

Отчет по лабораторной работе №.2

Студент: Перхуров В.А.

Группа: ИВМ-22

1. Постановка задачи

В процессе выполнения лабораторной работы необходимо выполнить следующие задачи:

1. Создать несколько maven-проектов:
 - a. Родительский
 - b. 2 Дочерних (1 дочерний проект зависит от второго)
2. Подключить внешнюю зависимость (из глобального репозитория) в один из проектов и продемонстрировать работу этой зависимости.
3. Создать maven-проект, который будет помещен в локальный репозиторий. Добавить этот проект как зависимость в проект из п.1
4. Создать maven-проект, упаковать его в jar-with-dependencies, продемонстрировать работу jar. Запустить проект с помощью maven.
5. Выполнить тест подключенной зависимости из п.2 с помощью junit

2. Разработка задачи

2.1 Структура проекта

Проект разделен на следующие директории:

docs

Данная документация

sources

Содержит сгенерированный jar-файл и три папки:

parent_project

Директория, где хранятся исходники родительского проекта.

doughter_porject_1

Директория, где хранятся исходники первого дочернего проекта.

doughter_porject_2

Директория, где хранятся исходники второго дочернего проекта.

3. Информация о реализации

3.1 Настройка проектов

Сначала были созданы необходимые директории. Далее были созданы три проекта (1 родительский и 2 дочерних) с помощью команды "mvn archetype:generate".

Параметры создания проектов указаны на следующем скриншоте:

```
Терминал - vitality_altermann@vitaliy-asusexpertbook:/mnt/7999679B46EE88E1/4_Projects/modern_technologies_of_industrial_s
Файл Правка Вид Терминал Вкладки Справка
8: 1.4
Choose a number: 8: 8
Define value for property 'groupId': ru.rsatu
Define value for property 'artifactId': parent_project
Define value for property 'version' 1.0-SNAPSHOT: :
Define value for property 'package' ru.rsatu: :
Confirm properties configuration:
groupId: ru.rsatu
artifactId: parent_project
version: 1.0-SNAPSHOT
package: ru.rsatu
Y: :

Терминал - vitality_altermann@vitaliy-asusexpertbook:/mnt/7999679B46EE88E1/4_Projects/modern_technologies_of_industrial_s
Файл Правка Вид Терминал Вкладки Справка
8: 1.4
Choose a number: 8: 8
Define value for property 'groupId': ru.rsatu
Define value for property 'artifactId': daughter_porject_1
Define value for property 'version' 1.0-SNAPSHOT: :
Define value for property 'package' ru.rsatu: :
Confirm properties configuration:
groupId: ru.rsatu
artifactId: daughter_porject_1
version: 1.0-SNAPSHOT
package: ru.rsatu
Y: :

Терминал - vitality_altermann@vitaliy-asusexpertbook:/mnt/7999679B46EE88E1/4_Projects/modern_technologies_of_industrial_s
Файл Правка Вид Терминал Вкладки Справка
8: 1.4
Choose a number: 8: 8
Define value for property 'groupId': ru.rsatu
Define value for property 'artifactId': daughter_porject_2
Define value for property 'version' 1.0-SNAPSHOT: :
Define value for property 'package' ru.rsatu: :
Confirm properties configuration:
groupId: ru.rsatu
artifactId: daughter_porject_2
version: 1.0-SNAPSHOT
package: ru.rsatu
Y: :
```

Далее были настроены связи проектов путём изменения pom-файлов.

