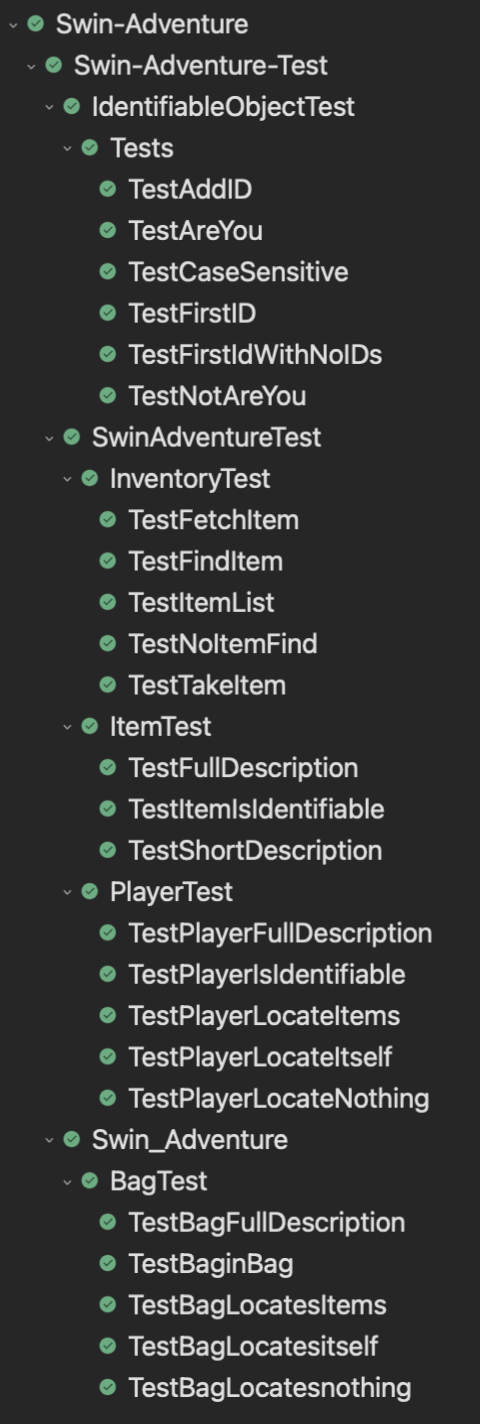
COS 20007

Task 5.2

## Duc Thuan Tran

*104330455*



# Folder Swin-Adventure

**8 printable files**

## (file list disabled)

### Swin-Adventure/Bag.cs

1. **using** System;

### namespace Swin\_Adventure

1. {

### public class Bag : Item

1. {
2. **private** Inventory \_inventory; 7

8 **public Bag**(string[] ids, string name, string description) : **base**(ids, name, description)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 9 |  |  | { |  | | | |
| 10 |  |  | \_inventory = **new** Inventory(); |
| 11 |  |  | } |
| 12 |  |  |  |
| 13 |  |  | **public** GameObject **Locate**(string id) |
| 14 |  |  | { |
| 15 |  |  | **if** (**this**.AreYou(id)) |
| 16 |  |  | { |
| 17 |  |  | **return this**; |
| 18 |  |  | } |
| 19 |  |  | **return** \_inventory.Fetch(id); |
| 20 |  |  | } |
| 21 |  |  |  |
| 22 |  |  | **public override** string FullDescription |
| 23 |  |  | { |
| 24 |  |  | **get** { **return** $"In the {Name} you can | see:\n" | + | \_inventory.ItemList; | } |
| 25 |  |  | } |  | | | |
| 26 |  |  |  |
| 27 |  |  | **public** Inventory Inventory |
| 28 |  |  | { |
| 29 |  |  | **get** { **return** \_inventory; } |
| 30 |  |  | } |
| 31 |  | } |  |
| 32 | } |  |  |
| 33 |  |  |  |
| 34 |  |  |  |

### Swin-Adventure/GameObject.cs

1. **using** System;
2. **using** System.Xml.Linq;

