

ANT - PROPERTY FILES

http://www.tutorialspoint.com/ant/ant_property_files.htm

Copyright © tutorialspoint.com

Setting properties directly in the build file is fine, if you are working with a handful of properties. However, for a large project, it makes sense to store the properties in a separate property file.

Storing the properties in a separate file offers the following benefits:

- It allows you to reuse the same build file, with different property settings for different execution environment. For example, build properties file can be maintained separately for DEV, TEST, and PROD environments.
- It is useful when you do not know the values for a property *in a particular environment* up-front. This allows you to perform the build in other environments where the property value is known.

There is no hard and fast rule, but typically the property file is named **build.properties** and is placed along-side the **build.xml** file. You could create multiple build properties files based on the deployment environments - such as **build.properties.dev** and **build.properties.test**.

The contents of the build property file are similar to the normal java property file. They contain one property per line. Each property is represented by a name and a value pair. The name and value pairs are separated by an equals = sign. It is highly recommended that the properties are annotated with proper comments. Comments are listed using the hash # character.

The following example shows a **build.xml** file and its associated **build.properties** file:

build.xml

```
<?xml version="1.0"?>
<project name="Hello World Project" default="info">

    <property file="build.properties"/>

    <target name="info">
        <echo>Apache Ant version is ${ant.version} - You are at ${sitename} </echo>
    </target>

</project>
```

build.properties

```
# The Site Name
sitename=www.tutorialspoint.com
buildversion=3.3.2
```

In the above example, **sitename** is a custom property which is mapped to the website name. You can declare any number of custom properties in this fashion. Another custom property listed in the above example is the **buildversion**, which, in this instance refers to the version of the build.

In addition to the above, Ant comes with a number of predefined build properties, which are listed in the previous section, but is represented below once again.

Properties	Description
ant.file	The full location of the build file.
ant.version	The version of the Apache Ant installation.
basedir	The basedir of the build, as specified in the basedir attribute of the project element.
ant.java.version	The version of the JDK that is used by Ant.

ant.project.name	The name of the project, as specified in the name attribute of the project element.
ant.project.default-target	The default target of the current project.
ant.project.invoked-targets	Comma separated list of the targets that were invoked in the current project.
ant.core.lib	The full location of the Ant jar file.
ant.home	The home directory of Ant installation.
ant.library.dir	The home directory for Ant library files - typically ANT_HOME/lib folder.

The example presented in this chapter uses the **ant.version** built-in property.

Loading [Mathjax]/jax/output/HTML-CSS/jax.js