
BibFormatter Documentation

Release latest

Dec 11, 2019

Contents

1	Documentation	3
1.1	Installation	3
1.2	Configuration	4
1.3	Commands	4
1.4	Examples	11
1.5	Common Questions	13
2	Links	15

BibFormatter is a program to clean up bib files.

- Doc is still in work

1.1 Installation

- *Requirements*
- *Installation*
 - *Windows*
 - *OS X & Linux*

1.1.1 Requirements

You need Java to run this program.

1.1.2 Installation

The newest version of BibFormatter can be found here:

Windows

Once you've downloaded the newest version follow these steps:

1. Double click `BibFormatter.jar` and wait until all files are generated
2. Edit the `run.bat` file. You can add commands in the third line after `commands=`
3. Drag and drop your bibliography file onto the `run.bat` file.

Check your log for errors. If you encounter errors, see the [FAQ](#) page.

OS X & Linux

Once you've downloaded the newest version follow these steps:

1. Double click `BibFormatter.jar` and wait until all files are generated
2. Edit the `run.sh` file. You can add commands in the third line after `commands=`
3. Drag and drop your bibliography file onto the `run.bat` file.

Check your log for errors. If you encounter errors, see the [FAQ](#) page.

1.2 Configuration

The `config.json` file gets generated (if not already existing) by clicking on the `BibFormatter.jar`

1.2.1 Configuration Files

Config file location: `./Data/config.json`

1.2.2 Settings

Setting	Default	Description
<code>entryOrder</code>	list of keys	The default list to order keys inside an entry
<code>typeOrder</code>	list of types	The default list to order entries inside a bibliography
<code>overridePublisher</code>	false	If set to true the <code>generatePublisher</code> command will override any already existing publisher value if a doi was found.
<code>defaultMonthFormat</code>	name	The default style to format the value of the month key (e.g. <code>name=jan number=01</code>)
<code>defaultPagesFormat</code>	double	The default style to format the value of the pages key (e.g. <code>double=-single=-</code>). Changes will effect the symbol between the two page numbers.
<code>encloseNumerals</code>	false	If set to true every value inside the bibliography will be enclosed with <code>{ }</code> even if its just a numeral.
<code>indentStyle</code>	tabs	Define the indent style of the keys and values.
<code>indentSpaceAmount</code>	3	The amount of spaces before each key.
<code>indentValueColumn</code>	12	The column every value will be placed at.
<code>writeEmptyEntries</code>	false	If set to true it will export keys with empty values.
<code>doiStyle</code>	raw	Set the default style to format the doi. (<code>raw/doi/proxy/url/prefix</code>)
<code>defaultCharacterMap</code>	unicode2late	Set the default character map file name.

1.3 Commands

- *General Commands*
- *Bibliography Commands*

- *Utility Commands*

Note: Arguments enclosed in [] are optional, those enclosed in < > are required.

Type represents things like article/book/inbook etc. and Key represents things like author/title/pages/doi etc.

Every command parameter argument can be written down in a .txt file and loaded by entering the exact path (useful if a path contains a space or for regex searches in certain commands). (e.g. `-createKey +t article C:/Types.txt +k title +v C:/Text.txt`)

Tip: You can access a command listing via the `-help` command.

`+type / +key / +match / +value` can be shortened to `+t / +k / +m / +v`

You can invert types and keys by surrounding them with ! (e.g. `+t !article!` will select every type except articles) # selects everything (e.g. `+t #` will select every type)

1.3.1 General Commands

-bibliography Everytime this command gets issued the currently loaded bibliography file will be saved (with all applied changes) and the new one will be loaded. Every command after this one will only effect the bibliography file loaded by this command (until `-b` is found again etc.).

