

```
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 namespace InventoryTest
10 {
11     [TestFixture]
12     public class Tests
13     {
14         private Inventory _inventory;
15         private Item _sword;
16         private Item _shield;
17
18         [SetUp]
19         public void Setup()
20         {
21             _inventory = new Inventory();
22             _sword = new Item(new string[] { "sword", "blade" },      ↗
23                             "Excalibur", "A strong sword");
24             _shield = new Item(new string[] { "shield", "safeguard" }, ↗
25                             "Aegis", "A strong shield");
26         }
27
28         [Test]
29         public void TestFindItem()
30         {
31             _inventory.Put(_sword);
32             Assert.IsTrue(_inventory.HasItem("sword"));
33         }
34
35         [Test]
36         public void TestNoItemFind()
37         {
38             Assert.IsFalse(_inventory.HasItem("spear"));
39         }
40
41         [Test]
42         public void TestFetchItem()
43         {
44             _inventory.Put(_sword);
45             Item fetchedItem = _inventory.Fetch("sword");
46             Assert.AreEqual(_sword, fetchedItem);
47             Assert.IsTrue(_inventory.HasItem("sword"));
48         }
49
50         [Test]
51         public void TestTakeItem()
52         {
53             _inventory.Put(_shield);
```

```
52         Item takenItem = _inventory.Take("shield");
53         Assert.AreEqual(takenItem, _shield);
54         Assert.IsFalse(_inventory.HasItem("shield"), "Inventory
    should not contain the shield because it has been taken");
55
56     }
57
58     public void TestItemList()
59     {
60         _inventory.Put(_sword);
61         _inventory.Put(_shield);
62         string expectedList = "\tExcalibur\n\tAegis";
63         Assert.AreEqual(expectedList, _inventory.ItemList);
64     }
65 }
66 }
```