Exercises

## Exercise 1: Inheritance

Create the following classes:

A User class



* Create a constructor which receives and stores **accessLevel** and **name**
* Overload the constructor, create a constructor which only receives name and sets the accessLevel to 0
* Use getter/setter methods to expose **name** as read/write and **accessLevel** as read only

Add two child classes, Student and Lecturer



* The method in the child classes should override the parent’s method
* The Lecturer always has an access level of 5
* The student always has an access level of 3
* Build appropriate constructors to meet the needs of this hierarchy
* EnterRoom() should write to console ClassName tries to enter the room, then it should call OpenDoor() and print if successful or not, If successful it should then call CloseDoor() and confirm this with a print statement.

Build the objects using parent references and test the relationship. Ask the User, the Student and the Lecturer objects to EnterRoom()

## Exercise 2 – Associations

Create another class, Room



* doorLocked should be initialised to true.
* OpenDoor(..) should open the door if the door is unlocked.
* If the accessLevel is >4 it should unlock and open the door.
* It should return the doorOpen status.
* CloseDoor() should set doorOpen to false and doorLocked to true
* Test your Room Object to make sure you have coded it correctly. Only an access level of 4 or above should successfully open the door.

The User uses the room. This is called an association.



There are two ways it can use the room. It can create the Room object or it can be given the room object. We will explore both.

1. **Create a Room object in User**

* In the EnterRoom() method, create a room object
  + Room room = new Room();
* We need to pass this room to OpenDoor() and CloseDoor() methods.
  + Overload the methods to receive a room (see below)
  + *OpenDoor(Room room)* and *CloseDoor(Room room)*
* OpenDoor(..) should be changed to call OpenDoor(..) in Room, likewise for CloseDoor(..)
  + *room.OpenDoor(accessLevel);*



* Test your design, who can open the room?

1. Receive a Room object

* Add an overload to EnterRoom(..) which can receive a Room object and uses that to call the open and close door methods



* Create the room object in Main(..) and send it to the User, Student and Lecturer objects
* Test to see if they can enter the room.

## Challenge

It would be nice for the lecturer to be able to unlock the room for the students. Using the existing classes (no changes to the classes) how would you tell the lecturer object to open the room? Test that the students can then enter the room.

## Exercise 3 Inheritance and Association

In this new scenario there are three types of rooms.

1. StaffRoom – access level 5
2. ClassRoom – access level 3
3. Toilet – access level 0

Change your design to incorporate this new hierarchy. Store the accessLevel in the relevant Room using constructors as you did with the student/lecturer constructors and their access level.



Now test your design again – who can access each type of room? Can you get the lecturer to unlock the classroom door for the user?