail.

In xUnit you can easily set each test name in the test case using the SetName function in a TestCaseData class.

Does xUnit have a similar function for this?

Not at this time.

Share Improve this answer Follow

edited Jun 20, 2020 at 9:12

answered Oct 14, 2017 at 23:15

Community Bot 1 82

Brad Wilson 65.7k 9 71 82

3 That's a shame. As it stands now I'll continue with NUnit. Hopefully they will fix this. – Ref Oct 15, 2017 at 8:29

@Ref Common it's not a shame but small limitation which can be easily compensated for by small piece of code - see below for my answer. Anyway xUnit is a test framework of choice for large number of most important in DotNet world repos. – Jacek Blaszczyński Oct 21, 2017 at 14:14

@JacekBlaszczyński Can you comment on what those reasons might be? – Ibrahim Jul 26, 2018 at 4:23

Add a comment

This worked for me:

```
public class TestScenario
{
    ...
    public override string ToString()
    {
        return $"Role: {Role}...";
    }
}

[Theory]
[ClassData(typeof(MyProvider))]
public void TestScenarios(TestScenario scenario)
{
    TestScenarioInternal(scenario);
}
```

Share Improve this answer Follow

answered Mar 16, 2021 at 17:49

crazybert 159 1 71 82

1 This was exactly what I was looking for, thanks! – MvRens Feb 2 at 13:43

Add a comment

Actually there is a working solution requiring some plumbing code which should work with unchanged tests. It requires implementation of custom TheoryAttribute, custom TheoryDiscoverer, and custom TestCase classes. Whole solution is available under MIT license in this repo [DjvuNet/DjvuNetShared.Tests](#)

Required files with implementations are: [DjvuTheoryAttribute](#), [DjvuTheoryDiscoverer](#), [DjvuNamedDataRowTestCase](#), [DjvuDataRowTestCase](#)

Usage is straightforward: compile above files either including them directly into test assembly or as a separate assembly and use them in code as [follows](#):

```
[DjvuTheory]
[ClassData(typeof(DjvuJsonDataSource))]
public void DirMChunk_Theory(DjvuJsonDocument doc, int index)
{
    int pageCount = 0;
    using (DjvuDocument document = DjvuNet.Tests.Util.GetTestDocument(index, out
    {
        DjvuNet.Tests.Util.VerifyDjvuDocument(pageCount, document);
        DjvuNet.Tests.Util.VerifyDjvuDocumentCtor(pageCount, document);

        // DirMChunk is present only in multi page documents
        // in which root form is of DjvuChunk type
        if (document.RootForm.ChunkType == ChunkType.Djvm)
        {
            DirMChunk dirM = ((DjvmChunk)document.RootForm).DirM;
            Assert.NotNull(dirM);

            Assert.True(dirM.IsBundled ? doc.Data.DirM.DocumentType == "bundled"
            var components = dirM.Components;
            Assert.Equal<int>(components.Count, doc.Data.DirM.FileCount);
        }
    }
}
```

One of the theory function arguments is not serializable in xUnit but despite that theory tests will be displayed individually and numbered. If first argument to theory function is of the string type it will be used as a name of the test besides being an argument to function invocation.

Credit for the idea goes to other developer - I have to find a link to his code - but it was reimplemented from scratch for DjvuNet project.

Share Improve this answer Follow

answered Oct 15, 2017 at 12:57

Jacek Blaszczyński 3,085 13 25

Add a comment

One readable and short way to do it would be:

```
public record AddNumbersTestCase(
    int Number1,
    int Number2,
    int ExpectedResult)
{
    private readonly AddNumbersTestCase[] TestCases =
    {
        new(0, 0, 0),
        new(0, 1, 1),
        new(1, 0, 1),
        new(1, 1, 2),
        new(2, 3, 5),
    };

    public static IEnumerable<object[]> TestCasesData =>
        TestCases.Select(testCase => new object[] { testCase });

    public override string ToString() =>
        $"{Number1} + {Number2} should equal {ExpectedResult}";
}

[Theory]
[MemberData(nameof(AddNumbersTestCases), MemberType = typeof(AddNumbersT
public void FindDayOfWeek(AddNumbersTestCase testCase)
{
    int result = testCase.Number1 + testCase.Number2;

    Assert.Equal(testCase.ExpectedResult, result);
}
```

Share Improve this answer Follow

answered Aug 2 at 14:24

Robert Synoradzki 1,506 12 18

Add a comment

Your Answer

B I

Sign up or log in

Post as a guest

Name

Email

Required, but never shown

Post Your Answer

By clicking "Post Your Answer", you agree to our terms of service, privacy policy and cookie policy

Not the answer you're looking for? Browse other questions tagged [asp.net-core](#) [xunit](#) or [ask your own question](#).

The Overflow Blog

- ✓ Satellite internet: More useful than sending a car into space (Ep. 470)
- ✓ Great engineering cultures are built on social learning communities

Featured on Meta

- Should we burninate the [master] tag?
- Announcing Design Accessibility Updates on SO

Related

- 23 Dynamic + link compilation error
- 2 Disposing TestCaseSource elements in NUnit test
- 15 Use AutoData and MemberData attributes in XUnit test
- 26 xUnit theory test using generics
- 3 XUnit and MSTest with ExpectedException Returning Different Results
- 6 xUnit Non-Static MemberData
- 13 xUnit Theory with async MemberData
- 6 Test Exception with Theory in XUnit
- 0 Cannot cast 'Microsoft.EntityFrameworkCore.Internal.IdentitySet' [Microsoft.AspNet.Core.Identity.IdentityUser] [System.Int32]

Hot Network Questions

- What kind of melee weapon is ideal to destroy a monster's core?
- NULL values appear when taking the sum of two columns taken from Lookup tables
- What play are these lines a reference to?
- Are pig testicles edible?
- Can a person be immune to all diseases and poisons
- When you velvet chicken, are you supposed to wash off the baking soda?
- How did the teacher know that Bobby had cheated?
- ERROR in every language
- Do smaller riders tend to need a wider saddle?
- Preview Animation?
- Throwing and catching a clock in general relativity
- Is it legal to own a photo of someone and spread it around without their permission?
- Movie where a woman goes to a world not known to humanity or to some distant star system through a secret project
- Why do compressed and uncompressed public keys have to produce different addresses?
- 'The Needle' human alien war against the Scetti (sweaty)
- What's the origin of process return status 0 meaning success?
- How can a root CA be kept offline and function
- What has China gained by doing military drills after Nancy Pelosi's visit to Taiwan?
- Word for when a derogatory form of address is used affectionately?
- What is a single word that describes "excessive" human desire for external validation" (i.e., popularity)?
- What is this icon called?
- Is it alright to kill off a character without setting up death flags?
- Modern researcher communicates ideas to Elizabethan inventor resulting in accelerating technology feedback loop

Question feed