3

### namespace Swin\_Adventure

1. {

### public class GameObject : IdentifiableObject

1. {
2. **private** string \_description;
3. **private** string \_name; 10

|  |  |  |
| --- | --- | --- |
| 11 | **public GameObject**(string[] ids, string | name, string description) : |
|  | **base**(ids) |  |
| 12 | { |  |
| 13 | \_description = description; |  |
| 14 | \_name = name; |  |
| 15 | } |  |
| 16 |  |  |
| 17 | **public** string Name |  |
| 18 | { |  |
| 19 | **get** { **return** \_name.ToLower(); } |  |
| 20 | } |  |
| 21 |  |  |
| 22 | **public** string ShortDescription |  |
| 23 | { |  |
| 24 | **get** { **return** $"a {\_name.ToLower()} | ({FirstId.ToLower()})"; } |
| 25 | } |  |
| 26 |  |  |
| 27 | **public virtual** string FullDescription |  |
| 28 | { |  |
| 29 | **get** { **return** \_description; } |  |
| 30 | } |  |
| 31 | } |  |
| 32 | } |  |
| 33 |  |  |
| 34 |  |  |

### Swin-Adventure/IdentifiableObjectClass.cs

1. **using** System;

### namespace Swin\_Adventure

1. {

### public class IdentifiableObject

1. {
2. **private** List<string> \_identifiers; 7
3. **public IdentifiableObject**(string[] idents)
4. {
5. \_identifiers = **new** List<string>(idents);
6. \_identifiers.AddRange(idents);

12 }

13

1. **public** bool **AreYou**(string id)
2. {
3. **return** \_identifiers.Contains(id.ToLower());

17 }

18

1. **public** string FirstId
2. {

### get

1. {
2. **if** (\_identifiers.Count == 0)
3. {
4. **return** "";

26 }

27 **return** \_identifiers[0];

28 }

29 }

30

31 **public void AddIdentifier**(string id)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 32 |  |  | { |  |
| 33 |  |  |  | \_identifiers.Add(id.ToLower()); |
| 34 |  |  | } |  |
| 35 |  | } |  |  |
| 36 | } |  | | |
| 37 |  |
| 38 |  |

### Swin-Adventure/Inventory.cs

1. **using** System;

### namespace Swin\_Adventure

1. {

### public class Inventory

1. {
2. **private** List<Item> \_items;

### public Inventory()

1. {
2. \_items = **new** List<Item>();

10 }

11

1. **public** bool **HasItem**(string id)
2. {
3. **foreach** (Item itm **in** \_items)
4. {
5. **if** (itm.AreYou(id))
6. {
7. **return** true;

19 }

20 }

21 **return** false;

22 }

23

1. **public void Put**(Item itm)
2. {
3. \_items.Add(itm);

27 }

28

1. **public** Item **Take**(string id)
2. {
3. Item itm = Fetch(id); 32

33 **if** (itm != null)

34 {

35 \_items.Remove(itm);

36 }

37

38 **return** itm;

39 }

40

41 **public** Item **Fetch**(string id)

42 {

43 **foreach** (Item itm **in** \_items)

44 {

45 **if** (itm.AreYou(id))

46 {

47 **return** itm;

48 }

49 }

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 50 |  |  | **return** null; |  |  | |
| 51 |  |  | } |  |
| 52 |  |  |  |  |
| 53 |  |  | **public** string ItemList |  |
| 54 |  |  | { |  |
| 55 |  |  | **get** |  |
| 56 |  |  | { |  |
| 57 |  |  | string list = ""; |  |
| 58 |  |  | **foreach** (Item item **in** | \_items) |
| 59 |  |  | { |  |
| 60 |  |  | list += "\t" + "a | " + item.Name + " (" + item.FirstId | + | ")\n"; |
| 61 |  |  | } |  |  |  |
| 62 |  |  | **return** list; |  |  |  |
| 63 |  |  | } |  |  |  |
| 64 |  |  | } |  |  |  |
| 65 |  | } |  |  |  |  |
| 66 | } |  |  |  |  |  |
| 67 |  |  |  |  |  |  |
| 68 |  |  |  |  |  |  |

