## FormatterInterface + format(array \$record) + formatBatch(array \$records) NormalizerFormatter + const SIMPLE\_DATE # \$dateFormat # \$maxNormalizeDepth # \$maxNormalizeItemCount construct(?string \$dateFormat=null) + format(array \$record) + formatBatch(array \$records) + getDateFormat() + setDateFormat(string \$dateFormat) + getMaxNormalizeDepth() + setMaxNormalizeDepth (int \$maxNormalizeDepth) + getMaxNormalizeItemCount() + setMaxNormalizeItemCount (int \$maxNormalizeItemCount) + setJsonPrettyPrint (bool \$enable) + addJsonEncodeOption (int \$option) + removeJsonEncodeOption (int \$option) # normalizeException (Throwable \$e, int \$depth=0) # toJson(\$data, bool \$ignoreErrors=false) # formatDate(\DateTimeInterface \$date) **JsonFormatter** + const BATCH\_MODE\_JSON + const BATCH MODE NEWLINES # \$batchMode # \$appendNewline # \$ignoreEmptyContextAndExtra # \$includeStacktraces + \_\_construct(int \$batchMode =self::BATCH\_MODE\_JSON, bool \$appendNewline=true, bool \$ignoreEmptyContextAndExtra =false, bool \$includeStacktraces=false) + getBatchMode() + isAppendingNewlines() + format(array \$record) + formatBatch(array \$records) + includeStacktraces (bool \$include=true) # formatBatchJson(array \$records) # formatBatchNewlines (array \$records) # normalize(\$data, int \$depth=0) # normalizeException (Throwable \$e, int \$depth=0) LogglyFormatter \_\_construct(int \$batchMode =self::BATCH\_MODE\_NEWLINES, bool \$appendNewline=false) + format(array \$record)