

Description

This task runs Checkstyle over specified Java files. The task has been tested using ANT 1.5. The latest version of checkstyle can be found at <http://checkstyle.sourceforge.net/> . This task is included in the checkstyle distribution.

Installation

The easiest way is to include `checkstyle-6.2-all.jar` in the classpath. This contains all the classes required to run Checkstyle. Alternatively, you must include the compile third party dependencies listed in [Project Dependencies](#) in the classpath.

To use the task in a build file, you will need the following taskdef declaration:

```
<taskdef resource="checkstyletask.properties"
  classpath="/path/to/checkstyle-6.2-all.jar"/>
```

Or, assuming that Checkstyle is in the global classpath (not recommended), then you will need the following taskdef declaration:

```
<taskdef resource="checkstyletask.properties"/>
```

Or if you use Ant 1.6 and later and assuming that Checkstyle is in the library search path, then you may use antlib feature of Ant (see <http://ant.apache.org/manual/Types/antlib.html> for more details). For example:

```
<project name="foo" ...
  xmlns:cs="antlib:com.puppcrawl.tools.checkstyle">
...
  <cs:checkstyle>
    ...
  </cs:checkstyle>
...
</project>
```

Parameters

Attribute	Description	Required
file	File to run checkstyle on.	One of either <i>file</i>

		or at least one nested <i>fileset</i> element
config	Specifies the location of the file that defines the configuration modules. The location can either be a filesystem location, or a name passed to the <code>ClassLoader.getResource()</code> method. See here for a description of how to define a configuration.	Exactly one of config or configURL
configURL	Specifies a URL that defines the configuration modules. See here for a description of how to define a configuration.	Exactly one of config or configURL
properties	Specifies a file that contains properties for expanded property values of the configuration. Ant properties (like <code>\${basedir}</code>) and nested property elements override the properties in this file.	No
failOnViolation	Specifies whether the build will continue even if there are violations. Defaults to "true".	No
failureProperty	The name of a property to set in the event of a violation.	No
maxErrors	The maximum number of errors that are tolerated before breaking the build or setting the failure property. Defaults to "0".	No
maxWarnings	The maximum number of warnings that are tolerated before breaking the build or setting the failure property. Defaults to "2147483647", i.e. <code>Integer.MAX_VALUE</code> .	No
classpath	The classpath to use when looking up classes. Defaults to the current classpath.	No
classpathref	The classpath to use when looking up classes, given as a reference to a path defined elsewhere.	No
omitIgnoredModules	For efficiency, Checkstyle does not invoke modules with a configured severity of "ignore" (since their output would be ignored anyway). A small number of modules may choose to log above their configured severity level and so always need to be invoked. This settings specifies that behaviour. Defaults to "true".	No

Note that the `packageNamesFile` parameter has been dropped for Checkstyle 5.0, because of significant changes regarding package name file handling. See for details.

Nested Elements

This task supports the nested elements `<fileset>`, `<classpath>`, `<formatter>`, and `<property>`.

The parameters for the `<formatter>` element are:

Attribute	Description	Required
type	The type of output to generate. The valid values are: <ul style="list-style-type: none"> <code>plain</code> - specifies the DefaultLogger <code>xml</code> - specifies the XMLLogger 	No

Defaults to "plain".

toFile	The file to write output to. Defaults to standard output. Note, there is no way to explicitly specify standard output.	No
useFile	Boolean that determines whether output should be sent to a file. Default is true.	No

A `<property>` element provides a property for [expanded property values](#) of the configuration. The parameters for the `<property>` element are:

Attribute	Description	Required
key	The key for the property.	Yes
value	The value of the property specified as a string.	Either <i>value</i> or <i>file</i>
file	The value of the property specified as a file. This is great for specifying file names relative to the ANT build file.	Either <i>value</i> or <i>file</i>

Examples

Run checkstyle with configuration file docs/sun_checks.xml on a single file

```
<checkstyle config="docs/sun_checks.xml" file="Check.java"/>
```

Run checkstyle on a set of Java files using site-wide configuration and an expanded property value

```
<checkstyle config="/path/to/site/sun_checks.xml">
  <fileset dir="src/checkstyle" includes="**/*.java"/>

  <!-- Location of cache-file. Something that is project specific -->
  <property key="checkstyle.cache.file" file="target/cachefile"/>
</checkstyle>
```

Run checkstyle on a set of files and output messages to standard output in plain format, and a file in XML format

```
<checkstyle config="docs/sun_checks.xml">
  <fileset dir="src/checkstyle" includes="**/*.java"/>
  <formatter type="plain"/>
  <formatter type="xml" toFile="build/checkstyle_errors.xml"/>
</checkstyle>
```

Run checkstyle with configuration file docs/sun_checks.xml on a file and provide a package names file

```
<checkstyle config="docs/sun_checks.xml"
    packageNamesFile="myPackageNames.xml"
    file="Check.java"/>
```

Run checkstyle in an automated build and send an email report if style violations are detected

```
<target name="checkstyle"
    description="Generates a report of code convention violations.">

    <checkstyle config="docs/sun_checks.xml"
        failureProperty="checkstyle.failure"
        failOnViolation="false">
        <formatter type="xml" tofile="checkstyle_report.xml"/>
        <fileset dir="src" includes="**/*.java"/>
    </checkstyle>

    <style in="checkstyle_report.xml" out="checkstyle_report.html" style="checkstyle.xsl"/>

</target>

<!-- run this target as part of automated build -->
<target name="checkstyle-nightly"
    depends="checkstyle"
    if="checkstyle.failure"
    description="Sends email if checkstyle detected code conventions violations.">

    <!-- use your own server and email addresses below. See Ant documentation for details -->

    <mail from="qa@some.domain"
        tolist="someone@some.domain,someoneelse@some.domain"
        mailhost="mailbox.some.domain"
        subject="Checkstyle violation(s) in project ${ant.project.name}"
        files="checkstyle_report.html"/>

</target>
```