### Swin-Adventure/Item.cs

1 **using** System;

2

### namespace Swin\_Adventure

1. {

### public class Item : GameObject

1. {
2. **public Item**(string[] idents, string name, string description) :

**base**(idents, name, description)

1. {

9

10 }

11 }

12 }

13

14

### Swin-Adventure/Player.cs

1. **using** System;

### namespace Swin\_Adventure

1. {

### public class Player : GameObject

1. {
2. **private** Inventory \_inventory; 7
3. **public Player**(string name, string description) : **base**(**new** string[] { "me", "inventory" }, name, description)
4. {
5. \_inventory = **new** Inventory();

11 }

12

13 **public** GameObject **Locate**(string id)

14 {

1. **if**(AreYou(id))
2. {

### return this;

18

19

20 }

21

22

23

24

25

26

}

**return** \_inventory.Fetch(id);

**public override** string FullDescription

{

### get

{

**return** "You are " + Name + ", " + **base**.FullDescription + ".\n"

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 27 |  |  | + "You | are carrying:\n" | + | Inventory.ItemList; |
| 28 |  | } |  |  |  |  |
| 29 | } |  |  |  |  |  |
| 30 |  |  |  |  |  |  |
| 31 |  | **public** Inventory | | Inventory | | |
| 32 |  | { | |  | | |
| 33 |  | **get**{ **return** | | \_inventory; } | | |
| 34 |  | } | |  | | |
| 35 | } |  | |  | | |
| 36 | } |  | |  | | |
| 37 |  |  | |  | | |
| 38 |  |  | |  | | |

### Swin-Adventure/Program.cs

|  |  |  |  |
| --- | --- | --- | --- |
| 1  2  3  4 | **namespace Swin\_Adventure**;  **class Program**  { | | |
| 5 |  | **static void Main**(string[] args) |  |
| 6 |  | { |  |
| 7 |  | Console.WriteLine("Duc Thuan Tran | - 104330455"); |
| 8 |  | } |  |
| 9 | } |  |  |
| 10 |  |  |  |
| 11 |  |  |  |

**Swin-Adventure/Swin-Adventure.csproj**

1 <**Project** Sdk="Microsoft.NET.Sdk">

2

### <PropertyGroup>

1. <**OutputType**>Exe</**OutputType**>
2. <**TargetFramework**>net7.0</**TargetFramework**>
3. <**RootNamespace**>Swin\_Adventure</**RootNamespace**>
4. <**ImplicitUsings**>enable</**ImplicitUsings**>
5. <**Nullable**>enable</**Nullable**>

### </PropertyGroup>

10

1. </**Project**>

12

# Folder IdentifiableObjectTest

1. **printable files**

## (file list disabled)

### IdentifiableObjectTest/BagTest.cs

* 1. **using** System;

### namespace Swin\_Adventure

* 1. {
  2. [TestFixture]

### public class BagTest

* 1. {
  2. **private** Bag \_bagTest1;
  3. **private** Bag \_bagTest2;
  4. **private** Item \_weaponTest;
  5. **private** Item \_armorTest; 11

1. [SetUp]

### public void SetUp()

14 {

1. \_bagTest1 = **new** Bag(**new** string[] { "bag1" }, "backpack", "It's spacious");
2. \_bagTest2 = **new** Bag(**new** string[] { "bag2" }, "suitcase", "It's compact"

);

1. \_weaponTest = **new** Item(**new** string[] { "weapon" }, "sword", "this is an Excalibur");
2. \_armorTest = **new** Item(**new** string[] { "armor" }, "shield", "this is a shield");

19

1. \_bagTest1.Inventory.Put(\_bagTest2);
2. \_bagTest1.Inventory.Put(\_weaponTest);
3. \_bagTest2.Inventory.Put(\_armorTest);

23 }

24

1. [Test]

### public void TestBagLocatesItems()

1. {
2. Assert.AreSame(\_weaponTest, \_bagTest1.Locate("weapon"));

29 }

30

1. [Test]

### public void TestBagLocatesitself()

1. {
2. Assert.AreSame(\_bagTest1, \_bagTest1.Locate("bag1"));

