



Дарья
Константин-
овна,
будьте с
нами нежны

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное
образовательное учреждение высшего образования
“Национальный исследовательский университет ИТМО”
ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ
и КОМПЬЮТЕРНОЙ ТЕХНИКИ

...



ЛАБОРАТОРНАЯ РАБОТА №3

по дисциплине
“МЕТОДЫ И СРЕДСТВА ПРОГРАММНОЙ ИНЖЕНЕРИИ”

Вариант: 343628.

выполнили:

Студенты группы Р32311
Птицын Максим Евгеньевич
Воронина Дарья Сергеевна

Преподаватель
Бострикова Дарья Константиновна





Дарья
Константин-
овна,
будьте с
нами нежны

...



Содержание

Задание:

2 build.xml:

21

3

4





1. Задание: Дарья

Константина

Сценарий для утилиты Apache Ant, реализующий компиляцию, тести-
рование и упаковку в jar-архив кода проекта из лабораторной работы №3 по дисциплине

"Веб-программирование".

Каждый этап должен быть выделен в отдельный блок сценария; все переменные и

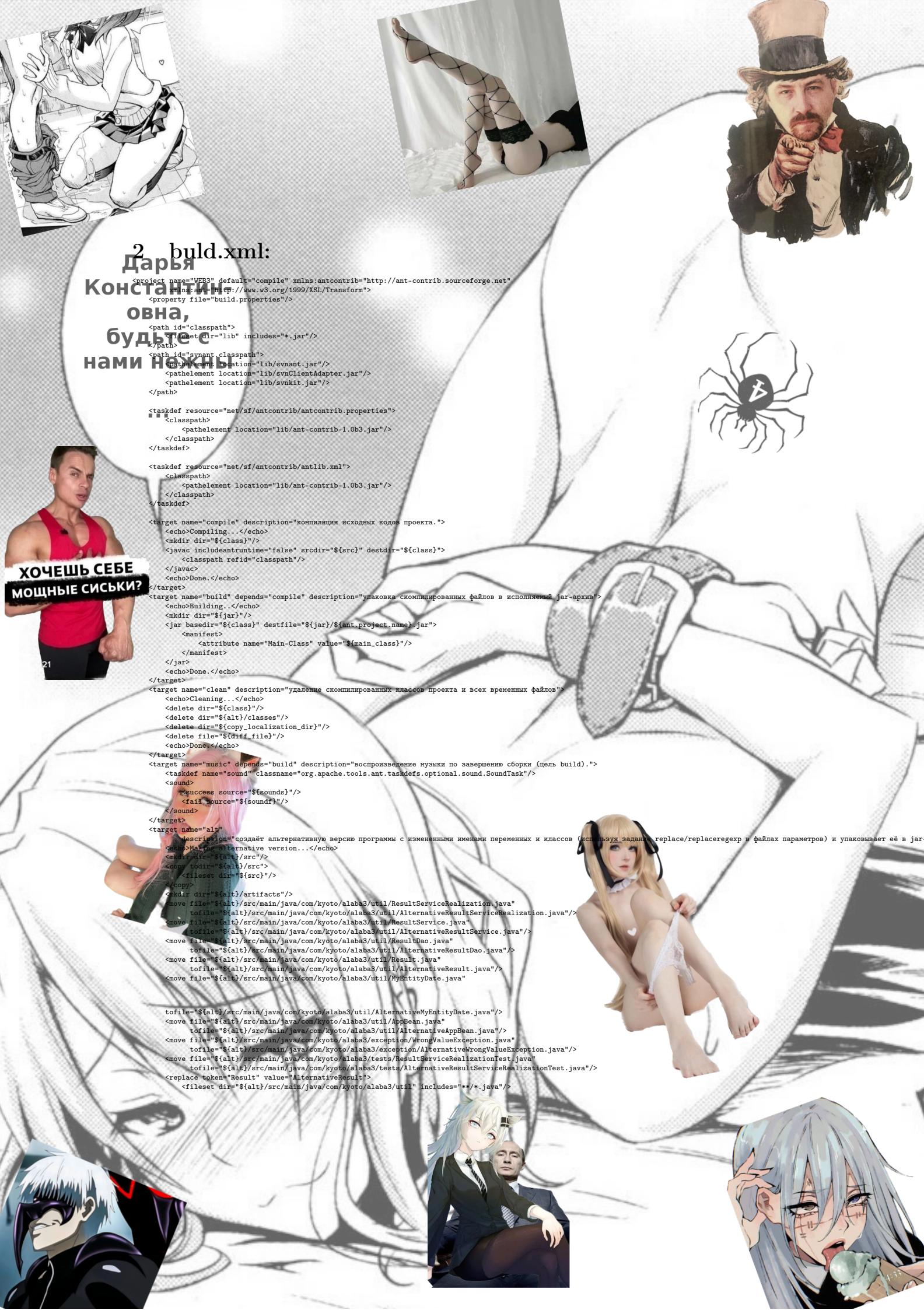
параметры используемые в сценарии, должны быть вынесены в отдельный файл па-

