Lab: Unit Testing

Problems for exercises and homework for the ["C# OOP" course @ SoftUni](#).

You can check your solutions here: https://judge.softuni.org/Contests/1844/Unit-Testing-Lab

1. Part I: Unit Testing Basics

Test Axe

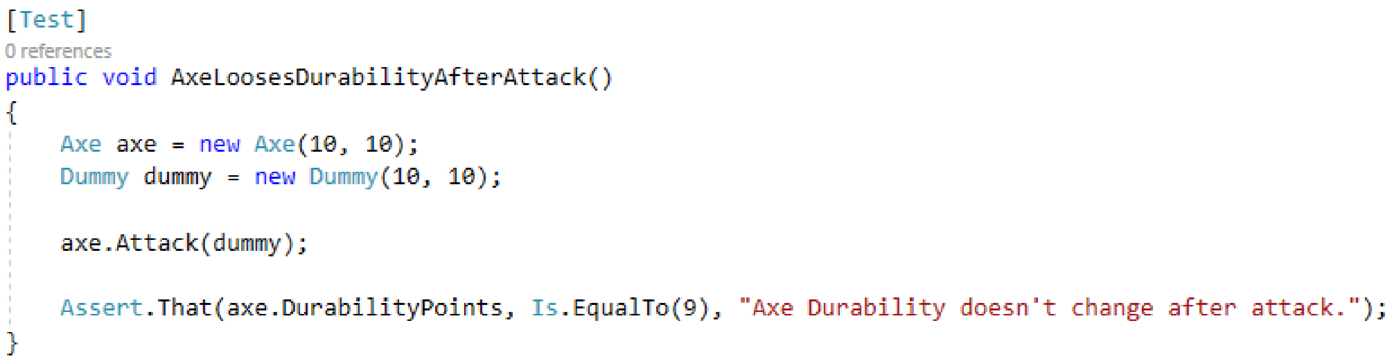
Load provided solution in Visual Studio. Add new project **Tests**

An **AxeTests** is given

Create the following tests:

- Test if weapon loses durability after each attack
- Test attacking with a broken weapon

Solution



```
[Test]
0 references
public void AxeLoosesDurabilityAfterAttack()
{
    Axe axe = new Axe(10, 10);
    Dummy dummy = new Dummy(10, 10);

    axe.Attack(dummy);

    Assert.That(axe.DurabilityPoints, Is.EqualTo(9), "Axe Durability doesn't change after attack.");
}
```

2. Test Dummy

A **DummyTests** is given

Create the following tests:

- Dummy loses health if attacked
- Dead Dummy throws an exception if attacked
- Dead Dummy can give XP
- Alive Dummy can't give XP

Hints

Follow the logic of the previous problem

3. Refactor Tests

Refactor the tests for **Axe** and **Dummy** classes

Make sure that:

- **Names** of test methods are **descriptive**

- You use **appropriate assertions** (assert equals vs assert true)
- You use **assertion messages**
- There are **no magic numbers**
- There is **no code duplication** (Don't Repeat Yourself)

Hints

Extract constants and private fields for **Axe** class

Create a method that executes **before each test**

Make use of constants and private fields, as well as add assertion messages

Follow the same logic for other test methods and **TestDummy** class