Haunted Horror House

Algorithm

5453835

4999406

Fall 2016

COP 2210

1. We start by creating the class HauntedHorrorHouse. Here, we will put the methods that run the entire game and it contains a sub class called Room.
2. We import JOptionPane to be able to ask the user for all of their input in the program.
3. In the class HauntedHorrorHouse, we have the variables String userName, boolean playerStatus, and boolean floorCheck.
4. Our constructor has setUserName() and exploreHauntedHouse(), which are the methods that we are using in this class.
5. Our first method, using public void setUserName (). This method is where we welcome the player into the game and ask for their name. The input is set as their userName for the rest of the game. This method also sets the player as alive by usng playerStatus = true and sets the person to be on the first floor using floorCheck = true.
6. Our second method houseMap shows the location of the user as they go through the house, it has different strings in each location that will have a ‘o’ character in which the user is currently located in. If floorCheck is true then the user is in the first floor and that will be printed out to the console. If it is false, then the second floor will be used.
7. The next method is the majority of what makes up the game, is the exploreHauntedHouse(). It has the variable String decision. We create 14 new objects. Each object is a room in the house. Then, for each of these rooms, we set up narrations for each one that asks what the user wants to do next.
8. Now we start on our loop, a while loop. This while loop will keep going as long as the playerStatus is true, which means the player is alive. It will exit the loop and display a game over message when the user dies.
9. We now have two main if and else statements. If the floorCheck is true, then the player is in floor one and will have all the options of floor one. Else, the player is on floor two.
10. In each floor, there are other if statements that show towards each room or object that we created beforehand. We use .getState to see if the object/room is active to be able to go through the narration of that room and its location with the houseMap method.
11. Each decision will either look at an object and then go through the same narration as before or be killed by that option by making playerStatus = false. Other decisions such as moving around will make the current object .setState false and .setState true for the next room. Then the user will go into the next narration and location for that room.
12. Going up the stairs will make the floorCheck false and viceversa. When put to go back to the front door, it is a different object then before, it is now frontDoorReturn and this has the ability for the user to exit the house and win the game.
13. Our sub-class Room, has variables String NAME, String narration, and Boolean state. It has a constructor that uses String newName and Boolean newState.
14. The methods in this class are getName(); setNarration(), getNarrations. We used these to set up each object/room. And it also has the methods setState and getState. That would activate and deactivated rooms in the house.
15. Now we have our Launcher class that has our main located in it.
16. We first made HHH to be reserved as a Haunted House object, before the main.
17. In the main, we make a Haunted House using the default constructor and then we use this new object to run our two methods that have the game, which are .setUserName and .exploreHauntedHouse.
18. Now the game runs using all of the user inputs. Now can the user make it alive?

Schedule for Assignment 4

Discuss and share previous assignments by 11/2/16

Decide what direction to take and whose project to be picked by 11/4/16

Meet up by 11/7/16

Start and finish flow chart by 11/9/16---4999406

Start on the HauntedHorrorHouse class by 11/8/16---4999406 and 5453835

Finish constructor by 11/9/16---4999406

Finish .setUserName method by 11/10/16---4999406 and 5453835

Finish houseMap by 11/11/16---5453835

Finish sub-class Room by 11/11/16---4999406

Finish method exploreHauntedHouse by 11/13/16---5453835

Finish javadocs by 11/13/16---4999406 and 5453835

Finish Algorithm by 11/14/16---5453835

Project all finished and done by 11/14/16