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

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.metadata.SocialContext

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

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

object UserItem {

val UserEntryNamespace: EntryNamespace = EntryNamespace("user")

}

case class UserItem(

override val id: Long,

override val sortIndex: Option[Long],

override val clientEventInfo: Option[ClientEventInfo],

override val feedbackActionInfo: Option[FeedbackActionInfo],

override val isMarkUnread: Option[Boolean],

displayType: UserDisplayType,

promotedMetadata: Option[PromotedMetadata],

socialContext: Option[SocialContext],

reactiveTriggers: Option[UserReactiveTriggers],

enableReactiveBlending: Option[Boolean])

extends TimelineItem {

override val entryNamespace: EntryNamespace = UserItem.UserEntryNamespace

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

}