NAME

gma go coredb – import/export entries in the GMA core SRD database

SYNOPSIS

(If using the full GMA core tool suite)

`gma go coredb` [options as described below...]

(Otherwise)

`coredb -h`

`coredb -help`

`coredb` [`-D` *flags*] [`-e` *file*] [`-f` [`!`]*regexp]* [`-I`] [`-i` *file*] [`-l` *file*] [`-preferences` *file*] [`-srd`] [`-t` *list*]

`coredb` [`-debug` *flags*] [`-export` *file*] [`-filter` [`!`]*regexp]* [`-ignore-case`] [`-import` *file*] [`-log` *file*] [`-preferences` *file*] [`-srd`] [`-type` *list*]

DESCRIPTION

`Coredb` is used for maintenance of the core (SRD) database. Once the initial load of community-published SRD data is performed via `gma initdb`, you will probably add a number of your own entries to the database for your own campaign. These are known as "local" entries, while the ones that are in the core game's SRD content which are licensed to be freely used (according to the applicable license) are known as "SRD" entries.

If run with the `-export` option, `coredb` will export the local entries from your core database to the specified output file, encoded in JSON. If run with the `-import` option, the opposite is done: the JSON-encoded file is read, and its contents added to the core database. If this file's contents match the name or ID (as appropriate) of an existing entry, that entry is updated in-place in the database.

OPTIONS

The command-line options described below have a long form (e.g., `-log`) and a short form (e.g., `-l`) which are equivalent. In either case, the option may be introduced with either one or two hyphens (e.g., `-log` or `--log`). Options which take parameter values may have the value separated from the option name by a space or an equals sign (e.g., `-log=`*path* or `-log` *path*), except for boolean flags which may be given alone (e.g., `-I`) to indicate that the option is set to "true" or may be given an explicit value which must be attached to the option with an equals sign (e.g., `-I=true` or `-I=false`).

You *may not* combine multiple single-letter options into a single composite argument, (e.g., the options `-I` and `-h` would need to be entered as two separate options, not as `-Ih`).

`-D`, `-debug` *flags*

This adds debugging messages to `coredb`'s output. The *flags* value is a comma-separated list of debug flag names, which may be any of the following:

`all` Enable all possible debugging flags.

`none` If you want to explicitly disable debugging, you can specify `none` for the *flags* value to override the previous flag list and effectively disable debugging. If `none` appears in a comma-separated flag list, it cancels all the previously-set flags, but any other flag names which occur after it will be set.

`misc` Show miscellaneous debugging messages.

`queries` Show SQL queries made to the database.

`-e`, `-export` *file*

Export the local entries from the database to the named JSON-encoded *file*.

`-f`, `-filter` [`!`]*regexp*

When importing or exporting, only include entries matching the regular expression *regexp*. If *regexp* begins with a "`!`" character, only entries which do *not* match the expression are included.

The *regexp* is matched against the `Code` and `Name` fields of the entries in question. If either matches, then the entry is included (or excluded). For language entries, the `Language` field is checked instead. For monsters in the bestiary, the `Code` and `Species` fields are checked.

`−I, −ignore−case`
Pattern matching via the `−filter` option should be done irrespective of case.

`−i, −import` *file*
Read the contents of the JSON-encoded *file*, importing them to the database.

`−l, −log` *file*
Directs log messages (including any debugging output) to the named *file* instead of the standard output. You may explicitly specify the standard output by using a single hyphen as the file name (e.g., “`−log=−`”).

`−preferences` *file*
Normally, `coredb` reads the default GMA preferences file to obtain the pathname of the core database file. This option specifies an alternative preferences file from which to get that pathname.

`−srd` Instead of exporting only the local entries and assuming imported entries are local, export SRD entries and assume imported entries are SRD.

`−t, −type` *list*
Entries exported will include only database entries of the specified type(s). When importing, any records in the import file which are not of the specified type(s) are skipped over. The *list* value is a comma-separated list of type names, which may be any of the following:

`all`	All entry types (the default).
`none`	No types. If `none` appears in a comma-separated type list, it cancels all the previously-set types, but any other type names which occur after it will be set.
`bestiary`	Monster entries. (The names `creature[s]` and `monster[s]` are also recognized.)
`class[es]`	Character class entries.
`feat[s]`	Feats.
`language[s]`	Languages.
`skill[s]`	Skills.
`spell[s]`	Spells.
`weapon[s]`	Weapons.

FILE FORMAT

The JSON file used for exporting and importing operations is as documented for the `util.CoreImport` function of the Go-GMA project (see *https://pkg.go.dev/github.com/MadScienceZone/go-gma/v5/util#CoreImport*). Each of the data structures marshalled into JSON for monsters, weapons, etc., are documented on the same page at that URL.

SEE ALSO

`gma`(6), `gma−importdb`(6), `gma−initdb`(6).

AUTHOR

Steve Willoughby / steve@madscience.zone.

HISTORY

The core database appeared in GMA Core version 7.0; this utility appeared in go-gma version 5.7.0.

BUGS

COPYRIGHT