SWINBURNE UNIVERSITY OF TECHNOLOGY

COS20007 OBJECT ORIENTED PROGRAMMING

2.4P - Case Study Iteration 1 - Identifiable Object

PDF generated at 16:05 on Monday 10th April, 2023

```csharp
using System;

namespace SwinAdventure
{
    public class IdentifiableObject
    {
        private List<string> _identifiers = new List<string>();


        public IdentifiableObject(string[] idents)
        {
            for (int i = 0; i < idents.Length; i++)
            {
                _identifiers.Add(idents[i].ToLower());
            }
        }


        public bool AreYou(string id)
        {
            foreach (string i in _identifiers)
            {
                if (i == id.ToLower())
                {
                    return true;
                }
            }

            return false;

        }

        public string FirstId
        {
            get
            {
                if (_identifiers.Count == 0)
                {
                    return "";
                }
                else
                {
                    return _identifiers.First();
                }
            }
        }

        public void Addidentifier(string id)
        {
            _identifiers.Add(id.ToLower());
        }

    }
```

```
54    }
55
56
57
```

```csharp
1   using System;
2   using System.Collections;
3   using System.Collections.Generic;
4   using System.Linq;
5   using NUnit.Framework;
6   using SwinAdventure;
7
8   namespace SwinAdventureTest
9   {
10      [TestFixture]
11
12      public class TestIdentifiableObject
13      {
14          private IdentifiableObject ObjectTest;
15          private string[] _array;
16          private string[] empty_array = new string[] { };
17          private IdentifiableObject emptyobject;
18
19
20          [SetUp]
21          public void SetUp()
22          {
23              _array = new string[] { "fred", "george", "bob" };
24              ObjectTest = new IdentifiableObject(_array);
25              emptyobject = new IdentifiableObject(empty_array);
26
27          }
28
29          [Test]
30          public void TestAreYou()
31          {
32              Assert.IsTrue(ObjectTest.AreYou("fred"));
33
34          }
35
36          [Test]
37          public void TestNotAreYou()
38          {
39              Assert.IsFalse(ObjectTest.AreYou("john"));
40
41          }
42
43          [Test]
44          public void TestCaseSensitive()
45          {
46              Assert.IsTrue(ObjectTest.AreYou("FreD"));
47
48          }
49
50          [Test]
51          public void TestFirstID()
52          {
53              Assert.AreEqual("fred", ObjectTest.FirstId);
```

```
54
55          }
56
57          [Test]
58          public void TestFirstIDWithNoIDs()
59          {
60              Assert.AreEqual("",emptyobject.FirstId);
61
62
63          }
64
65          [Test]
66          public void TestAddId()
67          {
68
69              ObjectTest.Addidentifier("billy");
70
71              Assert.IsTrue(ObjectTest.AreYou("billy"));
72              Assert.IsTrue(ObjectTest.AreYou("fred"));
73              Assert.IsTrue(ObjectTest.AreYou("george"));
74              Assert.IsTrue(ObjectTest.AreYou("bob"));
75
76          }
77      }
78  }
```