

Отчет по лабораторной работе №1

Изучение структуры Java кода, областей видимости и пакетов

Выполнила: Андреяшкина Мария

Группа: 6204-010302D

Оглавление

Задание 1.....3

Задание 2.....3

Задание 3.....3

Задание 4.....4

Задание 5.....6

Задание 6.....7

Задание 1

Изучение параметров компилятора `javac` и виртуальной машины Java

Приступила к изучению инструментов разработки Java. Запустила компилятор `javac` без параметров для ознакомления с доступными опциями, затем аналогично изучила параметры виртуальной машины Java.

Выполненные команды:

```
javac  
java
```

Вывод программ:

Обе команды отображали подробную справку по формату использования и доступным параметрам, что позволило понять основные возможности инструментов.

Задание 2

Создание и запуск первой Java-программы

Разработала базовую Java-программу, начав с минимальной структуры класса. Столкнулась с необходимостью корректного объявления метода `main`.

Исходный код:

```
class MyFirstClass {  
    public static void main(String[] args) {  
        System.out.println("Hello world!!!");  
    }  
}
```

Процесс компиляции и запуска:

```
javac MyFirstProgram.java  
java MyFirstClass
```

Вывод программы:

```
Hello world!!!
```

Наблюдения: Изначальные попытки запуска без метода `main` или с некорректной его сигнатурой приводили к ошибкам, что демонстрирует строгие требования JVM к точке входа программы.

Задание 3

Работа с аргументами командной строки

Модифицировала программу для обработки и отображения аргументов командной строки.

Исходный код:

```
class MyFirstClass {  
    public static void main(String[] s) {  
        for (int i = 0; i < s.length; i++)  
            System.out.println(s[i]);  
    }  
}
```

Пример запуска:

```
java MyFirstClass Java Programming Laboratory 2024
```

Вывод программы:

```
Java  
Programming  
Laboratory  
2024
```

Результат: Программа успешно обработала и отобразила все переданные аргументы, демонстрируя работу с массивом строк как параметром метода main.

Задание 4

Реализация класса с инкапсуляцией данных

Создала второй класс с приватными полями и методами доступа, реализовав принципы инкапсуляции. Программа генерирует таблицу умножения.

Основной класс:

```
class MyFirstClass {  
    public static void main(String[] s) {  
        Calculator calc = new Calculator(0, 0);  
  
        for (int i = 1; i <= 8; i++) {  
            for (int j = 1; j <= 8; j++) {  
                calc.setValue1(i);  
                calc.setValue2(j);  
                System.out.print(calc.getProduct() + " ");  
            }  
            System.out.println();  
        }  
    }  
}
```

```
}
```

Класс Calculator:

```
class Calculator {  
    private int value1;  
    private int value2;  
  
    public Calculator(int a, int b) {  
        value1 = a;  
        value2 = b;  
    }  
  
    public void setValue1(int num) {  
        value1 = num;  
    }  
  
    public void setValue2(int num) {  
        value2 = num;  
    }  
  
    public int getProduct() {  
        return value1 * value2;  
    }  
}
```

Вывод программы:

```
1 2 3 4 5 6 7 8  
2 4 6 8 10 12 14 16  
3 6 9 12 15 18 21 24  
4 8 12 16 20 24 28 32  
5 10 15 20 25 30 35 40  
6 12 18 24 30 36 42 48  
7 14 21 28 35 42 49 56  
8 16 24 32 40 48 56 64
```

Задание 5

Организация кода в пакеты

Вынесла функциональность в отдельный пакет, изучив особенности работы с пространствами имен в Java.

Структура пакета:

myfirstpackage/

└─ Calculator.java

Код в пакете:

```
package myfirstpackage;

public class Calculator {
    private int value1;
    private int value2;

    public Calculator(int a, int b) {
        value1 = a;
        value2 = b;
    }

    public void setValue1(int num) {
        value1 = num;
    }

    public void setValue2(int num) {
        value2 = num;
    }

    public int getProduct() {
        return value1 * value2;
    }
}
```

Основной класс с импортом:

```
import myfirstpackage.*;
```

```

class MyFirstClass {
    public static void main(String[] s) {
        Calculator calc = new Calculator(0, 0);

        for (int i = 1; i <= 8; i++) {
            for (int j = 1; j <= 8; j++) {
                calc.setValue1(i);
                calc.setValue2(j);
                System.out.print(calc.getProduct() + " ");
            }
            System.out.println();
        }
    }
}

```

Вывод программы:

Таблица умножения 8×8 (аналогично заданию 4)

Особенности: Потребовалось корректное объявление пакета и использование модификатора public для класса, а также соблюдение соответствия имени файла и имени public-класса.

Задание 6

Создание исполняемого JAR-архива

Упаковала программу в JAR-архив с указанием главного класса в манифесте.

Манифест-файл:

Manifest-Version: 1.0

Created-By: Andrejashkina

Main-Class: MyFirstClass

Процесс создания архив:

```
jar cfm myfirst.jar manifest.mf MyFirstClass.class myfirstpackage
```

Запуск из архива:

```
java -jar myfirst.jar
```

Вывод программы:

Таблица умножения 8×8, успешно запущенная из JAR-архива

Результат: Программа корректно работает после перемещения в отдельную директорию, демонстрируя портативность Java-приложений.

Выводы

В ходе лабораторной работы успешно освоила:

- Компиляцию и выполнение Java-программ через командную строку
- Структуру Java-классов и требования к методу `main`
- Обработку аргументов командной строки
- Принципы ООП, включая инкапсуляцию данных
- Организацию кода в пакеты и работу с пространствами имен
- Создание исполняемых JAR-архивов с манифестами

Все программы работают корректно, выводя ожидаемые результаты. Особую ценность представлял опыт решения практических проблем, таких как настройка `PATH` для JDK, работа с кодировками файлов и правильное оформление пакетов.