раметров; MANIFEST.MF должен содержать информацию о версии и о запускаемом

классе

Сценарий должен реализовывать следующие цели (targets):

- **compile** – компиляция исходных кодов проекта.
- **build** – компиляция исходных кодов проекта и их упаковка в исполняемый jar-архив. Компиляцию исходных кодов реализовать посредством вызова цели compile.
- **clean** – удаление скомпилированных классов проекта и всех временных файлов (если они есть).
- **test** – запуск junit-тестов проекта. Перед запуском тестов необходимо осуществить сборку проекта (цель build).
- **native2ascii** - преобразование native2ascii для копий файлов локализации (для тестирования сценария все строковые параметры необходимо вынести из классов в файлы локализации).
- **xml** - валидация всех xml-файлов в проекте.
- **music** - воспроизведение музыки по завершению сборки (цель build).
- **diff** - осуществляет проверку состояния рабочей копии, и, если изменения не касаются классов, указанных в файле параметров выполняет commit в репозиторий git.
- **alt** - создаёт альтернативную версию программы с изменёнными именами переменных и классов (используя задание replace/replaceregexp в файле параметров) и упаковывает её в jar-архив. Для создания jar-архива использует цель build.
- **history** - если проект не удается скомпилировать (цель compile), загружается предыдущая версия из репозитория svn. Операция повторяется до тех пор, пока проект не удастся собрать, либо не будет получена самая первая ревизия из репозитория. Если такая ревизия найдена, то формируется файл, содержащий результат операции diff для всех файлов, изменившихся в ревизии, следующей непосредственно за последней работающей.



2 build.xml:

**Дарья
Константина
овна,
будьте с
нами на**

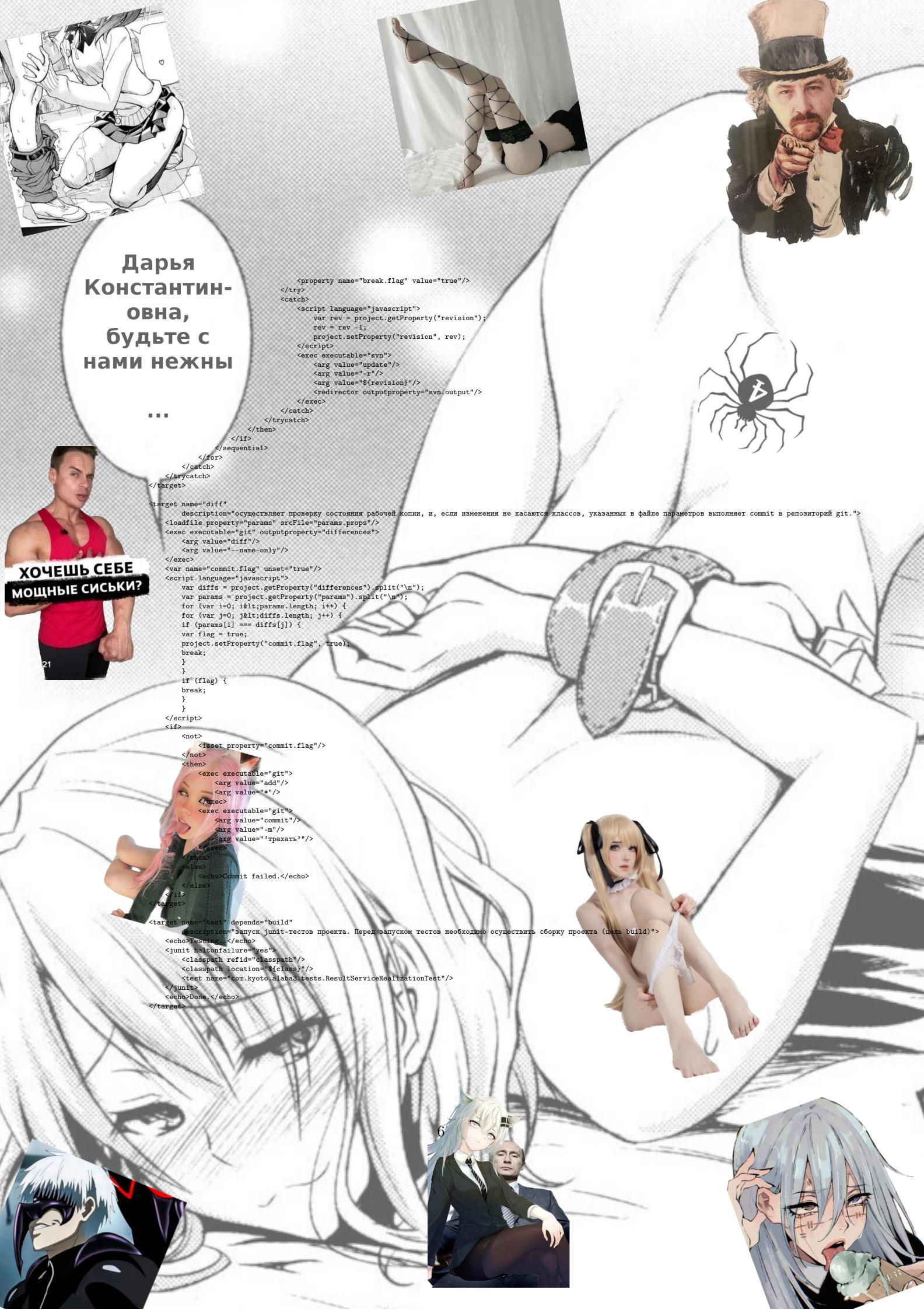
```
<project name="WEB3" default="compile" xmlns:antcontrib="http://ant-contrib.sourceforge.net"  
    xmlns:xsl="http://www.w3.org/1999/XSL/Transform">  
    <property file="build.properties"/>  
  
    <path id="classpath">  
        <pathelement location="lib/*.jar"/>  
    </path>  
    <path id="svnant_classpath">  
        <pathelement location="lib/svnant.jar"/>  
        <pathelement location="lib/svnClientAdapter.jar"/>  
        <pathelement location="lib/svnkit.jar"/>  
    </path>  
  
    <taskdef resource="net/sf/antcontrib/antcontrib.properties">  
        <classpath>  
            <pathelement location="lib/ant-contrib-1.0b3.jar"/>  
        </classpath>  
    </taskdef>  
  
    <taskdef resource="net/sf/antcontrib/antlib.xml">  
        <classpath>  
            <pathelement location="lib/ant-contrib-1.0b3.jar"/>  
        </classpath>  
    </taskdef>  
  
    <target name="compile" description="компиляция исходных кодов проекта.">  
        <echo>Compiling...</echo>  
        <mkdir dir="${class}" />  
        <javac includeantruntime="false" srcdir="${src}" destdir="${class}">  
            <classpath refid="classpath"/>  
        </javac>  
        <echo>Done.</echo>  
    </target>  
    <target name="build" depends="compile" description="упаковка скомпилированных файлов в исполняемый jar-архив">  
        <echo>Building...</echo>  
        <mkdir dir="${jar}" />  
        <jar basedir="${class}" destfile="${jar}/${ant.project.name}.jar">  
            <manifest>  
                <attribute name="Main-Class" value="${main_class}"/>  
            </manifest>  
        </jar>  
        <echo>Done.</echo>  
    </target>  
    <target name="clean" description="удаление скомпилированных классов проекта и всех временных файлов">  
        <echo>Cleaning...</echo>  
        <delete dir="${class}" />  
        <delete dir="${alt}/classes" />  
        <delete dir="${copy_localization_dir}" />  
        <delete file="${diff_file}" />  
        <echo>Done.</echo>  
    </target>  
    <target name="music" depends="build" description="воспроизведение музыки по завершению сборки (цель build).">  
        <taskdef name="sound" classname="org.apache.tools.ant.taskdefs.optional.sound.SoundTask"/>  
        <sound>  
            <success source="${sounds}"/>  
            <fail source="${soundfail}"/>  
        </sound>  
    </target>  
    <target name="alt" description="создаёт альтернативную версию программы с измененными именами переменных и классов (используя задание replace/replaceregexp в файлах параметров) и упаковывает её в jar-архив">  
        <echo>Making alternative version...</echo>  
        <mkdir dir="${alt}/src" />  
        <copy todir="${alt}/src" />  
            <fileset dir="${src}" />  
        <copy>  
            <mkjar dir="${alt}/artifacts" />  
            <move file="${alt}/src/main/java/com/kyoto/alaba3/util/ResultServiceRealization.java"  
                tofile="${alt}/src/main/java/com/kyoto/alaba3/util/AlternativeResultServiceRealization.java" />  
            <move file="${alt}/src/main/java/com/kyoto/alaba3/util/ResultService.java"  
                tofile="${alt}/src/main/java/com/kyoto/alaba3/util/AlternativeResultService.java" />  
            <move file="${alt}/src/main/java/com/kyoto/alaba3/util/ResultDao.java"  
                tofile="${alt}/src/main/java/com/kyoto/alaba3/util/AlternativeResultDao.java" />  
            <move file="${alt}/src/main/java/com/kyoto/alaba3/util/Result.java"  
                tofile="${alt}/src/main/java/com/kyoto/alaba3/util/AlternativeResult.java" />  
            <move file="${alt}/src/main/java/com/kyoto/alaba3/util/MyEntityDate.java"  
                tofile="${alt}/src/main/java/com/kyoto/alaba3/util/AlternativeMyEntityDate.java" />  
            <move file="${alt}/src/main/java/com/kyoto/alaba3/util/AppBean.java"  
                tofile="${alt}/src/main/java/com/kyoto/alaba3/util/AlternativeAppBean.java" />  
            <move file="${alt}/src/main/java/com/kyoto/alaba3/exception/WrongValueException.java"  
                tofile="${alt}/src/main/java/com/kyoto/alaba3/exception/AlternativeWrongValueException.java" />  
            <move file="${alt}/src/main/java/com/kyoto/alaba3/tests/resultServiceRealizationTest.java"  
                tofile="${alt}/src/main/java/com/kyoto/alaba3/tests/AlternativeResultServiceRealizationTest.java" />  
        <replace token="Result" value="AlternativeResult" />  
        <fileset dir="${alt}/src/main/java/com/kyoto/alaba3/util" includes="**/*.*" />  
    </target>
```



