

```
1
2 namespace IdentifiableObject
3 {
4     public class Tests
5     {
6         private Item _shield;
7         private Item _shovel;
8         private Inventory _my_invent;
9
10        [SetUp]
11        public void Setup()
12        {
13            _shield = new Item(new string[] { "shield" }, "a shield",
14                               "Shield level 1");
15            _shovel = new Item(new string[] { "shovel" }, "a shovel",
16                               "Shovel level 2");
17            _my_invent = new Inventory();
18        }
19
20        [Test]
21        public void TestFindItem()
22        {
23            _my_invent.Put(_shield);
24            Assert.IsTrue(_my_invent.HasItem(_shield.FirstId));
25        }
26
27        [Test]
28        public void TestNoItemFind()
29        {
30            Assert.IsFalse(_my_invent.HasItem(_shield.FirstId));
31        }
32
33        [Test]
34        public void TestFetchItem()
35        {
36            _my_invent.Put(_shovel);
37            Item fetch_item = _my_invent.Fetch(_shovel.FirstId);
38
39            Assert.AreEqual(fetch_item, _shovel);
40            Assert.IsTrue(_my_invent.HasItem(_shovel.FirstId));
41        }
42
43        [Test]
44        public void TestTakeItem()
45        {
46            _my_invent.Put(_shovel);
47            _my_invent.Take(_shovel.FirstId);
48
49            Assert.IsFalse(_my_invent.HasItem(_shovel.FirstId));
```

```
48     }
49
50     [Test]
51     public void TestItemList()
52     {
53         _my_invent.Put(_shield);
54         _my_invent.Put(_shovel);
55
56         Assert.AreEqual(_my_invent.ItemList, "a shield (shield)\n" + "a ↗
57             shovel (shovel)\n");
58     }
59 }
```