

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНОМУ
УНІВЕРСИТЕТУ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

Кафедра систем штучного інтелекту

Лабораторна робота №11
Параметризація в Java. Обробка параметризованих контейнерів
Мета з дисципліни.
«Об’єктно-орієнтоване програмування»

Виконала:
студентка групи КН-108
Гетьман Соломії
Викладач:
Гасько Р.Т.

Львів – 2018 р.

Мета

- Вивчення принципів параметризації в Java
- Розробка параметризованих класів та методів.
- Розширення функціональності параметризованих класів.

Задача

1. Створити власний клас-контейнер, що параметризується
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
4. Продемонструвати розроблену функціональність: створення контейнера, додавання елементів, видалення елементів, очищення контейнера, перетворення у масив, перетворення у рядок, перевірку наявності елементів
5. Розробити параметризовані методи
7. Продемонструвати розроблену функціональність (створення, управління та обробку власних контейнерів) в діалоговому та автоматичному режимах.

Задача

```
package
ua.lpnuai.oop.hetman11;

import java.util.Scanner;
public class Main {
    static void auto(){
        System.out.println("Creating list...");
        LinkedList<String> list = new LinkedList<String>();
        list.add("Hetman");
        list.add("Pona");
        list.add("Pankiv");
        list.add("Kit");
        System.out.println("Using toString:");
        System.out.println(list.toString());
        list.delete(2);
        System.out.println("Converting to Array...");
        Object[] arr = list.toArray();
        for (Object e: arr
            ) {
                System.out.println(e);
            }
    }
}
```

```

    }
    System.out.println("Cleaning list...");
    list.clean();
    System.out.println(list.toString());
}
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    System.out.println("Program started!");
    String in;
    LinkedList<String> list = new LinkedList<String>();
    while(true){
        in = sc.nextLine();
        switch(in.toLowerCase()){
            case "-auto":
                auto();
                System.exit(0);
                break;
            case "-add":
            case "--add":
                System.out.println("Enter data:");
                list.add(sc.nextLine());
                break;
            case "-r":
            case "--remove":
                System.out.println("Enter id:");
                list.delete(Integer.parseInt(sc.nextLine()));
                break;
            case "-p":
            case "--print":
                System.out.println(list.toString());
                break;
            case "-c":
            case "--clean":
                System.out.println("Cleaning the list...");
                list.clean();
                System.out.println("Cleaned.");
                break;
            case "-array":
            case "--array":
                System.out.println("Converting to Array...");
                Object[] arr = list.toArray();
                for (Object e: arr
                    ) {
                        System.out.println(e);
                    }
                break;
            case "-e":
            case "--exit":
                System.out.println("Bye, bye!");
                System.exit(0);
        }
    }
}

```

```
        break;
    default:
        System.out.println("Wrong command!");
        break;
    }
}
}
```