35 }

36

1. [Test]

### public void TestBagLocatesnothing()

1. {
2. Assert.IsNull(\_bagTest1.Locate("bag3"));

41 }

42

1. [Test]

### public void TestBagFullDescription()

1. {
2. Assert.AreEqual("In the backpack you can see:\n\ta suitcase (bag2)\n\ta sword (weapon)\n", \_bagTest1.FullDescription);

|  |  |  |  |
| --- | --- | --- | --- |
| 47 |  |  | } |
| 48 |  |  |  |
| 49 |  |  | [Test] |
| 50 |  |  | **public void TestBaginBag**() |
| 51 |  |  | { |
| 52 |  |  | Assert.AreSame(\_bagTest2, \_bagTest1.Locate("bag2")); |
| 53 |  |  | Assert.AreSame(\_weaponTest, \_bagTest1.Locate("weapon")); |
| 54 |  |  | Assert.IsNull(\_bagTest1.Locate("armor")); |
| 55 |  |  | } |
| 56 |  | } |  |
| 57 | } |  |  |
| 58 |  |  |  |
| 59 |  |  |  |

### IdentifiableObjectTest/IdentifiableObjectTest.cs

* 1. **using** NUnit.Framework;
  2. **using** Swin\_Adventure;

3

### namespace IdentifiableObjectTest

1. {

|  |  |  |
| --- | --- | --- |
| 6 |  |  |
| 7 | **internal class Tests** |
| 8 | { |
| 9 | **private** IdentifiableObject | \_test1; |
| 10 | **private** IdentifiableObject | \_test2; |
| 11 | **private** IdentifiableObject | \_test3; |
| 12 | **private** IdentifiableObject | \_test4; |
| 13 | **private** IdentifiableObject | \_test5; |
| 14 | **private** IdentifiableObject | \_test6; |
| 15 |  |  |
| 16 | [SetUp] |  |
| 17 | **public void Setup**() |  |
| 18 | { |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 19 |  | \_test1 | = | **new** | IdentifiableObject(**new** | string[] | { | "fred", | "bob" | }); |
| 20 |  | \_test2 | = | **new** | IdentifiableObject(**new** | string[] | { | "fred", | "bob" | }); |
| 21 |  | \_test3 | = | **new** | IdentifiableObject(**new** | string[] | { | "fred", | "bob" | }); |
| 22 |  | \_test4 | = | **new** | IdentifiableObject(**new** | string[] | { | "fred", | "bob" | }); |
| 23 |  | \_test5 | = | **new** | IdentifiableObject(**new** | string[] | { | }); |  |  |
| 24 |  | \_test6 | = | **new** | IdentifiableObject(**new** | string[] | { | "fred", | "bob" | }); |
| 25 | } |  |  |  |  |  |  |  |  |  |
| 26 |  |  |  |  |  |  |  |  |  |  |

|  |  |
| --- | --- |
| 27 | [Test] |
| 28 | **public void TestAreYou**() |
| 29 | { |
| 30 | Assert.IsTrue(\_test1.AreYou("fred")); |
| 31 | Assert.IsTrue(\_test1.AreYou("bob")); |
| 32 | } |
| 33 |  |
| 34 | [Test] |
| 35 | **public void TestNotAreYou**() |
| 36 | { |
| 37 | Assert.IsFalse(\_test2.AreYou("wilma")); |
| 38 | Assert.IsFalse(\_test2.AreYou("boby")); |
| 39 | } |
| 40 |  |
| 41 | [Test] |

