

Class Descriptions

1. LegalEntity

Purpose: This is an interface that defines the contract for any legal entity, requiring the implementation of methods to retrieve the address and VAT number.

Methods:

String getAddress(): Returns the address of the legal entity.

String getVatNumber(): Returns the VAT number of the legal entity.

```
package finalexam.task4;
```

```
public interface LegalEntity {  
    String getAddress();  
    String getVatNumber();  
}
```

2. Guard

Purpose: Represents a security guard with essential personal details.

Fields:

String name: The first name of the guard.

String surname: The surname of the guard.

String personalNumber: The personal identification number of the guard.

Constructor:

Guard(String name, String surname, String personalNumber): Initializes the guard object with the provided name, surname, and personal number.

Methods:

String toString(): Provides a string representation of the guard, including their name, surname, and personal number.

```
package finalexam.task4;
```

```
public class Guard {
```

```
    String name;
```

```
    String surname;
```

```
    String personalNumber;
```

```
    public Guard(String name, String surname, String personalNumber) {
```

```
        this.name = name;
```

```
        this.surname = surname;
```

```
        this.personalNumber = personalNumber;
```

```
    }
```

@Override

```
public String toString() {  
    return "Guard [Name=" + name + ", Surname=" + surname + ",  
    Personal Number=" + personalNumber + "];"  
}  
}
```

3. SecurityCompany

Purpose: Manages a list of guards, implements the LegalEntity interface, and provides methods to add, remove, save, and load guards.

Fields:

String address: The address of the security company.

String vatNumber: The VAT number of the security company.

List<Guard> guards: A list to store guard objects.

Constructor:

SecurityCompany(String address, String vatNumber): Initializes the security company with the provided address and VAT number.

Methods:

String getAddress(): Returns the address of the company.

String getVatNumber(): Returns the VAT number of the company.

void addGuard(Guard guard): Adds a guard to the list.

boolean deleteGuard(Guard guard): Removes a guard from the list.

void printGuards(): Prints the details of all guards in the list.

void saveGuardsList(String filename): Saves the list of guards to a file.

void loadGuardsList(String filename): Loads the list of guards from a file.

```
package finalexam.task4;
```

```
import java.io.*;
```

```
import java.util.ArrayList;
```

```
import java.util.List;
```

```
public class SecurityCompany implements LegalEntity {
```

```
    private String address;
```

```
    private String vatNumber;
```

```
    private List<Guard> guards = new ArrayList<>();
```

```
    public SecurityCompany(String address, String vatNumber) {
```

```
        this.address = address;
```

```
        this.vatNumber = vatNumber;
```

```
    }
```

@Override

```
public String getAddress() {  
    return address;  
}
```

@Override

```
public String getVatNumber() {  
    return vatNumber;  
}
```

```
public void addGuard(Guard guard) {  
    guards.add(guard);  
}
```

```
public boolean deleteGuard(Guard guard) {  
    return guards.remove(guard);  
}
```

```
public void printGuards() {
```

```
    for (Guard guard : guards) {  
        System.out.println(guard);  
    }  
}
```

```
public void saveGuardsList(String filename) {  
    try (PrintWriter writer = new PrintWriter(new  
FileWriter(filename))) {  
        for (Guard guard : guards) {  
            writer.println(guard.name + "," + guard.surname + "," +  
guard.personalNumber);  
        }  
    } catch (IOException e) {  
        System.out.println("An error occurred while saving the guards  
list.");  
        e.printStackTrace();  
    }  
}
```

```
public void loadGuardsList(String filename) {  
    guards.clear();
```

```
try (BufferedReader reader = new BufferedReader(new
FileReader(filename))) {

    String line;

    while ((line = reader.readLine()) != null) {

        String[] parts = line.split(",");

        if (parts.length == 3) {

            guards.add(new Guard(parts[0], parts[1], parts[2]));

        }

    }

} catch (IOException e) {

    System.out.println("An error occurred while loading the guards
list.");

    e.printStackTrace();

}

}
```