# Programming Advanced for QA – Regular Exam

# 13 April 2024

Submit your zip file here: <https://judge.softuni.org/Contests/Compete/Index/4553#1>

# 2. Unit Test: Sentence Analyzer

Test a given method which takes in **a string** and gets **the number of letters, digits, and special characters (including punctuation)**. The **whitespaces will be excluded**.

### Examples

|  |  |
| --- | --- |
| **Argument** | **Returned dictionary** |
| A | letters -> 1 time |
| Unit testing is the 1st step!!! | letters -> 22 times  digits -> 1 time  special characters -> 3 times |

The method is found in the SentenceAnalyzer.cs file:

Картина, която съдържа текст, екранна снимка, Шрифт, номер

Описанието е генерирано автоматично

You are given a **test** **file** SentenceAnalyzerTests.cs containing **4 empty tests**. Implement all tests:

Картина, която съдържа текст, екранна снимка, Шрифт, номер

Описанието е генерирано автоматично

**Note! You may need to test if result collection is equal to expected.**

When you are ready make sure your **tests run:**

**Картина, която съдържа текст, екранна снимка, Шрифт, линия

Описанието е генерирано автоматично**

**IMPORTANT:** **DO NOT REMOVE OR CHANGE ANY NAMESPACES AND USINGS.**

using NUnit.Framework;

using System.Collections.Generic;

namespace TestApp.Tests

{

[TestFixture]

public class SentenceAnalyzerTests

{

[Test]

public void Test\_Analyze\_EmptyString\_ReturnsEmptyDictionary()

{

// Arrange

string input = "";

Dictionary<string, int> expectedOutput = new Dictionary<string, int>();

// Act

Dictionary<string, int> result = SentenceAnalyzer.Analyze(input);

// Assert

CollectionAssert.AreEqual(expectedOutput, result);

}

[Test]

public void Test\_Analyze\_SingleLetter\_ReturnsDictionaryWithSingleLetterEntry()

{

// Arrange

string input = "A";

Dictionary<string, int> expectedOutput = new Dictionary<string, int>

{

{"letters", 1 }

};

// Act

Dictionary<string, int> result = SentenceAnalyzer.Analyze(input);

// Assert

CollectionAssert.AreEqual(expectedOutput, result);

}

[Test]

public void Test\_Analyze\_SingleDigit\_ReturnsDictionaryWithSingleDigitEntry()

{

// Arrange

string input = "5";

Dictionary<string, int> expectedOutput = new Dictionary<string, int>

{

{"digits", 1 }

};

// Act

Dictionary<string, int> result = SentenceAnalyzer.Analyze(input);

// Assert

CollectionAssert.AreEqual(expectedOutput, result);

}

[Test]

public void Test\_Analyze\_WholeSentence\_ReturnsDictionaryWithAllSymbolTypesCount()

{

// Arrange

string input = "Unit testing is the 1st step!!!";

Dictionary<string, int> expectedOutput = new Dictionary<string, int>

{

{"letters", 22 },

{"digits", 1 },

{"special characters", 3 }

};

// Act

Dictionary<string, int> result = SentenceAnalyzer.Analyze(input);

// Assert

CollectionAssert.AreEqual(expectedOutput, result);

}

}

}