

Лабораторна робота №8

Тема: Лямбда вирази. Функціональні інтерфейси. Посилання на методи. Stream API.

Мета роботи: практика роботи з лямбда виразами, функціональними інтерфейсами; використання посилань на методи та Stream API при розробці програм на Java.

Хід роботи:

Завдання 1. Створити консольний Java проект java_lab_8 з пакетом com.education.ztu.

Завдання 2. Описати власний функціональний інтерфейс Printable з методом void print() та написати лямбда вираз цього інтерфейсу.

Лістинг програми:

Main.java:

```
package com.education.ztu.Task2;

public class Main {
    public static void main(String[] args) {
        Printable printable = () -> System.out.println("This is a printable message!");
        printable.print();
    }
}
```

Printable.interface:

```
package com.education.ztu.Task2;

@FunctionalInterface
public interface Printable {
    void print();
}
```

Результат виконання програми:

This is a printable message!

Рис.1 Завдання 2

| | | | | | | | | |
|-----------|------|---------------------|--------|------|--|-------------------|------|---------|
| | | | | | ДУ «Житомирська політехніка».22.121.16.000 – Лр8 | | | |
| Змн. | Арк. | № докум. | Підпис | Дата | | | | |
| Розроб. | | Кохан Т.О | | | Звіт з лабораторної роботи №8 | Лім. | Арк. | Аркушів |
| Перевір. | | Піонтківський В. І. | | | | | 1 | 8 |
| Керівник | | | | | | ФІКТ Гр. ІПЗ-22-3 | | |
| Н. контр. | | | | | | | | |
| Зав. каф. | | Вакалюк Т.А. | | | | | | |

Завдання 3. Написати лямбда вирази для вбудованих функціональних інтерфейсів:

а) Створити лямбда вираз, який повертає значення true, якщо рядок можна привести до числа, використовуючи функціональний інтерфейс Predicate. Створити вираз лямбда, який перевіряє, що рядок можна привести до числа, використовуючи функціональний інтерфейс Predicate. Написати програму, яка перевіряє, що рядок можна привести до числа, використовуючи метод and() функціонального інтерфейсу Predicate.

б) Написати лямбда вираз, який приймає на вхід рядок і виводить на консоль повідомлення "Пара почалася о 8:30", "Пара закінчилася о 9:50". Використовуємо функціональний інтерфейс Consumer і метод за замовчуванням andThen.

в) Написати лямбда вираз, який виводить в консоль речення в з літерами у верхньому регістрі. Використовуємо функціональний інтерфейс Supplier.

г) Написати лямбда вираз, який приймає на вхід рядок з набором чисел через пробіл та повертає добуток цих чисел. Використовуємо функціональний інтерфейс Function.

Лістинг програми:

Main.java:

```
package com.education.ztu.Task3;

import java.util.function.Consumer;
import java.util.function.Function;
import java.util.function.Predicate;
import java.util.function.Supplier;

public class Main {
    public static void main(String[] args) {
        Predicate<String> isNumber = s -> {
            try {
                Integer.parseInt(s);
                return true;
            } catch (NumberFormatException e) {
                return false;
            }
        };

        System.out.println(isNumber.test("123"));
        System.out.println(isNumber.test("abc"));

        Predicate<String> isNumberAndPositive = isNumber.and(s ->
            Integer.parseInt(s) > 0);
        System.out.println(isNumberAndPositive.test("123"));
        System.out.println(isNumberAndPositive.test("-123"));
    }
}
```

Кохан Т.О.

Піонтківський В. І.

ДУ «Житомирська політехніка».22.121.16.000 – Лр8

Арк.

2

Змн.

Арк.

№ докум.

