

МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ
КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени

федеральное государственное бюджетное образовательное учреждение
высшего образования

МОСКОВСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ СВЯЗИ И
ИНФОРМАТИКИ

Кафедра «Математическая кибернетика и информационные технологии»

Отчёт по лабораторной работе № 6

по дисциплине «Информационные технологии и программирование»

Выполнил: студент группы БПИ2404 Журавлев Р. В.

Руководитель: Рыбаков Е. Д.

Москва 2025

Цель работы: Изучить работу с коллекциями и обобщениями (Generics) в языке программирования Java. Применить их на практике для закрепления навыков.

Задание 1. Написать программу, которая считывает текстовый файл и выводит на экран топ-10 самых часто встречающихся слов в этом файле. Для решения задачи использовать коллекцию Map, где ключом будет слово, а значением — количество его повторений в файле.

```
import java.io.File;
import java.io.FileNotFoundException;
import java.util.*;

public class TopWords {
    public static void main(String[] args) {

        File file = new File("text.txt");

        Scanner scanner = null;
        try {
            scanner = new Scanner(file);
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        }

        Map<String, Integer> map = new HashMap<>();

        while (scanner.hasNext()) {
            String word = scanner.next().toLowerCase();
            word = word.replaceAll("[^а-я-я]", "");

            if (!word.isEmpty()) {
                map.put(word, map.getOrDefault(word, 0) + 1);
            }
        }

        scanner.close();

        List<Map.Entry<String, Integer>> list = new ArrayList<>(map.entrySet());

        Collections.sort(list, new Comparator<Map.Entry<String, Integer>>() {
            @Override
            public int compare(Map.Entry<String, Integer> o1, Map.Entry<String, Integer> o2) {
                return o2.getValue().compareTo(o1.getValue());
            }
        });

        for (int i = 0; i < 10; i++) {
            Map.Entry<String, Integer> entry = list.get(i);
            System.out.println(entry.getKey() + ": " + entry.getValue());
        }
    }
}
```

Задание 2. Написать обобщенный класс Stack<T>, который реализует стек на основе массива. Класс должен иметь методы push для добавления элемента в стек, pop для удаления элемента из стека и peek для получения верхнего элемента стека без его удаления.

```
class Stack<T> {
    private T[] data;
    private int size;

    public Stack(int capacity) {
        data = (T[]) new Object[capacity];
        size = 0;
    }

    public void push(T element) {
        try {
            data[size] = element;
            size++;
        } catch (ArrayIndexOutOfBoundsException e) {
            System.out.println("Переполнение стека");
        }
    }

    public T pop() {
        if (size > 0) {
            T element = data[size - 1];

            data[size - 1] = null;
            size--;

            return element;
        } else {
            return null;
        }
    }

    public T peek() {
        if (size > 0) {
            return data[size - 1];
        } else {
            return null;
        }
    }
}

class Main {
    public static void main(String[] args) {
        Stack<Integer> stack = new Stack<> (3);
```

```
System.out.println(stack.pop());
System.out.println(stack.peek());

stack.push(16);
stack.push(2);
stack.push(7);

System.out.println(stack.pop());
System.out.println(stack.peek());

stack.push(4);

System.out.println(stack.peek());
}

}

}
```

Задание 3. Разработать программу для учета продаж в магазине. Программа должна позволять добавлять проданные товары в коллекцию, выводить список проданных товаров, а также считать общую сумму продаж и наиболее популярный товар. Использовать ArrayList для хранения списка проданных товаров. Использовать HashMap для хранения пар «товар – количество продаж».

```
import java.util.*;  
  
// Товар  
class Product {  
  
    private String name;  
    private int price;  
  
    public Product(String name, int price) {  
        this.name = name;  
        this.price = price;  
    }  
  
    public String getName() {  
        return name;  
    }  
  
    public int getPrice() {  
        return price;  
    }  
  
    @Override  
    public String toString() {  
        return name + " +" + price + "руб.";  
    }  
}  
  
class SalesManager {  
  
    private ArrayList<Product> soldProducts;  
  
    public SalesManager() {  
        soldProducts = new ArrayList<>();  
    }  
  
    // Добавление проданного товара  
    public void addSoldProduct(Product product) {  
        soldProducts.add(product);  
    }  
  
    // Вывод проданных товаров
```

```
public void printSoldProducts() {
    for (int i = 0; i < soldProducts.size(); i++) {
        System.out.println((i + 1) + ". " + soldProducts.get(i));
    }
}

// Общая сумма продаж
public int calculateTotalSales() {
    int total = 0;
    for (Product product : soldProducts) {
        total += product.getPrice();
    }
    return total;
}

// Наиболее популярный товар
public String findMostPopularProduct() {

    HashMap<String, Integer> productCount = new HashMap<>();

    for (Product product : soldProducts) {

        String productName = product.getName();

        if (productCount.containsKey(productName)) {
            productCount.put(productName, productCount.get(productName) + 1);
        } else {
            productCount.put(productName, 1);
        }
    }

    String mostPopular = "";
    int maxCount = 0;

    for (Map.Entry<String, Integer> entry : productCount.entrySet()) {
        if (entry.getValue() > maxCount) {
            maxCount = entry.getValue();
            mostPopular = entry.getKey();
        }
    }

    return mostPopular + " (" + maxCount + " продано)";
}

public static void main(String[] args) {

    SalesManager sales = new SalesManager();

    // Создание товаров
    Product milk = new Product("Молоко", 129);
    Product bread = new Product("Хлеб", 68);
```

```
Product cheese = new Product("Сыр", 279);

// Продажа товаров
sales.addSoldProduct(milk);
sales.addSoldProduct(bread);
sales.addSoldProduct(cheese);
sales.addSoldProduct(bread);

// Вывод проданных товаров
System.out.println("\nПроданные товары: ");
sales.printSoldProducts();

// Вывод суммы продаж
System.out.println("\nОбщая сумма продаж: " + sales.calculateTotalSales());

// Вывод самого популярного товара
System.out.println("\nСамый популярный товар: " +
sales.findMostPopularProduct());
}
```

Вывод: В ходе лабораторной работы были изучены и применены коллекции и обобщения в языке программирования Java.