package com.twitter.product\_mixer.component\_library.selector

import com.twitter.product\_mixer.component\_library.model.candidate.CursorCandidate

import com.twitter.product\_mixer.core.functional\_component.common.CandidateScope

import com.twitter.product\_mixer.core.functional\_component.common.SpecificPipelines

import com.twitter.product\_mixer.core.functional\_component.selector.Selector

import com.twitter.product\_mixer.core.functional\_component.selector.SelectorResult

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

import com.twitter.product\_mixer.core.model.common.presentation.CandidateWithDetails

import com.twitter.product\_mixer.core.model.common.presentation.ModuleCandidateWithDetails

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

import com.twitter.timelines.configapi.Param

/\*\*

\* Drop the module from the `result` if it doesn't contain enough item candidates.

\*

\* For example, for a given module, if minResultsParam is 3, and the results contain 2 items,

\* then that module will be entirely dropped from the results.

\*/

case class DropModuleTooFewModuleItemResults(

candidatePipeline: CandidatePipelineIdentifier,

minModuleItemsParam: Param[Int])

extends Selector[PipelineQuery] {

override val pipelineScope: CandidateScope = SpecificPipelines(candidatePipeline)

override def apply(

query: PipelineQuery,

remainingCandidates: Seq[CandidateWithDetails],

result: Seq[CandidateWithDetails]

): SelectorResult = {

val minModuleItemSelections = query.params(minModuleItemsParam)

assert(minModuleItemSelections > 0, "Min results must be greater than zero")

val updatedResults = result.filter {

case module: ModuleCandidateWithDetails

if pipelineScope.contains(module) && module.candidates.count { candidateWithDetails =>

!candidateWithDetails.candidate.isInstanceOf[CursorCandidate]

} < minModuleItemSelections =>

false

case \_ => true

}

SelectorResult(remainingCandidates = remainingCandidates, result = updatedResults)

}

}