

ТЕСТИРОВАНИЕ

<https://github.com/kontur-courses/testing>



ДОВЕРИЕ ТЕСТАМ

- Будет ли тест понятен ревьюеру?
- Можно ли быстро убедиться в корректности теста?
- Можно ли быстро понять, что он проверяет?

ТЕСТЫ КАК СПЕЦИФИКАЦИЯ

```
[TestFixture]
public class Superman_Should
{
    [Test]
    public void SaveKittenFromTree()
    {
        ...
        superman.Act();
        Assert.That(kitten.IsSaved());
    }

    [Test]
    public void WearRedBlueSuit_WhenAtWork()
    {
        ...
    }
    ...
}
```

ТЕСТЫ КАК СПЕЦИФИКАЦИЯ

```
[TestFixture]
public class Superman_Should
{
    [Test]
    public void SaveKittenFromTree()
    {...}

    [Test]
    public void WearRedBlueSuit_WhenAtWork()
    {...}
}
```

ИМЯ ТЕСТА КАК СПЕЦИФИКАЦИЯ

ЧТО ДОЛЖНО БЫТЬ В ИМЕНИ ТЕСТА?

System Under Test

ИМЯ КЛАССА, ИМЯ МЕТОДА

Условия

ВХОД, СОСТОЯНИЕ

Результат

ожидаемое поведение или требования для проверки

<http://java.dzone.com/articles/7-popular-unit-test-naming>

ИМЯ ТЕСТА КАК СПЕЦИФИКАЦИЯ

ParserTests.TestParse

ParserTests.Parse_Fails

ParserTests.Parse_BigNumbers

ParserTests.Parse_NumbersGreaterThanMaxInt

ParserTests.Fails_OnNegativeNumbers

ИМЯ ТЕСТА КАК СПЕЦИФИКАЦИЯ

`IsAdult_AgeLessThan18_False`

`ParseInt_Should.Fails_OnNonNumber`

`Stack_Should.BeEmpty_AfterCreation`

`When_MandatoryFieldsAreMissing_Expect_StudentAdmissionToFail`

ИМЯ ТЕСТА КАК СПЕЦИФИКАЦИЯ

- ✔ StringCalculator
 - ✔ returns zero on empty input
 - ✔ throws exception on negative numbers
 - ✔ lists all negatives in exception message when fail
- ▼ ✔ when coma separated numbers
 - ✔ sums from ""
 - ✔ sums from "42"
 - ✔ sums from "42,13"
 - ✔ sums from "1,2,3,4,5"
- ▼ ✔ when newline separated numbers
 - ✔ sums from "123"
- ▼ ✔ when delimiter from first special line
 - ✔ sums from "//;1;2;3"
 - ✔ sums from "//|4|5|6|1"

ИМЯ ТЕСТА КАК СПЕЦИФИКАЦИЯ



SAMPLES / SPECIFICATIONS /
STACK_SPECIFICATION.CS

ПРАВИЛЬНАЯ СТРУКТУРА ТЕСТА

Arrange

Act

Assert



SAMPLES / AAA /
ZIP_SHOULD.CS

АНТИПАТТЕРНЫ



SAMPLES /
ANTIPATTERNS

<https://habr.com/post/43761/>

ПОЧЕМУ НЕ ВСЕ ПИШУТ ТЕСТЫ?

ТЕСТ НАПИСАТЬ – КАК ЧАЙ ПОПИТЬ

ПИШЕМ ТЕСТЫ ЛЕГКО

БОРЬБА С ДУБЛИРОВАНИЕМ

СБОРКА И РАЗБОРКА ОКРУЖЕНИЯ

[OneTimeSetUp | OneTimeTearDown]

[SetUp | TearDown]

SETUP & TEARDOWN

```
[TestFixture]
public class Mailbox_Should
{
    private Mailbox mailbox;

    [SetUp]
    public void SetUp() { mailbox = new Mailbox(); }
    ...

    [TearDown]
    public void TearDown() { mailbox.Dispose(); }
}
```


OBJECT MOTHER & TEST DATA BUILDER



SAMPLES / TESTDATABUILDER /
TESTDATABUILDER.CS

PARAMETRIZED TESTS

ОНИ ЖЕ DATA DRIVEN



SAMPLES / PARAMETRIZED /
DOUBLE_SHOULD.CS

ДОПОЛНИТЕЛЬНЫЕ ТРЮКИ

ASSERT

```
Assert.AreEqual(expected, actual)
```

VS

```
Assert.AreEqual(actual, expected)
```

```
Assert.That(2+2, Is.EqualTo(4))
```

```
Assert.That(x, IResolveConstraint) // 0_o ?!?
```

SHOULD

```
(2+2).Should().Be(4)  
flag.Should().BeTrue()
```

```
new[] {1,2,3}  
    .Should().AllBeEquivalentTo(new [] {3,2,1});
```

```
new[] {1,2,3}  
    .Should().AllBeEquivalentTo(  
        new [] {1,2,3},  
        options => options.WithStrictOrdering());
```

FluentAssertions доступна через NuGet

ОЖИДАНИЕ ИСКЛЮЧЕНИЯ

```
Action action = () => { var z = x / y; };  
action.ShouldThrow<DivideByZeroException>();
```

ОГРАНИЧЕНИЕ ПО ВРЕМЕНИ

```
[Test, Timeout(1000)]  
public void ShouldDoInTimeout()  
{  
    ...  
}
```

ВЫБОР ТЕСТОВ ДЛЯ ПРОГОНА

```
[Test, Category("Smoke")]  
public void Test()  
{  
    ...  
}
```


LIVE TEMPLATES

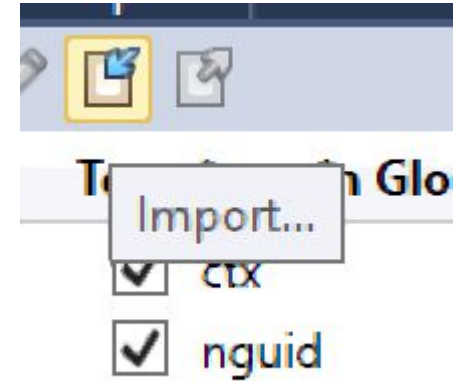
*Resharper → Tools → Templates Explorer
Import tests-templates.DotSettings*

tf — TestFixture

tt — Test

su — SetUp

td — TearDown



HOTKEYS

Ctrl+T+R

Ctrl+U+R

CHALLENGE

CHALLENGE

WordsStatistics



WordStatisticsXXX



Ctrl+F5



CHALLENGE /
WORDSSTATISTICS_TESTS.cs

CHALLENGE

РАЗБОР

Тесты по спецификации – это просто

Как взаимодействуют разные пункты спецификации?

CHALLENGE

РАЗБОР

Важно не только результат, но и время выполнения

CHALLENGE

РАЗБОР

Тесты не заменяют Code Review

Code Review не заменяет тесты

CHALLENGE

РАЗБОР

Большие цифры в лидерборде – плохо

Overspecification