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

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

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

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

import com.twitter.product\_mixer.core.model.common.CandidateWithFeatures

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

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

import com.twitter.stitch.Stitch

object ValidAdImpressionIdFilter extends Filter[PipelineQuery, AdsCandidate] {

override val identifier: FilterIdentifier = FilterIdentifier("ValidAdImpressionId")

override def apply(

query: PipelineQuery,

candidatesWithFeatures: Seq[CandidateWithFeatures[AdsCandidate]]

): Stitch[FilterResult[AdsCandidate]] = {

val (kept, removed) = candidatesWithFeatures

.map(\_.candidate)

.partition(candidate => candidate.adImpression.impressionString.exists(\_.nonEmpty))

Stitch.value(FilterResult(kept, removed))

}

}