

---

# **javaproperties-cli**

*Release 0.7.0*

**John T. Wodder II**

2020 Jul 16



## COMMANDS

1	<code>javaproperties</code>	3
2	<code>json2properties</code>	7
3	<code>properties2json</code>	9
4	Changelog	11
5	Installation	15
6	Quickstart	17
7	Indices and tables	19
	Python Module Index	21
	Index	23



[GitHub](#) | [PyPI](#) | [Documentation](#) | [Issues](#) | [Changelog](#)

`javaproperties-cli` is a wrapper around the `javaproperties` package (from which it was split off) that provides programs for basic command-line manipulation of Java `.properties` files, including getting, setting, & deleting values and converting to & from JSON.

Currently, the programs only support `.properties` files in the simple line-oriented format, not the XML variant.



## JAVAPROPERTIES

### 1.1 NAME

**javaproperties** — Basic manipulation of Java `.properties` files

### 1.2 SYNOPSIS

```
javaproperties get      [<OPTIONS>] <file> <key> ...
javaproperties select [<OPTIONS>] <file> <key> ...
javaproperties set   [<OPTIONS>] <file> <key> <value>
javaproperties delete [<OPTIONS>] <file> <key> ...
javaproperties format [<OPTIONS>] [<file>]
```

### 1.3 get

```
javaproperties get [<OPTIONS>] <file> <key> ...
```

Print out the values of the given keys in the given `.properties` file. Each value is printed out on a separate line with escape sequences interpolated.

If you want the output to also be in `.properties` format, see *select*.

#### 1.3.1 Options

**-d** <value>, **--default-value** <value>

Default value for undefined keys. If this option is not specified, keys requested on the command line that are not defined in either the main `.properties` file or the `--defaults` file will (unless the `--quiet` option is given) cause a warning to be printed to `stderr` and the command to exit with a failure status.

**-D** <file>, **--defaults** <file>

`.properties` file of default values. If this option is specified, keys requested on the command line that are not defined in the main `.properties` file will be looked up in this file.

**-e**, **--escaped**

Parse the keys and default value specified on the command line for `.properties`-style escape sequences (specifically, those supported by `javaproperties.unescape`)

- E** <encoding>, **--encoding** <encoding>  
Specifies the encoding of the input file(s); default value: `iso-8859-1` (a.k.a. Latin-1). Output always uses the locale's encoding.
- q**, **--quiet**  
New in version 0.7.0.  
Do not warn about or fail due to missing keys

## 1.4 select

```
javaproperties select [<OPTIONS>] <file> <key> ...
```

Print out the key-value entries in the given `.properties` file for the given keys. The output is in `.properties` format, reformatted as though by *format*.

### 1.4.1 Options

- A**, **--ascii**  
New in version 0.6.0.  
Escape all non-ASCII characters in the output with `\uXXXX` escape sequences. This overrides `--unicode`. This is the default behavior.
- d** <value>, **--default-value** <value>  
Default value for undefined keys. If this option is not specified, keys requested on the command line that are not defined in either the main `.properties` file or the `--defaults` file will (unless the `--quiet` option is given) cause a warning to be printed to `stderr` and the command to exit with a failure status.
- D** <file>, **--defaults** <file>  
`.properties` file of default values. If this option is specified, keys requested on the command line that are not defined in the main `.properties` file will be looked up in this file.
- e**, **--escaped**  
Parse the keys and default value specified on the command line for `.properties`-style escape sequences (specifically, those supported by `javaproperties.unescape`)
- E** <encoding>, **--encoding** <encoding>  
Specifies the encoding of the input and output files; default value: `iso-8859-1` (a.k.a. Latin-1)
- o** <file>, **--outfile** <file>  
Write output to this file instead of standard output
- s** <sep>, **--separator** <sep>  
Use <sep> as the key-value separator in the output; default value: `=`
- q**, **--quiet**  
New in version 0.7.0.  
Do not warn about or fail due to missing keys
- U**, **--unicode**  
New in version 0.6.0.  
Output non-ASCII characters literally, except for characters that are not supported by the output encoding, which are escaped with `\uXXXX` escape sequences. This overrides `--ascii`.



