HelperInterface

- + setHelperSet(HelperSet \$helperSet=null)
- + getHelperSet()
- + getName()

Helper

\$helperSet

- + setHelperSet(HelperSet \$helperSet=null)
- + getHelperSet()
- + static strlen(?string \$string)
- + static width(?string \$string)
- + static length(?string \$string)
- + static substr(?string \$string, int \$from, int \$length=null)
- + static formatTime(
 \$secs)
- + static formatMemory (int \$memory)
- + static strlenWithoutDecoration (OutputFormatterInterface \$formatter, ?string \$string)
- + static removeDecoration (OutputFormatterInterface \$formatter, ?string \$string)

DebugFormatterHelper

- + start(string \$id, string \$message, string \$prefix='RUN')
- + progress(string \$id, string \$buffer, bool \$error=false, string \$prefix='OUT', string \$errorPrefix='ERR')
- + stop(string \$id, string \$message, bool \$successful, string \$prefix='RES')
- + getName()

DescriptorHelper

- + __construct()
- + describe(OutputInterface \$output, ?object \$object, array \$options=[])
- + register(string \$format, DescriptorInterface \$descriptor)
- + getName()
- + getFormats()

FormatterHelper

- + formatSection(string \$section, string \$message, string \$style='info')
- + formatBlock(\$messages, string \$style, bool \$large=false)
- + truncate(string \$message, int \$length, string \$suffix='...')
- + getName()

InputAwareHelper

- # \$input
- + setInput(InputInterface \$input)

ProcessHelper

- + run(OutputInterface
 \$output, \$cmd, string
 \$error=null, callable
 \$callback=null, int \$verbosity
 =OutputInterface::VERBOSITY_VERY_VERBOSE)
- + mustRun(OutputInterface \$output, \$cmd, string \$error=null, callable \$callback=null)
- + wrapCallback(OutputInterface \$output, Process \$process, callable \$callback=null)
- + getName()

QuestionHelper

- + ask(InputInterface \$input, OutputInterface \$output, Question \$question)
- + getName()
- + static disableStty()
- # writePrompt(OutputInterface \$output, Question \$question)
- # formatChoiceQuestionChoices (ChoiceQuestion \$question, string \$tag)
- # writeError(OutputInterface \$output, \Exception \text{ \text{serror}})

SymfonyQuestionHelper

- # writePrompt(OutputInterface \$output, Question \$question)
- # writeError(OutputInterface \$output, \Exception \$error)