Description	Load a new .bib file. If no output path is specified the modified bib gets saved in the same folder where the original .bib file is located.
Shortcut	<code>-b</code>
Usage	<code>-bibliography <file> [output]</code>
<code><file></code>	Exact path to the bibliography file you want to load
<code>[output]</code>	The exact save path. If not specified this will be the current location of the loaded .bib file appended with <code>_formatted.bib</code>

-debug

Description	Debugs the modification process. If not specified debugging is set to no.
Shortcut	<code>-d</code>
Usage	<code>-debug {yes no}</code>

1.3.2 Bibliography Commands

-addEntry

Description	Adds every entry from other .bib files to the main bibliography file.
Shortcut	-ae
Usage	-addEntry [+type <types>] <+value <files>>
[+type <types>]	Only entries of this type get added. If not specified every type will be added
<+value <files>>	A list of all bibliography files. No spaces allowed

-checkType

Description	Checks if an entry is missing keys and exports the missing keys for every entry as a list.
Shortcut	-ct
Usage	-checkType [+type <types>] [+match <required>] [+value <style>]
[+type <types>]	Only entries of this type will be checked. If not specified every type will be checked
[+match <required>]	Specify a .json file which represents a map of required keys for every type. If not specified the default map will be loaded (. \Data\CheckFiles\valRequiredFields.json)
[+value <style>]	Specify the output style of the missing keys. (txt/json/html)
Styles	txt Exports the list of missing keys as plain txt
	json Exports the list of missing keys as json array
	html Exports the list of missing keys as a html document (e.g. BibChecker)

-createKey

Description	Creates a new key with a specified value.
Shortcut	-ck
Usage	-createKey [+type <types>] [+key <keys>] [+match <value>] [+value {yes no}]
[+type <types>]	Only entries of this type will be affected. If not specified every type will be affected
[+key <keys>]	Specify the keys you want to create. If multiple keys are listed they will all get the same value
[+match <value>]	Specify the value to put in the new key
[+value {yes no}]	Set to yes/y if you want to override any already existing key. If not specified overriding is set to no/n.

-formatDOI

Description	Searches for DOIs and adds the correct formatted value in the doi key.
Shortcut	-fd
Usage	-createKey [+type <types>] [+key <keys>] [+value <style>]
[+type <types>]	Only entries of this type will be affected. If not specified every type will be affected
[+key <keys>]	Specify the keys in which you want to search for DOIs. If no keys are specified only the doi key will be analysed.“
[+value <style>]	Specify the style to format the DOI after (raw/doi/proxy/url/prefix)
Styles	raw 10.1000/foobar
	doi doi:10.1000/foobar
	proxy https://doi.org/10.1000/foobar
	url \url{https://doi.org/10.1000/foobar}
	prefix 1000

-formatMonth

Description	Formats the month value into the described format.
Shortcut	-fm
Usage	-formatMonth [+type <types>] [+value <style>]
[+type <types>]	Only entries of this type will be affected. If not specified every type will be affected
[+value <style>]	Specify the style to format the month after (name/number). If no parameter is specified the standard format is name.
Styles	name jul
	number 07

-formatPages

Description	Formats the pages value into the described format.
Shortcut	-fp
Usage	-formatPages [+type <types>] [+value <style>]
[+type <types>]	Only entries of this type will be affected. If not specified every type will be affected
[+value <style>]	Specify the style to format the pages after (single/double). If no parameter is specified the standard format is double.
Styles	single 157-160
	double 157-160

-formatURL

Description	Formats a url found in a value into the correct format.
Shortcut	-fu
Usage	-createKey [+type <types>] [+key <keys>]
[+type <types>]	Only entries of this type will be affected. If not specified every type will be affected
[+key <keys>]	Only the keys specified here will be affected. If not specified every key will be affected

-fromAux

Description	Removes every entry in the currently loaded bibliography that is not cited in the .aux file.
Shortcut	-fa
Usage	-fromAux <+value <path>>
<+value <path>>	The exact path to the .aux file you want to load (no spaces allowed)

-generatePublisher

Description	If an entry contains a DOI it will search in the DOI prefix list after a matching publisher name and if found add this as value to a new publisher key.
Shortcut	-gp
Usage	-generatePublisher [+type <types>] [+key <keys>] [+match <dois>] [+value {y n}]
[+type <types>]	Only entries of this type will be affected. If not specified every type will be affected
[+key <keys>]	Specify the keys in which you want to search for DOIs. If no keys are specified only the doi key will be analysed.
[+match <dois>]	Specify the DOIs you want to accept. If no DOIs are specified every DOI will be accepted.
[+value {yes no}]	Set to yes/y if you want to override any already existing publisher value. If not specified overriding is set to no/n.

