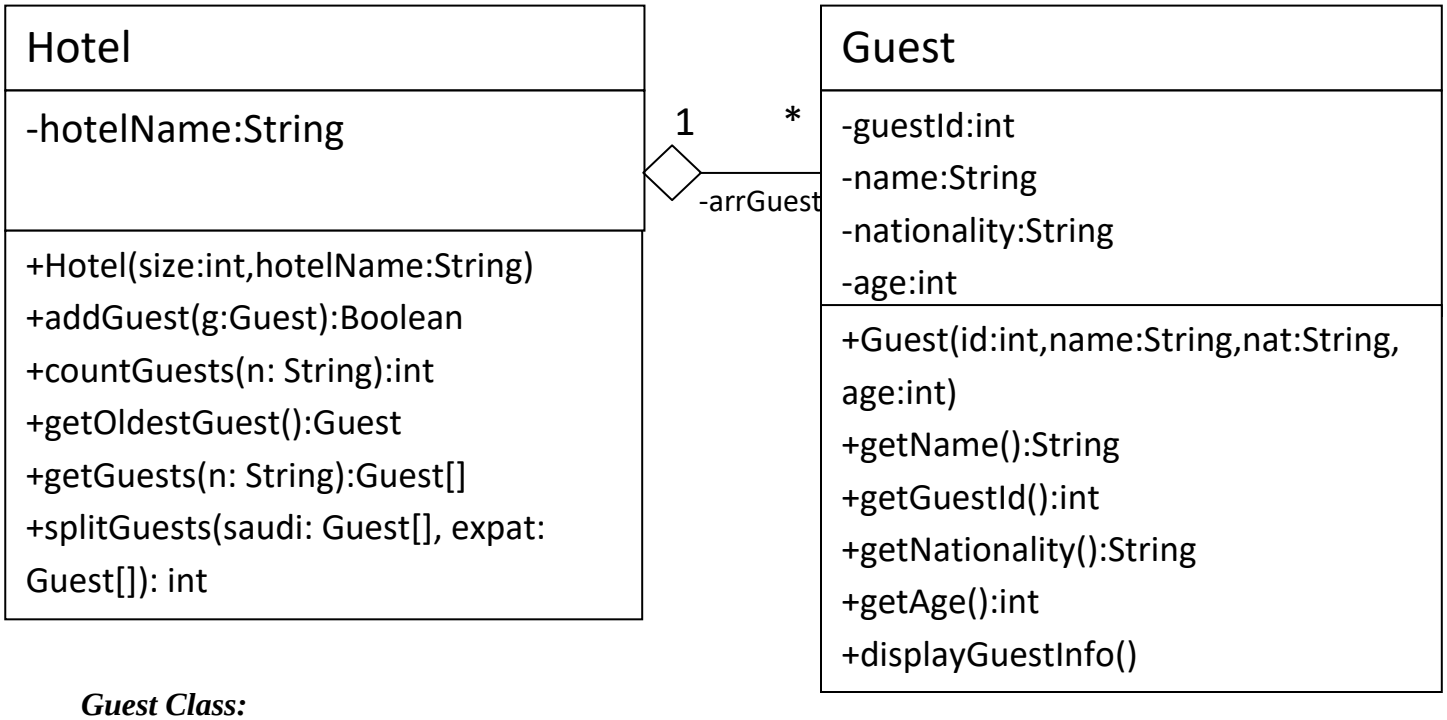
Lab: Relationship between Classes (Aggregation)

Create the classes along with the functionality given in the following UML Diagram. To understand the problem, please refer to the description given after the diagram.



Guest Class:

- Attributes:
  - guestId: unique id of the guest.
  - ***name***: the name of the guest.
  - ***nationality***: The nationality of the guest.
- Methods:
  - ***Guest(id:int,name: string, nationality: String, age: int)***: constructor
  - ***displayGuestinfo()***: this method displays all the attributes of the guest.
  - ***Getters***

Hotel Class:

- Attributes:
  - ***hotelName***: name of the hotel.

- Methods:
  - ***Hotel(size:int, name:String)***: constructor
  - ***addGuest(g:Guest):*** this method adds a guest to the hotel. It returns true if the ***Guest g*** is added; false otherwise.
  - ***countGuest(n:String)***: this method returns the number of guests of the hotel who are of nationality ***n***.
  - ***getOldestGuest():*** this method returns the oldest guest of the hotel.
  - ***getGuests(n:String):*** this method returns an array containing all guests of the hotel who are of nationality ***n***.
  - ***split(saudi: Guest[],expat: Guest[]):*** This method inserts all Saudi guests of the hotel into the array saudi. The other guests (non-Saudi guests) are inserted into the array expat. This method returns the number of non-Saudi guests inserted into the array expat.

Write a java program that will deal with hotel processing of different daily guests. Hotel can contain many guests. The java program will consist of two classes given in the above UML diagram and a test class that will display following Menu to execute different member functions of these classes.

To add a new guest, Enter 1

To get the number of guests of a given nationality, Enter 2

To get and display the oldest guest, Enter 3

To get and display all guests of a given nationality, enter 4

To view all Saudi guests in the Hotel, Enter 5

To Exit, Enter 0

Enter Your Option:____