cz.polankam.jjcron.remote

<<Cli>ent>>

- + shutdown(): void
- + isPaused(): boolean
- + pause(): void
- + unpause(): void
- + getTasks(): List<TaskDetail>
- + deleteTask(TaskDetail task): void
- + addTask(TaskMetadata task): void
- + reloadCrontab(): void
- + saveToCrontab(): void

TaskDetail

- + id: String
- + name: String
- + nextExecutionTime: LocalDateTime
- + metadata: TaskMetadata
- + stats: TaskStats

cz.polankam.jjcron.common

Exceptions

FormatException

CrontabTime

- second: CrontabTimeUnit
- minute: CrontabTimeUnit
- hour: CrontabTimeUnit
- dayOfMonth: CrontabTimeUnit
- month: CrontabTimeUnit
- dayOfWeek: CrontabTimeUnit
- + delay(LocaDateTime localNow): long
- + timeUnit(): TimeUnit

TaskMetadata

- time: CrontabTime
- command: String
- + time(): CrontabTime
- + command(): String

<<CrontabTimeUnit>>

- + isChanged(): boolean
- + delay(LocalDateTime current, int currentValue, boolean previousChanged): int
- + value(): CrontabTimeValue

CrontabTimeGeneralUnit

- # minValue: int
- # maxValue: int
- # period: int
- # unitStr: String
- # unit: CrontabTimeValue
- # parser: CrontabTimeValueParser
- # valueChanged: boolean
- isValueValid(int value): void
- checkValue(): void

CrontabTimeValue

- + valueType: CrontabTimeValueType
- + values: List<Integer>

CrontabTimeDayOfMonthUnit

CrontabTimeValueType: enum

TaskStats

+ record(LocalDateTime time, long duration): void

- lastExecutions: Deque<LocalDateTime>

- lastExecutionsDuration: Deque<Long>

+ getLastExecution(): LocalDateTime

+ getLastDuration(): Long

- + LIST
- + ASTERISK
- + PERIOD

<<CrontabTimeValueParser>>

+ parse(String value): CrontabTimeValue

CrontabTimeValueNamedParser

- namedValues: Map<String, Integer>