

Übungen zu Software Engineering

Wintersemester 2022/23

Blatt 7

Basis des Aufgabenblattes

In jeder Aufgabe dieses Übungsblattes sollen Sie je ein Build-Werkzeug auf die Java-Dateien `Main.java`, `A.java` und `B.java` anwenden. Laden Sie sich die Dateien dazu herunter und ordnen Sie sie entsprechend dem jeweiligen Build-Werkzeug korrekt in Ihrem Arbeitsverzeichnis an. Es empfiehlt sich, für jede Aufgabe ein neues Arbeitsverzeichnis zu erstellen. Die Klasse `Main.java` verfügt zudem über eine externe Abhängigkeit zu der Bibliothek `org.apache.commons.lang3`, die Sie sich u.a. hier herunterladen können: <https://mvnrepository.com/artifact/org.apache.commons/commons-lang3>.

Aufgabe 7.1: Make (25 Punkte)

Erstellen Sie ein `Makefile` innerhalb Ihres Arbeitsverzeichnisses. Mit dem Aufruf von `make` soll der Quellcode in das neue Verzeichnis `bin` kompiliert und aus den `.class`-Dateien eine ausführbare `jar`-Datei im Basis-Verzeichnis erzeugt werden. Mit dem Aufruf der `jar`-Datei soll die darin enthaltene `Main.class` ausgeführt werden. Die `.class`-Dateien sollen immer dann neu erzeugt werden, wenn sich die Quellcode-Dateien verändert haben. Analog dazu soll die `jar`-Datei neu erzeugt werden, wenn sich die `.class`-Dateien verändert haben.

Insgesamt soll Ihr `Makefile` mindestens folgende *targets* enthalten:

- jar:** Erzeugt die ausführbare `jar`-Datei `MyProject.jar` inklusive der Manifest-Datei und aller `.class`-Dateien, falls diese noch nicht existiert oder die `.class`-Dateien aktualisiert wurden. Die `org.apache.commons.lang3` Bibliothek soll dabei (in dieser Aufgabe) außerhalb der erzeugten Jar liegen.
- run:** Erzeugt `MyProject.jar` falls sie noch nicht vorliegt, oder die `.class`-Dateien aktualisiert wurden und führt sie aus.
- clean:** Löscht alle durch das `Makefile` bisher erzeugten Dateien.
- doc:** Erzeugt die Java-Dokumentation in dem Verzeichnis `doc`, solange sie noch nicht existiert oder die Quellcode-Dateien verändert wurden.

Neben dem von Ihnen erstellten `Makefile` und dem vorgegebenen Quellcode sollen alle weiteren Dateien nur durch das `Makefile` erzeugt werden.

Um zu überprüfen, ob Sie alle Java-Quellcode-Dateien korrekt ansprechen und kompilieren, sollten Sie diese mit der Option `-implicit:none` einzeln übersetzen.

Ein ausführliches Handbuch zu `make` finden Sie unter <http://www.gnu.org/software/make/manual/>

Aufgabe 7.2: Apache Ant (17 Punkte)

Erstellen Sie ein Ant-File, mit welchem sich die gleichen *targets* mit vergleichbarem Ergebnis wie in der Lösung des Makefiles von der vorherigen Aufgabe ausführen lassen. Allerdings soll im Gegensatz zu Aufgabe 7.1 für das Target `jar` die `org.apache.commons.lang3` Bibliothek in der Jar enthalten sein. Hierdurch soll sich die erstellte Jar ohne Abhängigkeiten zu weiteren Dateien ausführen lassen.

Aufgabe 7.3: Maven (21 Punkte)

Erstellen Sie ein Maven-Projekt mit dem gegebenen Quellcode als Inhalt. Passen Sie die `pom.xml` derart an, dass das Projekt explizit mit der Java-Version 11 kompiliert wird. Sorgen Sie auch dafür, dass beim Ausführen von `package` ein ausführbares `jar`-File mit der Klasse `Main` als `Main-class` erzeugt wird. Die Jar soll dabei die `org.apache.commons` Bibliothek beinhalten, wie es z.B. mit dem Plugin `maven-shade-plugin` möglich ist. Des Weiteren soll beim Ausführen des Befehls `mvn javadoc:javadoc` die JavaDoc Dokumentation für das Maven-Projekt generiert werden.

Aufgabe 7.4: Gradle (17 Punkte)

Erstellen Sie ein Gradle-Projekt mit dem gegebenen Quellcode als Inhalt. Passen Sie die Datei `build.gradle` derart an, dass das Projekt explizit mit der Java-Version 11 kompiliert wird. Dabei sollen Sie die Sprache Groovy zum Anpassen benutzen. Nutzen Sie zudem die Gradle Task `installDist`, um ein Startup-Skript für das Programm zu erzeugen. Erklären Sie Ihrem Tutor zudem, wie Sie die JavaDoc Dokumentation für das Projekt erzeugen können.

Aufgabe 7.5: Fragen (20 Punkte)

Ihr Tutor stellt Ihnen Fragen zur Veranstaltung.