-mergeBibliographies

Description	Add every entry or key from the specified .bib file that is not already in the loaded bibliography.
Shortcut	-mb
Usage	-mergeBibliographies [+type <types>] [+key <keys>] [+value <path>]
[+type <types>]	Only entries of this type will be added. If not specified every type will be added
[+key <keys>]	Only the keys specified here will be added. If not specified every key will be added
[+value <path>]	The exact path to the .bib file you want to load (no spaces allowed)

-orderKeys

Description	Orders every key in a bib file by a given list.
Shortcut	-ok
Usage	-orderKeys [+type <types>] [+key <keys>]
[+type <types>]	Only entries of this type will be affected. If not specified every type will be affected
[+key <keys>]	The order in which the keys will be ordered. If no keys are specified the standard order listed from the config file is used (<code>entryOrder</code>).

-orderTypes

Description	Orders every key in a bib file by a given list.
Shortcut	-ot
Usage	-orderTypes [+type <types>]
[+type <types>]	The order in which the entries will be ordered. If no types are specified the standard order listed from the config file is used (<code>typeOrder</code>).

-removeEntry

Description	Removes specified entries/keys from a bibliography.
Shortcut	-re
Usage	-removeEntry [+type <types>] [+key <keys>] [+match <match>] [+value {y n}]
[+type <types>]	Only entries of this type will be affected. If not specified every type will be affected
[+key <keys>]	Only the keys specified here will be affected. If not specified every key will be affected
[+match <match>]	Only the values that match this pattern will be affected (for regex please use a file here). If no match is specified every value will be affected.
[+value {yes no}]	Set to yes/y if you want to invert the whole statement. If not specified invert is set to no/n.

-replaceKey

Description	Replace a key with a new key.
Shortcut	-rk
Usage	-replaceKey [+type <types>] [+key <keys>] [+match <match>] [+value {y n}]
[+type <types>]	Only entries of this type will be affected. If not specified every type will be affected
[+key <keys>]	Only the keys specified here will be affected. If not specified every key will be affected
[+match <match>]	The value that has to match with the value of the key you want to replace. If no match is specified every value will be affected.
[+value {yes no}]	Set to yes/y if you want to override any already existing key. If not specified overriding is set to no/n.

-saveCapitals

Description	Enclose every uppercase letter. (``This is A test'' -> ``{T}his is {A} test``).
Shortcut	-sc
Usage	-saveCapitals [+type <types>] [+key <keys>] [+match <characters>]
[+type <types>]	Only entries of this type will be affected. If not specified every type will be affected
[+key <keys>]	Only the keys specified here will be affected. If not specified every key will be affected
[+match <characters>]	Only the characters specified here will be enclosed. If not specified every character will be enclosed

-saveSymbols

Description	Replaces every special character with the correct latex code.
Shortcut	-ss
Usage	-saveSymbols [+type <types>] [+key <keys>] [+value <characterMap>]
[+type <types>]	Only entries of this type will be affected. If not specified every type will be affected
[+key <keys>]	Only the keys specified here will be affected. If not specified every key will be affected
[+value <characterMap>]	The character map file name (.Data/<characterMap>.txt). If no map is specified the standard one is loaded (unicode2latex)

-setValue

Description	Set the specified value of a specified key in a specified typ to new value.
Shortcut	-sv
Usage	-setValue [+type <types>] [+key <keys>] [+match <match>] <+value <value>>
[+type <types>]	Only entries of this type will be affected. If not specified every type will be affected
[+key <keys>]	Only the keys specified here will be affected. If not specified every key will be affected
[+match <match>]	Only values matching this value will be affected (to escape special characters when using regex matching, use a file (e.g. +match C:/regexcode.txt)). If no matching string is specified the changes will effect every value.
<+value <value>>	The character map file name (.Data/<characterMap>.txt). If no map is specified the standard one is loaded (unicode2latex)

