package com.twitter.product\_mixer.core.functional\_component.common.alert.predicate

/\*\*

\* [[Predicate]]s will trigger if the metric's value is past the

\* `threshold` for `datapointsPastThreshold` or more datapoints

\* in a given `duration`

\*

\* @see [[https://docbird.twitter.biz/mon/reference.html#predicate Predicate]]

\*/

trait Predicate {

/\*\* @see [[https://docbird.twitter.biz/mon/reference.html#predicate OPERATOR]] \*/

val operator: Operator

/\*\* @see [[https://docbird.twitter.biz/mon/reference.html#predicate THRESHOLD]] \*/

val threshold: Double

/\*\*

\* The number of datapoints in a given duration beyond the threshold that will trigger an alert

\* @see [[https://docbird.twitter.biz/mon/reference.html#predicate DATAPOINTS]]

\*/

val datapointsPastThreshold: Int

/\*\*

\* @note if using a [[metricGranularity]] of [[Minutes]] then this must be >= 3

\* @see [[https://docbird.twitter.biz/mon/reference.html#predicate DURATION]]

\*/

val duration: Int

/\*\*

\* Specifies the metric granularity

\* @see [[https://docbird.twitter.biz/mon/reference.html#predicate DURATION]]

\*/

val metricGranularity: MetricGranularity

require(

datapointsPastThreshold > 0,

s"`datapointsPastThreshold` must be > 0 but got `datapointsPastThreshold` = $datapointsPastThreshold"

)

require(

datapointsPastThreshold <= duration,

s"`datapointsPastThreshold` must be <= than `duration.inMinutes` but got `datapointsPastThreshold` = $datapointsPastThreshold `duration` = $duration"

)

require(

metricGranularity != Minutes || duration >= 3,

s"Predicate durations must be at least 3 minutes but got $duration"

)

}

/\*\* [[ThroughputPredicate]]s are predicates that can trigger when the throughput is too low or high \*/

trait ThroughputPredicate extends Predicate