## 1.5 set

```
javaproperties set [<OPTIONS>] <file> <key> <value>
```

Set the value of <key> in the `.properties` file <file> to <value> and output the results. The other entries in the file (including comments, possibly not including the timestamp; see below) will be left as-is.

### 1.5.1 Options

**-A, --ascii**

New in version 0.6.0.

Escape all non-ASCII characters in the new key & value with `\uXXXX` escape sequences on output. This overrides `--unicode`. This is the default behavior.

**-e, --escaped**

Parse <key> and <value> for `.properties`-style escape sequences (specifically, those supported by `javaproperties.unescape`)

**-E <encoding>, --encoding <encoding>**

Specifies the encoding of the input and output files; default value: `iso-8859-1` (a.k.a. Latin-1)

**-o <file>, --outfile <file>**

Write output to this file instead of standard output

**-s <sep>, --separator <sep>**

Separate <key> and <value> in the output with <sep>; default value: `=`

**-T, --preserve-timestamp**

Do not modify the timestamp in the `.properties` file. By default, if a timestamp is found, it is updated to the current time, even if the rest of the file is unchanged.

**-U, --unicode**

New in version 0.6.0.

Output non-ASCII characters in the new key & value literally, except for characters that are not supported by the output encoding, which are escaped with `\uXXXX` escape sequences. This overrides `--ascii`.

## 1.6 delete

```
javaproperties delete [<OPTIONS>] <file> <key> ...
```

Remove all entries for the given keys from the given `.properties` file and output the results. The other entries in the file (including comments, possibly not including the timestamp; see below) will be left as-is.

## 1.6.1 Options

- e, --escaped**  
Parse the keys specified on the command line for `.properties`-style escape sequences (specifically, those supported by `javaproperties.unescape`)
- E <encoding>, --encoding <encoding>**  
Specifies the encoding of the input and output files; default value: `iso-8859-1` (a.k.a. Latin-1)
- o <file>, --outfile <file>**  
Write output to this file instead of standard output
- T, --preserve-timestamp**  
Do not modify the timestamp in the `.properties` file. By default, if a timestamp is found, it is updated to the current time, even if the rest of the file is unchanged.

## 1.7 format

```
javaproperties format [<OPTIONS>] [<file>]
```

Normalize the formatting of the given `.properties` file (or standard input if no file is given) and output the results. All comments, excess whitespace, invalid escapes, and duplicate keys are removed, and the entries are sorted lexicographically.

### 1.7.1 Options

- A, --ascii**  
New in version 0.6.0.  
  
Escape all non-ASCII characters in the output with `\uXXXX` escape sequences. This overrides `--unicode`. This is the default behavior.
- E <encoding>, --encoding <encoding>**  
Specifies the encoding of the input and output files; default value: `iso-8859-1` (a.k.a. Latin-1)
- o <file>, --outfile <file>**  
Write output to this file instead of standard output
- s <sep>, --separator <sep>**  
Use `<sep>` as the key-value separator in the output; default value: `=`
- U, --unicode**  
New in version 0.6.0.  
  
Output non-ASCII characters literally, except for characters that are not supported by the output encoding, which are escaped with `\uXXXX` escape sequences. This overrides `--ascii`.

## JSON2PROPERTIES

### 2.1 NAME

**json2properties** — Convert a JSON object to a Java `.properties` file

### 2.2 SYNOPSIS

```
json2properties [<OPTIONS>] [<infile> [<outfile>]]
```

### 2.3 DESCRIPTION

Convert a JSON file `infile` to a `.properties` file and write the results to `outfile`. If not specified, `infile` and `outfile` default to standard input and standard output, respectively.

