

```
1 using System.ComponentModel.Design;
2
3 namespace SwinAdventure
4 {
5     internal class Program
6     {
7         static void Main(string[] args)
8         {
9             Console.WriteLine("Welcome to the Adventure Game!");
10            Console.WriteLine("-----");
11            //Enter player name
12            Console.Write("Dear brave warrior, please enter your name: ");
13            string playername = Console.ReadLine();
14            //Enter player description
15            Console.WriteLine("Please enter player description:");
16            string playerdescription = Console.ReadLine();
17            Console.WriteLine("-----");
18            //Two items and add them to the player inventory
19            Player player = new Player(playername, playerdescription);
20            Item shovel = new Item(new string[] { "shovel" }, "A shovel", "A  ↗
                "An excavating shovel");
21            Item sword = new Item(new string[] { "sword" }, "A sword", "A  ↗
                sharp sword ");
22            player.Inventory.Put(shovel);
23            player.Inventory.Put(sword);
24            // Create a bag and add it to player's inventory
25            Bag backpack = new Bag(new string[] { "backpack" }, "A  ↗
                backpack", "A useful backpack");
26            player.Inventory.Put(backpack);
27            //Create another item and add it to the bag
28            Item book = new Item(new string[] { "book" }, "A book", "A  ↗
                thick book");
29            backpack.Inventory.Put(book);
30            Item gem = new Item(new string[] { "gem" }, "A gem", "A shiny  ↗
                gem");
31            backpack.Inventory.Put(gem);
32            Item shield = new Item(new string[] { "shield" }, "A shield",  ↗
                "A bronze shield");
33            backpack.Inventory.Put(shield);
34            Item gun = new Item(new string[] { "gun" }, "A gun", "A  ↗
                powerful gun");
35            backpack.Inventory.Put(gun);
36            Item map = new Item(new string[] { "map" }, "A map", "A  ↗
                detailed map");
37            backpack.Inventory.Put(map);
38
39            LookCommand lookCommand = new LookCommand();
40            Console.WriteLine($"Hello, {player.Name}!\n
                {player.FullDescription}");
```

```
41 Console.WriteLine("-----");
42 //Main
43 while (true)
44 {
45     Console.WriteLine("Please enter command: ");
46
47     string commandLine = Console.ReadLine();
48     string[] commandWords = commandLine.ToLower().Split(' ');
49     if (string.IsNullOrEmpty(commandLine))
50     {
51         Console.WriteLine("Please enter a command.");
52         continue;
53     }
54
55     else if (commandWords.Length > 0 && commandWords[0] == "look" && commandWords.Length == 3) ➤
56     {
57         string result = lookCommand.Execute(player, ➤
58             commandWords);
59         Console.WriteLine(result);
60     }
61     else if (commandWords.Length > 0 && commandWords[0] == "look" && commandWords.Length == 5) ➤
62     {
63         string result = lookCommand.Execute(player, ➤
64             commandWords);
65         Console.WriteLine(result);
66     }
67     else if (commandLine == "Exit" || commandLine == "exit")
68     {
69         Console.WriteLine("Exiting the game. Goodbye!");
70         break;
71     }
72     else if (commandWords.Length > 0 && commandWords[0] == "help") ➤
73     {
74         Console.WriteLine("Available commands:\n" +
75             "1. look at [item]\n" +
76             "2. look at [item] in [container]\n" ➤
77             +
78             "3. exit - to exit the game\n" +
79             "4. help - to see this message ➤
80             again");
81     }
82     else
83     {
84         Console.WriteLine("Unknown command. Please try ➤
85             again.");
86     }
87 }
```

```
82
83         }
84
85     }
86 }
87
88
```