package com.twitter.product\_mixer.component\_library.experiments.metrics

// Base trait for all placeholder values

sealed trait Named {

def name: String

}

case class Const(override val name: String) extends Named

// contains only client event patterns

case class CEPattern(

override val name: String,

client: String = "",

page: String = "",

section: String = "",

component: String = "",

element: String = "",

action: String = "",

strainer: String = "")

extends Named {

override def toString: String = {

"\"" + client + ":" + page + ":" + section + ":" + component + ":" + element + ":" + action + "\""

}

}

case class Topic(

override val name: String,

topicId: String = "")

extends Named

object PlaceholderConfig {

type PlaceholderKey = String

type Placeholder = Seq[Named]

type PlaceholdersMap = Map[PlaceholderKey, Placeholder]

}