# C# Masterclass Exam - 01 December 2019

# Overview

You will be tasked to solve three different problems. You have skeletons for all three tasks. For the first one you will need to create a custom testing framewok using reflection. The second problem is synchronous http server which you have to refactor it, so that it works asynchronously. The last task is bonus, where you will need to impelment the deprecated class PrivateObject.

# Setup

* **Do not modify the interfaces, classes or their namespaces**
* **Do not** violate your **interface** **implementations** by adding **more public methods** or **properties** in the concrete class than the interface has defined

# Task 1: CTF Framework (50 points)

You are given folders, which contain **classes** and **methods**. You have to **implement** their **functionality**.

There are **4** folders: **Asserts**, **Attributes**, **Exceptions**, and **TestRunner**.

### Asserts

The **Asserts** folder should contain **one class** "CTFAssert", and three methods which are used for assertion:

public static void AreEqual(object a, object b)

public static void AreNotEqual(object a, object b)

public static void Throws<T>(Func<bool> condition)

* You will need to add constraint, where T is inherit from the **Exception** type

**NOTE**: If the objects are not equal, throw an exception of type **TestException**

### Attributes

The **Attributes** folder should contains **two classes**:

public class CTFTestClassAttribute

* This attribute will target classes

public class CTFTestMethodAttribute

* This attribute will target methods

### Exceptions

The **Exception** folder should contain one class which is "TestException"

public class TestException

### TestRunner

The **TestRunner** folder should contain one **class** "Runner" with one **method** "Run" which accepts **assemblyPath** and returns **string** as a result:

* public string Run(string assemblyPath)

"Run" method should **fetch** all classes which have "CTFTestClassAttribute" **attribute**, with all of his methods that have "CTFTestMethodAttribute" and **execute** them. It should have **three cases**. Store all of the cases and return them as a string.

If the method has passed successfully:

$"Class: {testClassName} Method: {testMethodName} - passed!"

If the method has failed:

$"Class: {testClassName} Method: {testMethodName} - failed!"

If unexpected error occurs:

$"Unexpected error occurred in {testMethodName}!"

You can test it with the following code:

string assemblyPath = string.Empty;

Runner runner = new Runner();

string result = runner.Run(assemblyPath);

Console.WriteLine(result);

For assembly path you can set the Calculator.Tests library.

**NOTE**: Do not use **Assembly.Load**. Use **Assembly.LoadFrom** instead.

# Task 2: Async Want to be (50 points)

You are given **working** http server and your task is to **refactor** the http server to work **asynchronously**.

**NOTE: This problem is evaluated by hand!**

# Task 3: Deprecated Class (25 points) Bonus

You are given a skeleton which contains three classes: "StartUp", "PrivateObject" and "Summator".

The purpose of the "PrivateObject" is to test private methods.

You have to create "PrivateObject" class which contains the following things:

A constructor:

public PrivateObject(object obj)

A method:

public object Invoke(string methodName, params object[] parameters)

The "Invoke" method must find the **private** method with the given name, which can be **static** or **non-static** and invoke it with the given **parameters** and return the result as an **object**.

You can test it with the following code:

var summator = new Summator();

var privateObject = new PrivateObject(summator);

var result = privateObject.Invoke("GetSum", 12, 13);