Дар Константина овна будьте нами ненави

```
</replace>
<replace token="Result" value="AlternativeResult">
    <fileset dir="${alt}/src/main/java/com/kyoto/alaba3/tests" includes="**/*.java"/>
</replace>
<replace token="MyEntityDate" value="AlternativeMyEntityDate">
    <fileset dir="${alt}/src/main/java/com/kyoto/alaba3/util" includes="**/*.java"/>
</replace>
<replace token="MyEntityDate" value="AlternativeMyEntityDate">
    <fileset dir="${alt}/src/main/java/com/kyoto/alaba3/tests" includes="**/*.java"/>
</replace>
<replace token="AppBean" value="AlternativeAppBean">
    <fileset dir="${alt}/src/main/java/com/kyoto/alaba3/util" includes="**/*.java"/>
</replace>
<replace token="WrongValueException" value="AlternativeWrongValueException">
    <fileset dir="${alt}/src/main/java/com/kyoto/alaba3/util" includes="**/*.java"/>
</replace>
<replace token="WrongValueException" value="AlternativeWrongValueException">
    <fileset dir="${alt}/src/main/java/com/kyoto/alaba3/tests" includes="**/*.java"/>
</replace>
<replace token="WrongValueException" value="AlternativeWrongValueException">
    <fileset dir="${alt}/src/main/java/com/kyoto/alaba3/exception" includes="**/*.java"/>
</replace>
<antcall target="compile">
    <param name="src" value="${alt}/src"/>
    <param name="class" value="${alt}/classes"/>
</antcall>
<antcall target="build" inheritRefs="false">
    <param name="src" value="${alt}/src"/>
    <param name="jar" value="${alt}/artifacts"/>
    <param name="class" value="${alt}/classes"/>
    <param name="main_class" value="${alt}/src/main/java/com/kyoto/alaba3/util/AlternativeAppBean.java"/>
</antcall>
<echo>Done.</echo>
</target>
<target name="xml" description="валидация всех xml-файлов в проекте.">
    <xmllvalidate file="src/main/webapp/WEB-INF/faces-config.xml"/>
    <xmllvalidate file="src/main/webapp/WEB-INF/web.xml"/>
    <xmllvalidate file="src/main/resources/META-INF/beans.xml"/>
    <xmllvalidate file="src/main/resources/META-INF/persistence.xml"/>
    <xmllvalidate file="src/main/resources/hibernate.cfg.xml"/>
</target>
<target name="history">
    <description>если проект не удаётся скомпилировать (цель compile), загружается предыдущая версия из репозитория svn. Операция повторяется до тех пор, пока проект не удастся собрать.</description>
    <!-- Set variables -->
    <exec executable="svn" outputproperty="revision">
        <arg value="info"/>
        <arg value="--show-item"/>
        <arg value="revision"/>
    </exec>
    <!-- Try to compile the project -->
    <trycatch>
        <try>
            <antcall target="compile"/>
        </try>
        <catch>
            <exec executable="svn">
                <arg value="update"/>
                <arg value="-r"/>
                <arg value="${revision}"/>
                <redirector outputproperty="svn.output"/>
            </exec>
            <uiname>break.flag</uiname>
            <param name="i" begin="${revision}" end="0" step="-1">
            <sequential>
                <if>
                    <not>
                        <isset property="break.flag"/>
                    </not>
                    <then>
                        <echo>Revision: ${revision}</echo>
                    <trycatch>
                        <try>
                            <antcall target="compile"/>
                            <property name="prev_revision" value="0"/>
                            <script language="javascript">
                                var rev = project.getProperty("revision");
                                rev = rev - 1;
                                project.setProperty("prev_revision", rev);
                            </script>
                            <exec executable="svn">
                                <arg value="--diff"/>
                                <arg value="-r/>
                                <arg value="${prev_revision};${revision}"/>
                                <arg value="--summarize"/>
                                <redirector output="${diff_file}"/>
                            </exec>
                        </try>
                    </trycatch>
                </if>
            </sequential>
        </catch>
    </trycatch>
</target>
```





