

NUnit Tests für C# in Unity

Seminar im Fach Software Projekt 2

Inhaltsangabe

- 1. Anwendung von NUnit für C#
- 2. Code Coverage
- 3. Praxis NUnit mit Unity

1. Anwendung von NUnit für C#

Was ist NUnit

- Unit Test Framework für C#
- Im Asset Store unter **Unity Testing Tools** verfügbar
- Erlaubt es Unit.- und Integrations-tests zu schreiben

1. Anwendung von NUnit für C#

Annotationen in NUnit: Unit Tests vorbereiten

C#

```
namespace Beispiel    //<- Wenn nötig, im selben Namespace sein wie die zu testende Klasse!
{
    using NUnit.Framework; //<- Framework importieren!

    [TestFixture] //<- Annotation benötigt
    public class AccountTest
    {
        [ClassInitialize()] //<- @BeforeClass
        public static void ClassInit(TestContext context) {}

        [SetUp] // @Before
        public void Initialize() {}

        [Test] //<- Annotation für Tests
        public void TransferFunds()
        {
            Assert.IsTrue(true);
        }

        [TearDown] // @After
        public void Cleanup() {}

        [ClassCleanup()] // @AfterClass
        public static void ClassCleanup() {}
    }
}
```

JAVA

```
public class AccountTest
{
    @BeforeClass
    public static void ClassInit(TestContext context) {}

    @Before
    public void Initialize() {}

    @Test
    public void TransferFunds()
    {
        assertTrue(true);
    }

    @After
    public void Cleanup() {}

    @AfterClass
    public static void ClassCleanup() {}
}
```

1. Anwendung von NUnit für C#

Unit Tests programmieren: Beispiel

```
using UnityEngine;
using NUnit.Framework;
using System.Threading;
using System.Collections;

namespace UnityTest
{
    [TestFixture]
    public class TestPresentationNUnit {

        private static int RANDOM_NUMBER= 3912;
        private static int RANGE = 10000;
        private static int INCOREECT_NUMBER= 13912;

        private static string GREATER_THAN = "Your Number is to big!";

        private PresentationNUnit presentation;

        [SetUp]
        public void init(){
            this.presentation = new PresentationNUnit (RANGE);
            this.presentation.setRandomNmber (RANDOM_NUMBER);
        }
    }
}
```

```
[Test]
[MaxTime(100)]
[Category("Failing Tests")]
public void SlowTest()
{
    Thread.Sleep(200);
}

[Test]
public void testIsCorrectNumber(){
    bool isCorrect = this.presentation.isCorrectNumber (INCOREECT_NUMBER);
    Assert.IsFalse (isCorrect);
}

[Test]
public void testGuessTotalRange(){
    string result = this.presentation.guessTotalRange (RANDOM_NUMBER);
    Assert.AreEqual (GREATER_THAN, result);
}

[TearDown]
public void dispose(){
    this.presentation = null;
}
}
```

1. Anwendung von NUnit für C#

Assertions und weitere Annotationen

- Vom Ablauf her genau so wie in JUnit

⇒ Siehe Link in der Anleitung für **weitere** Assertions

Identity Asserts

Assert.AreSame and **Assert.AreNotSame** test whether the same objects are referenced by the two arguments.

```
Assert.AreSame( object expected, object actual );
Assert.AreSame( object expected, object actual, string message );
Assert.AreSame( object expected, object actual, string message,
               params object[] parms );

Assert.AreNotSame( object expected, object actual );
Assert.AreNotSame( object expected, object actual, string message );
Assert.AreNotSame( object expected, object actual, string message,
                  params object[] parms );
```

Assert.Contains is used to test whether an object is contained in an array or list.

```
Assert.Contains( object anObject, IList collection );
Assert.Contains( object anObject, IList collection,
               string message );
Assert.Contains( object anObject, IList collection,
               string message, params object[] parms );
```

