

ЗВІТ

до лабораторної роботи № 2

На тему: *“Робота з колекціями в Java”*

З дисципліни: *“Кросплатформне програмування”*

Лектор:

доцент каф. ПЗ
Дяконюк Л. М.

Виконав:

ст. гр. ПЗ-35
Сточанський М.А.

Прийняв:

асист. каф. ПЗ
Шкраб Р.Р.

« ____ » _____ 2020 р.

Σ= ____

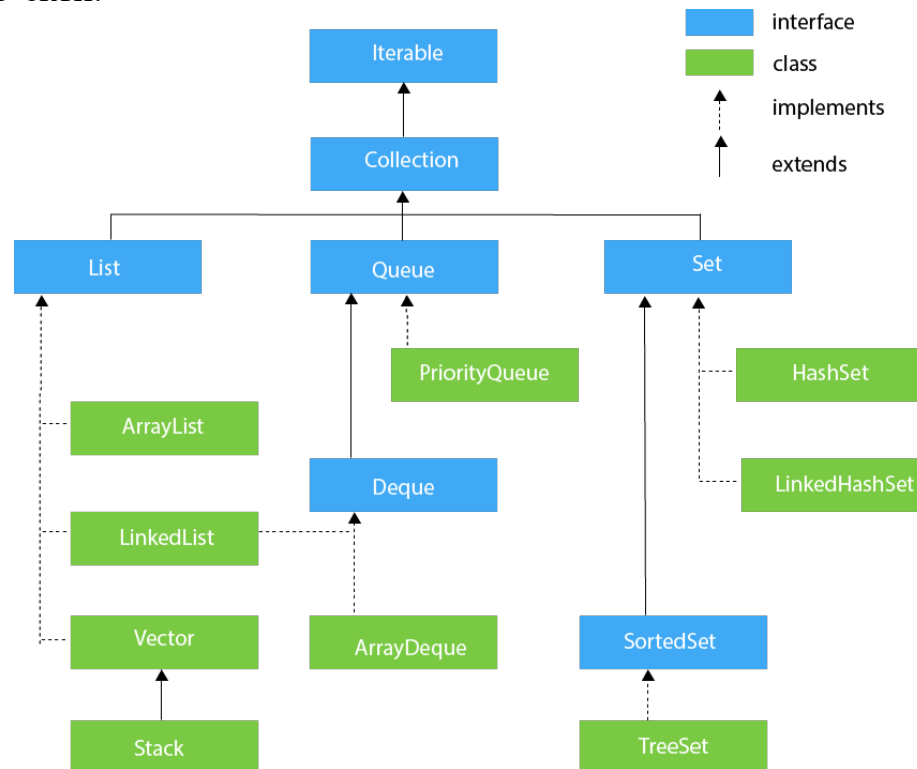
Тема роботи: Робота з колекціями в Java.

Мета роботи: Освоїти навички роботи з колекціями мови програмування Java.

Теоретичні відомості

Колекції в мові програмування Java поділяються на 3 основні типи:

- List – список впорядкованих елементів та може зберігати дублікати.
- Queue – список, який працює за принципом FIFO (first-in-first-out).
- Set – невпорядкований набір даних, який тримає тільки унікальні об'єкти.



Індивідуальне завдання

Задано інформацію про продукти, яка включає назву, дату виготовлення, кінцеву дату споживання та ціну.

1. Здійснити переоцінку всіх продуктів на 10 відсотків, кінцева дата споживання яких є через 3 дні від біжучої дати. Видрукувати таблицю, в якій в першій колонці розташовані кінцеві дати, в другій – колекція всіх продуктів з заданою кінцевою датою.
2. Вилучити з попередньої таблиці всі продукти з заданою назвою.
3. З 2 різних файлів зчитати 2 вихідні набори інформації про продукти. Визначити спільну колекцію унікальних за назвою продуктів, які є тільки в першому, або тільки в другому наборі, вилучивши з неї елементи дата виготовлення яких є не в біжучому році.

Код програми

Executor.java

```
package execution;

import data.dateFormatter.DateFormatter;
import data.products.ProductsRepository;
import data.products.ProductsRepositoryImpl;
import data.products.models.Product;

import java.util.*;
import java.util.stream.Collectors;
import java.util.stream.Stream;

public class Executor {

    private final ProductsRepository repository = new ProductsRepositoryImpl();

    private final DateFormatter datesFormatter = DateFormatter.getInstance();

    private final Printer printer = Printer.getInstance();
    private final AppScanner scanner = new AppScanner();

    public void execute() {
        printer.printStep("First step");
        Map<Date, List<Product>> map = performFirstStep();
        printer.printMap(map);

        printer.printStep("Second step");
        performSecondStep(map);
        printer.printMap(map);

        printer.printStep("Third step");
        List<Product> products = performThirdStep();
        printer.printList(products, "Combined and filtered");
    }

    private Map<Date, List<Product>> performFirstStep() {
        List<Product> products = repository.getProductsFromFirstFile();

        products.forEach(product -> {
            Long currentTime = System.currentTimeMillis();
            if (datesFormatter.daysBetweenTimestamps(product.getEndDate().getTime(), currentTime) < 3) {
                product.setPrice((float) (product.getPrice() - product.getPrice() * 0.1));
            }
        });

        return products.stream().collect(Collectors.groupingBy(Product::getEndDate));
    }

    private void performSecondStep(Map<Date, List<Product>> input) {

        printer.println("Please enter product name to delete");

        String nameToDelete = scanner.getLine();

        input.forEach((date, products) -> products.removeIf(product ->
product.getName().equals(nameToDelete)));

        List<Date> keysToRemove = input.entrySet().stream()
            .filter(entry -> entry.getValue().isEmpty())
            .map(Map.Entry::getKey)
            .collect(Collectors.toList());

        keysToRemove.forEach(input::remove);
    }

    private List<Product> performThirdStep() {
        List<Product> firstList = repository.getProductsFromFirstFile();
        List<Product> secondList = repository.getProductsFromSecondFile();

        printer.printList(firstList, "First list");
        printer.printList(secondList, "Second list");

        Calendar calendar = Calendar.getInstance();
        int currentYear = calendar.get(Calendar.YEAR);

        return Stream.concat(firstList.stream(), secondList.stream())
    }
}
```

```

        .filter(item -> !(listContainsItem(firstList, item) && listContainsItem(secondList, item)))
        .filter(item -> {
            calendar.setTime(item.getManufacturedDate());
            return calendar.get(Calendar.YEAR) == currentYear;
        })
        .collect(Collectors.toList());
    }
}
private boolean listContainsItem(List<Product> list, Product product) {
    return list.stream().anyMatch(item -> item.compareTo(product) == 0);
}
}

