КИЇВСЬКИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ «КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

**Лабораторна робота №1**

З інженерії програмного забезпечення

**Тема**: «Підготовка програмного проекту»

залікова книжка № 3223

Виконав:

студент ІІ курсу ФІОТ

група ІО-32

Попенко Руслан

Перевірив:

Антонюк А. І.

Київ-2014

**Мета:** Отримання базових навичок з використання мови XML. Вивчення структури типового програмного проекту, форматів стандартних файлів опису проекту. Вивчення формату JAR. Здобуття навичок з використання засобів автоматизації процесу збірки програмних проектів на мові Java - Apache ANT (Another Neat Tool). Розробка програмного проекту на основі типового прикладу.

1. **Завдання.**

1. Вивчити синтаксис та структуру мови XML. Вільно володіти поняттями тег, елемент, атрибут. Вміти редагувати XML-файли у текстовому редакторі.

2. Ознайомитись з призначенням і структурою архівів JAR. Вміти формувати архіви JAR та запускати на виконання Java-класи з архіву JAR за допомогою засобів командної строки. Знати призначення підпису архіву JAR.

3. Ознайомитись з засобом автоматизації збірки проектів ANT. Вивчити призначення і структуру файлу build.xml, його структурні компоненти - цілі, завдання, залежності, тощо.

4. З сайту лабораторії завантажити архів типового проекту для середовища Eclipse (template.zip). Ознайомитися з структурою каталогів типового проекту, XML-файлами опису проекту (.project, .classpath, build.xml).

5. Імпортувати типовий проект до робочого простору (workspace) середовища Eclipse. В пакеті com.lablll запустити на виконання (run) клас TestMain. Ознайомитися з засобами використання ANT у середовищі Eclipse - редагування файлу build.xml, виконання цілей у відображенні (View) ANT. Виконати цілі ANT з середовища Eclipse та з командного рядка.

6. Модифікувати типовий проект для його використання у наступних лабораторних роботах. Обов’язково замінити назву і автора проекту в файлах опису проекту (.project, build.xml). Модифікувати файл build.xml таким чином, щоб у ньому були всі потрібні цілі, вільно володіти призначенням кожної цілі.

7. Розробити та перевірити в eclipse нову ціль в файлі build.xml згідно варіанту.

Мій варіант **1**

**Видалити з проекту всі файли з розширеннями tmp, jar,class крім тих, що починаються з літери “а”.**

1. **Роздруківки файлів**

**build.xml**

<project name="EngyneerSoftware" default="signJAR" basedir=".">

<!-- ================== Application Properties =================== -->

<property name="app.name" value="EngyneerSoftware" />

<property name="app.version" value="0.1" />

<property name="app.title" value="Template Application" />

<property name="app.author" value="Ruslan Popenko" />

<property name="app.company" value="Lab111 - http://www.comsys.ntu-kpi.kiev.ua/lab111" />

<property name="sourceDir" value="src/main/java" />

<property name="outputDir" value="out/eclipse-classes" />

<property name="buildDir" value="out/build" />

<property name="resourceDir" value="src/res" />

<property name="libDir" value="lib" />

<!-- ================== JAR Properties =================== -->

<property name="jar.mainClass" value="com.lab111.TestMain" />

<property name="jar.name" value="template.jar" />

<property name="jar.keyStore" value="${basedir}/out/tempKey.store" />

<property name="jar.keyPass" value="telpat" />

<property name="jar.keyAlias" value="tempAlias" />

<!-- ==================== Compilation Control Options ================== -->

<property name="compile.debug" value="true" />

<property name="compile.deprecation" value="false" />

<property name="compile.optimize" value="true" />

<path id="compile.classpath">

<fileset dir="${libDir}">

<include name="\*.jar" />

</fileset>

</path>

<!-- ==================== Clean Target ============================== -->

<target name="clean" description="Clean build dirs">

<delete dir="${buildDir}" />

</target>

<!-- ==================== Prepare Target =========================== -->

<target name="prepare" depends="clean" description="Prepare build dirs">

<mkdir dir="${buildDir}" />

<mkdir dir="${buildDir}/output" />

<mkdir dir="${buildDir}/web-apps" />

</target>

<!-- ==================== Compile Target ============================ -->

<target name="compile" depends="prepare" description="Compile Java sources">

<javac srcdir="${sourceDir}" destdir="${buildDir}/output"

