

SWINBURNE UNIVERSITY OF TECHNOLOGY

COS20007 OBJECT ORIENTED PROGRAMMING

2.4P - Case Study Iteration 1 - Identifiable Object

PDF generated at 15:49 on Sunday 12th 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 IdentifiableObject
8  {
9      public class IdentifiableObject
10     {
11         private List<string> _identifiers = new List<string>();
12
13         public IdentifiableObject(string[] idents)
14         {
15             foreach (string s in idents)
16             {
17                 _identifiers.Add(s.ToLower());
18             }
19         }
20
21         public bool AreYou(string id)
22         {
23             return _identifiers.Contains(id.ToLower());
24         }
25
26         public string FirstID
27         {
28             get
29             {
30                 return _identifiers.First();
31             }
32         }
33
34
35         public void AddIdentifier(string id)
36         {
37             _identifiers.Add(id.ToLower());
38         }
39     }
40 }
41
42
```

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6  using NUnit.Framework;
7
8  namespace IdentifiableObject
9  {
10     [TestFixture]
11     public class IdentifiableObjectTest
12     {
13
14         [Test]
15         public void TestAreYou()
16         {
17             string[] testStringArray = new string[] { "Test", "CheckOrder" };
18             IdentifiableObject testObject = new IdentifiableObject(testStringArray);
19             Assert.IsTrue(testObject.AreYou("CheckOrder"));
20         }
21
22         [Test]
23         public void TestNotAreYou()
24         {
25             string[] testStringArray = new string[] { "Test", "CheckOrder" };
26             IdentifiableObject testObject = new IdentifiableObject(testStringArray);
27             Assert.IsFalse(testObject.AreYou("bob"));
28         }
29
30         [Test]
31         public void TestCaseSensitive()
32         {
33             string[] testStringArray = new string[] { "Test", "CheckOrder" };
34             IdentifiableObject testObject = new IdentifiableObject(testStringArray);
35             Assert.IsTrue(testObject.AreYou("test"));
36         }
37
38         [Test]
39         public void TestFirstID()
40         {
41             string[] testStringArray = new string[] { "Test", "CheckOrder" };
42             IdentifiableObject testObject = new IdentifiableObject(testStringArray);
43             StringAssert.AreEqualIgnoringCase("Test", testObject.FirstID);
44         }
45
46         [Test]
47         public void TestAddID()
48         {
49             string[] testStringArray = new string[] { "Fred", "Bob" };
50             IdentifiableObject testObject = new IdentifiableObject(testStringArray);
51             testObject.AddIdentifier("wilma");
52             Assert.IsTrue(testObject.AreYou("Fred"));
53             Assert.IsTrue(testObject.AreYou("bob"));
```

```
54         Assert.IsTrue(testObject.AreYou("wilma"));
55     }
56 }
57 }
```

