

Міністерство освіти і науки України  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»  
Інститут комп'ютерних технологій, автоматики та метрології

Кафедра ЕОМ



**Звіт**

**До лабораторної роботи №7**

**З дисципліни: «Кросплатформні засоби програмування»**

**На тему «Параметризоване програмування»**

**Варіант №26**

Виконав: ст. гр. КІ-36

Бродський Б.А.

Перевірив: доцент кафедри ЕОМ

Іванов Ю.С.

Львів – 2022

**Мета:** оволодіти навиками параметризованого програмування мовою Java.

### ЗАВДАННЯ

1. Створити параметризований клас, що реалізує предметну область задану варіантом. Клас має містити мінімум 4 методи опрацювання даних включаючи розміщення та виймання елементів. Парні варіанти реалізують пошук мінімального елементу, непарні – максимального. Написати на мові Java та налагодити програму-драйвер для розробленого класу, яка мстить мінімум 2 різні класи екземпляри яких розмішуються у

8

екземплярі розробленого класу-контейнеру. Програма має розміщуватися в пакеті Група.Прізвище.Lab6 та володіти коментарями, які дозволять автоматично згенерувати документацію до розробленого пакету.

2. Автоматично згенерувати документацію до розробленого пакету.
3. Скласти звіт про виконану роботу з приведенням тексту програми, результату її виконання та фрагменту згенерованої документації.
4. Дати відповідь на контрольні запитання.

## 26. Банка

### Виконання:

#### Лістинг програми:

```
public interface Item
{
    double getVolume();
    void PrintInfo();
}
public class Food implements Item
{
    private double volume;
    private String type;
    private String expirationDate;

    public Food(double volume, String type, String expirationDate) {
        this.volume = volume;
        this.type = type;
        this.expirationDate = expirationDate;
    }
}
```

```

    public void setVolume(double volume) {
        this.volume = volume;
    }

    public String getType() {
        return type;
    }

    public void setType(String type) {
        this.type = type;
    }

    public String getExpirationDate() {
        return expirationDate;
    }

    public void setExpirationDate(String expirationDate) {
        this.expirationDate = expirationDate;
    }

    @Override
    public double getVolume() {
        return volume;
    }

    @Override
    public void PrintInfo() {
        System.out.println("Food{ volume - " + volume + "; food type " + type +
"; expiration date - " + expirationDate + " }");
    }
}
public class Liquid implements Item
{
    private double volume;
    private String type;

    public Liquid(double volume, String type) {
        this.volume = volume;
        this.type = type;
    }

    public void setVolume(double volume) {
        this.volume = volume;
    }

    public String getType() {
        return type;
    }

    public void setType(String type) {
        this.type = type;
    }

    @Override
    public double getVolume() {
        return volume;
    }

    @Override

```

```

        public void PrintInfo() {
            System.out.println("Liquid{ volume = " + volume + "; liquid type " +
type + " }");
        }
    }
import java.util.ArrayList;
import java.util.List;

public class Jar<T extends Item>
{
    private final List<T> arr;

    public Jar()
    {
        arr = new ArrayList<T>();
    }

    public T FindMin()
    {
        if (arr.isEmpty())
        {
            return null;
        }
        else
        {
            T min = arr.get(0);
            for (int i = 1; i < arr.size(); i++)
            {
                if (arr.get(i).getVolume() < min.getVolume())
                {
                    min = arr.get(i);
                }
            }
            return min;
        }
    }

    public void AddItem(T item)
    {
        arr.add(item);
        var index = arr.indexOf(item);
        System.out.print("Element was added: index = " + index + "; ");
        item.PrintInfo();
    }

    public void DeleteItem(int i)
    {
        System.out.print("Element on index " + i + " was deleted: ");
        arr.get(i).PrintInfo();
        arr.remove(i);
    }

    public void PrintAllItems()
    {
        for(T el : arr)
        {
            el.PrintInfo();
        }
    }
}

```

```

    }
}
public class Main
{
    public static void main(String[] args)
    {
        Jar<? super Item> jar = new Jar<>();

        jar.AddItem(new Liquid(56.7, "Jucy"));
        jar.AddItem(new Liquid(12, "Coke"));
        jar.AddItem(new Food(521, "Apple", "10-10-2023"));
        jar.AddItem(new Food(23, "Burger", "10-09-2022"));
        jar.AddItem(new Liquid(433, "Sprite"));
        jar.AddItem(new Food(43, "Meat", "12-12-2022"));
        jar.AddItem(new Liquid(5435, "Fanta"));
        jar.AddItem(new Food(511, "Eggs", "10-11-2022"));

        System.out.println("Min element is: " );
        jar.FindMin().PrintInfo();
    }
}

```

## Результати:

```

Element was added: index - 0; Liquid{ volume - 56.7; liquid type Jucy }
Element was added: index - 1; Liquid{ volume - 12.0; liquid type Coke }
Element was added: index - 2; Food{ volume - 521.0; food type Apple; expiration date - 10-10-2023 }
Element was added: index - 3; Food{ volume - 23.0; food type Burger; expiration date - 10-09-2022 }
Element was added: index - 4; Liquid{ volume - 433.0; liquid type Sprite }
Element was added: index - 5; Food{ volume - 43.0; food type Meat; expiration date - 12-12-2022 }
Element was added: index - 6; Liquid{ volume - 5435.0; liquid type Fanta }
Element was added: index - 7; Food{ volume - 511.0; food type Eggs; expiration date - 10-11-2022 }
Min element is:
Liquid{ volume - 12.0; liquid type Coke }

Process finished with exit code 0
|

```

**Висновок:** оволодів навиками параметризованого програмування мовою Java.