package com.twitter.home\_mixer.product.for\_you.side\_effect

import com.twitter.finagle.mtls.authentication.ServiceIdentifier

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

import javax.inject.Inject

import javax.inject.Singleton

@Singleton

case class ServedCandidateFeatureKeysKafkaSideEffectBuilder @Inject() (

injectedServiceIdentifier: ServiceIdentifier) {

def build(

sourceIdentifiers: Set[CandidatePipelineIdentifier]

): ServedCandidateFeatureKeysKafkaSideEffect = {

val topic = injectedServiceIdentifier.environment.toLowerCase match {

case "prod" => "tq\_ct\_served\_candidate\_feature\_keys"

case \_ => "tq\_ct\_served\_candidate\_feature\_keys\_staging"

}

new ServedCandidateFeatureKeysKafkaSideEffect(topic, sourceIdentifiers)

}

}