The JSON document must be an object with scalar (i.e., string, numeric, boolean, and/or null) values; anything else will result in an error.

Key ordering is preserved in the output by default (unless the `--sort-keys` option is given), and numeric, boolean, & null values are output using their JSON representations; e.g., the input:

```
{  
  "yes": true,  
  "no": "false",  
  "nothing": null  
}
```

becomes:

```
#Mon Sep 26 18:57:44 UTC 2016  
yes=true  
no=false  
nothing=null
```

Changed in version 0.7.0: Key ordering is now preserved by default instead of always being sorted

## 2.4 OPTIONS

**-A, --ascii**

New in version 0.6.0.

Escape all non-ASCII characters in the output with `\uXXXX` escape sequences. This overrides `--unicode`. This is the default behavior.

**-c <comment>, --comment <comment>**

New in version 0.5.0.

Show the given string as a comment at the top of the output

**-E <encoding>, --encoding <encoding>**

Use `<encoding>` as the output encoding; default value: `iso-8859-1` (a.k.a. Latin-1)

**-s <sep>, --separator <sep>**

Use `<sep>` as the key-value separator in the output; default value: `=`

**-S, --sort-keys**

New in version 0.7.0.

Sort entries in output by key

**-U, --unicode**

New in version 0.6.0.

Output non-ASCII characters literally, except for characters that are not supported by the output encoding, which are escaped with `\uXXXX` escape sequences. This overrides `--ascii`.

## PROPERTIES2JSON

### 3.1 NAME

**properties2json** — Convert a Java `.properties` file to JSON

### 3.2 SYNOPSIS

```
properties2json [<OPTIONS>] [<infile> [<outfile>]]
```

### 3.3 DESCRIPTION

Convert a `.properties` file `infile` to a JSON object and write the results to `outfile`. If not specified, `infile` and `outfile` default to standard input and standard output, respectively. The output is encoded in UTF-8.

Changed in version 0.6.0: Output encoding is now always UTF-8 instead of being determined by the locale.

Changed in version 0.7.0: Key ordering is now preserved by default instead of always being sorted

### 3.4 OPTIONS

**-A, --ascii**

New in version 0.6.0.

Escape all non-ASCII characters in the output with `\uXXXX` escape sequences. This overrides `--unicode`. This is the default behavior.

**-E <encoding>, --encoding <encoding>**

Specifies the encoding of the input file; default value: `iso-8859-1` (a.k.a. Latin-1)

**-S, --sort-keys**

New in version 0.7.0.

Sort entries in output by key

**-U, --unicode**

New in version 0.6.0.

Output non-ASCII characters literally. This overrides `--ascii`.



## CHANGELOG

### 4.1 v0.7.0 (in development)

- **properties2json** and **json2properties** now preserve the input ordering of keys by default; use the new `--sort-keys` option to get the old behavior
- Drop support for Python 3.4
- Gave **javaproperties**' **get** and **select** subcommands `--quiet` options for suppressing warnings about missing keys

### 4.2 v0.6.1 (2020-03-09)

- Require `javaproperties 0.7.*`

### 4.3 v0.6.0 (2020-03-02)

- Require `javaproperties 0.6.*`
- Use `'javapropertiesreplace'` error handler when opening output files
- Gave **javaproperties**' **format**, **select**, and **set** subcommands `--ascii` and `--unicode` options for controlling the escaping of non-ASCII characters in output
- Gave **properties2json** and **json2properties** `--ascii` and `--unicode` options for controlling the escaping of non-ASCII characters in output
- **properties2json** now always outputs UTF-8
- By default, comments output by **json2properties** now have all non-ASCII characters escaped rather than all non-Latin-1 characters

## 4.4 v0.5.0 (2020-01-24)

- Include installation instructions, GitHub links, and changelog in the Read the Docs site
- Gave `json2properties` a `--comment <comment>` option
- Support Python 3.8