Підпис

Дата

```

Consumer<String> startLesson = s -> System.out.println("Пара почалася о 8:30");
Consumer<String> endLesson = s -> System.out.println("Пара закінчилася о 9:50");

startLesson.andThen(endLesson).accept("Розклад");
Supplier<String> sentence = () -> "вчимо java".toUpperCase();
System.out.println(sentence.get());

Function<String, Integer> productOfNumbers = s -> {
    String[] numbers = s.split(" ");
    int product = 1;
    for (String num : numbers) {
        product *= Integer.parseInt(num);
    }
    return product;
};
System.out.println(productOfNumbers.apply("1 2 3 4"));
}

```

Результат виконання програми:

```

true
false
true
false
Пара почалася о 8:30
Пара закінчилася о 9:50
ВЧИМО JAVA
24

```

Рис.2 Завдання 3

Завдання 4. Stream API.

- Створити стрім з масиву Product з полями name, brand, price, count.
- Отримати всі бренди та вивести в консоль. (map)
- Отримати 2 товари ціна яких менше тисячі. (filter, limit)
- Отримати суму всіх видів товарів, що є на складі. (reduce)
- Згрупувати товари по бренду (Collectors.groupingBy())
- Відсортувати товари за зростанням ціни та повернути масив (sorted, Collectors)
- За бажанням дописати функціонал, що використовує інші методи стрімів.

Лістинг програми:

Main.java:

| | | | | | | |
|------|------|---------------------|--------|------|--|------|
| | | Кохан Т.О. | | | ДУ «Житомирська політехніка».22.121.16.000 – Лр8 | Арк. |
| | | Піонтківський В. І. | | | | 3 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

```

package com.education.ztu.Task4;

import java.util.Arrays;
import java.util.Comparator;
import java.util.List;
import java.util.Map;
import java.util.stream.Collectors;

public class Main {
    public static void main(String[] args) {
        List<Product> products = Arrays.asList(
            new Product("TV", "LG", 15000.00, 20),
            new Product("Mobile", "Samsung", 7300.00, 100),
            new Product("MacBook", "Apple", 35999.00, 10),
            new Product("Headphones", "Samsung", 500.00, 50),
            new Product("Phone case", "Apple", 925.00, 300)
        );

        System.out.println("All brands:");
        products.stream()
            .map(Product::getBrand)
            .distinct()
            .forEach(System.out::println);

        System.out.println("\nProducts with price less than 1000 (limited to 2):");
        products.stream()
            .filter(product -> product.getPrice() < 1000)
            .limit(2)
            .forEach(System.out::println);

        int totalCount = products.stream()
            .mapToInt(Product::getCount)
            .reduce(0, Integer::sum);
        System.out.println("\nTotal count of products: " + totalCount);

        System.out.println("\nProducts grouped by brand:");
        Map<String, List<Product>> groupedByBrand = products.stream()
            .collect(Collectors.groupingBy(Product::getBrand));
        groupedByBrand.forEach((brand, productList) -> {
            System.out.println(brand + ": " + productList);
        });

        System.out.println("\nProducts sorted by price:");
        List<Product> sortedProducts = products.stream()
            .sorted(Comparator.comparingDouble(Product::getPrice))
            .collect(Collectors.toList());
        sortedProducts.forEach(System.out::println);

        System.out.println("\nProducts with price between 500 and 15000:");
        products.stream()
            .filter(product -> product.getPrice() >= 500 &&
product.getPrice() <= 15000)
            .forEach(System.out::println);
    }
}

```

Product.java;

```

package com.education.ztu.Task4;

public class Product {
    String name;
    String brand;
    double price;
}

```

Кохан Т.О.

Піонтківський В. І.

Змн.

Арк.

№ докум.

Підпис

Дата

ДУ «Житомирська політехніка».22.121.16.000 – Лр8

Арк.

