```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace Iteration3
8  {
9      public class Bag : Item
10     {
11         private Inventory _inventory;
12         public Bag(string[] ids, string name,string desc):base(ids, name,
             desc)
13         {
14             _inventory = new Inventory();
15         }
16         public GameObject Locate(string id)
17         {
18             if (AreYou(id))
19                 {
20                     return this;
21                 }
22             else if (_inventory.HasItem(id))
23                 {
24                     return _inventory.Fetch(id);
25                 }
26             else
27                 {
28                     return null;
29                 }
30         }
31         public override String FullDescription
32         {
33             get
34             {
35                 return $"{this.name}, containing:\n" + _inventory.ItemList;
36
37             }
38         }
39
40         public Inventory Inventory
41         {
42             get { return _inventory; }
43         }
44     }
45 }
46
```

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace Iteration3
8  {
9      public class GameObject : IdentifiableObject
10     {
11         private string _description;
12         private string _name;
13         public GameObject(string[] ids, string name, string desc) : base
             (ids)
14         {
15             _description = desc;
16             _name = name;
17         }
18         public string name
19         {
20             get { return _name; }
21         }
22         public string ShortDescription
23         {
24             get { return $"{_name} ({FirstID})"; }
25         }
26         public virtual string FullDescription
27         {
28             get { return _description; }
29         }
30     }
31
32 }
33
```

```
 1  using System;
 2  using System.Collections.Generic;
 3  using System.Linq;
 4  using System.Text;
 5  using System.Threading.Tasks;
 6
 7  namespace Iteration3
 8  {
 9      public class Inventory
10      {
11          private List<Item> _items;
12          public Inventory()
13          {
14              _items = new List<Item>();
15          }
16          public bool HasItem(string id)
17          {
18              foreach (var item in _items)
19              {
20                  if (item.AreYou(id))
21                  {
22                      return true;
23                  }
24              }
25              return false;
26          }
27          public void Put(Item i)
28          {
29              _items.Add(i);
30          }
31          public Item Take(string id)
32          {
33              foreach (var item in _items)
34              {
35                  if (item.AreYou(id))
36                  {
37                      _items.Remove(item);
38                      return item;
39                  }
40
41              }
42              return null;
43          }
44          public Item Fetch(string id)
45          {
46              foreach (var item in _items)
47              {
48                  if (item.AreYou(id))
49                  {
```

```
50                      return item;
51
52                  }
53              }
54              return null;
55          }
56          public string ItemList
57          {
58              get
59              {
60                  string listItem = "";
61                  foreach (Item i in _items)
62                  {
63                      listItem = listItem + i.ShortDescription + "\n";
64                  }
65                  return listItem;
66              }
67          }
68      }
69  }
70
71
72
73
```

```
 1  using System;
 2  using System.Collections.Generic;
 3  using System.Linq;
 4  using System.Text;
 5  using System.Threading.Tasks;
 6
 7  namespace Iteration3
 8  {
 9      public class IdentifiableObject
10      {
11          private List<string> _identifiers;
12
13          public IdentifiableObject(string[] idents)
14          {
15              _identifiers = new List<string>();
16              foreach (string ident in idents)
17              {
18                  _identifiers.Add(ident.ToLower());
19              }
20
21
22          }
23
24          public bool AreYou(string name)
25          {
26              foreach (string idents in _identifiers)
27              {
28                  if (idents.ToLower() == name.ToLower())
29                  {
30                      return true;
31                  }
32              }
33
34              return false;
35          }
36
37          public string FirstID
38          {
39              get
40              {
41                  if (_identifiers.Count == 0)
42                  {
43                      return "";
44                  }
45                  else
46                  {
47                      return _identifiers.First();
48
49                  }
```

```
50              }
51          }
52
53          public void AddIdentifier(string id)
54          {
55              _identifiers.Add(id.ToLower());
56          }
57      }
58  }
59
```

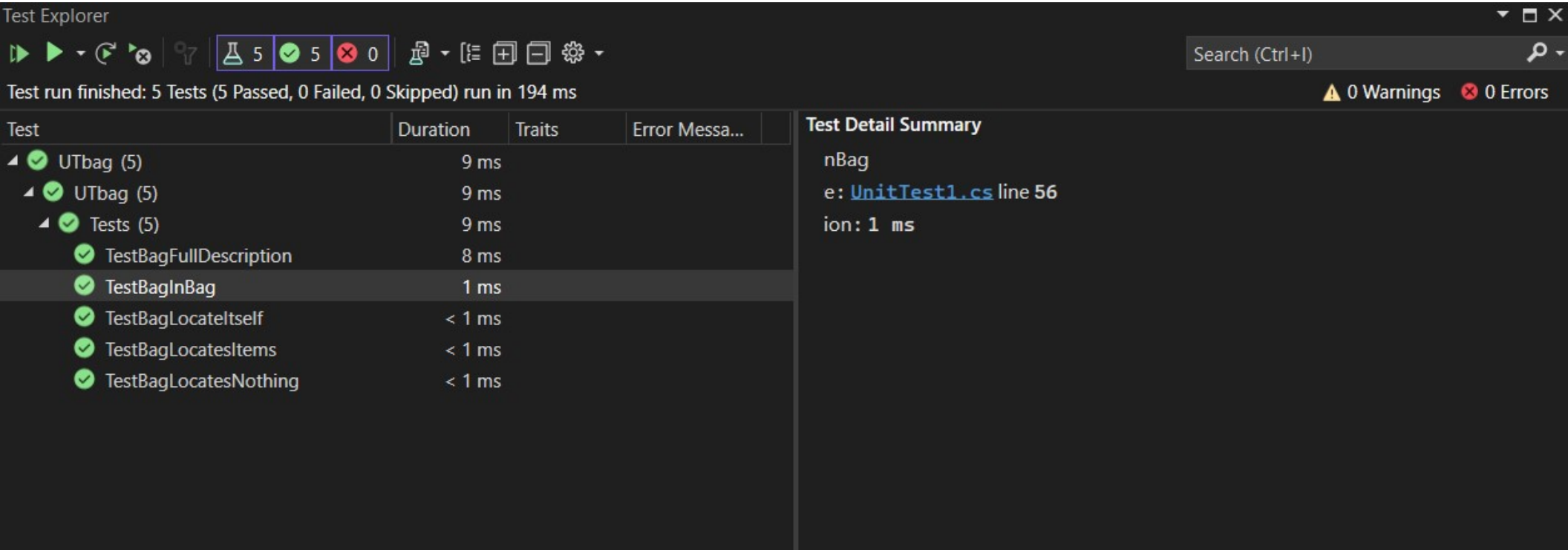
```
 1  using System;
 2  using System.Collections.Generic;
 3  using System.Linq;
 4  using System.Text;
 5  using System.Threading.Tasks;
 6
 7  namespace Iteration3
 8  {
 9      public class Item : GameObject
10      {
11          public Item(string[] idents, string name, string desc) : base
              (idents, name, desc)
12          {
13
14          }
15      }
16  }
17
18
```

