General Overview

The provided Java classes form a simple system for managing hotel rooms, encapsulated within the Hotel class. These classes offer basic functionalities for adding, removing, saving, and loading room data. Additionally, the system supports retrieving information about the hotel's address and VAT number, adhering to the LegalEntity interface.

Detailed Functionality and Goal of Each Class

LegalEntity

The LegalEntity interface defines the contract for any legal entity that can provide an address and a VAT number. It includes two methods: getAddress, which returns the entity's address, and getVatNumber, which returns the VAT number.

Room

The Room class represents a room in a hotel with a room number and type. It contains fields for the room number (int roomNumber) and type (String type). The class provides methods to get and set these values and a toString method to return a string representation of the room. Functionality includes creating instances of Room with specific details, modifying these details, and representing the room as a string.

Hotel

The Hotel class represents a hotel containing a list of rooms and implements the LegalEntity interface. It includes fields for the hotel's address (String address), VAT number (String vatNumber), and a list of rooms (List<Room> rooms). Methods provided by the class include getAddress and getVatNumber for retrieving the hotel's address and VAT number, respectively; addRoom for adding a room to the hotel's list; removeRoom for removing a room from the list; getRooms for retrieving the list of rooms; and saveRoomsToFile and loadRoomsFromFile for saving and loading the list of rooms to and from a file. The toString method returns a string representation of the hotel, including its rooms. The functionality allows for adding and removing rooms, retrieving and representing the room list, and saving and loading room data for persistence.

Sample Test Scenarios

1. Adding Rooms: Add three different rooms (Single, Double, Suite) to the hotel. The rooms list should contain the three added rooms.

2. Removing Rooms: Remove one room from the hotel. The rooms list should no longer contain the removed room.

3. Saving and Loading Rooms: Save the current list of rooms to a file and then load the rooms from the file. The rooms list should be identical before and after the save/load operations.

4. Clearing and Reloading Rooms: Clear the current rooms list and then load the rooms from a previously saved file. The rooms list should be restored to the state it was in when it was saved.

Important Methods

1. addRoom(Room room): Adds the specified room to the hotel's list of rooms. This method is used to expand the hotel's inventory of rooms.

2. removeRoom(Room room): Removes the specified room from the hotel's list of rooms. This method is used to remove rooms that are no longer available or required.

3. saveRoomsToFile(String filename): Serializes the list of rooms and saves it to the specified file. This method is used to persist the current state of the hotel's rooms for later retrieval.

4. loadRoomsFromFile(String filename): Deserializes the list of rooms from the specified file and restores it to the hotel's rooms list. This method is used to restore the hotel's rooms from a previously saved state.