Apache Ant Java build tool

CONCEPTS

Ant reads build.xml in the current directory. Run with no target and it runs the project's default target.

Project build.xml, default target
- **Target** named step, depends on others
  - **Task** javac, jar, copy ...

Properties are immutable: the first definition wins, so -D on the command line overrides build.xml.

RUN

ant
Run the project's default target

ant *<target>* **[***<target>***...]**
Run targets in order

ant -projecthelp / ant -p
List targets that have a description

ant -buildfile *<file>* **/ ant -f** *<file>*
Use a different build file

ant -find [*<file>***] / ant -s [***<file>***]**
Search parent dirs for the build file

ant -version
Print Ant version

ant -diagnostics
Print Java, classpath and Ant diagnostics

ant -help / ant -h
Show command line options

ant -keep-going / ant -k
Continue with targets not depending on a failure

PROPERTIES AND OUTPUT

ant -D*<name>***=***<value>*
Set a property from the command line

ant -quiet / ant -q
Less output

ant -verbose / ant -v
Extra output

ANT_OPTS="-Xmx1g" ant
Pass options to the JVM

BUILD FILE BASICS

<project name="<n>" **default="***<target>***" basedir="."***>*
Root element

<target name="<n>" **depends="***<a>***,***<b>***" description="***<text>***"***>*
Define a target

<property name="<n>" **value="***<v>***"/>**
Define a property

<property file="build.properties"/>
Load properties from a file

${*<name>***}**
Reference a property

<path id="<id>"><fileset dir="lib" includes=".jar"/></path>*
Reusable classpath

<fileset dir="<d>" **includes="**/*.java" excludes="***<pat>***"/>**
Select files by pattern

COMMON TASKS

*<mkdir dir="<d>"***/>**
Create a directory

*<delete dir="<d>"***/>**
Remove a directory tree

*<copy todir="<d>"><fileset dir="<s>"***/>***</copy>***
Copy files

<javac srcdir="src" destdir="build/classes" classpathref="<id>" **includeantruntime="false"/>**
Compile Java sources

<jar destfile="dist/<n>.jar" **basedir="build/classes"/>**
Build a jar

*<manifest><attribute name="Main-Class" value="<cls>"***/>***</manifest>***
Set jar manifest (inside <jar>)

<java classname="<cls>" **classpathref="***<id>***" fork="true"/>**
Run a Java class

<junit printsummary="yes" haltonfailure="yes">
Run JUnit tests