

9. Параметризація в Java

- **Мета:** Вивчення принципів параметризації в *Java*.
- Розробка параметризованих класів та методів.

1 ВИМОГИ

1.1 Розробник

Інформація про розробника:

- Шарма Олександр Раджнішович
- НТУ “ХПІ” КІТ1196
- Варіант 25(2)

1.2 Загальне завдання

1. Створити власний клас-контейнер, що параметризується ([Generic Type](#)), на основі [зв'язних списків](#) для реалізації колекції domain-об'єктів [лабораторної роботи №7](#).
2. Для розроблених класів-контейнерів забезпечити можливість використання їх об'єктів у циклі [foreach](#) в якості джерела даних.
3. Забезпечити можливість збереження та відновлення колекції об'єктів: 1) за допомогою стандартної серіалізації; 2) не використовуючи протокол серіалізації.
4. Продемонструвати розроблену функціональність: створення контейнера, додавання елементів, видалення елементів, очищення контейнера, перетворення у масив, перетворення у рядок, перевірку на наявність елементів.
5. Забороняється використання контейнерів (колекцій) з [Java Collections Framework](#).

1.3 Задача

Кадрове агентство. Дані про вакансії: фірма; спеціальність; умови праці; оплата; вимоги до фахівця - набір необов'язкових властивостей у вигляді “спеціальність, стаж, освіта”.

2 ОПИС ПРОГРАМИ

2.1 Засоби ООП

Композиція, інкапсуляція.

