COS 20007: Object Oriented Programming

Pass Task 7.1: Case Study — Iteration 5

Show Wai Yan/105293041

# Program.cs

namespace SwinAdventure

{

public class Program

{

public static void Main(string[] args)

{

// Configurations

string helpCommand =

$"Here is the List of command\n\t- look at me: Display what you are carrying in your inventory\n\t- look at <item> [?in <container>]: Get description of that item,which inside in the container\n\t- quit/exit: Halt the program\n";

// Getting Player's Name and Description

string PlayerName = "";

string PlayerDescription = "";

Console.WriteLine("Write Your Name, Traveller!");

Console.Write("NAME -> ");

PlayerName = Console.ReadLine();

Console.WriteLine("How about Your description, Traveller!");

Console.Write("Description -> ");

PlayerDescription = Console.ReadLine();

// Object Configurations

Player me = new Player(PlayerName, PlayerDescription); // Create Player

// Create two items and put these to player's inventory

Item sword = new Item(

new string[] { "sword", "bronze sword" },

"Bronze Sword",

"A shiny bronze sword"

);

Item shield = new Item(

new string[] { "shield", "wooden shield" },

"Wooden Shield",

"A tough wooden shield"

);

me.Inventory.Put(sword);

me.Inventory.Put(shield);

// Create a bag and put it to player's inventory

Bag myBag = new Bag(

new string[] { "bag", "backpack" },

"Leather Bag",

"A sturdy leather bag to carry items"

);

me.Inventory.Put(myBag);

// Create another item and add it to the bag

Item potion = new Item(

new string[] { "potion", "health potion" },

"Health Potion",

"A magical red potion that restores health"

);

myBag.Inventory.Put(potion);

// Game Loop

Console.WriteLine("Write '-h' for helper");

while (true)

{

string command = "";

Console.Write("Command -> ");

command = Console.ReadLine().ToLower();

if (command == "exit" || command == "quit")

{

Console.WriteLine("Take the rest, Traveller!");

return;

}

else if (command == "-h")

{

Console.WriteLine(helpCommand);

}

else

{

Console.WriteLine(new LookCommand().Execute(me, command.Split(' ')));

}

}

}

}

}

# Screenshot of the console output showing all valid and invalid commands

A screenshot of a computer program

AI-generated content may be incorrect.