

10. Обробка параметризованих контейнерів

Мета: Розширення функціональності параметризованих класів.

1 ВИМОГИ

1.1 Розробник

Інформація про розробника:

- Кононенко Дмитро Олексійович
- НТУ “ХПІ” 1.KIT102.8a
- Варіант 8

1.2 Загальне завдання

- Використовуючи програму рішення завдання лабораторної роботи №9:
- Розробити параметризовані методи (Generic Methods) для обробки колекцій об'єктів згідно прикладної задачі.
- Продемонструвати розроблену функціональність (створення, управління та обробку власних контейнерів) в діалоговому та автоматичному режимах.
- Автоматичний режим виконання програми задається параметром командного рядка -auto. Наприклад, java ClassName -auto.
- В автоматичному режимі діалог з користувачем відсутній, необхідні данні генеруються, або зчитуються з файлу.
- Забороняється використання алгоритмів з Java Collections Framework.

1.3 Задача

Варіант 8. Автостанція. Сортування за номером рейсу, за часом відправлення, за кількістю вільних місць.

2 ОПИС ПРОГРАМИ

2.1 Засоби ООП

У даній програмі присутні об'єктно-орієнтовані методи: Інкапсуляція – захист даних від неправомірного користування та поліморфізм.

2.2 Ієрархія та структура даних

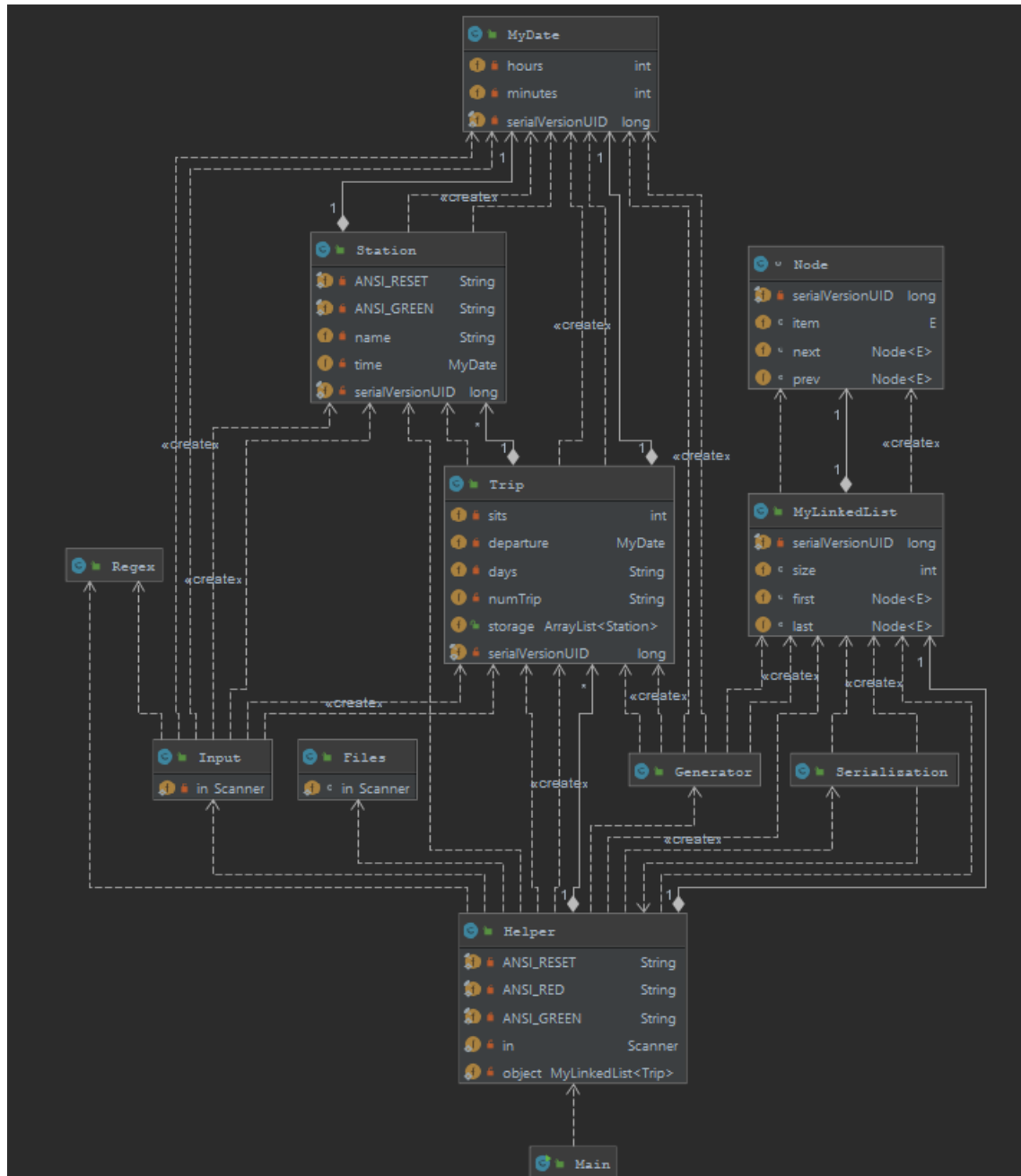


Рисунок 1 – Діаграма класів

2.3 Важливі фрагменти програми

```
public void sort(Comparator<? super E> c, E[] arr) {  
    Object[] a = arr;  
    Arrays.sort(a, (Comparator) c);  
    ListIterator<E> i = this.listIterator();  
    for (Object e : a) {  
        i.next();  
        i.set((E) e);  
    }  
}
```

Рисунок 2 — сортування в списку(саме використання знаходиться в класі Helper)

```
case 2:  
    input.sort(new Comparator<Trip>() {  
        @Override  
        public int compare(Trip p1, Trip p2) {  
            if (p1.getSits() != p2.getSits()) {  
                return p1.getSits() - p2.getSits();  
            }  
            return 0;  
        }  
    });  
    break;  
case 3:  
    input.sort(new Comparator<Trip>() {  
        @Override  
        public int compare(Trip p1, Trip p2) {  
            if (p1.getDeparture().getToCompare() != p2.getDeparture().getToCompare()) {  
                return p1.getDeparture().getToCompare() - p2.getDeparture().getToCompare();  