Листинг 1. Листинг pom-файла родительского проекта parent_project

```

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi=
"http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>ru.rsatu</groupId>
  <artifactId>parent_project</artifactId>
  <version>1.0-SNAPSHOT</version>
  <packaging>pom</packaging>

  <name>parent_project</name>
  <!-- FIXME change it to the project's website -->
  <url>http://www.example.com</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
  </properties>

  <dependencies>
    <!-- https://mvnrepository.com/artifact/org.apache.commons/commons-math3 -->
    <dependency>
      <groupId>org.apache.commons</groupId>
      <artifactId>commons-math3</artifactId>
      <version>3.6.1</version>
    </dependency>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>4.11</version>
      <scope>test</scope>
    </dependency>
  </dependencies>

  <build>
    <pluginManagement><!-- lock down plugins versions to avoid using Maven defaults
(may be moved to parent pom) -->
      <plugins>
        <!-- clean lifecycle, see https://maven.apache.org/ref/current/maven-
core/lifecycles.html#clean_Lifecycle -->
        <plugin>
          <artifactId>maven-clean-plugin</artifactId>
          <version>3.1.0</version>
        </plugin>
        <!-- default lifecycle, jar packaging: see
https://maven.apache.org/ref/current/maven-core/default-
bindings.html#Plugin_bindings_for_jar_packaging -->

```

```

<plugin>
  <artifactId>maven-resources-plugin</artifactId>
  <version>3.0.2</version>
</plugin>
<plugin>
  <artifactId>maven-compiler-plugin</artifactId>
  <version>3.8.0</version>
</plugin>
<plugin>
  <artifactId>maven-surefire-plugin</artifactId>
  <version>2.22.1</version>
</plugin>
<plugin>
  <artifactId>maven-jar-plugin</artifactId>
  <version>3.0.2</version>
</plugin>
<plugin>
  <artifactId>maven-install-plugin</artifactId>
  <version>2.5.2</version>
</plugin>
<plugin>
  <artifactId>maven-deploy-plugin</artifactId>
  <version>2.8.2</version>
</plugin>
<!-- site lifecycle, see https://maven.apache.org/ref/current/maven-
core/lifecycles.html#site_Lifecycle -->
<plugin>
  <artifactId>maven-site-plugin</artifactId>
  <version>3.7.1</version>
</plugin>
<plugin>
  <artifactId>maven-project-info-reports-plugin</artifactId>
  <version>3.0.0</version>
</plugin>
</plugins>
</pluginManagement>
</build>
</project>

```

В этом файле были проведены следующие изменения:

1. С помощью параметра "<packaging>...</packaging>" данный проект будет собран в "pom-проект".
2. Была подключена математическая библиотека из внешнего репозитория "commons-math3".

Листинг 2. Листинг pom-файла первого дочернего проекта daughter_porject_1

```
<?xml version="1.0" encoding="UTF-8"?>
```

```

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi=
"http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <parent>
    <groupId>ru.rsatu</groupId>
    <artifactId>parent_project</artifactId>
    <version>1.0-SNAPSHOT</version>
  </parent>

  <groupId>ru.rsatu</groupId>
  <artifactId>doughter_porject_1</artifactId>
  <version>1.0-SNAPSHOT</version>

  <name>doughter_porject_1</name>
  <!-- FIXME change it to the project's website -->
  <url>http://www.example.com</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
  </properties>

  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>4.11</version>
      <scope>test</scope>
    </dependency>
  </dependencies>

  <build>
    <pluginManagement><!-- lock down plugins versions to avoid using Maven defaults
(may be moved to parent pom) -->
      <plugins>
        <!-- clean lifecycle, see https://maven.apache.org/ref/current/maven-
core/lifecycles.html#clean_Lifecycle -->
        <plugin>
          <artifactId>maven-clean-plugin</artifactId>
          <version>3.1.0</version>
        </plugin>
        <!-- default lifecycle, jar packaging: see
https://maven.apache.org/ref/current/maven-core/default-
bindings.html#Plugin_bindings_for_jar_packaging -->
        <plugin>
          <artifactId>maven-resources-plugin</artifactId>
          <version>3.0.2</version>

