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

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

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

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

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

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

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

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

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

import com.twitter.product\_mixer.core.model.marshalling.response.urt.timeline\_module.ModuleDisplayType

import com.twitter.product\_mixer.core.model.marshalling.response.urt.timeline\_module.ModuleFooter

import com.twitter.product\_mixer.core.model.marshalling.response.urt.timeline\_module.ModuleHeader

import com.twitter.product\_mixer.core.model.marshalling.response.urt.timeline\_module.ModuleMetadata

import com.twitter.product\_mixer.core.model.marshalling.response.urt.timeline\_module.ModuleShowMoreBehavior

sealed trait TimelineEntry

extends HasEntryIdentifier

with HasSortIndex

with HasExpirationTime

with PinnableEntry

with ReplaceableEntry

with MarkUnreadableEntry

trait TimelineItem extends TimelineEntry with HasClientEventInfo with HasFeedbackActionInfo

case class ModuleItem(

item: TimelineItem,

dispensable: Option[Boolean],

treeDisplay: Option[ModuleItemTreeDisplay])

case class TimelineModule(

override val id: Long,

override val sortIndex: Option[Long],

override val entryNamespace: EntryNamespace,

override val clientEventInfo: Option[ClientEventInfo],

override val feedbackActionInfo: Option[FeedbackActionInfo],

override val isPinned: Option[Boolean],

items: Seq[ModuleItem],

displayType: ModuleDisplayType,

header: Option[ModuleHeader],

footer: Option[ModuleFooter],

metadata: Option[ModuleMetadata],

showMoreBehavior: Option[ModuleShowMoreBehavior])

extends TimelineEntry

with HasClientEventInfo

with HasFeedbackActionInfo

with ContainsFeedbackActionInfos {

override def feedbackActionInfos: Seq[Option[FeedbackActionInfo]] = {

items.map(\_.item.feedbackActionInfo) :+ feedbackActionInfo

}

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

}

trait TimelineOperation extends TimelineEntry