

Лабораторная работа №1

Задание на лабораторную работу

В процессе написания тестовых заданий ознакомиться со структурой исходного кода для Java, изучить особенности областей видимости и использования пакетов.

Рекомендуется каждое следующее задание выполнять в новом каталоге (например, Task1, Task2 и так далее).

Лабораторная работа выполняется в командной строке.

Задание 1

Запустите компилятор `javac` без параметров и ознакомьтесь с форматом задания параметров компилятора.

Запустите программу `java` без параметров и ознакомьтесь с форматом задания параметров запуска виртуальной машины Java (JVM).

Задание 2

Создайте файл `MyFirstProgram.java`, содержащий исходный код одного класса с именем `MyFirstClass`, обеспечивающий вывод на экран вашей фамилии.

Откомпилируйте файл с помощью компилятора `javac`. Для этого необходимо выполнить команду `"javac MyFirstProgram.java"`. Запустите полученный файл класса на выполнение с помощью команды `"java <Имя класса>"`.

Обратите внимание на то, что на вход компилятора необходимо подавать имя файла с расширением, а на вход JVM – имя класса (без расширения).

Также не забывайте использовать опцию `-classpath`, если файл вашего класса находится не в текущем каталоге.

Задание 3

Измените исходный код таким образом, чтобы ваша фамилия была выведена в столбик по буквам. Буквы фамилии должны задаваться как параметры командной строки.

Откомпилируйте и запустите программу, добавив в командную строку ряд аргументов. Например, следующим образом: `"java MyFirstClass И в а н о в"`.

Задание 4

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

- ⌚ имеет поле, которое описывает одномерный массив целых чисел;
- ⌚ метод для получения значения элемента массива;
- ⌚ метод для модификации значения элемента массива;
- ⌚ конструктор, создающий объект массива заданного размера и инициализирующий его элементы случайными числами;
- ⌚ метод для получения среднего арифметического (возвращает значение среднего арифметического, а не выводит его на экран);
- ⌚ метод, обеспечивающий вывод элементов массива на экран.

Код метода `main()` `MyFirstClass` должен обеспечивать создание объекта типа `MySecondClass`, замену первого элемента массива на заданное число, вычисление среднего арифметического, вывод на экран значения среднего арифметического, вывод на экран элементов массива.

Добейтесь работоспособности программы.

Задание 5

Вынесите код класса `MySecondClass` без изменений в отдельный файл с именем `MyFirstPackage.java`, и поместите его в поддиректорию `MyFirstPackage`. Добейтесь работоспособности программы.

Задание 6

Запустите программу `jar`, предназначенную для создания архивов, и ознакомьтесь с форматом задания ключей для формирования архивов.

Скопируйте в рабочую папку, сохранив структуру каталогов, только файлы с расширением `class`, полученные в результате выполнения задания 5.

Создайте файл `manifest.mf`, содержащий следующий код:

```
--- manifest.mf -----
```

```
Manifest-Version: 1.0
```

```
Created-By: <Ваши фамилии>
```

```
Main-Class: MyFirstClass
```

```
-----
```

Обратите внимание на то, что после имени класса надо обязательно поставить символ новой строки.

Создайте архив `myfirst.jar`, включив в него полученные ранее файлы классов и указав созданный вами манифест-файл. Запустите его на выполнение с помощью команды `"java -jar myfirst.jar"`.

Вопросы

1. Три направления развития платформы Java.
2. Характерные особенности языка Java.
3. Три принципа ООП. Пример.
4. Достоинства и недостатки ООП.
5. Классы и объекты. Свойства объектов. Пример.
6. Члены класса. Модификаторы объявления класса.
7. Пакеты.
8. Пространства имен.
9. Модуль компиляции.
10. Поля.
11. Методы. Метод `main`.
12. Модификаторы доступа.
13. Создание объектов. Конструкторы.
14. Блоки инициализации. Статическая инициализация.
15. Комментарии. Простые типы. Массивы.
16. Операторы. Циклы.