```

```

</plugin>
<plugin>
  <artifactId>maven-compiler-plugin</artifactId>
  <version>3.8.0</version>
</plugin>
<plugin>
  <artifactId>maven-surefire-plugin</artifactId>
  <version>2.22.1</version>
</plugin>
<plugin>
  <artifactId>maven-jar-plugin</artifactId>
  <version>3.0.2</version>
</plugin>
<plugin>
  <artifactId>maven-install-plugin</artifactId>
  <version>2.5.2</version>
</plugin>
<plugin>
  <artifactId>maven-deploy-plugin</artifactId>
  <version>2.8.2</version>
</plugin>
<!-- site lifecycle, see https://maven.apache.org/ref/current/maven-
core/lifecycles.html#site_Lifecycle -->
<plugin>
  <artifactId>maven-site-plugin</artifactId>
  <version>3.7.1</version>
</plugin>
<plugin>
  <artifactId>maven-project-info-reports-plugin</artifactId>
  <version>3.0.0</version>
</plugin>
</plugins>
</pluginManagement>
</build>
</project>

```

В этом файле с помощью параметра "<parent>...</parent>" был указан родительский pom-проект.

Листинг 3. Листинг pom-файла второго дочернего проекта daughter_porject_2

```

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi=
"http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <parent>
    <groupId>ru.rsatu</groupId>

```

```

    <artifactId>parent_project</artifactId>
    <version>1.0-SNAPSHOT</version>
</parent>

<groupId>ru.rsatu</groupId>
<artifactId>doughter_porject_2</artifactId>
<version>1.0-SNAPSHOT</version>
<packaging>jar</packaging>

<name>doughter_porject_2</name>
<!-- FIXME change it to the project's website -->
<url>http://www.example.com</url>

<properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
</properties>

<dependencies>
    <dependency>
        <groupId>ru.rsatu</groupId>
        <artifactId>doughter_porject_1</artifactId>
        <version>1.0-SNAPSHOT</version>
        <scope>compile</scope>
    </dependency>
    <dependency>
        <groupId>junit</groupId>
        <artifactId>junit</artifactId>
        <version>4.11</version>
        <scope>test</scope>
    </dependency>
</dependencies>

<build>
    <pluginManagement><!-- lock down plugins versions to avoid using Maven defaults
(may be moved to parent pom) -->
        <plugins>
            <!-- clean lifecycle, see https://maven.apache.org/ref/current/maven-
core/lifecycles.html#clean_Lifecycle -->
            <plugin>
                <artifactId>maven-clean-plugin</artifactId>
                <version>3.1.0</version>
            </plugin>
            <!-- default lifecycle, jar packaging: see
https://maven.apache.org/ref/current/maven-core/default-
bindings.html#Plugin_bindings_for_jar_packaging -->
            <plugin>
                <artifactId>maven-resources-plugin</artifactId>
                <version>3.0.2</version>
            </plugin>

```

```

<plugin>
  <artifactId>maven-compiler-plugin</artifactId>
  <version>3.8.0</version>
</plugin>
<plugin>
  <artifactId>maven-surefire-plugin</artifactId>
  <version>2.22.1</version>
</plugin>
<plugin>
  <artifactId>maven-jar-plugin</artifactId>
  <version>3.0.2</version>
</plugin>
<plugin>
  <artifactId>maven-install-plugin</artifactId>
  <version>2.5.2</version>
</plugin>
<plugin>
  <artifactId>maven-deploy-plugin</artifactId>
  <version>2.8.2</version>
</plugin>
<!-- site lifecycle, see https://maven.apache.org/ref/current/maven-
core/lifecycles.html#site_Lifecycle -->
<plugin>
  <artifactId>maven-site-plugin</artifactId>
  <version>3.7.1</version>
</plugin>
<plugin>
  <artifactId>maven-project-info-reports-plugin</artifactId>
  <version>3.0.0</version>
</plugin>
<plugin>
  <artifactId>maven-assembly-plugin</artifactId>
  <configuration>
    <archive>
      <manifest>
        <mainClass>ru.rsatu.DougherTwoApp</mainClass>
      </manifest>
    </archive>
    <descriptorRefs>
      <descriptorRef>jar-with-dependencies</descriptorRef>
    </descriptorRefs>
  </configuration>
  <executions>
    <execution>
      <id>make-assembly</id> <!-- this is used for inheritance merges -->
      <phase>package</phase> <!-- bind to the packaging phase -->
      <goals>
        <goal>single</goal>
      </goals>
    </execution>
  </executions>

```



```
        </plugin>
    </plugins>
</pluginManagement>
</build>
</project>
```

В этом файле были проведены следующие изменения:

1. С помощью параметра "<packaging>...</packaging>" данный проект будет собран "jar-файл".
2. Установлена зависимость от первого дочернего проекта "dougher_porject_1".
3. Добавлен плагин для формирования jar-файла. Данный плагин сработает для цели "single".

3.2 Реализация классов для запуска

В предыдущем пункте maven автоматически сгенерировал main-классы проектов. В данном пункте добавим в них реализацию.

Листинг 4. Листинг класса DougherOneApp

```
package ru.rsatu;
import org.apache.commons.math3.stat.descriptive.DescriptiveStatistics;
import java.util.*;

public class DougherOneApp
{
    public static void main( String[] args )
    {
        System.out.println( "RUN DougherOneApp" );
    }

    public static double GetMean( Integer[] values )
    {
        DescriptiveStatistics ds = new DescriptiveStatistics();
        for( int i=0; i < values.length; i++ )
        {
            ds.addValue( values[i] );
        }

        return ds.getMean();
    }
}
```

Класс ru.rsatu.DougherOneApp содержит метод для нахождения среднего арифметического из введенного набора чисел.

```
package ru.rsatu;
import ru.rsatu.DougherOneApp;

import java.util.*;

public class DougherTwoApp
{
    public static void main( String[] args )
    {
        System.out.println( "RUN DougherTwoApp" );

        System.out.println( "Введите размер массива: " );
        Scanner scan = new Scanner( System.in );
        int size = scan.nextInt();

        System.out.println( "Введите числа для нахождения среднего арифметического: "
);
        ArrayList<Integer> values_list = new ArrayList<>();
        for( int i = 0; i < size; i++ )
        {
            values_list.add( scan.nextInt() );
        }

        Integer[] values = new Integer[values_list.size()];
        values = values_list.toArray(values);
        System.out.printf( "Результат: %f", DougherOneApp.GetMean( values ) );
    }
}
```

Класс `ru.rsatu.DougherTwoApp` содержит `main`-метод, из которого вызывается метод `GetMean` класса `DougherOneApp`.

3.3 Реализация классов для тестов

В предыдущем пункте `maven` автоматически сгенерировал классы тестов. В данном пункте добавим в один из них реализацию.

```
package ru.rsatu;

import static org.junit.Assert.assertTrue;
import org.apache.commons.math3.stat.descriptive.DescriptiveStatistics;

import org.junit.Test;

/**
 * Unit test for simple DougherTwoApp.
 */
public class DougherTwoAppTest
{
    /**
     * Rigorous Test :-)
     */
    @Test
    public void shouldAnswerWithTrue()
    {
        assertTrue( true );
    }

    @Test
    public void testAbs()
    {
        double[] values = new double[]{1,3,5};
        DescriptiveStatistics ds = new DescriptiveStatistics();
        for( int i = 0; i < 3; i++ )
        {
            ds.addValue(values[i]);
        }
        int real_result = (int)ds.getMean();
        int valide_result = 3;
        assertTrue( real_result == valide_result );
    }
}
```

3.4 Сборка с помощью maven

Сборка проектов выполняется следующей командой из терминала:

Листинг 7. Команда для сборки maven-проектов

```
mvn install
```

Результат выполнения сборки из терминала представлены на следующих скриншотах:

```
Терминал - vitaliy_altermann@vitaliy-asusexpertbook:/mnt/7999679B46EE88E1/4_Projects/modern_tech...
Файл Правка Вид Терминал Вкладки Справка

[vitaliy_altermann@vitaliy-asusexpertbook parent_project]$ mvn install
[INFO] Scanning for projects...
[INFO]
[INFO] -----< ru.rsatu:parent_project >-----
[INFO] Building parent_project 1.0-SNAPSHOT
[INFO] -----[ pom ]-----
[INFO]
[INFO] --- maven-install-plugin:2.5.2:install (default-install) @ parent_project ---
[INFO] Installing /mnt/7999679B46EE88E1/4_Projects/modern_technologies_of_industrial_software_development/lr_2/sources/parent_project/pom.xml to /home/vitaliy_altermann/.m2/repository/ru/rsatu/parent_project/1.0-SNAPSHOT/parent_project-1.0-SNAPSHOT.pom
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 0.280 s
[INFO] Finished at: 2022-09-29T11:12:02+03:00
[INFO]
[vitaliy_altermann@vitaliy-asusexpertbook parent_project]$
```

Figure 1. Сборка родительского проекта

```
Терминал - vitaliy_altermann@vitaliy-asusexpertbook:/mnt/7999679B46EE88E1/4_Projects/modern_technologies_of_industrial_software_development/lr_2/sources/d...
Файл Правка Вид Терминал Вкладки Справка

[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory /mnt/7999679B46EE88E1/4_Projects/modern_technologies_of_industrial_software_development/lr_2/sources/dou...
[INFO] --- maven-compiler-plugin:3.8.0:testCompile (default-testCompile) @ daughter_project_1 ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 1 source file to /mnt/7999679B46EE88E1/4_Projects/modern_technologies_of_industrial_software_development/lr_2/sources/dou...
[INFO] --- maven-surefire-plugin:2.22.1:test (default-test) @ daughter_project_1 ---
[INFO]
[INFO] T E S T S
[INFO]
[INFO] Running ru.rsatu.DaughterOneAppTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.034 s - in ru.rsatu.DaughterOneAppTest
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] --- maven-jar-plugin:3.0.2:jar (default-jar) @ daughter_project_1 ---
[INFO] Building jar: /mnt/7999679B46EE88E1/4_Projects/modern_technologies_of_industrial_software_development/lr_2/sources/dou...
[INFO] --- maven-install-plugin:2.5.2:install (default-install) @ daughter_project_1 ---
[INFO] Installing /mnt/7999679B46EE88E1/4_Projects/modern_technologies_of_industrial_software_development/lr_2/sources/dou...
[INFO] Installing /mnt/7999679B46EE88E1/4_Projects/modern_technologies_of_industrial_software_development/lr_2/sources/dou...
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 1.970 s
[INFO] Finished at: 2022-09-29T11:20:45+03:00
[INFO]
[vitaliy_altermann@vitaliy-asusexpertbook daughter_project_1]$

Терминал - vitaliy_altermann@vitaliy-asusexpertbook:/mnt/7999679B46EE88E1/4_Projects/modern_technologies_of_industrial_software_development/lr_2/sources/d...
Файл Правка Вид Терминал Вкладки Справка

[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory /mnt/7999679B46EE88E1/4_Projects/modern_technologies_of_industrial_software_development/lr_2/sources/dou...
[INFO] --- maven-compiler-plugin:3.8.0:testCompile (default-testCompile) @ daughter_project_2 ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 1 source file to /mnt/7999679B46EE88E1/4_Projects/modern_technologies_of_industrial_software_development/lr_2/sources/dou...
[INFO] --- maven-surefire-plugin:2.22.1:test (default-test) @ daughter_project_2 ---
[INFO]
[INFO] T E S T S
[INFO]
[INFO] Running ru.rsatu.DaughterTwoAppTest
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.06 s - in ru.rsatu.DaughterTwoAppTest
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] --- maven-jar-plugin:3.0.2:jar (default-jar) @ daughter_project_2 ---
[INFO] Building jar: /mnt/7999679B46EE88E1/4_Projects/modern_technologies_of_industrial_software_development/lr_2/sources/dou...
[INFO] --- maven-install-plugin:2.5.2:install (default-install) @ daughter_project_2 ---
[INFO] Installing /mnt/7999679B46EE88E1/4_Projects/modern_technologies_of_industrial_software_development/lr_2/sources/dou...
[INFO] Installing /mnt/7999679B46EE88E1/4_Projects/modern_technologies_of_industrial_software_development/lr_2/sources/dou...
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 1.919 s
[INFO] Finished at: 2022-09-29T11:20:50+03:00
[INFO]
[vitaliy_altermann@vitaliy-asusexpertbook daughter_project_2]$
```

