

9. Параметризація в Java

Мета: Вивчення принципів параметризації в Java.

Розробка параметризованих класів та методів.

1 ВИМОГИ

1.1 Розробник

Інформація про розробника:

- Кабак Олександр Русланович
- НТУ “ХПІ” 1.KIT102.8a
- Варіант 5

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 наявності елементів.
5. Забороняється використання контейнерів (колекцій) з Java Collections Framework.

1.3 Задача

5. Прикладна галузь: Довідник покупця. Торгівельна точка: назва; адреса; телефони (кількість не обмежена); спеціалізація; час роботи (з зазначенням днів тижня).

2 ОПИС ПРОГРАМИ

2.1 Засоби ООП

У даній програмі присутні:

- 1) Двозв'язний список, що параметризується;
- 2) Збереження та відновлення об'єктів без протоколу серіалізації та з ним.

2.2 Важливі фрагменти програми

```
public class List<E> extends AbstractSequentialList<E> implements java.util.List<E>, Serializable {
    private int size;
    private List.Node<E> first;
    private List.Node<E> last;

    public List() { this.size = 0; }
    void linkLast(E e) {...}
    void linkBefore(E e, List.Node<E> succ) {...}
    E unlink(List.Node<E> x) {...}
    public boolean contains(Object o) { return this.indexOf(o) >= 0; }
    public int size() { return this.size; }
    public void clear() {...}
    public void add(int index, E element) {...}
    public E remove(int index) {...}
    public Object[] toArray() {...}
    public void writeExternalizable(ObjectOutputStream s) throws IOException {...}
    public void readExternalizable(ObjectInputStream s) throws IOException, ClassNotFoundException {...}
    public String toString() {...}
}
```

Рис. 1 – Функціонал списка по заданню

```
public void writeExternalizable(ObjectOutputStream s) throws IOException {
    s.defaultWriteObject();
    s.writeInt(this.size);

    for (List.Node x = this.first; x != null; x = x.next) {
        s.writeObject(x.item);
    }
}

public void readExternalizable(ObjectInputStream s) throws IOException, ClassNotFoundException {
    s.defaultReadObject();
    int size = s.readInt();

    for (int i = 0; i < size; ++i) {
        this.linkLast((E) s.readObject());
    }
}
```

Рис. 2 – Реалізація збереження даних та їх зчитування з файла

3 ВАРІАНТИ ВИКОРИСТАННЯ

Програма дозволяє створювати об'єкт – список магазинів – , що заносяться у запис каталогу. Користувач може додавати інші магазини до списку, видаляти елементи вибірково, а також очистити весь масив одним викликом відповідної кнопки меню. Також присутня можливість серіалізувати/десеріалізувати об'єкти з файлу.

Справочник покупателя

1. Список торговых магазинов
2. Добавить торговый магазин
3. Убрать торговый магазин списка
4. Очистить весь список
5. Завершить работу

Выберите опцию:

1

Список магазинов:

Название: Toys

Адресс: str Pushkina 12

Номер(а) телефона(ов):[635123]

Расписание работы магазина:

Пн:8:30-18:30

Вт:8:30-18:30

Ср:8:30-18:30

Чт:8:30-18:30

Пт:8:30-18:30

Сб:8:30-18:30

Вс:8:30-18:30

Название: Closes Flowey

Адресс: str Puskina 13

Номер(а) телефона(ов):[321231]

Расписание работы магазина:

Пн:8:30-18:30

Вт:8:30-18:30

Ср:8:30-18:30

Чт:8:30-18:30

Пт:8:30-18:30

Сб:8:30-18:30

Вс:8:30-18:30

Рис. 3 – Результати виведення списку об'єктів(після створення 2 об'єктів)

```
try (ObjectOutputStream oos = new ObjectOutputStream(new FileOutputStream( name: "person.dat"))) {  
    stores.writeObject(oos);  
} catch (Exception ex) {  
    System.out.println(ex.getMessage());  
}
```

Рисунок 4 – Приклад запису у файл

```
try (ObjectInputStream ois = new ObjectInputStream(new FileInputStream( name: "C:\\Greed\\Study\\Java\\Lab9\\person.dat"))) {  
    stores = (List<Store>) ois.readObject();  
    System.out.println(stores);  
} catch (Exception ex) {  
    System.out.println(ex.getMessage());  
}
```

```
"C:\Greed\Programs\IntelliJ IDEA Community Edition 2019.3.4\jbr\bin\java.exe"  
  
Название: Toys  
Адресс: str. Pushkina 12  
Номер(а) телефона(ов):[123654, 123655]  
Расписание работы магазина:  
    Пн:8:30-18:30  
    Вт:8:30-18:30  
    Ср:8:30-18:30  
    Чт:8:30-18:30  
    Пт:8:30-18:30  
    Сб:8:30-18:30  
    Вс:8:30-18:30  
  
Справочник покупателя  
1. Список торговых магазинов  
2. Добавить торговый магазин  
3. Убрать торговый магазин списка  
4. Очистить весь список  
5. Завершить работу  
  
Выберите опцию:
```

Рисунок 5 – Приклад зчитування інформації

Висновки: даній лабораторній роботі розроблено та реалізовано двозв'язний список, реалізовано управління списком *domain*-об'єктів за допомогою меню(спілкування з користувачем), розроблено та використано 2 варіанти серіалізації, розроблено метод для перегляду списку за допомогою *foreach*.