Contents

Preface

Audience iv
Documentation Accessibility iv
Diversity and Inclusion iv
Related Documents iv
Conventions v

1 JavaDoc Tool

JavaDoc Features 1-1
This guide provides information about using the JavaDoc tool.

Audience

This document is intended for JavaDoc tool users. Users who are developing JavaDoc content should also see the Documentation Comment Specification for the Standard Doclet for detailed information required to create javadoc content.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers’ existing technologies and the need to ensure continuity of service as Oracle’s offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Related Documents

- See JDK 17 Documentation.
- javadoc Command man page for users running the tool to generate API documentation
- Documentation Comment Specification for the Standard Doclet (JDK 17) for authors writing content API docs
- Javadoc Search Specification for authors writing content API docs
Conventions

The following text conventions are used in this document:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>boldface</strong></td>
<td>Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.</td>
</tr>
<tr>
<td><em>italic</em></td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td><strong>monospace</strong></td>
<td>Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>
1

JavaDoc Tool

The JavaDoc tool is a program that reads a collection of source files into an internal form. The structure is: (source-files)->[javadoc-tool:doclet]->(generated files).

The tool uses a JavaDoc doclet in a manner similar to a pluggable back end to analyze the internal form of the source files and generate output files.

The content of the generated files is specific to the doclet. The standard doclet generates HTML output, but a different doclet could generate other output, such as a report of misspelled words or grammatical errors.

If you specify a doclet other than the standard doclet, then the descriptions in this guide might not apply to the operation of that doclet or the files (if any) that are generated.

To use the JavaDoc tool, you must:

• Use source code that contains Java documentation comments.
• Run the javadoc tool with a doclet to analyze the documentation comments and any other special tags. If you don’t specify a doclet in the command, the Standard Doclet is used by default.

In addition to the descriptions in this guide, JavaDoc tool users and content developers should use the following documentation:

• Doc Comment Tag Specification (for authors writing content API docs)
• Man page (for users running the tool to generate API docs)
• The Help page in the generated docs bundle and associated specs, such as the Search Specification (for end-user readers of API docs)

JavaDoc Features

JavaDoc features include the following: JavaDoc search, support for generating HTML output, support for documentation comments in module systems, and a new Doclet API introduced in JDK 9.

Search

When the JavaDoc tool runs the standard doclet, it generates output that enables users to search the generated documentation for elements and additional key phrases defined in the generated API documentation. Search results include matching characters from any position in the search string. The Search facility can also provide page redirection based on user selection.

Note:

See the Javadoc Search Specification for detailed information about using Search.
Module System

The `javadoc` tool supports documentation comments in module declarations. Some JavaDoc command-line options enable you to specify the set of modules to document and generate a new summary page for any modules being documented. It has new command-line options to configure the set of modules to be documented and generates a new summary page for any modules being documented.

**Note:**

Users running the JavaDoc tool to generate API documentation should see the `javadoc Command` man page for detailed information.

HTML Support

The standard doclet only generates HTML output.

**Note:**

Authors writing content for API docs should see the Documentation Comment Specification for the Standard Doclet (JDK 17) for detailed information about Syntax and Tagging requirements for generating JavaDoc HTML files.

Doclet API

The Doclet API uses powerful APIs that represent all of the latest language features.

**Note:**

See `Module jdk.javadoc` for detailed information.