4

```

int count;

public Product(String name, String brand, double price, int count) {
    this.name = name;
    this.brand = brand;
    this.price = price;
    this.count = count;
}

public String getName() {
    return name;
}

public String getBrand() {
    return brand;
}

public double getPrice() {
    return price;
}

public int getCount() {
    return count;
}

@Override
public String toString() {
    return "Product{name='" + name + "', brand='" + brand + "', price=" +
price + ", count=" + count + '}';
}
}

```

Результат виконання програми:

```

All brands:
LG
Samsung
Apple

Products with price less than 1000 (limited to 2):
Product{name='Headphones', brand='Samsung', price=500.0, count=50}
Product{name='Phone case', brand='Apple', price=925.0, count=300}

Total count of products: 480

Products grouped by brand:
Apple: [Product{name='MacBook', brand='Apple', price=35999.0, count=10}, Product{name='Phone case', brand='Apple', price=925.0, count=300}]
Samsung: [Product{name='Mobile', brand='Samsung', price=7300.0, count=100}, Product{name='Headphones', brand='Samsung', price=500.0, count=50}]
LG: [Product{name='TV', brand='LG', price=15000.0, count=20}]

```

Products sorted by price:

```

Product{name='Headphones', brand='Samsung', price=500.0, count=50}
Product{name='Phone case', brand='Apple', price=925.0, count=300}
Product{name='Mobile', brand='Samsung', price=7300.0, count=100}
Product{name='TV', brand='LG', price=15000.0, count=20}
Product{name='MacBook', brand='Apple', price=35999.0, count=10}

```

Products with price between 500 and 15000:

```

Product{name='TV', brand='LG', price=15000.0, count=20}
Product{name='Mobile', brand='Samsung', price=7300.0, count=100}
Product{name='Headphones', brand='Samsung', price=500.0, count=50}
Product{name='Phone case', brand='Apple', price=925.0, count=300}

```

Рис.3 Завдання 4

| | | | | | | |
|------|------|---------------------|--------|------|--|------|
| | | Кохан Т.О. | | | ДУ «Житомирська політехніка».22.121.16.000 – Лр8 | Арк. |
| | | Піонтківський В. І. | | | | 5 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

Завдання 5. Посилання на методи чи конструктори. В попередньому завданні, де це можливо, виклики переробити на посилання на методи чи конструктори.

Лістинг програми:

Main.java:

```
package com.education.ztu.Task5;

import com.education.ztu.Task4.Product;

import java.util.Arrays;
import java.util.Comparator;
import java.util.List;
import java.util.Map;
import java.util.stream.Collectors;

public class Main {
    public static void main(String[] args) {
        List<com.education.ztu.Task4.Product> products = Arrays.asList(
            new com.education.ztu.Task4.Product("TV", "LG", 15000.00, 20),
            new com.education.ztu.Task4.Product("Mobile", "Samsung",
7300.00, 100),
            new com.education.ztu.Task4.Product("MacBook", "Apple",
35999.00, 10),
            new com.education.ztu.Task4.Product("Headphones", "Samsung",
500.00, 50),
            new Product("Phone case", "Apple", 925.00, 300)
        );

        System.out.println("All brands:");
        products.stream()
            .map(Product::getBrand)
            .distinct()
            .forEach(System.out::println);

        System.out.println("\nProducts with price less than 1000 (limited to
2):");
        products.stream()
            .filter(product -> product.getPrice() < 1000)
            .limit(2)
            .forEach(System.out::println);

        int totalCount = products.stream()
            .mapToInt(Product::getCount)
            .reduce(0, Integer::sum);
        System.out.println("\nTotal count of products: " + totalCount);

        System.out.println("\nProducts grouped by brand:");
        Map<String, List<Product>> groupedByBrand = products.stream()
            .collect(Collectors.groupingBy(Product::getBrand));
        groupedByBrand.forEach((brand, productList) -> {
            System.out.println(brand + ": " + productList);
        });

        System.out.println("\nProducts sorted by price:");
        List<Product> sortedProducts = products.stream()
            .sorted(Comparator.comparingDouble(Product::getPrice))
            .collect(Collectors.toList());
    }
}
```

Кохан Т.О.

