Title	**<u>Coworking space System</u>**
Description	This project manages a system for coworking space Your system has 2 types of users: visitor, admin It contains different types of meeting rooms
Deliverables	1. Class diagram for the project (Printed) 2. Java project includes all classes and functions in the description 3. Documentation that contains a. your own system description b. input and output scenarios
Bonus	GUI
Description	Your system has 3 types of rooms: General room Meeting room and Teaching room. It has 2 General rooms, 3 Meeting rooms and 3 teaching rooms. Your visitor is one of 3 types: general, formal, instructor The system has one admin. The admin manages all entities. A visitor can login to the system using name and password (check if it is the first time to use the system, ask to register and take the rest of needed information to make an account) The admin can login using name: “admin” and password: “admin” The system has a leader board for all types of visitors and the visitor with the highest number of hours has a free 7 hours for a month in a room compatible with the type of the visitor. A visitor has: name, password, ID, Type A visitor can: ● Register ● Log in ● Make reservation: reserve one slot or more, let him choose from a list of slots (as long as these slots are available and in the current month of reservation) Hint: Let the visitor choose from room slots that goes with the **visitor type** General visitor: general room slot

Formal visitor: meeting room slot
Instructor visitor: teaching room slot

- Cancel reservation: delete reservation, free the slot and do not forget to apply penalty (extra fees) for the next reservation.
- Update reservation: update reservation time or update reservation date. make all the checkups you need to make sure the update is valid.
- Display available slots: display times and dates
- Reward system for free 1 hour after finishing 6 hours this rule apply for all types of visitor except the instructor.

  The instructor can have a free 1 hour but after finishing 12 hours.

General room has: name, ID, number_of_visitors (max of 20), list of slots, list of visitors
Meeting room has: name, ID, number_of_visitors (max of 10), list of slots, list of visitors
Teaching room has: name, ID, projector_type, board_type, instructor_name, number_of_visitors (max of 10), list of slots, list of visitors

A slot has time, date and fees

The admin has: name, password
The admin can:

- Log in
- Add slots
- Delete any entity
- Display all available slots for all rooms (room type + room ID + slot details <date and time>)
- Display all visitor’s data
- Display all rooms’ data
- Display all instructor’s data
- Calculate and display the total amount of money from general room reservation

	● Calculate and display the total amount of money from meeting room reservation ● Calculate and display the total amount of money from teaching room reservation ● Update any data of any entity **Don’t forget to handle deletions and updates: if any update is done in any case in any entity, it should be updated for the whole system**
Notes	1. You should implement all concepts of OOP. 2. Each member **MUST** work on at least one of the required classes besides file processing or GUI. (**Individual marks**) 3. The evaluation will be mainly based on the student’s ability to use and apply OOP concepts and the explanation of the code. 4. You must deliver the Class Diagram for the project. 5. You must apply exception handling 6. Using Files is mandatory (Not Database) 7. Any project must have at least **8 classes** 8. Regarding **files:** ● You must have only two functions for file reading and writing. ● You should read data once at the beginning of your run then do your operations and access the code then save in files at the end of your program.