package com.twitter.home\_mixer.functional\_component.filter

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.UniversalNoun

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

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

import com.twitter.stitch.Stitch

import com.twitter.timelines.configapi.FSBoundedParam

case class DropMaxCandidatesFilter[Candidate <: UniversalNoun[Any]](

maxCandidatesParam: FSBoundedParam[Int])

extends Filter[PipelineQuery, Candidate] {

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

override def apply(

query: PipelineQuery,

candidates: Seq[CandidateWithFeatures[Candidate]]

): Stitch[FilterResult[Candidate]] = {

val maxCandidates = query.params(maxCandidatesParam)

val (kept, removed) = candidates.map(\_.candidate).splitAt(maxCandidates)

Stitch.value(FilterResult(kept, removed))

}

}