package com.twitter.home\_mixer.product.following.model

import com.twitter.adserver.thriftscala.HomeTimelineType

import com.twitter.adserver.thriftscala.TimelineRequestParams

import com.twitter.home\_mixer.model.HomeAdsQuery

import com.twitter.dspbidder.commons.{thriftscala => dsp}

import com.twitter.home\_mixer.model.request.DeviceContext

import com.twitter.home\_mixer.model.request.HasDeviceContext

import com.twitter.home\_mixer.model.request.HasSeenTweetIds

import com.twitter.home\_mixer.model.request.FollowingProduct

import com.twitter.onboarding.task.service.{thriftscala => ots}

import com.twitter.product\_mixer.component\_library.model.cursor.UrtOrderedCursor

import com.twitter.product\_mixer.component\_library.pipeline.candidate.flexible\_injection\_pipeline.transformer.HasFlipInjectionParams

import com.twitter.product\_mixer.core.feature.featuremap.FeatureMap

import com.twitter.product\_mixer.core.model.marshalling.request.\_

import com.twitter.product\_mixer.core.pipeline.HasPipelineCursor

import com.twitter.product\_mixer.core.pipeline.PipelineQuery

import com.twitter.timelines.configapi.Params

case class FollowingQuery(

override val params: Params,

override val clientContext: ClientContext,

override val pipelineCursor: Option[UrtOrderedCursor],

override val requestedMaxResults: Option[Int],

override val debugOptions: Option[DebugOptions],

override val features: Option[FeatureMap],

override val deviceContext: Option[DeviceContext],

override val seenTweetIds: Option[Seq[Long]],

override val dspClientContext: Option[dsp.DspClientContext])

extends PipelineQuery

with HasPipelineCursor[UrtOrderedCursor]

with HasDeviceContext

with HasSeenTweetIds

with HasFlipInjectionParams

with HomeAdsQuery {

override val product: Product = FollowingProduct

override def withFeatureMap(features: FeatureMap): FollowingQuery =

copy(features = Some(features))

override val timelineRequestParams: Option[TimelineRequestParams] =

Some(TimelineRequestParams(homeTimelineType = Some(HomeTimelineType.HomeLatest)))

// Fields below are used for FLIP Injection in Onboarding Task Service (OTS)

override val displayLocation: ots.DisplayLocation = ots.DisplayLocation.HomeLatestTimeline

override val rankingDisablerWithLatestControlsAvailable: Option[Boolean] = None

override val isEmptyState: Option[Boolean] = None

override val isFirstRequestAfterSignup: Option[Boolean] = None

override val isEndOfTimeline: Option[Boolean] = None

override val timelineId: Option[Long] = None

}