1.3.3 Utility Commands

-help Not yet implemented!	
Description	Displays help for BibFormatter commands
Shortcut	-h
Usage	-help [+match <command>] [+value <value>]
[+match <command>]	The command to retrieve help for. If no command is specified a list of all commands will be displayed
[+value <value>]	The page to retrieve

1.4 Examples

- *Save every capital E in every article title*
- *Remove every key except author title year pages publisher month place*

1.4.1 Save every capital E in every article title

```
-b example.bib -sc +t article +k title +m E
```

Loads the bibliography `example.bib`

The `-sc` will load the save capitals command

The `+t article` parameter will only select entries of typ article

The `+k title` parameter will only select title keys

The `+m E` parameter will only select capital E letters

Example: Enclosing capital E in article titles

Before:

```
@Article{parsons:1990,
  author = {Parsons, Terence},
  title  = {Events in the Semantics of English},
  year   = {1990},
  month  = {01},
  place  = {Cambridge, MA},
}
```

After:

```
@Article{parsons:1990,
  author = {Parsons, Terence},
  title  = {{E}vents in the Semantics of {E}nglish},
  year   = {1990},
  month  = {01},
  place  = {Cambridge, MA},
}
```

1.4.2 Remove every key except author title year pages publisher month place

```
-b example.bib -re +k author title year pages publisher month place +v y
```

Loads the bibliography `example.bib`

The `-re` parameter will load the remove entry command

The `+k author title year pages publisher month place` parameter will only select keys of type `title year pages publisher month place`

The `+v y` parameter will invert the key selection, meaning that only `title year pages publisher month place` will not be removed

Example: Remove every key except author title year pages publisher month place

Before:

```
@Article{vanderDoes1991,
  author = {van der Does, Jaap},
  title  = {A generalized quantifier logic for naked infinitives},
  journal = {Linguistics and Philosophy},
  year   = {1991},
  volume = {14},
  number = {3},
  pages  = {241--294},
  month  = {Jun},
  issn   = {1573-0549},
  day    = {01},
  doi    = {10.1007/BF00627404},
  url    = {https://doi.org/10.1007/BF00627404},
}
```

```
@Book{kroch1974semantics,
```

```
  title = {The Semantics of English},
  publisher = {Massachusetts Institute of Technology, Department of Foreign
  ↳ Literatures and Linguistics},
```

12

```
  year   = {1974},
  author = {Kroch, Anthony S.},
  series = {MIT working papers in linguistics},
  url    = {https://books.google.de/books?id=FcsvvwEACAAJ},
}
```


After:

```
@Article{vanderDoes1991,
  author = {van der Does, Jaap},
  title  = {A generalized quantifier logic for naked infinitives},
  year   = {1991},
  pages  = {241--294},
  month  = {Jun},
}

@Book{kroch1974semantics,
  title      = {The Semantics of Scope in English},
  publisher  = {Massachusetts Institute of Technology, Department of Foreign_
↳Literatures and Linguistics},
  year       = {1974},
  author     = {Kroch, Anthony S.},
}
```

1.5 Common Questions

- *General*
 - *Who needs this program?*
 - *Will there be a UI*
 - *Is this program free?*
 - *Does this program corrupt my bibliography integrity?*

1.5.1 General

Who needs this program?

There are several bibliography programs out there (e.g. JabRef/Citavi) that can manage .bib files. But this program was designed to apply specific changes to a large amount of entries in huge bibliography files (>20k lines). This tool is meant for fast and complex changes at large scale.

Will there be a UI

It is planned but there is no release date yet.

Is this program free?

Yes

Does this program corrupt my bibliography integrity?

This program has been tested on numerous huge bibliography files (100k+ lines) without problems. But this program is still under heavy development so I highly recommend creating a backup of your bibliography before using this tool.

CHAPTER 2

Links

- [GitHub](#)