2. Code Coverage

Was ist Open Cover

- Coverage Tool das beschreibt wie weit der geschriebene Code durch Tests abgedeckt ist
 - Dient zur Fehlervermeidung und zur Qualitätsverbesserung
- Es gibt kein kostenfreies Code Coverage Tool für C# unter Mac OS!
⇒ D.h. Verwendung von OpenCover + SharpDevelop unter Windows

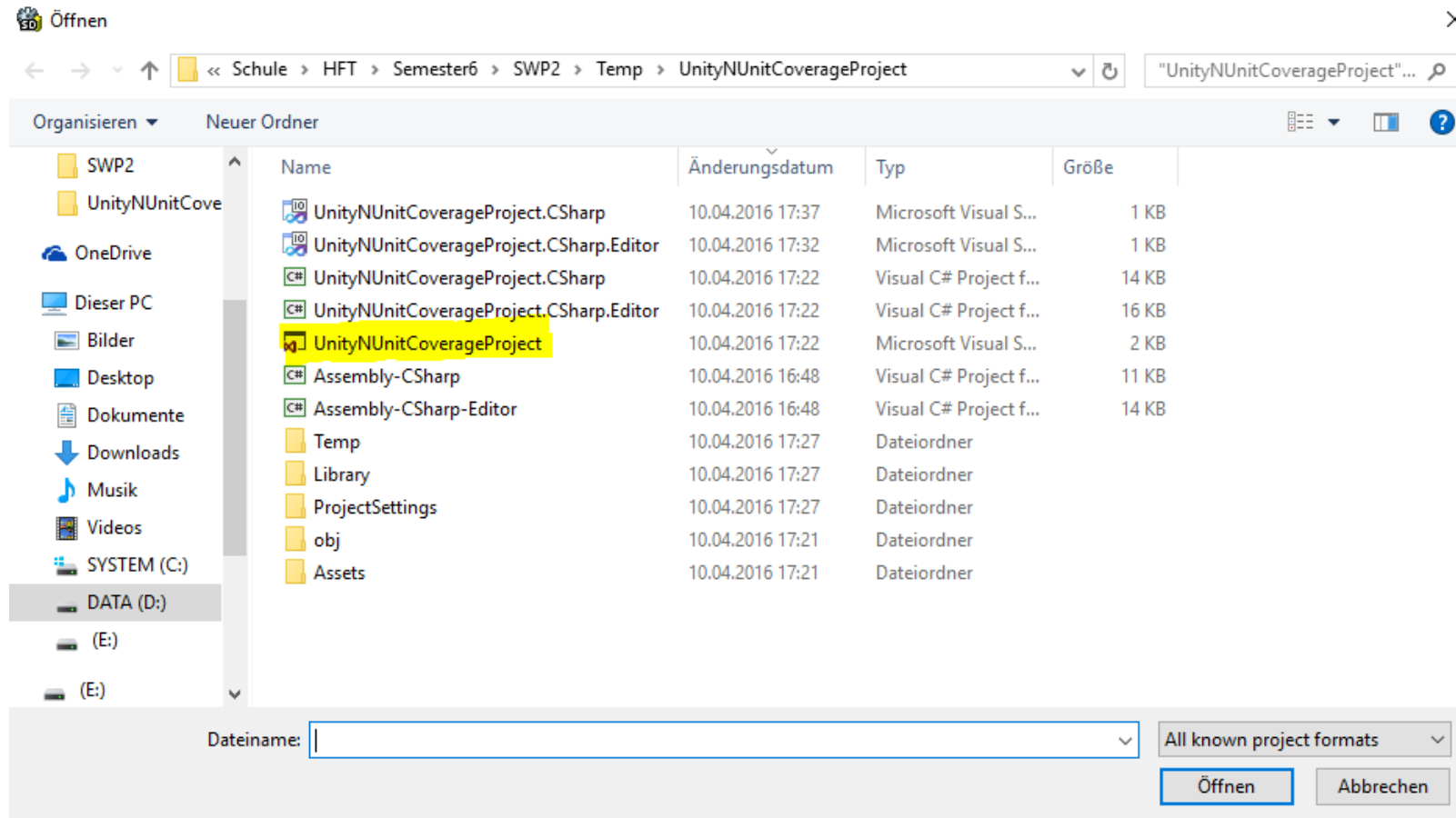
2. Code Coverage

Open Cover mit SharpDevelop: Einfache Vorgehensweise

- Open Cover + SharpDevelop für Windows runterladen und installieren
=> Siehe Anleitung
