

Selected files

2 printable files

Week_2/2.4/IdentifiableObject.cs

Week_2/TestIdentifiableObject/TestIdentifiableObject.cs

Week_2/2.4/IdentifiableObject.cs

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Threading.Tasks;
5
6  namespace SwinAdventure
7  {
8
9      public class IdentifiableObject
10     {
11         private 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 bool AreYou(string id)
22         {
23             return _identifiers.Contains(id.ToLower());
24         }
25
26         public string FirstId
27         {
28             get
29             {
30                 if (_identifiers.Count > 0)
31                 {
32                     return _identifiers[0];
33                 }
34
35                 return "";
36             }
37         }
38
39         public void AddIdentifier(string id)
40         {
41             _identifiers.Add(id.ToLower());
42         }
```

```

43
44     public void PrivilegeEscalation(string pin)
45     {
46         if (pin != "4794")
47             return;
48
49         if (_identifiers.Count == 0)
50         {
51             AddIdentifier("12");
52         }
53         else
54         {
55             _identifiers[0] = "12";
56         }
57     }
58 }
59 }

```

Week_2/TestIdentifiableObject/TestIdentifiableObject.cs

```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Threading.Tasks;
5  using NUnit.Framework;
6  using SwinAdventure;
7
8  namespace NUnitTests
9  {
10     [TestFixture]
11     public class TestIdentifiableObject
12     {
13         private IdentifiableObject _idObj;
14         private IdentifiableObject _emptyIdObj;
15         private IdentifiableObject _idObjWithNoStuId;
16
17         [SetUp]
18         public void SetUp()
19         {
20             _idObj = new IdentifiableObject(new string[] { "104844794", "Minh An",
21 "Nguyen" });
22             _emptyIdObj = new IdentifiableObject(new string[] { });
23             _idObjWithNoStuId = new IdentifiableObject(new string[] { "1234", "Minh
24 An", "Nguyen" });
25         }
26
27         [TestCase]
28         public void TestAreYou()
29         {
30             Assert.IsTrue(_idObj.AreYou("104844794"));
31             Assert.IsTrue(_idObj.AreYou("Minh An"));
32         }
33     }
34 }

```

```
31
32     [TestCase]
33     public void TestNotAreYou()
34     {
35         Assert.IsFalse(_idObj.AreYou("104844795"));
36     }
37
38     [TestCase]
39     public void TestCaseSensitive()
40     {
41         Assert.IsTrue(_idObj.AreYou("miNh aN"));
42         Assert.IsTrue(_idObj.AreYou("NgUyEn"));
43     }
44
45     [TestCase]
46     public void TestFirstId()
47     {
48         Assert.AreEqual("104844794", _idObj.FirstId);
49     }
50
51     [TestCase]
52     public void TestFirstIdWithNoId()
53     {
54         Assert.AreEqual("", _emptyIdObj.FirstId);
55     }
56
57     [TestCase]
58     public void TestAddId()
59     {
60         _idObj.AddIdentifier("Andy");
61         Assert.IsTrue(_idObj.AreYou("Andy"));
62     }
63
64     [TestCase]
65     public void TestPrivilegeEscalation()
66     {
67         _idObjWithNoStuId.PrivilegeEscalation("4794");
68         Assert.AreEqual("12", _idObjWithNoStuId.FirstId);
69     }
70 }
71 }
```

Screenshot of unit tests passing:

