



УНИВЕРСИТЕТ ИТМО

Факультет программной
инженерии и компьютерной техники

Лабораторная работа № 3
по дисциплине
«Методы и средства программной инженерии»

Вариант 55222

Преподаватель: Покид Александр Владимирович

Выполнили: Геллер Л. А., Блинова М. А.

Группа: Р3230

Санкт-Петербург
2021 г.

Задание

Сценарий должен реализовывать следующие цели (targets):

1. **compile** -- компиляция исходных кодов проекта.
2. **build** -- компиляция исходных кодов проекта и их упаковка в исполняемый jar-архив. Компиляцию исходных кодов реализовать посредством вызова цели **compile**.
3. **clean** -- удаление скомпилированных классов проекта и всех временных файлов (если они есть).
4. **test** -- запуск junit-тестов проекта. Перед запуском тестов необходимо осуществить сборку проекта (цель **build**).
5. **xml** - валидация всех xml-файлов в проекте.
6. **team** - осуществляет получение из svn-репозитория 4 предыдущих ревизий, их сборку (по аналогии с основной) и упаковку получившихся jar-файлов в zip-архив. Сборку реализовать посредством вызова цели **build**.

Build.xml

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

<project default="build" name="Lab3" >

    <property file="build.properties"/>

    <target name="compile">
        <mkdir dir="${compiled.classes}"/>
        <path id="compile.classpath">
            <fileset dir="${lib.dir}" includes="**/*.jar"/>
        </path>
        <javac includeantruntime="false" destdir="${compiled.classes}">
            <src path="${src}"/>
            <classpath>
                <pathelement location="${compiled.classes}"/>
                <path refid="compile.classpath"/>
            </classpath>
        </javac>
    </target>

    <target name="compile-tests">
        <mkdir dir="${compiled.tests}"/>
        <path id="compile-tests.classpath">
            <fileset dir="${lib.dir}" includes="**/*.jar"/>
        </path>
        <javac includeantruntime="false" destdir="${compiled.tests}">
            <src path="${test.root}"/>
            <classpath>
                <pathelement location="${compiled.classes}"/>
                <path refid="compile-tests.classpath"/>
            </classpath>
        </javac>
    </target>
```

```

    <target name="build" depends="compile, compile-tests">
        <jar index="true" basedir="${compiled.classes}"
            destfile="${out.jar}">
            <zipgroupfilesset dir="${lib.dir}" includes="*.jar"/>
            <manifest>
                <attribute name="Created-By" value="mi"/>
                <attribute name="Manifest-Version" value="${version}"/>
                <attribute name="Main-Class" value="Main"/>
            </manifest>
        </jar>
    </target>

    <target name="clean">
        <delete dir="${compiled}"/>
    </target>

    <target name="test" depends="build">
        <mkdir dir="reports"/>
        <path id="test.classpath">
            <fileset dir="${lib.dir}" includes="**/*.jar"/>
        </path>
        <junit printsummary="on" fork="true">
            <formatter type="xml"/>
            <classpath>
                <pathelement location="${compiled.tests}"/>
                <pathelement location="${compiled.classes}"/>
                <path refid="test.classpath"/>
            </classpath>
            <batchtest todir="reports">
                <fileset dir="${test.root}" includes="*Test*.java"/>
            </batchtest>
        </junit>
    </target>

    <target name="team">

        <mkdir dir="${out}"/>
        <delete dir="${out.previous}"/>
        <mkdir dir="${out.previous}"/>

        <typedef resource="${svnantlib}">
            <classpath>
                <pathelement location="${svnant.jar}"/>
                <pathelement location="${svnclientadapter.jar}"/>
                <pathelement location="${svnjavahl.jar}"/>
            </classpath>
        </typedef>

        <svn>
            <update dir="${trunk}" revision="HEAD"/>
        </svn>

        <svn>
            <update dir="${trunk}" revision="PREV"/>
        </svn>
        <antcall target="build">
            <param name="compiled.classes" value="${svn.compiled.classes}"/>
            <param name="out.jar" value="${out.previous}/VERSION4.jar"/>
            <param name="src" value="${svn.src}"/>
            <param name="version" value="4.0"/>
        </antcall>

        <svn>
            <update dir="${trunk}" revision="PREV"/>

```

```

</svn>
<antcall target="build">
  <param name="compiled.classes" value="${svn.compiled.classes}"/>
  <param name="out.jar" value="${out.previous}/VERSION3.jar"/>
  <param name="src" value="${svn.src}"/>
  <param name="version" value="3.0"/>
</antcall>

