

---

# **javaproperties-cli**

***Release 0.4.0***

**2017 Apr 22**



---

## Commands

---

<b>1</b>	<b>javaproperties</b>	<b>3</b>
<b>2</b>	<b>json2properties</b>	<b>7</b>
<b>3</b>	<b>properties2json</b>	<b>9</b>
<b>4</b>	<b>Indices and tables</b>	<b>11</b>
	<b>Python Module Index</b>	<b>13</b>



`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

---

### NAME

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

### 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>]
```

### 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](#)*.

### 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 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.

## 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*.

## 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 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: `=`

## 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.



## Options

- 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.

## 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.

## 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.

## 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 and converted to ASCII with `javaproperties.escape`.

## Options

- 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: `=`

---

## json2properties

---

### NAME

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

### SYNOPSIS

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

### DESCRIPTION

Convert a JSON file `infile` to a `.properties` file and write the results to `outfile`. If not specified, `infile` and `outfile` default to `sys.stdin` and `sys.stdout`, 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.

Output is sorted by key, 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
no=false
nothing=null
yes=true
```

## OPTIONS

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

Use <encoding> as the output encoding; default value: `iso-8859-1` (a.k.a. Latin-1). (As all output is *currently* always pure ASCII, this option is not very useful, but there are plans to make it useful.)

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

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

---

## properties2json

---

### NAME

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

### SYNOPSIS

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

### DESCRIPTION

Convert a `.properties` file `infile` to a JSON object and write the results to `outfile`. If not specified, `infile` and `outfile` default to `sys.stdin` and `sys.stdout`, respectively.

### OPTIONS

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



## CHAPTER 4

---

### Indices and tables

---

- `genindex`
- `search`





### j

- `javaproperties_cli`, 1
- `javaproperties_cli.__main__`, 1
- `javaproperties_cli.fromjson`, 6
- `javaproperties_cli.tojson`, 8



## Symbols

- D <file>, -defaults <file>
  - javaproperties-get command line option, 3
  - javaproperties-select command line option, 4
- E <encoding>, -encoding <encoding>
  - javaproperties-delete command line option, 5
  - javaproperties-format command line option, 6
  - javaproperties-get command line option, 4
  - javaproperties-select command line option, 4
  - javaproperties-set command line option, 5
  - json2properties command line option, 8
  - properties2json command line option, 9
- T, -preserve-timestamp
  - javaproperties-delete command line option, 5
  - javaproperties-set command line option, 5
- d <value>, -default-value <value>
  - javaproperties-get command line option, 3
  - javaproperties-select command line option, 4
- e, -escaped
  - javaproperties-delete command line option, 5
  - javaproperties-get command line option, 4
  - javaproperties-select command line option, 4
  - javaproperties-set command line option, 5
- o <file>, -outfile <file>
  - javaproperties-delete command line option, 5
  - javaproperties-format command line option, 6
  - javaproperties-select command line option, 4
  - javaproperties-set command line option, 5
- s <sep>, -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

## J

- javaproperties (command), 1
- javaproperties-delete command line option
  - E <encoding>, -encoding <encoding>, 5
  - T, -preserve-timestamp, 5

- e, -escaped, 5
- o <file>, -outfile <file>, 5
- javaproperties-format command line option
  - E <encoding>, -encoding <encoding>, 6
  - o <file>, -outfile <file>, 6
  - s <sep>, -separator <sep>, 6
- javaproperties-get command line option
  - D <file>, -defaults <file>, 3
  - E <encoding>, -encoding <encoding>, 4
  - d <value>, -default-value <value>, 3
  - e, -escaped, 4
- javaproperties-select command line option
  - D <file>, -defaults <file>, 4
  - E <encoding>, -encoding <encoding>, 4
  - d <value>, -default-value <value>, 4
  - e, -escaped, 4
  - o <file>, -outfile <file>, 4
  - s <sep>, -separator <sep>, 4
- javaproperties-set command line option
  - E <encoding>, -encoding <encoding>, 5
  - T, -preserve-timestamp, 5
  - e, -escaped, 5
  - o <file>, -outfile <file>, 5
  - s <sep>, -separator <sep>, 5
- javaproperties\_cli (module), 1
- javaproperties\_cli.\_\_main\_\_ (module), 1
- javaproperties\_cli.fromjson (module), 6
- javaproperties\_cli.tojson (module), 8
- json2properties (command), 6
- json2properties command line option
  - E <encoding>, -encoding <encoding>, 8
  - s <sep>, -separator <sep>, 8

## P

- properties2json (command), 8
- properties2json command line option
  - E <encoding>, -encoding <encoding>, 9