Figure 2. Сборка дочерних проектов

На рисунке 2 видно, что помимо сборки были выполнены junit-тесты. В первом дочернем проекте был выполнен один базовый тест, а во втором - два (базовый и новый).

В результате данной сборки в директориях проектов появились папки "target", в которых расположены jar-файлы. Запустить эти файлы с помощью команды jar не получится, т.к. вся информация о классах содержится в maven. Для создания jar-файла, который можно запускать через команду jar, была выполнена сборка с помощью следующей команды:

Листинг 8. Команда для сборки независимого jar-файла

```
mvn compile assembly:single
```

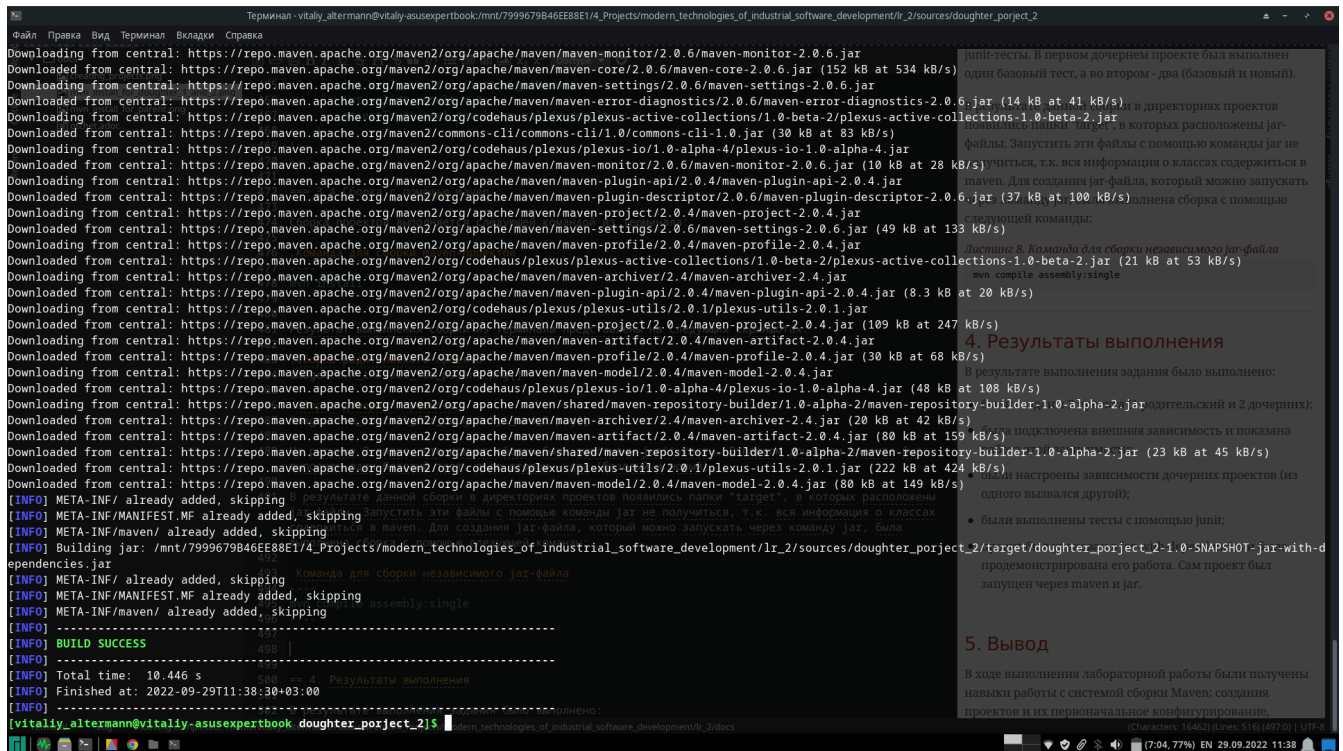


Figure 3. Сборка независимого jar-файла

В результате выполнения сборки независимого jar-файла были подтянуты все зависимости и загружены в один jar-файл (doughter_porject_2-1.0-SNAPSHOT-jar-with-dependencies.jar).