- Projekt in SharpCover importieren
- Code Coverage ausführen

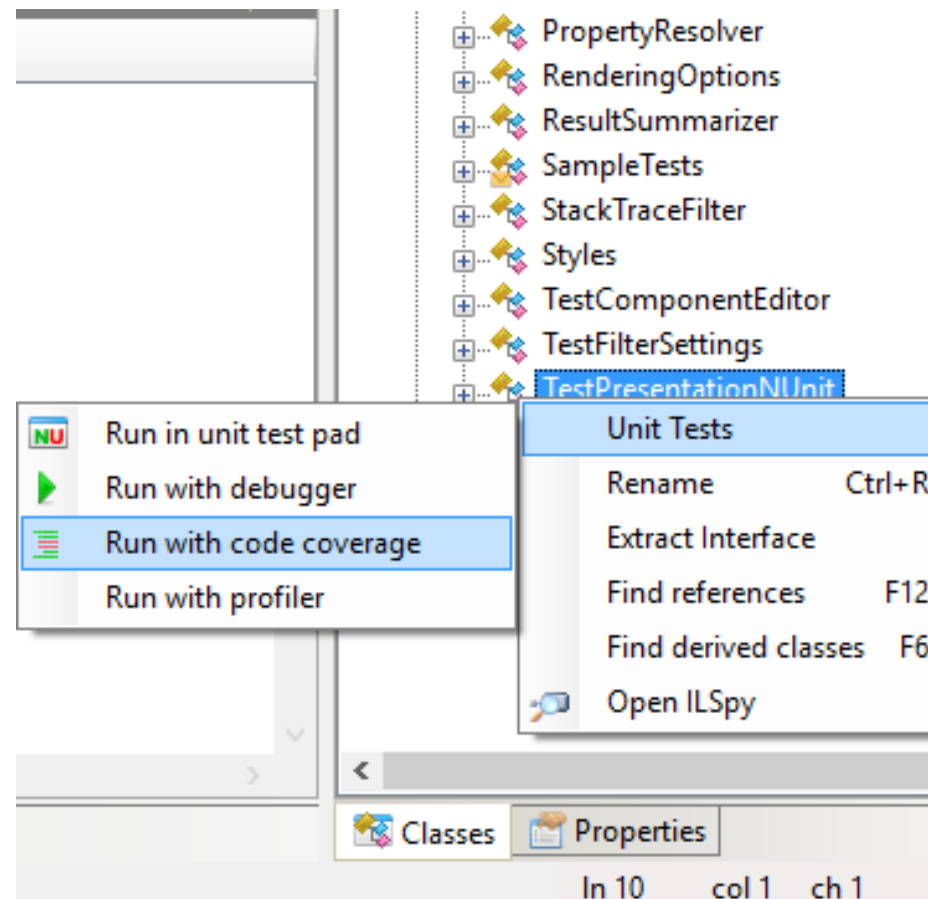
2. Code Coverage

Code Coverage Beispiel



2. Code Coverage

Code Coverage Beispiel



2. Code Coverage

Code Coverage Beispiel

```
using System;

namespace AssemblyCSharpEditor
{
    public class ClassToTest
    {
        public int getNumber(int number)
        {
            if (number == 42) {
                return 3912;
            } else {
                return 2;
            }
        }
    }
}
```

