```
 1  using System;
 2  using System.Collections.Generic;
 3  using System.Linq;
 4  using System.Text;
 5  using System.Threading.Tasks;
 6  using SwinAdventure;
 7
 8  namespace SwinAdventureTest
 9  {
10      public class LookCommandtest //Change from internal to public
11      {
12          Item gem;
13          Bag bag;
14          Player player;
15          LookCommand look;
16          [SetUp]
17          public void SetUp()
18          {
19              look = new LookCommand();
20
21              player = new Player("Truong Ngoc Gia Hieu", "A brave
                  Swinburne Warrior");
22              bag = new Bag(new string[] { "bag" }, "Hieu's Bag", "A small
                  Bag");
23
24              gem = new Item(new string[] { "gem" }, "a gem", "A bright red
                  gem");
25
26              player.Inventory.Put(bag);
27          }
28          [Test]
29          public void TestLookAtGem()
30          {
31              player.Inventory.Put(gem);
32
33              string output = look.Execute(player, new string[] { "look",
                  "at", "gem" });
34              string expected = "A bright red gem";
35              Assert.AreEqual(expected, output);
36          }
37
38          [Test]
39          public void TestLookAtMe()
40          {
41              string output = look.Execute(player, new string[] { "look",
                  "at", "inventory" });
42              string expected = "Truong Ngoc Gia Hieu, A brave Swinburne
                  Warrior\nList of Items that you have:\nHieu's Bag: bag\n";
43              Assert.AreEqual(expected, output);
```

```
44          }
45
46          [Test]
47          public void TestLookAtGemInMe()
48          {
49              player.Inventory.Put(gem);
50
51              string output = look.Execute(player, new string[] { "look",
                  "at", "gem", "in", "inventory" });
52              string expected = "A bright red gem";
53              Assert.AreEqual(expected, output);
54          }
55
56          [Test]
57          public void TestLookAtUnk()
58          {
59              // Test looking at an unknown item
60              string output = look.Execute(player, new string[] { "look",
                  "at", "gem" });
61              string expected = "I cannot find the gem in the Truong Ngoc
                  Gia Hieu";
62              Assert.AreEqual(expected, output);
63          }
64
65          [Test]
66          public void TestLookatGemInNoBag()
67          {
68              Player player1 = new Player("NPC", "Robot Warrior");
69
70              string output = look.Execute(player1, new string[] { "look",
                  "at", "gem", "in", "bag" });
71              string expected = "I cannot find the gem";
72              Assert.AreEqual(expected, output);
73          }
74
75          [Test]
76          public void TestLookAtGemInBag()
77          {
78              // Put the gem in the bag
79              bag.Inventory.Put(gem);
80
81              // Test looking at the gem in the bag
82              string output = look.Execute(player, new string[] { "look",
                  "at", "gem", "in", "bag" });
83              string expected = "I cannot find the gem";
84              Assert.AreEqual(expected, output);
85          }
86
87          [Test]
```

```
 88          public void TestInvalidLookCommands()
 89          {
 90              string output1 = look.Execute(player, new string[] { "look",
                   "around" });
 91              string output2 = look.Execute(player, new string[]
                   { "hello" });
 92              string expected = "I don't know how to look like that";
 93
 94              Assert.AreEqual(expected, output1);
 95              Assert.AreEqual(expected, output2);
 96          }
 97
 98          [Test]
 99          public void TestLookAtNoGemInBag()
100          {
101              // Create an empty bag
102              Bag emptyBag = new Bag(new string[] { "emptybag" }, "Empty
                   Bag", "An empty bag");
103
104              // Put the empty bag in the player's inventory
105              player.Inventory.Put(emptyBag);
106
107              // Test looking at the gem in the empty bag
108              string output = look.Execute(player, new string[] { "look",
                   "at", "gem", "in", "emptybag" });
109              string expected = "I cannot find the gem";
110              Assert.AreEqual(expected, output);
111          }
112      }
113  }
114
```