package com.twitter.product\_mixer.component\_library.premarshaller.urt.builder

import com.twitter.product\_mixer.core.model.marshalling.response.urt.AddEntriesTimelineInstruction

import com.twitter.product\_mixer.core.model.marshalling.response.urt.TimelineEntry

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

/\*\*

\* Iterates over all the [[TimelineEntry]] passed it and creates `addEntry` entries in the URT for

\* any entries which are not pinned and not replaceable(cursors are replaceable)

\*

\* This is because pinned entries always show up in the `pinEntry` section, and replaceable entries

\* will show up in the `replaceEntry` section.

\*/

case class AddEntriesWithPinnedAndReplaceInstructionBuilder[Query <: PipelineQuery](

override val includeInstruction: IncludeInstruction[Query] = AlwaysInclude)

extends UrtInstructionBuilder[Query, AddEntriesTimelineInstruction] {

override def build(

query: Query,

entries: Seq[TimelineEntry]

): Seq[AddEntriesTimelineInstruction] = {

if (includeInstruction(query, entries)) {

val entriesToAdd = entries

.filterNot(\_.isPinned.getOrElse(false))

.filter(\_.entryIdToReplace.isEmpty)

if (entriesToAdd.nonEmpty) Seq(AddEntriesTimelineInstruction(entriesToAdd))

else Seq.empty

} else

Seq.empty

}

}