Піонтківський В. І.

Змн.

Арк.

№ докум.

Підпис

Дата

ДУ «Житомирська політехніка».22.121.16.000 – Лр8

Арк.

6

```

        sortedProducts.forEach(System.out::println);

        System.out.println("\nProducts with price between 500 and 15000:");
        products.stream()
                .filter(product -> product.getPrice() >= 500 &&
product.getPrice() <= 15000)
                .forEach(System.out::println);
    }
}

```

Product.java:

```

package com.education.ztu.Task5;

public class Product {
    String name;
    String brand;
    double price;
    int count;

    public Product(String name, String brand, double price, int count) {
        this.name = name;
        this.brand = brand;
        this.price = price;
        this.count = count;
    }

    public String getName() {
        return name;
    }

    public String getBrand() {
        return brand;
    }

    public double getPrice() {
        return price;
    }

    public int getCount() {
        return count;
    }

    @Override
    public String toString() {
        return "Product{name='" + name + "', brand='" + brand + "', price=" +
price + ", count=" + count + "'}";
    }
}

```

Результат виконання програми:

```

All brands:
LG
Samsung
Apple

Products with price less than 1000 (limited to 2):
Product{name='Headphones', brand='Samsung', price=500.0, count=50}
Product{name='Phone case', brand='Apple', price=925.0, count=300}

Total count of products: 480

Products grouped by brand:
Apple: [Product{name='MacBook', brand='Apple', price=35999.0, count=10}, Product{name='Phone case', brand='Apple', price=925.0, count=300}]
Samsung: [Product{name='Mobile', brand='Samsung', price=7300.0, count=100}, Product{name='Headphones', brand='Samsung', price=500.0, count=50}]
LG: [Product{name='TV', brand='LG', price=15000.0, count=20}]

```

| | | | | | | |
|------|------|---------------------|--------|------|--|------|
| | | Кохан Т.О. | | | ДУ «Житомирська політехніка».22.121.16.000 – Лр8 | Арк. |
| | | Піонтківський В. І. | | | | 7 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

```

Products sorted by price:
Product{name='Headphones', brand='Samsung', price=500.0, count=50}
Product{name='Phone case', brand='Apple', price=925.0, count=300}
Product{name='Mobile', brand='Samsung', price=7300.0, count=100}
Product{name='TV', brand='LG', price=15000.0, count=20}
Product{name='MacBook', brand='Apple', price=35999.0, count=10}

Products with price between 500 and 15000:
Product{name='TV', brand='LG', price=15000.0, count=20}
Product{name='Mobile', brand='Samsung', price=7300.0, count=100}
Product{name='Headphones', brand='Samsung', price=500.0, count=50}
Product{name='Phone case', brand='Apple', price=925.0, count=300}

```

Рис.4 Завдання 5

Завдання 6. Використання Optional та його методів. Знайти максимальне значення з масиву чисел, в іншому випадку повернути рядок «Числа відсутні».

Лістинг програми:

Main.java:

```

package com.education.ztu.Task6;

import java.util.Arrays;
import java.util.Optional;

public class Main {
    public static void main(String[] args) {
        Integer[] numbers = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

        Optional<Integer> maxNumber = Arrays.stream(numbers)
            .max(Integer::compareTo);

        String result = maxNumber
            .map(String::valueOf)
            .orElse("Числа відсутні");

        System.out.println(result);
    }
}

```

Результат виконання програми:

```

10

```

Рис.5 Завдання 6

Висновок: під час виконання лабораторної роботи я попрактикувала роботу з лямбда виразами, функціональними інтерфейсами; навчилася використовувати посилання на методи та Stream API при розробці програм на Java.

| | | | | | | |
|------|------|---------------------|--------|------|--|------|
| | | Кохан Т.О. | | | ДУ «Житомирська політехніка».22.121.16.000 – Лр8 | Арк. |
| | | Піонтківський В. І. | | | | 8 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |