

Отчет по лабораторной работе №.1

Студент: Перхуров В.А.

Группа: ИВМ-22

1. Постановка задачи

В процессе выполнения лабораторной работы необходимо выполнить следующие задачи:

1. Создать исполняемый java класс, который выводит текстовую строку на экран
2. Разработать два класса. Первый класс вызывает 2 метода из второго класса (статический и не статический)
3. Создать jar-файл, хранящий элементы из предыдущего задания

2. Разработка задачи

2.1 Структура проекта

Проект разделен на следующие директории:

docs

Данная документация

sources

Содержит manifest.mf, сгенерированный jar-файл и две папки:

build

Сборочная директория, где хранятся собранные class-файлы.

ru/rsatu

Иерархия исходников, где хранятся main-класс и вспомогательные классы в виде java-файлов.

lr1

Содержит main-класс в файле ConsoleApp.java, из которого вызывается 1 статический метод и несколько обычных из созданного объекта класса Concat.java.

utils

Содержит 2 вспомогательных класса: Consts.java и Concat.java. Класс Consts содержит константы, а класс Concat - методы конкатенации и вывода в консоль.

3. Информация о реализации

Сначала были созданы необходимые директории. Далее был создан исполняемый (ConsoleApp) и вспомогательные (Consts и Concat) классы.

Листинг 1. Листинг класса ConsoleApp

```
package ru.rsatu.l1r1;

import ru.rsatu.utils.Consts;
import ru.rsatu.utils.Concat;

public class ConsoleApp {
    public static void main(String[] args)
    {
        Concat concat = new Concat();
        concat.Add(Consts.EN_HELLO_WORLD);
        concat.Add(" - ");
        concat.Add(Consts.RU_HELLO_WORLD);
        Concat.Print( concat );
    }
}
```

Листинг 2. Листинг класса Consts

```
package ru.rsatu.utils;

public class Consts {
    public static String EN_HELLO_WORLD = "Hello world!";
    public static String RU_HELLO_WORLD = "Привет мир!";
}
```

Листинг 3. Листинг класса Concat

```
package ru.rsatu.utils;

public class Concat {
    public static void Print(Concat cnct)
    {
        System.out.println(cnct.Get());
    }

    public void Add(String str)
    {
        mTxt += str;
    }

    public String Get()
    {
        return mTxt;
    }

    private String mTxt = "";
}
```

Для компиляции файла через терминал использовалась команда "javac". Для компиляции результирующих файлов *.class использовал ключ "-d" с указанием сборочной директории build.

Листинг 4. Команда javac

```
javac -d build ru/rsatu/lr1/ConsoleApp.java
```

Для запуска файла из терминала использовалась команда "java", но т.к. объектные файлы были сгенерированы в отдельной сборочной директории для запуска был использован ключ "-classpath" с указанием сборочной директории build.

Листинг 5. Команда java

```
java -classpath build ru.rsatu.lr1.ConsoleApp
```

Результат работы скомпилированного файла

Листинг 6. Результат работы

```
Hello world! - Привет мир!
```

Далее был создан jar-файл. Для его создания был создан файл manifest.mf, в котором указан главный исполняемый класс ConsoleApp

Листинг 7. Листинг manifest.mf

```
Manifest-Version: 1.0  
Created-By: Vitaliy Perhurov  
Main-Class: ru.rsatu.lr1.ConsoleApp
```

После этого была выполнена команда по сборке архива jar и запуск jar-файла

Листинг 8. Команда по сборке архива

```
jar cvmf manifest.mf lr1.jar -C ./build ru
```

Листинг 9. Запуск jar-файла

```
java -jar lr1.jar
```

В результате выполнения выводится следующее:

Листинг 10. Результат выполнения

```
Hello world! - Привет мир!
```

4. Результаты выполнения

В результате выполнения задания было выполнено:

- созданы классы ConsoleApp в директории ru/rsatu/lr1, Concat и Consts в директории ru/rsatu/utils;
- созданные классы были скомпилированы в папку build с помощью команды javac с ключом "-d";
- был запущен исполняемый класс через терминал с помощью команды java с ключом "-classpath";
- для последующего создания исполняемого jar-файла был создан файл manifest.mf;
- с помощью команды jar был создан исполняемый jar-файл;
- был запущен jar-файл с помощью команды java с ключом "-jar".

Вся последовательность продемонстрирована на следующем скриншоте:

```
Терминал - vitality_altermann@vitaliy-asusexpertbook:/mnt/7999679B46EE88E1/4_Projects/modern_technologies_of_industrial_softwar
Файл Правка Вид Терминал Вкладки Справка
[vitaliy_altermann@vitaliy-asusexpertbook sources]$ javac -d build ru/rsatu/lr1/ConsoleApp.java
[vitaliy_altermann@vitaliy-asusexpertbook sources]$ java -cp classpath build ru/rsatu/lr1/ConsoleApp
Hello world! - Привет мир!
[vitaliy_altermann@vitaliy-asusexpertbook sources]$ jar cvmf manifest.mf lr1.jar -C ./build ru
added manifest
adding: ru/(in = 0) (out= 0)(stored 0%)
adding: ru/rsatu/(in = 0) (out= 0)(stored 0%)
adding: ru/rsatu/lr1/(in = 0) (out= 0)(stored 0%)
adding: ru/rsatu/lr1/ConsoleApp.class(in = 560) (out= 365)(deflated 34%)
adding: ru/rsatu/utills/(in = 0) (out= 0)(stored 0%)
adding: ru/rsatu/utills/Concat.class(in = 1095) (out= 566)(deflated 48%)
adding: ru/rsatu/utills/Consts.class(in = 400) (out= 307)(deflated 23%)
[vitaliy_altermann@vitaliy-asusexpertbook sources]$ java -jar lr1.jar
Hello world! - Привет мир!
[vitaliy_altermann@vitaliy-asusexpertbook sources]$
```

5. ВЫВОД

В результате выполнения лабораторной работы получены навыки по компиляции классов, запуску проектов, созданию jar-файлов с помощью командной строки.