|  |  |  |  |
| --- | --- | --- | --- |
| 42 |  |  | **public void TestCaseSensitive**() |
| 43 |  |  | { |
| 44 |  |  | Assert.IsTrue(\_test3.AreYou("FRED")); |
| 45 |  |  | Assert.IsTrue(\_test3.AreYou("bOB")); |
| 46 |  |  | } |
| 47 |  |  |  |
| 48 |  |  | [Test] |
| 49 |  |  | **public void TestFirstID**() |
| 50 |  |  | { |
| 51 |  |  | Assert.AreEqual("fred", \_test4.FirstId); |
| 52 |  |  | } |
| 53 |  |  |  |
| 54 |  |  | [Test] |
| 55 |  |  | **public void TestFirstIdWithNoIDs**() |
| 56 |  |  | { |
| 57 |  |  | Assert.AreEqual("", \_test5.FirstId); |
| 58 |  |  | } |
| 59 |  |  |  |
| 60 |  |  | [Test] |
| 61 |  |  | **public void TestAddID**() |
| 62 |  |  | { |
| 63 |  |  | \_test6.AddIdentifier("wilma"); |
| 64 |  |  | Assert.IsTrue(\_test6.AreYou("fred")); |
| 65 |  |  | Assert.IsTrue(\_test6.AreYou("bob")); |
| 66 |  |  | Assert.IsTrue(\_test6.AreYou("wilma")); |
| 67 |  |  | } |
| 68 |  | } |  |
| 69 | } |  |  |
| 70 |  |  |  |
| 71 |  |  |  |

### IdentifiableObjectTest/InventoryTest.cs

1. **using** System;
2. **using** Swin\_Adventure;

3

### namespace SwinAdventureTest

1. {
2. [TestFixture]

### public class InventoryTest

8 {

1. **private** Inventory \_inventoryTest;
2. **private** Item \_weaponTest;
3. **private** Item \_armorTest; 12
4. [SetUp]

### public void SetUp()

1. {
2. \_inventoryTest = **new** Inventory();
3. \_weaponTest = **new** Item(**new** string[] { "weapon" }, "sword", "this is an Excalibur");
4. \_armorTest = **new** Item(**new** string[] { "armor" }, "shield", "this is a shield");

19

1. \_inventoryTest.Put(\_weaponTest);
2. \_inventoryTest.Put(\_armorTest);

22 }

23

24 [Test]

(weapon)\na shield (armor)\n");

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 25 | **public void TestFindItem**() | | |  |
| 26 | { | | |
| 27 | Assert.IsTrue(\_inventoryTest.HasItem("weapon")); | | |
| 28 | Assert.IsTrue(\_inventoryTest.HasItem("armor")); | | |
| 29 | } | | |
| 30 |  | | |
| 31 | [Test] | | |
| 32 | **public void TestNoItemFind**() | | |
| 33 | { | | |
| 34 | Assert.IsFalse(\_inventoryTest.HasItem("axe")); | | |
| 35 | Assert.IsFalse(\_inventoryTest.HasItem("helmet")); | | |
| 36 | } | | |
| 37 |  | | |
| 38 | [Test] | | |
| 39 | **public void TestFetchItem**() | | |
| 40 | { | | |
| 41 | Assert.IsTrue(\_weaponTest == \_inventoryTest.Fetch("weapon")); | | |
| 42 | Assert.IsTrue(\_inventoryTest.HasItem("weapon")); | | |
| 43 |  | | |
| 44 | Assert.IsTrue(\_armorTest == \_inventoryTest.Fetch("armor")); | | |
| 45 | Assert.IsTrue(\_inventoryTest.HasItem("armor")); | | |
| 46 | } | | |
| 47 |  | | |
| 48 | [Test] | | |
| 49 | **public void TestTakeItem**() | | |
| 50 | { | | |
| 51 | Assert.IsTrue(\_weaponTest == \_inventoryTest.Take("weapon")); | | |
| 52 | Assert.IsFalse(\_inventoryTest.HasItem("weapon")); | | |
| 53 |  | | |
| 54 | Assert.IsTrue(\_armorTest == \_inventoryTest.Take("armor")); | | |
| 55 | Assert.IsFalse(\_inventoryTest.HasItem("armor")); | | |
| 56 | } | | |
| 57 |  | | |
| 58 | [Test] | | |
| 59 | **public void TestItemList**() | | |
| 60 | { | | |
| 61 | Assert.IsTrue(\_inventoryTest.ItemList.Replace("\t", "") == "a | | | sword |
| 62 |  |  | } | |
| 63 |  | } |  | |
| 64 | } |  |  | |
| 65 |  |  |  | |
| 66 |  |  |  | |