## 4.5 v0.4.1 (2018-09-18)

- Drop support for Python 2.6 and 3.3
- Support `javaproperties` 0.5.0

## 4.6 v0.4.0 (2017-04-22)

- Split off the command-line programs from `javaproperties` into a separate package, `javaproperties-cli`

## 4.7 v0.3.0 (2017-04-13)

- Added the `PropertiesFile` class for preserving comments in files [#1]
- The `ordereddict` package is now required under Python 2.6

## 4.8 v0.2.1 (2017-03-20)

- **Bugfix** to `javaproperties` command: Don't die horribly on missing non-ASCII keys
- PyPy now supported

## 4.9 v0.2.0 (2016-11-14)

- Added a `javaproperties` command for basic command-line manipulating of `.properties` files
- Gave `json2properties` a `--separator` option
- Gave `json2properties` and `properties2json` `--encoding` options
- Exported the `java_timestamp()` function
- `to_comment()` now converts CR LF and CR line endings inside comments to LF
- Some minor documentation improvements



## 4.10 v0.1.0 (2016-10-02)

Initial release



## INSTALLATION

Just use `pip` (You have `pip`, right?) to install `javaproperties-cli` and its dependencies:

```
pip install javaproperties-cli
```



## QUICKSTART

```
javaproperties get <file> <key> ...
```

Output the values of the given keys in the given `.properties` file

```
javaproperties select <file> <key> ...
```

Output the key-value pairs for the given keys in the given `.properties` file

```
javaproperties set <file> <key> <value>
```

Set `<key>` in `<file>` to `<value>` and output the result

```
javaproperties delete <file> <key> ...
```

Output the given `.properties` file with the given keys deleted

```
javaproperties format [<file>]
```

Reformat the given `.properties` file, removing comments & extraneous whitespace and putting keys in sorted order

```
json2properties [<infile> [<outfile>]]
```

Convert a JSON object to a `.properties` file

```
properties2json [<infile> [<outfile>]]
```

Convert a `.properties` file to a JSON object



## INDICES AND TABLES

- genindex
- search





## PYTHON MODULE INDEX

### j

javaproperties\_cli, 1  
javaproperties\_cli.\_\_main\_\_, 1  
javaproperties\_cli.fromjson, 6  
javaproperties\_cli.tojson, 8



## Symbols

- A
  - javaproperties-format command line option,6
  - javaproperties-select command line option,4
  - javaproperties-set command line option,5
  - json2properties command line option,8
  - properties2json command line option,9
- D <file>
  - javaproperties-get command line option,3
  - javaproperties-select command line option,4
- E <encoding>
  - javaproperties-delete command line option,6
  - javaproperties-format command line option,6
  - javaproperties-get command line option,3
  - javaproperties-select command line option,4
  - javaproperties-set command line option,5
  - json2properties command line option,8
  - properties2json command line option,9
- S
  - json2properties command line option,8
  - properties2json command line option,9
- T
  - javaproperties-delete command line option,6
  - javaproperties-set command line option,5
- U
  - javaproperties-format command line option,6
  - javaproperties-select command line option,4
  - javaproperties-set command line option,5
  - json2properties command line option,8
  - properties2json command line option,9
- ascii
  - javaproperties-format command line option,6
  - javaproperties-select command line option,4
  - javaproperties-set command line option,5
  - json2properties command line option,8
  - properties2json command line option,9
- comment <comment>
  - json2properties command line option,8
- default-value <value>
  - javaproperties-get command line option,3
  - javaproperties-select command line option,4
- defaults <file>
  - javaproperties-get command line option,3
  - javaproperties-select command line option,4
- encoding <encoding>
  - javaproperties-delete command line option,6
  - javaproperties-format command line option,6
  - javaproperties-get command line option,3

```

javaproperties-select command line
    option,4
javaproperties-set command line
    option,5
json2properties command line
    option,8
properties2json command line
    option,9
--escaped
javaproperties-delete command line
    option,6
javaproperties-get command line
    option,3
javaproperties-select command line
    option,4
javaproperties-set command line
    option,5
--outfile <file>
javaproperties-delete command line
    option,6
javaproperties-format command line
    option,6
javaproperties-select command line
    option,4
javaproperties-set command line
    option,5
--preserve-timestamp
javaproperties-delete command line
    option,6
javaproperties-set command line
    option,5
--quiet
javaproperties-get command line
    option,4
javaproperties-select command line
    option,4
--separator <sep>
javaproperties-format command line
    option,6
javaproperties-select command line
    option,4
javaproperties-set command line
    option,5
json2properties command line
    option,8
--sort-keys
json2properties command line
    option,8
properties2json command line
    option,9
--unicode
javaproperties-format command line
    option,6
javaproperties-select command line
    option,4
javaproperties-set command line
    option,5
json2properties command line
    option,8
properties2json command line
    option,9
-c <comment>
json2properties command line
    option,8
-d <value>
javaproperties-get command line
    option,3
javaproperties-select command line
    option,4
-e
javaproperties-delete command line
    option,6
javaproperties-get command line
    option,3
javaproperties-select command line
    option,4
javaproperties-set command line
    option,5
-o <file>
javaproperties-delete command line
    option,6
javaproperties-format command line
    option,6
javaproperties-select command line
    option,4
javaproperties-set command line
    option,5
-q
javaproperties-get command line
    option,4
javaproperties-select command line
    option,4
-s <sep>
javaproperties-format command line
    option,6
javaproperties-select command line
    option,4
javaproperties-set command line
    option,5
json2properties command line
    option,8

