Rocket Class

A new class named Rocket is created to store the number of rocket parts that the user is currently holding by declaring instance variables of rocket plan, rocket body and rocket engine.

This class contains 5 methods.3 methods each returning the number of a particular rocket part that the user is currently holding,1 method accepting the name of rocket part as input parameter and increase the number of the respective rocket part by 1 and the last method accepting the name of rocket part as input parameter as well but decrease the number of the respective rocket part by 1.

This class is created to keep track of the number of rocket parts that the player is holding and make changes to the value if there is any exchanging,dropping and picking up of rocket part in the game.

This class inherits Item class which allows the rocket object to be stored in the Item class type list named inventory in Actor.java in game engine.

Key Class

Another new class named Key is created to store the number of key that the user is holding by declaring an instance variable which records the number of key that the user currently has.

This class contains 3 methods.1 for retrieving the number of key that the user currently holding and the other 2 each responsible for increasing and decreasing the number of keys by 1 as the player pick up or uses the key.

This class inherits Item class and this enable the object of this class to be placed inside the list of Item class type named inventory in Actor.java.

Door Class

This class contains a constructor which determine the symbol that represents a door.

This class inherited Ground class to overwrite the canActorEnter method to determine whether the player can enter the room by checking the number of key that the player holding. If the key is more than 0, the method which responsible for decrementing the number of keys by 1 inside the key class will be called and a Boolean value of True will be returned. If the player does not have any key in the inventory, Boolean value of False will be returned.

Miniboss Class and Goon Class

Miniboss class inherits Grunt class because they have similar attacking behaviour but different damage value and hit point.The values can be modified by overwriting the method in charge of the damage and hit point.The miniboss moving behaviour needs to be overwrote as well as it will be staying at the same position throughout the entire game instead of following player.

Goon class inherits Grunt class as well because they have similar attacking and moving behaviour but with higher damage value.The damage value need to be overwrote and the moving behavior in the Grunt class can be reused.

Goon class contains an extra method to generate a string to insult player which will be invoked at a probability of 10%.

Ninja Class

This class inherits Actor class instead of inheriting any of the other enemy classes because it has a unique behavior.

The method of Actor will be reused to set the name,symbol displayed on the map,priority,hit point and maximum hit point of Ninja.

This class contains a method which will determine whether player is within 5 squares from them.It will return a boolean value of True if it is,else returning False.If it return True,it will move one space away from the player followed by invoking the Stunt class which contains the method to check this player condition.If the player is not stunned,the player will be stunned for two turn,otherwise no stunt action will be performed.

Q Class

This class inherits Actor class.

The method of Actor will be reused to set the name,symbol displayed on the map,priority,hit point and maximum hit point of Q.

This main reason of inheriting Actor class is to allow this class to access the item type list named inventory.The list contains the rocket object which will be used to get the number of rocket parts that the player is currently holding.

This class contains method to check whether the player is around of it.If it is,it will check whether the player is holding a rocket plan.If the player is holding,remove method inside rocket class is called to subtract the number of rocket plan by 1 followed by calling the adding method to increase the number of rocket body by 1.After performing the exchange of rocket parts,Q will be removed from the map.

If the number of rocket plans that the player is currently holding is equals to 0,this method will print a string of “I can give you something that will help, but I’m going to need the plans” and access the player’s inventory to check for the number of rocket plan that the player has. If the player has more than 0 rocket plan,the string of “Hand them over, I don’t have all day!” will be printed.

Stunt class

This class inherits SkipTurnAction class to reuse the method of skipping the player’s turn when player enter 5 to skip their turn.This method will be overwrote to skip the player turn without accepting player input and the skipping round will be increased to 2.

RocketPad Class

This class inherits Actor class to allow it to access the instance variable named inventory inside the Actor class.Inventory is an item type list which contains the rocket object and by accessing it,the number of rocket parts that the player is currently holding can be obtained for further use by the methods of RocketPad Class.

This class is invoked when the player is stepping on the rocket pad on the map. This class contains methods to check the number of rocket plans, rocket body and rocket engine in the inventory of player. If the number of each of the rocket part is more than or equal to 1,then the player win the game and exits the program.