

Лабораторная работа № 1

Выполнила: Рапп Ксения
Александровна
группа 6204-010302D

Оглавление

Задание 1	3
Задание 2	3
Задание 3	4
Задание 4	4-6
Задание 5	6-7
Задание 6	7-8

Задание 1

Я запустила компилятор `javac` без параметров и ознакомилась с форматом задания параметров компилятора. Затем запустила программу `java` без параметров и ознакомилась с форматом задания параметров запуска виртуальной машины Java (JVM).

Задание 2

Я создала файл `MyFirstProgram.java`, содержащий исходный код одного пустого класса с именем `MyFirstClass`:

`MyFirstProgram.java`

```
class MyFirstClass {  
}
```

Затем откомпилировала его с помощью команды `javac MyFirstProgram.java`. Потом запустила полученный файл класса на выполнение с помощью команды `java MyFirstClass`. В результате получила ошибку:

Error: Main method not found in class MyFirstClass, please define the main method as:

```
public static void main(String[] args)  
or a JavaFX application class must extend  
javafx.application.Application
```

Добавила в класс метод `main()`:

`MyFirstProgram.java`

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

Откомпилировала и запустила программу на выполнение (`javac MyFirstProgram.java` и `java MyFirstClass`). Я получила ошибку:

Error: Main method is not static in class MyFirstClass, please define the main method as:

```
public static void main(String[] args)
```

После данной ошибки сделала метод статическим, снова откомпилировала и запустила программу. В результате получила вывод в консоли:

```
Hello world!!!
```

Задание 3

Я заменила текст метода `main()` на следующий:
(*MyFirstClass.main*)

```
for (int i = 0; i < s.length; i++)  
    System.out.println(s[i]);
```

Откомпилировала и запустила программу, добавив в командную строку ряд аргументов:

```
javac MyFirstProgram.java ---> java MyFirstClass 35 11 66 99 76543
```

на экране командной строки появились аргументы в столбик:

```
35  
11  
66  
99  
76543
```

Задание 4

В том же файле `MyFirstProgram.java` после описания класса `MyFirstClass` добавила описание второго класса `MySecondClass`, реализующего следующую функциональность:

- имеет два приватных поля типа `int` (`firstNum`, `secondNum`);
- методы для получения и модификации их значений (`getFirstNum`, `getSecondNum`, `setFirstNum`, `setSecondNum`);
- конструктор, создающий объект и инициализирующий значения полей;
- метод с возвращаемым типом `int`, реализующий над этими числами умножение (`multiply`).

```
class MySecondClass {  
    // приватные поля  
    private int firstNum;  
    private int secondNum;  
  
    // конструктор  
    public MySecondClass(int first, int second) {  
        this.firstNum = first;  
        this.secondNum = second;  
    }  
}
```

```

    }

    // метод для получения первого числа
    public int getFirstNum() {
        return firstNum;
    }

    // метод для установки первого числа
    public void setFirstNum(int value) {
        this.firstNum = value;
    }

    // метод для получения второго числа
    public int getSecondNum() {
        return secondNum;
    }

    // метод для установки второго числа
    public void setSecondNum(int value) {
        this.secondNum = value;
    }

    // метод для умножения чисел
    public int multipl() {
        return firstNum * secondNum;
    }
}

```

Код метода `MyFirstClass.main()` при этом заменила на следующий:
`MyFirstClass.main`

```

class MyFirstClass {
    public static void main(String[] s) {
        MySecondClass o = new MySecondClass(0, 0); // создаем объект класса
        MySecondClass

        int i, j;
        for (i = 1; i <= 8; i++) {
            for (j = 1; j <= 8; j++) {
                o.setFirstNum(i);           // установка 1 числа
                o.setSecondNum(j);         // установка 2 числа
                System.out.print(o.multipl()); // умножаем и выводим
                System.out.print(" ");
            }
        }
    }
}

```

```

    }
    System.out.println(); // переход на новую строку
}
}

```

Откомпилировала и запустила программу. После ввода команды `java MyFirstClass`, на экране появилось следующее:

```

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

Я удалила все откомпилированные байт-коды классов. Вынесла код класса `MySecondClass` без изменений в отдельный файл с именем `MyFirstPackage.java`, и поместила его в поддиректорию `myfirstpackage`, откомпилировала командой `javac MyFirstPackage.java`. Затем откомпилировала файл `MyFirstProgram.java`. На экране появилась ошибка:

```

MyFirstProgram.java:4: error: cannot find symbol
    MySecondClass o = new MySecondClass(0, 0);
    ^
symbol:   class MySecondClass
location: class MyFirstClass
MyFirstProgram.java:4: error: cannot find symbol
    MySecondClass o = new MySecondClass(0, 0);
    ^
symbol:   class MySecondClass
location: class MyFirstClass
2 errors

```

Затем добавила в начало исходного кода в файле `MyFirstProgram.java` следующий код: `MyFirstProgram.java`
`import myfirstpackage.*;`

Снова откомпилировала `MyFirstProgram.java`, получила ошибку:

MyFirstProgram.java:4: error: cannot access MySecondClass

```
    MySecondClass o = new MySecondClass(0, 0);
```

^

bad class file: .\myfirstpackage\MySecondClass.class

class file contains wrong class: MySecondClass

Please remove or make sure it appears in the correct subdirectory of the classpath.

MyFirstProgram.java:4: error: cannot find symbol

```
    MySecondClass o = new MySecondClass(0, 0);
```

^

symbol: class MySecondClass

location: class MyFirstClass

2 errors

Далее я изменила код файла MyFirstPackage.java. Добавила в начало файла package myfirstpackage; а также сделала класс MySecondClass общедоступным, используя модификатор доступа public:

```
package myfirstpackage;
```

```
public class MySecondClass
```

Затем появилась ошибка:

MyFirstPackage.java:2: error: class MySecondClass is public, should be declared in a file named MySecondClass.java

```
public class MySecondClass {
```

^

1 error

После этого я переименовала название файла MyFirstPackage.java на название MySecondClass.java, так как имя файла должно точно совпадать с именем public-класса. Затем я снова откомпилировала данный файл, откомпилировала файл MyFirstProgram.java и запустила программу командой java MyFirstClass, все прошло успешно.

Задание 6

Я запустила программу jar, предназначенную для создания архивов, и ознакомилась с форматом задания ключей для формирования архивов. Скопировала в рабочую папку, сохранив структуру каталогов, только файлы с расширением class, полученные в результате выполнения задания 5.

Создала файл manifest.mf, содержащий следующий код:
manifest.mf

```
Manifest-Version: 1.0  
Created-By: <Rapp>  
Main-Class: MyFirstClass
```

С помощью команды "C:\Program Files\Java\jdk-25\bin\jar.exe" cfm myfirst.jar manifest.mf MyFirstClass.class myfirstpackage/MySecondClass.class я создала архив myfirst.jar, включив в него полученные ранее файлы классов и указав соданный мной манифест-файл. Затем переместила полученный файл в другую директорию (поддиректорию MyJar) и запустила его на выполнение с помощью команд `cd MyJar` → `java -jar myfirst.jar`:

```
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
```