

**Министерство образования Республики Беларусь**  
**Учреждение образования**  
**«Брестский государственный технический университет»**  
**Кафедра ИИТ**

**Лабораторная работа №3**

По дисциплине: «ССП»

Вариант 20

**Выполнил:** Студент

3 курса

Группы ПО-8

Соколов С.Д.

**Проверил:**

Крощенко А.А

**Брест 2024**

## Лабораторная работа №3

**Цель работы:** научиться создавать и использовать классы в программах на языке программирования Java.

**Задание 1:** Реализовать простой класс.

Требования к выполнению

- Реализовать пользовательский класс по варианту.
- Создать другой класс с методом `main`, в котором будут находиться примеры использования пользовательского класса.

Для каждого класса

- Создать поля классов
- Создать методы классов
- Добавьте необходимые `get` и `set` методы (по необходимости)
- Укажите соответствующие модификаторы видимости
- Добавьте конструкторы
- Переопределить методы `toString()` и `equals()`

Множество символов переменной мощности – Предусмотреть возможность пересечения двух множеств, вывода на печать элементов множества, а так же метод, определяющий, принадлежит ли указанное значение множеству. Класс должен содержать методы, позволяющие добавлять и удалять элемент в/из множества. Конструктор должен позволить создавать объекты с начальной инициализацией. Реализацию множества осуществить на базе структуры `ArrayList`. Реализовать метод `equals`, выполняющий сравнение объектов данного типа.

**Код программы: `main.java`**

```
import java.util.ArrayList;
import java.util.Objects;

public class Main {
    public static void main(String[] args) {
        VariablePowerSet set1 = new VariablePowerSet();
        set1.addElement('a');
        set1.addElement('b');
        set1.addElement('c');

        VariablePowerSet set2 = new VariablePowerSet();
```

```

        set2.addElement('b');
        set2.addElement('c');
        set2.addElement('d');

        VariablePowerSet intersectionSet = set1.intersection(set2);

        System.out.println("Set 1: " + set1);
        System.out.println("Set 2: " + set2);
        System.out.println("Intersection: " + intersectionSet);
    }
}

```

## VariablePowerSet.java

```

import java.util.ArrayList;
import java.util.Objects;

public class VariablePowerSet {
    private ArrayList<Character> set;

    public VariablePowerSet() {
        this.set = new ArrayList<>();
    }

    public VariablePowerSet(ArrayList<Character> set) {
        this.set = new ArrayList<>(set);
    }

    public void addElement(Character element) {
        if (!this.set.contains(element)) {
            this.set.add(element);
        }
    }

    public void removeElement(Character element) {
        this.set.remove(element);
    }

    public boolean isElementBelong(Character element) {
        return this.set.contains(element);
    }

    public VariablePowerSet intersection(VariablePowerSet anotherSet) {
        VariablePowerSet intersectionSet = new VariablePowerSet();
        for (Character element : this.set) {
            if (anotherSet.isElementBelong(element)) {
                intersectionSet.addElement(element);
            }
        }
        return intersectionSet;
    }

    @Override
    public boolean equals(Object o) {
        if (this == o) return true;
        if (o == null || getClass() != o.getClass()) return false;
        VariablePowerSet that = (VariablePowerSet) o;
        return Objects.equals(set, that.set);
    }

    @Override
    public int hashCode() {
        return Objects.hash(set);
    }
}

```

```
@Override
public String toString() {
    return "VariablePowerSet{" +
        "set=" + set +
        '}';
}
```

### Результат работы программы:

```
C:\Users\semen\.jdk\openjdk-22.0.1\bin\java.exe
Set 1: VariablePowerSet{set=[a, b, c]}
Set 2: VariablePowerSet{set=[b, c, d]}
Intersection: VariablePowerSet{set=[b, c]}

Process finished with exit code 0
```

**Задание 2:** Разработать автоматизированную систему на основе некоторой структуры данных, манипулирующей объектами пользовательского класса. Реализовать требуемые функции обработки данных.

#### 10) Частотный словарь

Составить программу, которая формирует англо-русский словарь. Словарь должен содержать английское слово, русское слово и количество обращений к слову. Программа должна:

- обеспечить начальный ввод словаря (по алфавиту) с конкретными значениями счетчиков обращений;
- формирует новое дерево, в котором слова отсортированы не по алфавиту, а по количеству обращений.
- Реализовать возможность добавления новых слов, удаления существующих, поиска нужного слова, выполнять просмотр обоих вариантов словаря

## Код программы: Word.java

```
import java.util.*;

class Word {
    String english;
    String russian;
    int count;

    Word(String english, String russian, int count) {
        this.english = english;
        this.russian = russian;
        this.count = count;
    }
}

class Dictionary {
    List<Word> words = new ArrayList<>();

    void addWord(String english, String russian, int count) {
        words.add(new Word(english, russian, count));
    }

    void removeWord(String english) {
        words.removeIf(word -> word.english.equals(english));
    }

    Word findWord(String english) {
        for (Word word : words) {
            if (word.english.equals(english)) {
                return word;
            }
        }
        return null;
    }

    void printAlphabetical() {
        words.sort(Comparator.comparing(word -> word.english));
        for (Word word : words) {
            System.out.println(word.english + " - " + word.russian + " - " +
word.count);
        }
    }

    void printByCount() {
        words.sort(Comparator.comparing(word -> -word.count));
        for (Word word : words) {
            System.out.println(word.english + " - " + word.russian + " - " +
word.count);
        }
    }
}
```

## Main.java

```
public class Main {
    public static void main(String[] args) {
        Dictionary dictionary = new Dictionary();

        // Добавление слов
        dictionary.addWord("hello", "привет", 10);
        dictionary.addWord("world", "мир", 20);
        dictionary.addWord("java", "ява", 15);

        // Печать словаря в алфавитном порядке
        System.out.println("Словарь в алфавитном порядке:");
        dictionary.printAlphabetical();
    }
}
```

```

        // Печать словаря по количеству обращений
        System.out.println("\nСловарь по количеству обращений:");
        dictionary.printByCount();

        // Поиск слова
        Word foundWord = dictionary.findWord("hello");
        if (foundWord != null) {
            System.out.println("\nНайдено слово: " + foundWord.english + " - " +
foundWord.russian + " - " + foundWord.count);
        }

        // Удаление слова
        dictionary.removeWord("world");

        // Печать словаря в алфавитном порядке после удаления слова
        System.out.println("\nСловарь в алфавитном порядке после удаления слова:");
        dictionary.printAlphabetical();
    }
}

```

## Результаты работы программы:

```

C:\Users\semen\.jdk\openjdk-22.0.1\bin\java.exe "-javaagent:C:\Progr
Словарь в алфавитном порядке:
hello - привет - 10
java - ява - 15
world - мир - 20

Словарь по количеству обращений:
world - мир - 20
java - ява - 15
hello - привет - 10

Найдено слово: hello - привет - 10

Словарь в алфавитном порядке после удаления слова:
hello - привет - 10
java - ява - 15

Process finished with exit code 0

```

**Вывод:** научился создавать и использовать классы в программах на языке программирования Java.