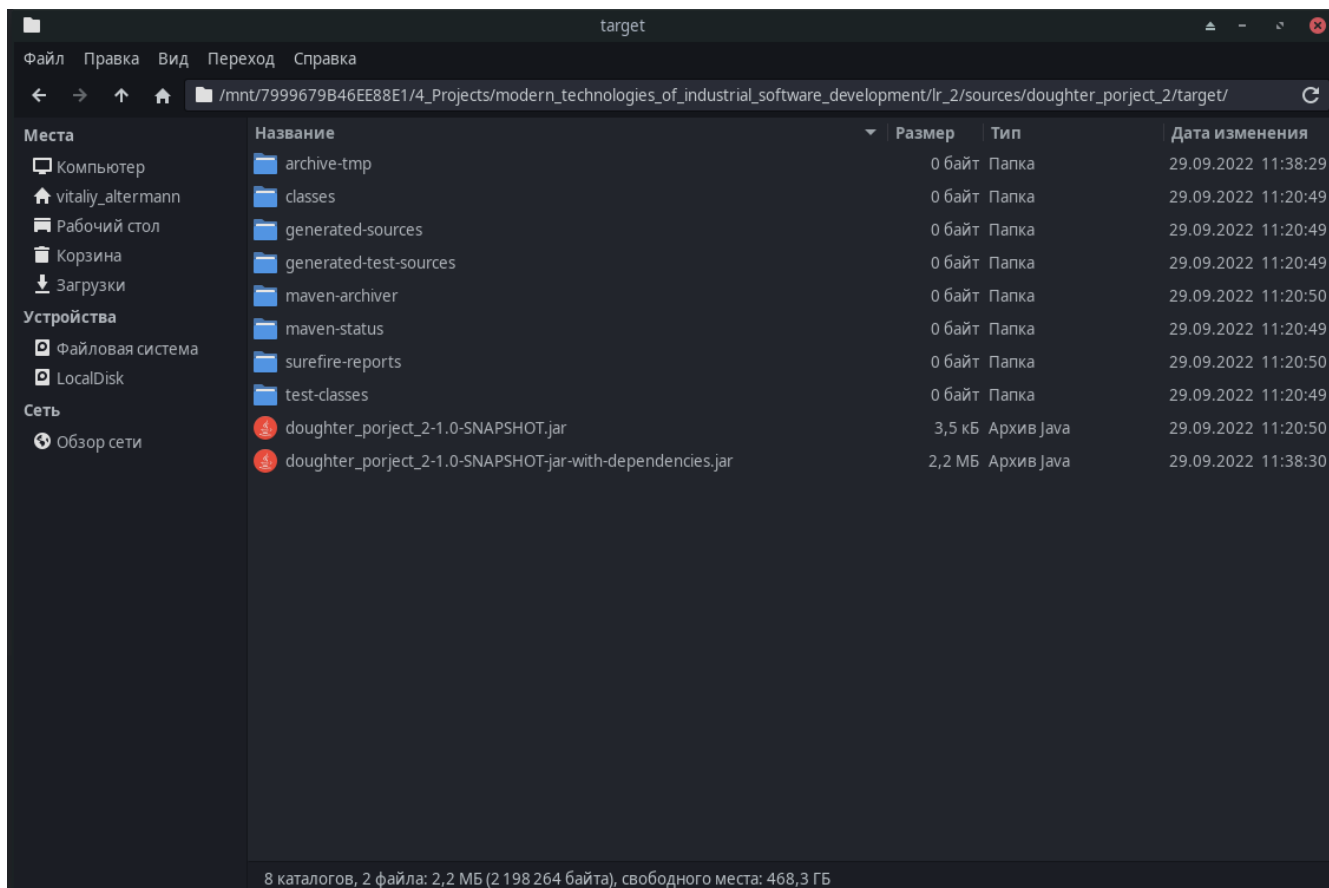


Figure 4. Результат сборки независимого jar-файла

3.5 Запуск итоговой программы с помощью maven

Запуск проета возможно произвести из maven, для этого необходимо выполнить команду:

Листинг 9. Команда для запуска программы через maven

```
mvn exec:java -Dexec.mainClass="ru.rsatu.DougherTwoApp"
```

В этой команде явно указывается класс, который необходимо запустить. В результате работы программа вывела на экран строки:

```
Терминал - vitality_altermann@vitaliy-asusexpertbook: /mnt/7999679B46EE8E1/4_Projects/modern_technologies_of_industrial_software_development/lr_2/sources/dough
Файл Правка Вид Терминал Вкладки Справка
[vitaliy_altermann@vitaliy-asusexpertbook daughter_porject_2]$ mvn exec:java -Dexec.mainClass="ru.rsatu.DoughterTwoApp"
[INFO] Scanning for projects...
[INFO]
[INFO] -----< ru.rsatu:daughter_porject_2 >-----
[INFO] Building daughter_porject_2 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- exec-maven-plugin:3.1.0:java (default-cli) @ daughter_porject_2 ---
RUN DoughterTwoApp
Введите размер массива:
3
Введите числа для нахождения среднего арифметического:
1 3 5
Результат: 3,000000[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 29.092 s
[INFO] Finished at: 2022-09-29T11:59:08+03:00
[INFO] -----
[vitaliy_altermann@vitaliy-asusexpertbook daughter_porject_2]$
```

Figure 5. Результат работы программа, запущенной через maven

3.6 Запуск итоговой программы с помощью java -jar

Запуск проета возможно произвести из java -jar, для этого необходимо выполнить команду:

