

```
1 using _4._2P;
2 using System;
3 using System.Collections.Generic;
4 using System.Linq;
5 using System.Threading.Tasks;
6 using NUnit.Framework;
7 using System.Numerics;
8
9
10 namespace NUnitTests
11 {
12     [TestFixture]
13     public class TestIdentifiableObject
14     {
15         private IdentifiableObject _objectIdentifier;
16         private IdentifiableObject _emptyObjectIdentifier;
17         private IdentifiableObject _objectIdentifierNoStuId;
18
19         [SetUp]
20         public void Setup()
21         {
22             _objectIdentifier = new IdentifiableObject(new string[] { "104772183", "Khoi Nguyen", "Pham" });
23             _emptyObjectIdentifier = new IdentifiableObject(new string[] { });
24             _objectIdentifierNoStuId = new IdentifiableObject(new string[] { "1111", "Khoi Nguyen", "Pham" });
25         }
26
27         [TestCase]
28         public void TestAreYou()
29         {
30             Assert.IsTrue(_objectIdentifier.AreYou("104772183"));
31             Assert.IsTrue(_objectIdentifier.AreYou("Khoi Nguyen"));
32         }
33
34         [TestCase]
35         public void TestNotAreYou()
36         {
37             Assert.IsFalse(_objectIdentifier.AreYou("nonexistent identifier"));
38         }
39
40
41
42         [TestCase]
43         public void TestFirstId()
44         {
45             Assert.AreEqual("104772183", _objectIdentifier.FirstId);
46         }
47
48         [TestCase]
49         public void TestFirstIdWithNoId()
```

```
50     {
51         Assert.AreEqual("", _emptyObjectIdentifier.FirstId);
52     }
53
54     [TestCase]
55     public void TestAddId()
56     {
57         _objectIdentifier.AddIdentifier("Nevan");
58         Assert.IsTrue(_objectIdentifier.AreYou("Nevan"));
59     }
60
61     [TestCase]
62     public void TestCaseSensitivity()
63     {
64         Assert.IsTrue(_objectIdentifier.AreYou("khOI NGUYEN"));
65         Assert.IsTrue(_objectIdentifier.AreYou("PhAM"));
66     }
67
68     [TestCase]
69     public void TestPrivilegeEscalation()
70     {
71
72
73         string correctPin = "2183";
74         string expectedFirstId = "7";
75
76
77         _objectIdentifierNoStuId.PrivilegeEscalation(correctPin);
78
79         // Assert
80         Assert.AreEqual(expectedFirstId,
81             _objectIdentifierNoStuId.FirstId);
82     }
83
84
85
86
87 }
```