            }  
            return 0;  
        }  
    });  
    break;
```

Рисунок 3 — створення компараторів для сортування

3 ВАРІАНТИ ВИКОРИСТАННЯ

Програма дозволяє створювати об'єкти – “записи в розкладі”, що заносяться у запис каталогу, тобто створюється масив об'єктів. Користувач може додавати об'єкти до масиву, видаляти елементи вибірково, а також очистити увесь масив одним викликом відповідної кнопки меню. Також присутня можливість серіалізувати/десеріалізувати об'єкти з файлу. Якщо програма починає свою роботу з параметром “-auto\ -a”, то перший крок програми зчитування з файлу.

```
Hello, you are now in menu.....
List of settings:
0 - Exit
1 - Show data
2 - Insert
3 - Remove
4 - Sort
5 - Clear
6 - to String
7 - to Array
8 - Save
9 - Download
10 - Search
Select: 4
List of settings:
0 - Exit
1 - Sort by number of trip
2 - Sort by number of sits
3 - Sort by time of department
4 - Show data
Select:
```

Рисунок 4 — обираємо режим сортування

```
List of settings:
0 - Exit
1 - Sort by number of trip
2 - Sort by number of sits
3 - Sort by time of department
4 - Show data
Select: 3
List of settings:
0 - Exit
1 - Sort by number of trip
2 - Sort by number of sits
3 - Sort by time of department
4 - Show data
Select: 4
```

Рисунок 5 — обираємо режим сортування(та виводимо на екран)

```
Number of trip: 657EP
Number of sits: 32
Working days: fri
Time of department: 00:01

Number of trip: 608GJ
Number of sits: 61
Working days: thur
Time of department: 00:15

Number of trip: 210JE
Number of sits: 15
Working days: tues
Time of department: 01:02

Number of trip: 414NY
Number of sits: 12
Working days: thur
Time of department: 01:19

Number of trip: 12JE
Number of sits: 55
Working days: wed
Time of department: 01:37

Number of trip: 67EB
Number of sits: 20
Working days: sun
Time of department: 01:39

Number of trip: 462GA
Number of sits: 55
Working days: sun
Time of department: 02:03
```

Рисунок 6 — Результат сортування

ВИСНОВКИ

В даній лабораторній роботі було розроблено та реалізовано класи та методи відповідно прикладної галузі, реалізував управління списком domain-об'єктів, а методи сортування та обробка початкових параметрів.