Листинг 10. Команда для запуска программы через java -jar

```
java -jar java -jar lab2.3-1.0-SNAPSHOT-jar-with-dependencies.jar
```

В этой команде указывается jar-файл с вложенными зависимостями. В результате работы программа вывела на экран строки:

```
Терминал - vitality_altermann@vitaliy-asusexpertbook: /mnt/7999679B46EE8E1/4_Projects/modern_technologies_of_industrial_software_development/lr_2/sources/doughte
Файл Правка Вид Терминал Вкладки Справка
[vitaliy_altermann@vitaliy-asusexpertbook target]$ java -jar daughter_porject_2-1.0-SNAPSHOT-jar-with-dependencies.jar
RUN DoughterTwoApp
Введите размер массива:
3
Введите числа для нахождения среднего арифметического:
1 3 5
Результат: 3,000000[vitaliy_altermann@vitaliy-asusexpertbook target]$
```

Figure 6. Результат работы программа, запущенной через java -jar

4. Результаты выполнения

В результате выполнения задания было выполнено:

- были созданы 3 проекта (1 родительский и 2 дочерних);
- была подключена внешняя зависимость и показана работа этой зависимости;
- были настроены зависимости дочерних проектов (из одного вызвался другой);
- были выполнены тесты с помощью junit;
- проект был упакован в jar-with-dependencies и была продемонстрирована его работа. Сам проект был запущен через maven и jar.

5. Вывод

В ходе выполнения лабораторной работы были получены навыки работы с системой сборки Maven: создания проектов и их первоначальное конфигурирование, добавление как внешних, так и локальных зависимостей, создания и выполнения модульных тестов, сборки единого файла, содержащего все зависимости, а также запуска проектов.