package com.twitter.unified\_user\_actions.client.config

import com.twitter.inject.Test

class KafkaConfigsSpec extends Test {

test("configs should be correct") {

val states = Seq(

(

KafkaConfigs.ProdUnifiedUserActions,

Constants.UuaProdEnv,

Constants.UuaKafkaTopicName,

Constants.UuaKafkaProdClusterName),

(

KafkaConfigs.ProdUnifiedUserActionsEngagementOnly,

Constants.UuaProdEnv,

Constants.UuaEngagementOnlyKafkaTopicName,

Constants.UuaKafkaProdClusterName),

(

KafkaConfigs.StagingUnifiedUserActions,

Constants.UuaStagingEnv,

Constants.UuaKafkaTopicName,

Constants.UuaKafkaStagingClusterName),

(

KafkaConfigs.StagingUnifiedUserActionsEngagementOnly,

Constants.UuaStagingEnv,

Constants.UuaEngagementOnlyKafkaTopicName,

Constants.UuaKafkaStagingClusterName)

)

states.foreach {

case (actual, expectedEnv, expectedTopic, expectedClusterName) =>

assert(expectedEnv == actual.environment.name, s"in $actual")

assert(expectedTopic == actual.topic, s"in $actual")

assert(expectedClusterName == actual.cluster.name, s"in $actual")

case \_ =>

}

}

}