```

## J

```

javaproperties (command), 1
javaproperties_cli (module), 1
javaproperties_cli.__main__ (module), 1
javaproperties_cli.fromjson (module), 6
javaproperties_cli.tojson (module), 8

```

```

javaproperties-delete command line
  option
  -E <encoding>,6
  -T,6
  --encoding <encoding>,6
  --escaped,6
  --outfile <file>,6
  --preserve-timestamp,6
  -e,6
  -o <file>,6
javaproperties-format command line
  option
  -A,6
  -E <encoding>,6
  -U,6
  --ascii,6
  --encoding <encoding>,6
  --outfile <file>,6
  --separator <sep>,6
  --unicode,6
  -o <file>,6
  -s <sep>,6
javaproperties-get command line option
  -D <file>,3
  -E <encoding>,3
  --default-value <value>,3
  --defaults <file>,3
  --encoding <encoding>,3
  --escaped,3
  --quiet,4
  -d <value>,3
  -e,3
  -q,4
javaproperties-select command line
  option
  -A,4
  -D <file>,4
  -E <encoding>,4
  -U,4
  --ascii,4
  --default-value <value>,4
  --defaults <file>,4
  --encoding <encoding>,4
  --escaped,4
  --outfile <file>,4
  --quiet,4
  --separator <sep>,4
  --unicode,4
  -d <value>,4
  -e,4
  -o <file>,4
  -q,4
  -s <sep>,4
javaproperties-set command line option
  -A,5
  -E <encoding>,5
  -T,5
  -U,5
  --ascii,5
  --encoding <encoding>,5
  --escaped,5
  --outfile <file>,5
  --preserve-timestamp,5
  --separator <sep>,5
  --unicode,5
  -e,5
  -o <file>,5
  -s <sep>,5
json2properties (command),6
json2properties command line option
  -A,8
  -E <encoding>,8
  -S,8
  -U,8
  --ascii,8
  --comment <comment>,8
  --encoding <encoding>,8
  --separator <sep>,8
  --sort-keys,8
  --unicode,8
  -c <comment>,8
  -s <sep>,8
P
properties2json (command),8
properties2json command line option
  -A,9
  -E <encoding>,9
  -S,9
  -U,9
  --ascii,9
  --encoding <encoding>,9
  --sort-keys,9
  --unicode,9

```