debug="${compile.debug}"

deprecation="${compile.deprecation}"

optimize="${compile.optimize}"

target="1.7"

source="1.7">

<classpath refid="compile.classpath" />

</javac>

<copy todir="${buildDir}/output">

<fileset dir="${sourceDir}" excludes="\*\*/\*.java" />

</copy>

</target>

<!-- ==================== Create JAR archive Target ==================== -->

<target name="createJAR" depends="compile" description="Create JAR archive">

<jar destfile="${buildDir}/${jar.name}" basedir="${outputDir}">

<manifest>

<attribute name="Created-By" value="${app.author} - (${app.company})" />

<attribute name="Built-By" value="${user.name}" />

<attribute name="Main-Class" value="${jar.mainClass}" />

<section name="${app.name}">

<attribute name="Specification-Title" value="${app.title}" />

<attribute name="Specification-Version" value="${app.version}" />

<attribute name="Specification-Vendor" value="${app.company}" />

<attribute name="Implementation-Title" value="${app.name}" />

<attribute name="Implementation-Version" value="${app.version}" />

<attribute name="Implementation-Vendor" value="${app.company}" />

</section>

</manifest>

</jar>

</target>

<!-- ==================== Generate Key for JAR signing Target ============ -->

<target name="generateKey" description="Generates Key for JAR signing">

<delete failonerror="false" file="${jar.keyStore}" />

<genkey keystore="${jar.keyStore}" alias="${jar.keyAlias}" storepass="${jar.keyPass}" validity="720" keyalg="RSA">

<dname>

<param name="CN" value="${app.company}" />

<param name="OU" value="${app.title}" />

<param name="O" value="${app.company}" />

</dname>

</genkey>

</target>

<!-- ==================== FixStyle Target ================== -->

<target name="fixstyle" description="Fix Style in source code">

<fixcrlf srcdir="${basedir}" tab="remove" tablength="2" includesfile="fixstyle.list" />

</target>

<!-- ==================== JAR Singing Target ================================== -->

<target name="signJAR" depends="createJAR,generateKey" description="Signing JAR archive">

<exec dir="${buildDir}" executable="jarsigner">

<arg line="-keystore ${jar.keyStore} -storepass ${jar.keyPass} ${jar.name} ${jar.keyAlias}" />

</exec>

<delete file="${buildDir}/myKeystore" />

</target>

<!-- ==================== Make zip Target =========== -->

<target name="make-zip-project" description="Zip all project tree from basedir">

<tstamp />

<zip destfile="../${app.name}-${DSTAMP}-${TSTAMP}.zip" basedir="${basedir}" excludes="out/\*\*" />

</target>

<!-- ==================== Make lab delete target ====================== -->

<target name="delete-files" description="Delete jar, tmp, class file except first a in name">

<delete>

<fileset dir="." includes="\*\*/\*.jar" excludes="\*\*/a\*.jar" />

<fileset dir="." includes="\*\*/\*.tmp" excludes="\*\*/a\*.tmp" />

<fileset dir="." includes="\*\*/\*.class" excludes="\*\*/a\*.class" />

</delete>

</target>

</project>

**.project**

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

<**projectDescription**>

<**name**>EngyneerSoftware</**name**>

<**comment**></**comment**>

<**projects**>

</**projects**>

<**buildSpec**>

<**buildCommand**>

<**name**>org.eclipse.jdt.core.javabuilder</**name**>

<**arguments**>

</**arguments**>

</**buildCommand**>

<**buildCommand**>

<**name**>com.atlassw.tools.eclipse.checkstyle.CheckstyleBuilder</**name**>

<**arguments**>

</**arguments**>

</**buildCommand**>

</**buildSpec**>

<**natures**>

<**nature**>org.eclipse.jdt.core.javanature</**nature**>

<**nature**>com.atlassw.tools.eclipse.checkstyle.CheckstyleNature</**nature**>

</**natures**>

</**projectDescription**>

**.classpath**

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

<**classpath**>

<**classpathentry** kind=*"src"* path=*"src/main/java"*/>

<**classpathentry** kind=*"src"* path=*"src/test/java"*/>

<**classpathentry** kind=*"con"* path=*"org.eclipse.jdt.launching.JRE\_CONTAINER"*/>

<**classpathentry** kind=*"output"* path=*"out/eclipse-classes"*/>

</**classpath**>