Swinburne University of Technology

COS20007 Object Oriented Programming

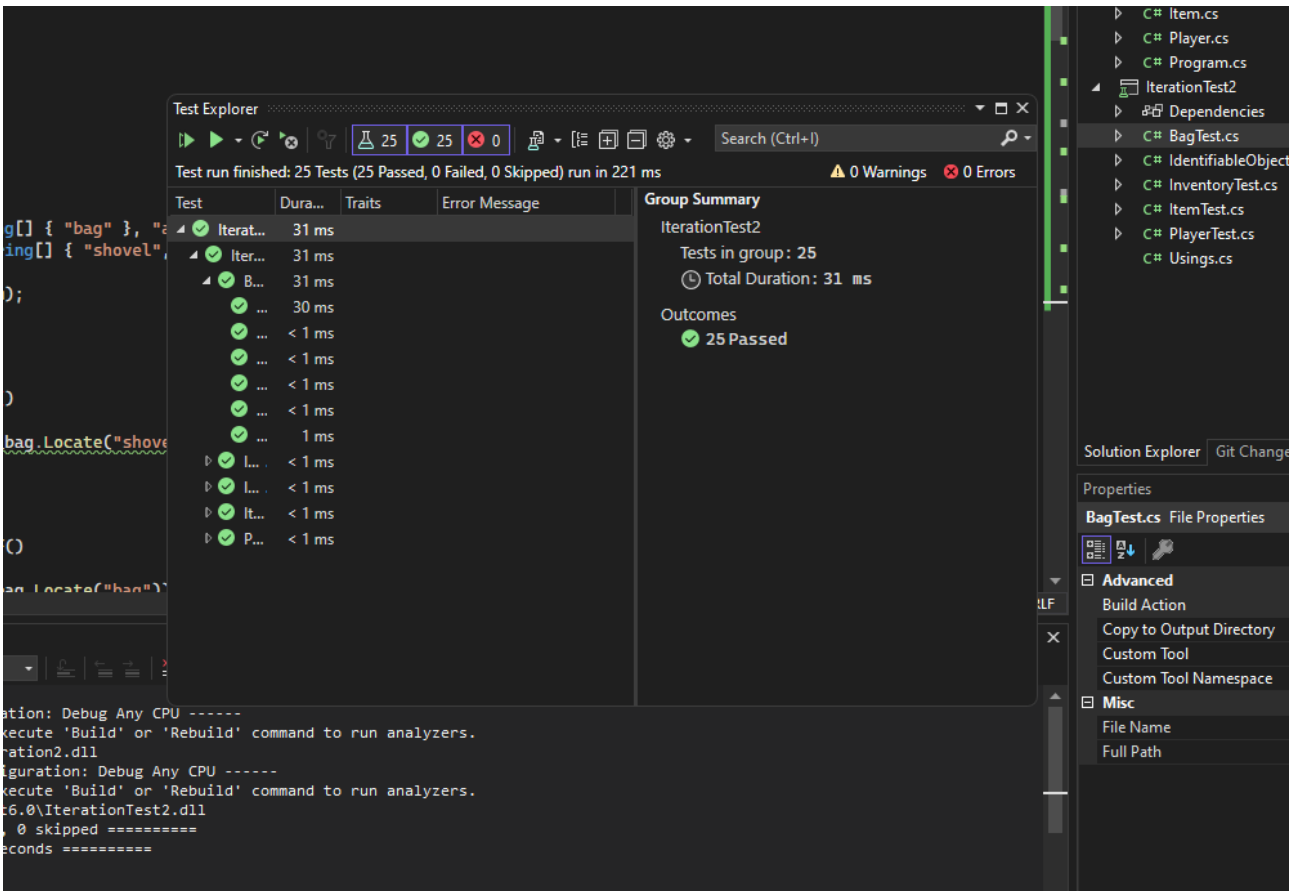
Case Study - Iteration 3 - Bags

PDF generated at 01:47 on Monday 4th September, 2023

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Iteration2
{
    public class Bag : Item
    {
        private Inventory _inventory;

        public Bag(string[] ids, string name, string desc) : base(ids, name, desc)
        {
            _inventory = new Inventory();
        }

        public GameObject Locate(string id)
        {
            if (AreYou(id))
            {
                return this;
            }
            else
            {
                return _inventory.Fetch(id);
            }
        }

        public override string FullDescription
        {
            get
            {
                return $"In the {this.Name} \nYou can see:\n{_inventory.ItemList}";
            }
        }

        public Inventory Inventory
        {
            get
            {
                return _inventory;
            }
        }
    }
}
```

```
using Iteration2;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace IterationTest2
{
    public class BagTest
    {
        private Bag _bag;
        private Item _item;


        [SetUp()]
        public void SetUp()
        {
            _bag = new Bag(new string[] { "bag" }, "a bag", "This is a bag");
            _item = new Item(new string[] { "shovel", "spade" }, "a shovel", "This is okay...");

            _bag.Inventory.Put(_item);
        }

        [Test()]
        public void BagLocatesItems()
        {
            Assert.AreEqual(_item, _bag.Locate("shovel"));
        }

        [Test()]
        public void BagLocatesItself()
        {
            Assert.AreEqual(_bag, _bag.Locate("bag"));
        }

        [Test()]
        public void BagLocatesNothing()
        {
            Assert.AreEqual(null, _bag.Locate("grenade"));
        }

        [Test()]
        public void BaginBag()
        {
            Bag _sack = new Bag(new string[] { "sack" }, "a sack", "This is a sack");

            _bag.Inventory.Put(_sack);
            //test bag2 in bag1
            Assert.AreEqual(_sack, _bag.Locate("sack"));

            //test bag1 locate other items in bag1
```

```
53              //Assert.AreEqual(_item, _bag.Locate("spade"));
54
55              //bag1 cannot locate items in bag2
56              Item _item = new Item(new string[] { "katana", "knife" }, "a katana", "a
    ↪  knife");
57              Assert.AreEqual(null, _bag.Locate("katana"));
58          }
59
60          [Test()]
61          public void BagInBag2()
62          {
63              //test bag1 locate other items in bag1
64              Assert.AreEqual(_item, _bag.Locate("spade"));
65          }
66
67          [Test()]
68          public void FullDescription()
69          {
70              Assert.AreEqual("In the a bag \nYou can see:\n" +
    ↪  _bag.Inventory.ItemList, _bag.FullDescription);
71          }
72      }
73  }
```


Test Explorer
25
25
0
Search (Ctrl+I)
Test run finished: 25 Tests (25 Passed, 0 Failed, 0 Skipped) run in 221 ms
0 Warnings
0 Errors
Test
Dura...
Traits
Error Message
Group Summary
IterationTest2
Tests in group: 25
Total Duration: 31 ms
Outcomes
25 Passed
Item.cs
Player.cs
Program.cs
IterationTest2
Dependencies
BagTest.cs
IdentifiableObject
InventoryTest.cs
ItemTest.cs
PlayerTest.cs
Usings.cs
Solution Explorer
Properties
BagTest.cs File Properties
Advanced
Build Action
Copy to Output Directory
Custom Tool
Custom Tool Namespace
Misc
File Name
Full Path