2.2 Ієрархія та структура даних

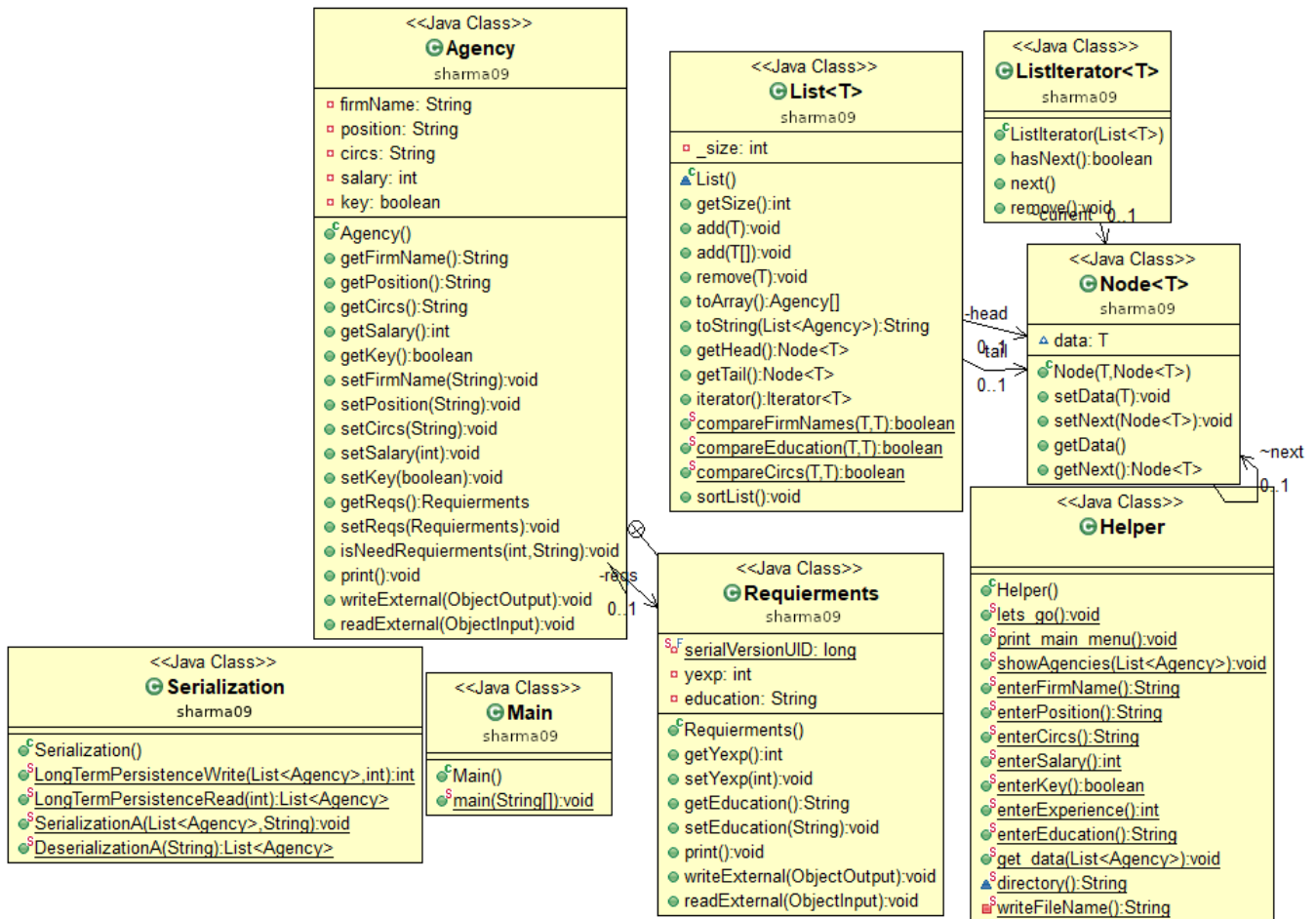


Рисунок 1 – діаграма класів

2.3 Важливі фрагменти програми

```
public class Serialization {
    public static int LongTermPersistenceWrite(List<Agency> object,int size) throws FileNotFoundException {

        XMLEncoder encoder = new XMLEncoder(
            new BufferedOutputStream(
                new FileOutputStream(Helper.directory())));

        for (Agency agency: object) {
            encoder.writeObject(agency);
        }

        try(FileWriter writer = new FileWriter("notes3.txt", false))
        {
            writer.write(size);
            writer.flush();
        }
        catch(IOException ex){
            System.out.println(ex.getMessage());
        }
        size = object.getSize();
        encoder.close();
        return size;
    }

    public static List<Agency> LongTermPersistenceRead(int size) throws FileNotFoundException {
        XMLDecoder decoder = new XMLDecoder(
            new BufferedInputStream(
                new FileInputStream(Helper.directory())));

        List<Agency> object = new List<Agency>();
        System.out.println(size);
        for(int i = 0; i < size; i++) {
            object.add((Agency)decoder.readObject());
        }
    }
}
```

Рисунок 2 – нестандартний протокол серіалізації/десеріалізації

3 ВАРІАНТИ ВИКОРИСТАННЯ

Програма створена для роботи з прикладною задачею. Для коректної роботи були реалізовані методи введення та отримання даних, також дані приховані від користувача, щоб не порушувати суттєвість об'єкту.

```

Информация об этой фирме:
Название фирмы : Luxoft
Должность : Senior SoftDev
Условия работы: NICE
Зарплата : 2000
Дополнительные требования:
Опыт работы : 2
Образование : Higher
1.Enter data
2.Show current data
3.Clear all
4.Save data
5.Recover data
6.Exit
4
Enter file name: zzz.xml
1) .                2) ..                3) .settings        4) bin
5) src
C:\Users\{Dean}\eclipse-workspace\hulevych_andrii> 1
1.Enter data
2.Show current data
3.Clear all
4.Save data
5.Recover data
6.Exit

```

Рисунок 3 – результати серіалізації

```

Enter data first
1.Enter data
2.Show current data
3.Clear all
4.Save data
5.Recover data
6.Exit
5
Enter file name: zzz.xml
1) .                2) ..                3) .settings        4) bin
5) src
C:\Users\{Dean}\eclipse-workspace\hulevych_andrii> 1
1.Enter data
2.Show current data
3.Clear all
4.Save data
5.Recover data
6.Exit
2
Информация об этой фирме:
Название фирмы : Luxoft
Должность : Senior SoftDev
Условия работы: NICE
Зарплата : 2000
Дополнительные требования:
Опыт работы : 2
Образование : Higher
1.Enter data
2.Show current data
3.Clear all
4.Save data
5.Recover data
6.Exit

```

Рисунок 3 – результати десеріалізації

ВИСНОВКИ

В даній лабораторній роботі було розроблено клас-контейнер для роботи з прикладною задачею та реалізовано відповідні методи. Набуто навичок об'єктно-орієнтованого підходу та з параметризації на платформі Java SE.