Дарья Константин- овна, будьте с нами нежны

```
<property name="break.flag" value="true"/>
</try>
<catch>
<script language="javascript">
    var rev = project.getProperty("revision");
    rev = rev - 1;
    project.setProperty("revision", rev);
</script>
<exec executable="svn">
    <arg value="update"/>
    <arg value="r"/>
    <arg value="${revision}"/>
    <redirector outputproperty="svn.output"/>
</exec>
</catch>
</then>
</if>
</sequential>
</for>
</catch>
</trycatch>
</target>

<target name="diff">
    description="осуществляет проверку состояния рабочей копии, и, если изменения не касаются классов, указанных в файле параметров выполняет commit в репозиторий git."
    <loadfile property="params" srcFile="params.props"/>
    <exec executable="git" outputproperty="differences">
        <arg value="diff"/>
        <arg value="--name-only"/>
    </exec>
    <var name="commit.flag" unset="true"/>
    <script language="javascript">
        var diffs = project.getProperty("differences").split("\n");
        var params = project.getProperty("params").split("\n");
        for (var i=0; i<params.length; i++) {
            for (var j=0; j<diffs.length; j++) {
                if (params[i] === diffs[j]) {
                    var flag = true;
                    project.setProperty("commit.flag", true);
                    break;
                }
            }
            if (flag) {
                break;
            }
        }
    </script>
<if>
    <not>
        <isset property="commit.flag"/>
    </not>
    <then>
        <exec executable="git">
            <arg value="add"/>
            <arg value="*"/>
        </exec>
        <exec executable="git">
            <arg value="commit"/>
            <arg value="-m"/>
            <arg value="--traxhax'"/>
        </exec>
    </then>
    <else>
        <echo>Commit failed.</echo>
    </else>
</if>
</target>

<target name="test" depends="build">
    description="запуск junit-тестов проекта. Перед запуском тестов необходимо осуществить сборку проекта (шель build)">
    <echo>Testing...</echo>
    <junit haltonfailure="yes">
        <classpath refid="classpath"/>
        <classpath location="${class}" />
        <test name="com.kyoto.alaba3.tests.ResultServiceRealizationTest"/>
    </junit>
    <echo>Done.</echo>
</target>
```



Дары
Константина
овна,
будьте с
нами нежны

```
<target name="native2ascii">
<description>преобразование native2ascii для копий файлов локализации (для тестирования сценария все строковые параметры необходимо вынести из классов в файлы локализации).</description>
<mkdir dir="${copy_localization_dir}" />
<echo Native2ASCII coding...</echo>
<native2ascii encoding="EUCJIS" src="${localization_dir}" dest="${copy_localization_dir}" includes="**/*.loc"/>
<echo>Done.</echo>
</project>
```



21