```

Протокол виконання

```

First step

13.12.2020 :::: name='Кааа', manufacturedDate=13.12.2018, endDate=13.12.2020, price=50.11 \\\
03.03.2021 :::: name='Кетчун', manufacturedDate=11.06.2020, endDate=03.03.2021, price=21.3 \\\
14.06.2021 :::: name='Перец_мелений', manufacturedDate=14.05.2019, endDate=14.06.2021, price=5.12 \\\
25.10.2021 :::: name='Чай', manufacturedDate=25.03.2019, endDate=25.10.2021, price=40.54 \\\
05.05.2023 :::: name='Лавровий_лист', manufacturedDate=29.08.2020, endDate=05.05.2023, price=8.11 \\\
24.10.2020 :::: name='Майонез', manufacturedDate=11.02.2020, endDate=24.10.2020, price=30.869999 \\\ name='Масно', manufacturedDate=24.07.2020, endDate=24.10.2020, price=40.788 \\\
25.10.2020 :::: name='Магарин', manufacturedDate=23.07.2020, endDate=25.10.2020, price=21.105001 \\\ name='Хніб_6інні', manufacturedDate=21.10.0020, endDate=25.10.2020, price=19.917 \\\

Second step

Please enter product name to delete
Кетчун

13.12.2020 :::: name='Кааа', manufacturedDate=13.12.2018, endDate=13.12.2020, price=50.11 \\\
14.06.2021 :::: name='Перец_мелений', manufacturedDate=14.05.2019, endDate=14.06.2021, price=5.12 \\\
25.10.2021 :::: name='Чай', manufacturedDate=25.03.2019, endDate=25.10.2021, price=40.54 \\\
05.05.2023 :::: name='Лавровий_лист', manufacturedDate=29.08.2020, endDate=05.05.2023, price=8.11 \\\
24.10.2020 :::: name='Майонез', manufacturedDate=11.02.2020, endDate=24.10.2020, price=30.869999 \\\ name='Масно', manufacturedDate=24.07.2020, endDate=24.10.2020, price=40.788 \\\
25.10.2020 :::: name='Магарин', manufacturedDate=23.07.2020, endDate=25.10.2020, price=21.105001 \\\ name='Хніб_6інні', manufacturedDate=21.10.0020, endDate=25.10.2020, price=19.917 \\\

Third step

FIRST LIST
name='Кетчун', manufacturedDate=11.06.2020, endDate=03.03.2021, price=21.3
name='Чай', manufacturedDate=25.03.2019, endDate=25.10.2021, price=40.54
name='Кааа', manufacturedDate=13.12.2018, endDate=13.12.2020, price=50.11
name='Майонез', manufacturedDate=11.02.2020, endDate=24.10.2020, price=34.3
name='Перец_мелений', manufacturedDate=14.05.2019, endDate=14.06.2021, price=5.12
name='Лавровий_лист', manufacturedDate=29.08.2020, endDate=05.05.2023, price=8.11
name='Масно', manufacturedDate=24.07.2020, endDate=24.10.2020, price=45.32
name='Магарин', manufacturedDate=23.07.2020, endDate=25.10.2020, price=23.45
name='Хніб_6інний', manufacturedDate=21.10.0020, endDate=25.10.2020, price=22.13
SECOND LIST
name='Вафлі', manufacturedDate=11.03.2019, endDate=01.11.2020, price=15.3
name='Печиво', manufacturedDate=01.08.2019, endDate=31.10.2020, price=25.3
name='Шоколад', manufacturedDate=04.07.2020, endDate=04.12.2020, price=23.19
name='Хніб_6інний', manufacturedDate=18.10.2020, endDate=25.10.2020, price=21.12
name='Чай', manufacturedDate=12.12.2018, endDate=12.12.2020, price=31.12
COMBINED AND FILTERED
name='Кетчун', manufacturedDate=11.06.2020, endDate=03.03.2021, price=21.3
name='Майонез', manufacturedDate=11.02.2020, endDate=24.10.2020, price=34.3
name='Лавровий_лист', manufacturedDate=29.08.2020, endDate=05.05.2023, price=8.11
name='Масно', manufacturedDate=24.07.2020, endDate=24.10.2020, price=45.32
name='Магарин', manufacturedDate=23.07.2020, endDate=25.10.2020, price=23.45
name='Шоколад', manufacturedDate=04.07.2020, endDate=04.12.2020, price=23.19

Process finished with exit code 0

```

Висновок

У результаті виконання лабораторної роботи було розроблено програму, яка опрацьовує різні колекції мови Java, а саме перетворює список в мапу, видаляє задані елементи з мапи та об'єднує 2 списки різних продуктів, виключаючи ті, які повторюються та фільтруючи результат.