package com.twitter.product\_mixer.component\_library.pipeline.candidate.ads

import com.twitter.adserver.{thriftscala => ads}

import com.twitter.product\_mixer.component\_library.model.candidate.ads.AdsCandidate

import com.twitter.product\_mixer.component\_library.model.query.ads.AdsQuery

import com.twitter.product\_mixer.core.functional\_component.candidate\_source.CandidateSource

import com.twitter.product\_mixer.core.functional\_component.common.alert.Alert

import com.twitter.product\_mixer.core.functional\_component.decorator.CandidateDecorator

import com.twitter.product\_mixer.core.functional\_component.feature\_hydrator.BaseCandidateFeatureHydrator

import com.twitter.product\_mixer.core.functional\_component.filter.Filter

import com.twitter.product\_mixer.core.functional\_component.gate.BaseGate

import com.twitter.product\_mixer.core.functional\_component.transformer.CandidatePipelineResultsTransformer

import com.twitter.product\_mixer.core.functional\_component.transformer.DependentCandidatePipelineQueryTransformer

import com.twitter.product\_mixer.core.model.common.identifier.CandidatePipelineIdentifier

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

import com.twitter.product\_mixer.core.pipeline.candidate.DependentCandidatePipelineConfig

import com.twitter.timelines.configapi.FSParam

import com.twitter.timelines.configapi.decider.DeciderParam

class AdsDependentCandidatePipelineConfig[Query <: PipelineQuery with AdsQuery](

override val identifier: CandidatePipelineIdentifier,

override val enabledDeciderParam: Option[DeciderParam[Boolean]],

override val supportedClientParam: Option[FSParam[Boolean]],

override val gates: Seq[BaseGate[Query]],

override val candidateSource: CandidateSource[

ads.AdRequestParams,

ads.AdImpression

],

override val filters: Seq[Filter[Query, AdsCandidate]],

override val postFilterFeatureHydration: Seq[

BaseCandidateFeatureHydrator[Query, AdsCandidate, \_]

],

override val decorator: Option[CandidateDecorator[Query, AdsCandidate]],

override val alerts: Seq[Alert],

adsDisplayLocationBuilder: AdsDisplayLocationBuilder[Query],

urtRequest: Option[Boolean],

getOrganicItemIds: GetOrganicItemIds,

countNumOrganicItems: CountNumOrganicItems[Query],

) extends DependentCandidatePipelineConfig[

Query,

ads.AdRequestParams,

ads.AdImpression,

AdsCandidate

] {

override def queryTransformer: DependentCandidatePipelineQueryTransformer[

Query,

ads.AdRequestParams

] = AdsDependentCandidatePipelineQueryTransformer(

adsDisplayLocationBuilder = adsDisplayLocationBuilder,

getOrganicItemIds = getOrganicItemIds,

countNumOrganicItems = countNumOrganicItems,

urtRequest = urtRequest

)

override val resultTransformer: CandidatePipelineResultsTransformer[

ads.AdImpression,

AdsCandidate

] = AdsCandidatePipelineResultsTransformer

}