

Introduction

The goal of this project is to design a simple and effective management system for a bakery. The system allows the management of staff (in the context of a paper factory) and customers (for the bakery). The project demonstrates the use of object-oriented programming principles, including encapsulation, inheritance, and serialization, to manage data persistence and retrieval.

Class Descriptions and Functionalities

1. LegalEntity Interface

- **Functionality:** This interface defines the basic contract for any legal entity, ensuring that any class implementing this interface will provide methods to get the address and VAT number.
- **Goal:** To standardize the way legal entities such as businesses or organizations handle their address and VAT information.

2. Person Class

- **Functionality:** Represents a person with basic attributes such as name, surname, and personal number. Implements `Serializable` to allow instances to be serialized and deserialized.
- **Goal:** To model individual persons who can be managed within a system (e.g., as staff in a factory).

3. PaperFactory Class

- **Functionality:** Manages a list of staff members (Persons). Provides methods to add, delete, save, and load staff members to and from a file.
- **Goal:** To demonstrate basic CRUD (Create, Read, Update, Delete) operations and data persistence for managing staff in a factory.

4. FactoryTester Class

- **Functionality:** Tests the functionality of the `PaperFactory` class. It adds persons to the factory, saves the list to a file, clears the list, and then loads it back from the file.
- **Goal:** To ensure the `PaperFactory` methods work correctly and demonstrate their usage.

5. Customer Class

- **Functionality:** Represents a customer with attributes such as name, email, and phone number. Implements `Serializable` to allow instances to be serialized and deserialized.
- **Goal:** To model individual customers for the bakery.

6. Bakery Class

- **Functionality:** Manages a list of customers. Provides methods to add, delete, save, and load customers to and from a file. Implements the `LegalEntity` interface to provide address and VAT number information.
- **Goal:** To manage customer data for a bakery, demonstrating similar CRUD operations and data persistence as the `PaperFactory` class.

7. BakeryTester Class

- **Functionality:** Tests the functionality of the `Bakery` class. It adds customers to the bakery, saves the list to a file, clears the list, and then loads it back from the file.

- **Goal:** To ensure the `Bakery` methods work correctly and demonstrate their usage.

Conclusion

This project demonstrates the design and implementation of a simple management system for a bakery and a factory. The use of interfaces, serialization, and basic CRUD operations provides a solid foundation for understanding object-oriented programming and data persistence techniques. The `FactoryTester` and `BakeryTester` classes serve as effective tools for validating the functionality of the system.