



GlpiCommandInterface

+ mustCheckMandatoryRequirements()

+ requiresUpToDateDb()

+ERROR\_PLUGIN\_IMPORT\_FAILED

+ERROR\_PLUGIN\_VERSION\_OR\_DATA\_INVALID

AbstractCommand

# \$db

# \$input

# \$output

# \$requires\_db

# \$requires\_db\_up\_to\_date

# \$progress\_bar

+ mustCheckMandatoryRequirements()

+ requiresUpToDateDb()

# initialize(InputInterface \$input, OutputInterface \$output)

# writeInOutputWithProgress Bar( \$messages, ProgressBar \$progress\_bar, \$verbosity =OutputInterface::VERBOSITY\_NORMAL)

# outputSessionBufferedMessages (\$levels\_to\_output=[INFO, WARNING, ERROR])

# outputWarningOnMissingOptionnal Requirements()

# askForConfirmation (bool \$default\_to\_yes=true)

# warnAboutExecutionTime()

# iterate(iterable \$iterable, ?callable \$message\_callback=null, ?int \$count=null)

# outputMessage(string \$message, int \$verbosity =OutputInterface::VERBOSITY\_NORMAL)

AbstractPluginToCoreCommand

# configure()

# initialize(InputInterface \$input, OutputInterface \$output)

# execute(InputInterface \$input, OutputInterface \$output)

# checkPlugin()

# handleImportError (\$message, ProgressBar \$progress\_bar=null, bool \$prevent\_exit=false)

# getMatchingElementId (string \$itemtype, array \$criteria)

# defineTargetItem(string \$source\_itemtype, int \$source\_id, string \$target\_itemtype, int \$target\_id)

# getTargetItem(string \$source\_itemtype, int \$source\_id)

# storeItem(string \$itemtype, ?int \$existing\_item\_id, array \$input, ProgressBar \$progress\_bar=null)

# storeInfocomForItem (CommonDBTM \$item, array \$infocom\_input, ProgressBar \$progress\_bar=null)

# getPluginKey()

# getRequiredMinimalPlugin Version()

# getRequiredDatabasePlugin Fields()

# migratePlugin()