
C:\assignments\OOP\Tasks\Ta
Enter Player Name:
Thenura
Enter Player Description:
Dulnath
Command:
look at pen in bag
a black pen
Command:

```
1  namespace Iteration5
2  {
3      public class Program
4      {
5          static void Main(string[] args)
6          {
7              string name, description;
8              Item item1, item2, item3;
9              Bag bag0;
10             LookCommand command = new LookCommand();
11
12             item1 = new Item(new string[] { "computer" }, "a computer", "a
                 small computer");
13             item2 = new Item(new string[] { "bottle" }, "a bottle", "a
                 white water bottle");
14             item3 = new Item(new string[] { "pen" }, "a pen", "a black
                 pen");
15
16             bag0 = new Bag(new string[] { "bag" }, "a bag", "black color
                 bag");
17
18             Console.WriteLine("Enter Player Name: ");
19             name = Console.ReadLine();
20
21             Console.WriteLine("Enter Player Description: ");
22             description = Console.ReadLine();
23
24             Player Player1 = new Player(name, description);
25             Player1.Inventory.Put(item1);
26             Player1.Inventory.Put(item2);
27             Player1.Inventory.Put(bag0);
28             bag0.Inventory.Put(item3);
29
30             while (true)
31             {
32                 Console.WriteLine("Command: ");
33                 string userInput = Console.ReadLine();
34
35                 if (userInput == "stop")
36                 {
37                     break;
38                 }
39                 else
40                 {
41                     Console.WriteLine(command.Execute(Player1,
                       userInput.Split()));
42                 }
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 Iteration5
 8  {
 9      public class Bag : Item, IhaveInventory
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
23              {
24                  return _inventory.Fetch(id);
25              }
26          }
27          public override string FullDescription
28          {
29              get
30              {
31                  return $"In the {this.name} you can see:\n" +
                      _inventory.ItemList;
32
33              }
34          }
35
36          public Inventory Inventory
37          {
38              get { return _inventory; }
39          }
40      }
41  }
42
```

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace Iteration5
8  {
9      public abstract class Command : IdentifiableObject
10     {
11         public Command(string[] ids) : base(ids)
12         {
13
14         }
15         public abstract string Execute(Player p, string[] text);
16     }
17 }
18
```

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace Iteration5
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 Iteration5
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  }
```

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace Iteration5
8  {
9      public interface IhaveInventory
10     {
11         public string name
12         {
13             get;
14         }
15         public GameObject Locate(string id);
16
17     }
18 }
19
```

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace Iteration5
8  {
9      public class Inventory
10     {
11         private List<Item> _items = new List<Item>();
12         public Inventory()
13         {
14
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 Iteration5
 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  }
```

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace Iteration5
8  {
9      public class LookCommand : Command
10     {
11         public LookCommand() : base(new string[] { "look" })
12         {
13         }
14
15         public override string Execute(Player p, string[] text)
16         {
17             if ((text.Length != 3) && (text.Length != 5))
18             {
19                 return "I don't know how to look like that";
20             }
21             else if (text[0] != "look")
22             {
23                 return "Error in look input";
24             }
25             else if (text[1] != "at")
26             {
27                 return "What do you want to look at?";
28             }
29
30             if ((text.Length == 5) && (text[3] != "in"))
31             {
32                 return "What do you want to look in?";
33             }
34
35             String itemId = text[2];
36             IhaveInventory container = p;
37
38             if (text.Length == 5)
39             {
40                 container = FetchContainer(p, text[4]);
41                 if (container == null)
42                 {
43                     return $"I cannot find the {text[4]}";
44                 }
45             }
46
47             return LookAtIn(itemId, container);
48         }
49
```

```
50          private IhaveInventory FetchContainer(Player p, string containerId)
51          {
52              return p.Locate(containerId) as IhaveInventory;
53          }
54
55          private string LookAtIn(string thingId, IhaveInventory container)
56          {
57              var item = container.Locate(thingId);
58              if (item != null)
59              {
60                  return item.FullDescription;
61              }
62              else
63              {
64                  return $"I can't find the {thingId}";
65              }
66          }
67      }
68  }
69
```

```
 1  using System;
 2  using System.Collections.Generic;
 3  using System.Linq;
 4  using System.Text;
 5  using System.Threading.Tasks;
 6
 7  namespace Iteration5
 8  {
 9      public class Player : GameObject, IhaveInventory
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 {this.name}. You are carrying:\n" +
                      _inventory.ItemList;
32              }
33          }
34          public Inventory Inventory
35          {
36              get
37              {
38                  return _inventory;
39              }
40          }
41
42      }
43  }
44
```