package com.twitter.product\_mixer.core.model.marshalling.response.urt.operation

import com.twitter.product\_mixer.core.model.marshalling.response.urt.metadata.ClientEventInfo

import com.twitter.product\_mixer.core.model.marshalling.response.urt.metadata.FeedbackActionInfo

import com.twitter.product\_mixer.core.model.marshalling.response.urt.operation.CursorOperation.CursorEntryNamespace

import com.twitter.product\_mixer.core.model.marshalling.response.urt.EntryNamespace

import com.twitter.product\_mixer.core.model.marshalling.response.urt.TimelineEntry

import com.twitter.product\_mixer.core.model.marshalling.response.urt.TimelineItem

/\*\*

\* CursorItem should only be used for Module cursors

\* For timeline cursors, see

\* [[com.twitter.product\_mixer.core.model.marshalling.response.urt.operation.CursorOperation]]

\*/

case class CursorItem(

override val id: Long,

override val sortIndex: Option[Long],

override val clientEventInfo: Option[ClientEventInfo],

override val feedbackActionInfo: Option[FeedbackActionInfo],

value: String,

cursorType: CursorType,

displayTreatment: Option[CursorDisplayTreatment])

extends TimelineItem {

override val entryNamespace: EntryNamespace = CursorEntryNamespace

override lazy val entryIdentifier: String =

s"$entryNamespace-${cursorType.entryNamespace}-$id"

override def withSortIndex(sortIndex: Long): TimelineEntry = copy(sortIndex = Some(sortIndex))

}