COS20007: Object Oriented Programming  
Pass Task 2.4: Case Study — Iteration 1: Identifiable Object

Show Wai Yan/105293041

# IdentifiableObject.cs

using System.Collections.Generic;

namespace SwinAdventure

{

public class IndentifiableObject

{

// Fields

private List<string> \_identifiers = new List<string>();

// Constructor

public IndentifiableObject(string[] idents)

{

foreach (string s in idents)

{

AddIdentifier(s);

}

}

//Methods

public bool AreYou(string id)

{

return \_identifiers.Contains(id.ToLower());

}

public void AddIdentifier(string id)

{

\_identifiers.Add(id.ToLower());

}

public void PrivilegeEscalation(string pin)

{

if (pin == "3041")

{

\_identifiers[0] = "105293041";

}

}

// Propertry

public string FirstId

{

get

{

if (\_identifiers.Count == 0) return "";

return \_identifiers.First();

}

}

}

}

# TestIdentifiableObject.cs

using NUnit.Framework;

using NUnit.Framework.Legacy;

using SwinAdventure;

namespace SwinAdventure.Tests

{

[TestFixture]

public class IndentifiableObjectTest

{

private IndentifiableObject testObject = new IndentifiableObject(new string[] { "105293041", "Show Wai Yan", "Miss Siti" });

private IndentifiableObject testObjectEmpty = new IndentifiableObject(new string[] { });

[Test]

public void TestAreYou()

{

string testString = "105293041";

ClassicAssert.True(testObject.AreYou(testString));

}

[Test]

public void TestNotAreYou()

{

string testString = "1O5293O41";

ClassicAssert.False(testObject.AreYou(testString));

}

[Test]

public void TestCaseSensitive()

{

string testString = "inTI";

testObject.AddIdentifier(testString);

ClassicAssert.True(testObject.AreYou(testString));

}

[Test]

public void TestFirstID()

{

string firstIndentifier = "105293041";

ClassicAssert.AreEqual(firstIndentifier, testObject.FirstId);

}

[Test]

public void TestFirstIdWithNoIDs()

{

string noIndentifier = "";

ClassicAssert.AreEqual(noIndentifier, testObjectEmpty.FirstId);

}

[Test]

public void TestAddID()

{

string[] testStrings = { "JavaScript", "Linux", "C#", "Rust", "DotNet" };

foreach (string s in testStrings)

{

testObject.AddIdentifier(s);

}

foreach (string s in testStrings)

{

ClassicAssert.True(testObject.AreYou(s));

}

}

[Test]

public void TestPrivilegeEscalation()

{

string tutorialID = "105293041";

string lastFourDigitStudentID = "3041";

testObject.PrivilegeEscalation(lastFourDigitStudentID);

ClassicAssert.AreEqual(tutorialID, testObject.FirstId);

}

}

}

# Screenshot of the Test Explorer showing your unit test running