```
1  namespace Iteration3
2  {
3      internal class Program
4      {
5          static void Main(string[] args)
6          {
7              Console.WriteLine("Hello, World!");
8          }
9      }
10 }
11
```

```
1  using Iteration3;
2
3  namespace UTbag
4  {
5      public class Tests
6      {
7          Bag b1;
8          Bag b2;
9          Item shovel = new Item(new string[] { "shovel" }, "a shovel", "This
               is a shovel");
10         Item sword = new Item(new string[] { "sword" }, "a sword", "This is
               a sword");
11         Item book = new Item(new string[] { "book" }, "a book", "This is a
             small book");
12         Item pc = new Item(new string[] { "pc" }, "a pc", "This is a small
             computer");
13
14         [SetUp]
15         public void Setup()
16         {
17             b1 = new Bag(new string[] { "bag" }, "a bag", "This is a bag");
18             b2 = new Bag(new string[] { "bag1" }, "a bag1", "This is a
                 bag1");
19             b1.Inventory.Put(shovel);
20             b1.Inventory.Put(sword);
21             b2.Inventory.Put(book);
22             b2.Inventory.Put(pc);
23         }
24
25         [Test]
26         public void TestBagLocatesItems()
27         {
28             Assert.IsTrue(b1.Inventory.HasItem("sword"));
29             Assert.IsTrue(b1.Inventory.HasItem("shovel"));
30
31
32             Assert.IsTrue(b1.Locate(sword.FirstID) == sword);
33             Assert.IsTrue(b1.Locate(shovel.FirstID) == shovel);
34
35         }
36         [Test]
37         public void TestBagLocateItself()
38         {
39             Assert.IsTrue(b1.Locate(b1.FirstID) == b1);
40             Assert.IsTrue(b1.Locate("bag") == b1);
41         }
42
43         [Test]
44         public void TestBagLocatesNothing()
```

```
45          {
46              Assert.IsTrue(b2.Locate(shovel.FirstID) == null);
47          }
48
49          [Test]
50          public void TestBagFullDescription()
51          {
52              Assert.AreEqual(b1.FullDescription, "a bag, containing:\na
                  shovel (shovel)\na sword (sword)\n");
53          }
54
55          [Test]
56          public void TestBagInBag()
57          {
58              b1.Inventory.Put(b2);
59              Assert.IsTrue(b1.Locate(b2.FirstID) == b2);
60              Assert.IsTrue(b1.Locate(sword.FirstID) == sword);
61              Assert.IsFalse(b2.Locate(shovel.FirstID) == shovel);
62
63              Assert.AreEqual(b1.FullDescription, "a bag, containing:\na
                  shovel (shovel)\na sword (sword)\na bag1 (bag1)\n");
64              Assert.AreEqual(b2.FullDescription, "a bag1, containing:\na
                  book (book)\na pc (pc)\n");
65
66
67          }
68
69      }
70  }
```

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace Iteration3
8  {
9      public class Player : GameObject
10     {
11         private Inventory _inventory;
12         public Player(string name, string desc) : base(new string[] { "me",
               "inventory" }, name, desc)
13         {
14             _inventory = new Inventory();
15         }
16         public GameObject Locate(string id)
17         {
18             if (AreYou(id))
19             {
20                 return this;
21             }
22             else
23             {
24                 return _inventory.Fetch(id);
25             }
26         }
27         public override string FullDescription
28         {
29             get
30             {
31                 return $"You are {name} {base.FullDescription}. You are
                       carrying \n" + _inventory.ItemList;
32             }
33         }
34         public Inventory Inventory
35         {
36             get => _inventory;
37         }
38
39     }
40 }
41
```


Test Explorer
5
5
0
Search (Ctrl+I)
Test run finished: 5 Tests (5 Passed, 0 Failed, 0 Skipped) run in 194 ms
0 Warnings
0 Errors
Test
Duration
Traits
Error Messa...
UTbag (5)
9 ms
UTbag (5)
9 ms
Tests (5)
9 ms
TestBagFullDescription
8 ms
TestBagInBag
1 ms
TestBagLocateItself
< 1 ms
TestBagLocatesItems
< 1 ms
TestBagLocatesNothing
< 1 ms
Test Detail Summary
nBag
e: UnitTest1.cs line 56
ion: 1 ms