### IdentifiableObjectTest/ItemTest.cs

1. **using** System;
2. **using** Swin\_Adventure;

3

### namespace SwinAdventureTest

1. {
2. [TestFixture]

### public class ItemTest

8 {

9 **private** Item \_itemTest; 10

11

1. [SetUp]

### public void Setup()

|  |  |
| --- | --- |
| 14 | { |
| 15 | \_itemTest = **new** Item(**new** string[] { "weapon" }, "sword", "This is an Excalibur"); |
| 16 |  |
| 17 | } |
| 18 |  |
| 19 | [Test] |
| 20 | **public void TestItemIsIdentifiable**() |
| 21 | { |
| 22 | Assert.IsTrue(\_itemTest.AreYou("weapon")); |
| 23 | } |
| 24 |  |
| 25 | [Test] |
| 26 | **public void TestShortDescription**() |
| 27 | { |
| 28 | Assert.IsTrue(\_itemTest.ShortDescription == "a sword (weapon)"); |
| 29 | } |
| 30 |  |
| 31 | [Test] |
| 32 | **public void TestFullDescription**() |
| 33 | { |

**IdentifiableObjectTest/PlayerTest.cs**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 34 | Assert.IsTrue(\_itemTest.FullDescription | == | "This | is | an | Excalibur"); |
| 35 } |  |  |  |  |  |  |
| 36 } |  |  |  |  |  |  |
| 37 } |  |  |  |  |  |  |
| 38 |  |  |  |  |  |  |
| 39 |  |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| 1  2  3  4  5 | **using** System;  **using** Swin\_Adventure;  **namespace SwinAdventureTest**  { | |
| 6 | [TestFixture] |  |
| 7 | **public class PlayerTest** |  |
| 8 | { |  |
| 9 | **private** Player \_playerTest; |  |
| 10 | **private** Item \_weaponTest; |  |
| 11 | **private** Item \_armorTest; |  |
| 12 |  |  |
| 13 | [SetUp] |  |
| 14 | **public void Setup**() |  |
| 15 | { |  |
| 16 | \_playerTest = **new** Player("thuan", | "dan choi"); |

1. \_weaponTest = **new** Item(**new** string[] { "weapon" }, "sword", "this is an Excalibur");
2. \_armorTest = **new** Item(**new** string[] { "armor" }, "shield", "this is a shield");

19

20

21

1. \_playerTest.Inventory.Put(\_weaponTest);
2. \_playerTest.Inventory.Put(\_armorTest);

24 }

25

1. [Test]

### public void TestPlayerIsIdentifiable()

|  |  |  |
| --- | --- | --- |
| 28 | { |  |
| 29 |  | Assert.IsTrue(\_playerTest.AreYou("me")); |
| 30 |  | Assert.IsTrue(\_playerTest.AreYou("inventory")); |
| 31 | } |  |
| 32 |  |  |

1. [Test]

### public void TestPlayerLocateItems()

1. {
2. Assert.IsTrue(\_playerTest.Locate("weapon") == \_weaponTest);
3. Assert.IsTrue(\_playerTest.Locate("armor") == \_armorTest); 38
4. Assert.IsTrue(\_playerTest.Inventory.HasItem("weapon"));
5. Assert.IsTrue(\_playerTest.Inventory.HasItem("armor"));

41 }

42

1. [Test]

### public void TestPlayerLocateItself()

1. {
2. Assert.IsTrue(\_playerTest == \_playerTest.Locate("me"));
3. Assert.IsTrue(\_playerTest == \_playerTest.Locate("inventory"));

48 }

49

1. [Test]

### public void TestPlayerLocateNothing()

1. {
2. Assert.IsTrue(\_playerTest.Locate("helmet") == null);

54 }

55

1. [Test]

### public void TestPlayerFullDescription()

1. {
2. Assert.IsTrue(\_playerTest.FullDescription == "You are thuan, dan choi.\nYou are carrying:\n\ta sword (weapon)\n\ta shield (armor)\n");

60 }

61 }

62 }

63

64