<svn>
  <update dir="${trunk}" revision="PREV"/>
</svn>
<antcall target="build">
  <param name="compiled.classes" value="${svn.compiled.classes}"/>
  <param name="out.jar" value="${out.previous}/VERSION2.jar"/>
  <param name="src" value="${svn.src}"/>
  <param name="version" value="2.0"/>
</antcall>

<svn>
  <update dir="${trunk}" revision="PREV"/>
</svn>
<antcall target="build">
  <param name="compiled.classes" value="${svn.compiled.classes}"/>
  <param name="out.jar" value="${out.previous}/VERSION1.jar"/>
  <param name="src" value="${svn.src}"/>
  <param name="version" value="1.0"/>
</antcall>

<zip destfile="${previous.zip}" basedir="${out.previous}"/>
<property name="compiled.classes" value="build/classes"/>
<property name="out.jar" value="out/Lab3.jar"/>
<property name="src" value="src"/>

</target>

<target name="xml">
  <xmlvalidate failonerror="no" lenient="yes">
    <fileset dir="." includes="**/*.xml"/>
  </xmlvalidate>
</target>

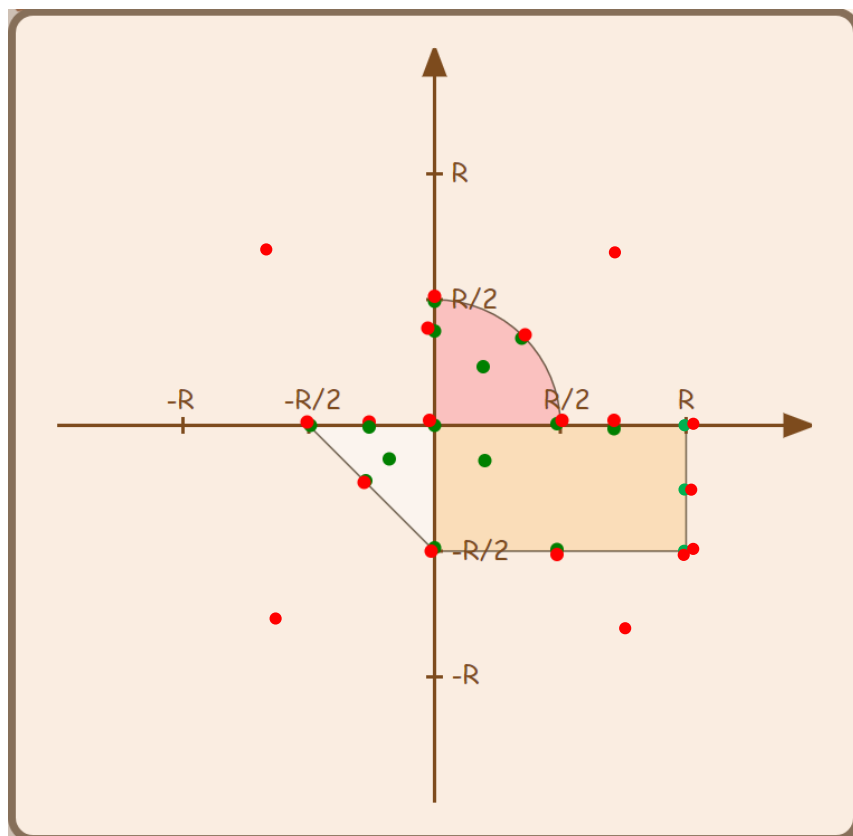
</project>

```

build.properties

```
compiled=build
compiled.classes=build/classes
compiled.tests=build/tests
src=src
test.root=test/main/java
out=out
out.jar=out/Lab3.jar
out.previous=out/previous
previous.zip=out/previous/PREVIOUS.zip
version=1.0
reports=reports
trunk=svn/trunk
svn.src=svn/trunk/src
svn.compiled.classes=svn/trunk/build/classes
svnantlib=org/tigris/subversion/svnant/svnantlib.xml
lib.dir=lib
svnant.jar=lib/svnant.jar
svnclientadapter.jar=lib/svnclientadapter-1.3.jar
svnjavahl.jar=lib/svnjavahl-1.3.jar
```

Тестовое покрытие



Вывод

При выполнении данной лабораторной работы мы изучили основы работы с Apache Ant и написали свои первые junit-тесты