

ECS414U/A Miniproject form

Queen Mary University of London

2021/22

Name	Shady Abushady
Student ID	210280593
Submitted file name	miniprojectOOPShadyAbushady210280593.zip
Level of this program (1, 2, 3, Extra)	2

Brief description of the program. Write the chosen theme and a high-level overview of the features (two or three sentences should suffice).

Adventure game.

You have an assignment due in 8 hours and have to navigate the library and interact with different characters in order to complete the assignment in time and win the game.

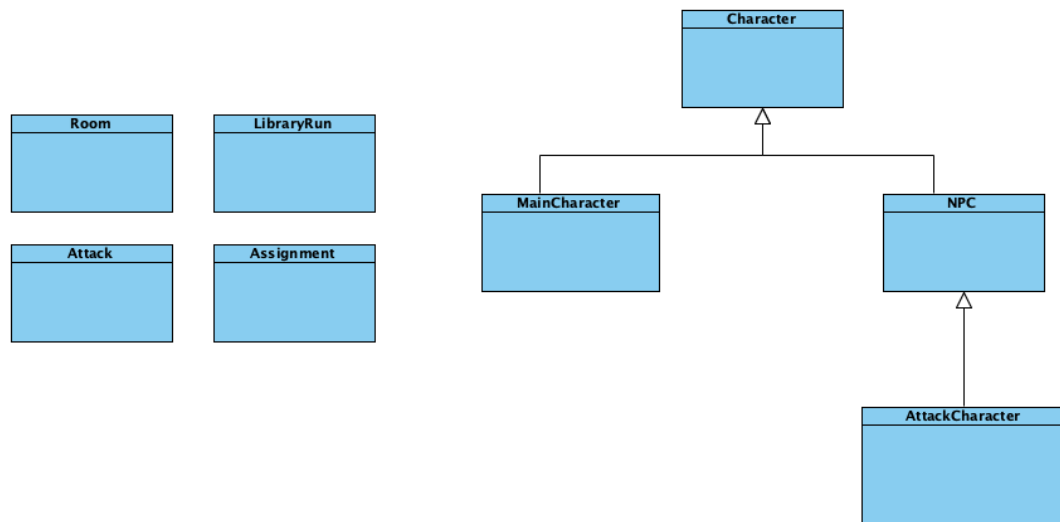
The game is text based and uses Scanner. The game uses inheritance and polymorphism in order to create its characters and there is exception handling (mostly if else statements but one try catch statement)

List all your source code files, and briefly describe their roles. Add as many rows as necessary. Mark the main file used for compilation in bold.

File name	Description
Assignment.java	This is the assignment, it holds information for the progress and time left, which decide when the game is over and whether the user has won or lost.
Attack.java	This is the attack, it holds information on how much damage is done per attack and the name of the attack. The methods in this class allow the main character to attack an AttackCharacter and vice versa and also allows the main character to escape a fight.
AttackCharacter.java	This is an NPC. However, it is special because it has health points and an attack meaning it is able to fight.
Character.java	This is the character superclass. It contains a name and get name method
LibraryRun.java	This is the main class. It contains the main method and executes the game.
MainCharacter.java	This is the main character. It holds an attack and an assignment. It has a direction method which allows the user to control the character and move from room to room.

NPC.java	This is the NPC class. It's a subclass of Character but a superclass of AttackCharacter. It has a message and a method that prints out said message.
Room.java	This is the room class. It contains the method assignNPC which creates a new NPC (30% chance) or an AttackCharacter (70% chance) every time it is called and replaces the existing instance variable. In order to randomise the creation of each character the class contains multiple arrays that help fill out the variables needed to create an NPC or an AttackCharacter.

Class diagram, in the format specified in the instructions.



Usage instructions. Describe briefly what features are available to the user and how to use them. If File I/O is used, list and describe the files involved.

The user can input the name of their character. Then the user has to choose a direction to go in, this will bring them into a new room. They then will either face a fighting character or a peaceful character. A fighting character will initiate a fight and after the first round of attacks the user can choose between escaping and continuing the fight. If the user chooses to escape they will have to go to the next room. If the user chooses to fight they will fight until the time for the assignment runs out and the game is over or until the NPC is defeated and the user has to go into the next room. If a peaceful character spawns the user gets a tip for the game and then moves on to the next room. The game continues until either the time runs out on the assignment or the assignment is complete.

Other comments.

`cd <directory pathname>`

(On my machine the directory pathname is:

`/Users/shadyabushady/Desktop/miniprojectOOPShadyAbushady210280593/project)`

Compilation command:

`javac LibraryRun.java`

Execution command:

`java LibraryRun`