Unit 1 – Compulsory assignment

Using the information of the Restaurant database that you created last unit, build the following Java application:

The database is corrupted, and the company has exported the data into .csv files to store the information until the database is recovered:

- The information is stored in text files, enclosed to the assignment: *customers.csv* and *orders.csv*

- The application should have the following options:
  - List all customers
  - List all orders
  - List orders of a specific customer or a specific date (List orders screen - Filter)
  - Append a new order: Make sure that the customer exist
  - Delete a customer: If it has any orders, ask the user, and if so, delete the orders before deleting the customer

- Use a constructor for the model classes to convert the line into an object, and a toStringTextFile() method to do the opposite. Example:

```
public class Customer {
    private int id;
    private String firstName;
    private String lastName;
    private String email;
    private String phone;
    private LocalDate dob;

    public Customer(String fileLine) {
        String[] elemArray = fileLine.split(";");
        this.id=Integer.parseInt(elemArray[0]);
        this.firstName=elemArray[1];
        this.lastName=elemArray[2];
        this.email=elemArray[3];
        this.phone=elemArray[4];
        this.dob=LocalDate.parse(elemArray[5]);
    }
    public String toStringTextFile() {
        return id + ";" +  firstName + ";" + lastName
                + ";" + email + ";" + phone +  ";" + dob;
    }
```

- The file paths are stored into a Properties file.

Use java.NIO Files class (*readAllLines* and *write* methods for customers file, *newBufferedReader* and *newBufferedWriter* for orders file). Use singleton patterns for the Properties and Yaml files. Use a layered architecture.