

SWINBURNE UNIVERSITY OF TECHNOLOGY

COS20007 OBJECT ORIENTED PROGRAMMING

2.4P - Case Study Iteration 1 - Identifiable Object

PDF generated at 15:05 on Wednesday 15th March, 2023

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace SwinAdventure
8  {
9      public class IdentifiableObject
10     {
11         List<string> _identifiers = new List<string>();
12
13         public IdentifiableObject(string[] idents)
14         {
15             foreach (string id in idents)
16             {
17                 AddIdentifier(id);
18             }
19         }
20
21         public string FirstId
22         {
23             get
24             {
25                 if (_identifiers.Count == 0)
26                 {
27                     return "";
28                 }
29                 else
30                 {
31                     return _identifiers.First();
32                 }
33             }
34         }
35
36         public bool AreYou(string id)
37         {
38             return _identifiers.Contains(id.ToLower());
39         }
40
41         public void AddIdentifier(string id)
42         {
43             _identifiers.Add(id.ToLower());
44         }
45     }
46 }
47 }
```

```
1  using NUnit.Framework;
2  using NUnit.Framework.Internal;
3  using SwinAdventure;
4
5  namespace IdentifiableObjectTest
6  {
7      [TestFixture]
8      public class IdentifiableObjectTest
9      {
10         IdentifiableObject id;
11         IdentifiableObject emptyId;
12
13         [SetUp]
14         public void SetUp()
15         {
16             id = new IdentifiableObject(new string[] { "bob", "fred" });
17             emptyId = new IdentifiableObject(new string[] { });
18         }
19
20         [Test]
21         public void TestAreYou()
22         {
23             Assert.That(id.AreYou("bob"), Is.True);
24         }
25
26         [Test]
27         public void TestNotAreYou()
28         {
29             Assert.That(id.AreYou("wilma"), Is.False);
30         }
31         [Test]
32         public void TestCaseSensitive()
33         {
34             Assert.That(id.AreYou("bOB"), Is.True);
35         }
36         [Test]
37         public void TestFirstId()
38         {
39             Assert.That(id.FirstId, Is.SameAs("bob"));
40         }
41         [Test]
42         public void TestFirstIdWithNoIds()
43         {
44             Assert.That(emptyId.FirstId, Is.SameAs(""));
45         }
46         [Test]
47         public void TestAddId()
48         {
49             id.AddIdentifier("billy");
50             Assert.That(id.AreYou("billy"), Is.True);
51         }
52     }
53 }
```

