package com.twitter.home\_mixer.product

import com.twitter.home\_mixer.model.request.FollowingProduct

import com.twitter.home\_mixer.model.request.ForYouProduct

import com.twitter.home\_mixer.model.request.ListRecommendedUsersProduct

import com.twitter.home\_mixer.model.request.ListTweetsProduct

import com.twitter.home\_mixer.model.request.ScoredTweetsProduct

import com.twitter.home\_mixer.model.request.SubscribedProduct

import com.twitter.home\_mixer.product.following.FollowingProductPipelineConfig

import com.twitter.home\_mixer.product.for\_you.ForYouProductPipelineConfig

import com.twitter.home\_mixer.product.list\_recommended\_users.ListRecommendedUsersProductPipelineConfig

import com.twitter.home\_mixer.product.scored\_tweets.ScoredTweetsProductPipelineConfig

import com.twitter.home\_mixer.product.list\_tweets.ListTweetsProductPipelineConfig

import com.twitter.home\_mixer.product.subscribed.SubscribedProductPipelineConfig

import com.twitter.inject.Injector

import com.twitter.product\_mixer.core.product.guice.ProductScope

import com.twitter.product\_mixer.core.product.registry.ProductPipelineRegistryConfig

import javax.inject.Inject

import javax.inject.Singleton

@Singleton

class HomeProductPipelineRegistryConfig @Inject() (

injector: Injector,

productScope: ProductScope)

extends ProductPipelineRegistryConfig {

private val followingProductPipelineConfig = productScope.let(FollowingProduct) {

injector.instance[FollowingProductPipelineConfig]

}

private val forYouProductPipelineConfig = productScope.let(ForYouProduct) {

injector.instance[ForYouProductPipelineConfig]

}

private val scoredTweetsProductPipelineConfig = productScope.let(ScoredTweetsProduct) {

injector.instance[ScoredTweetsProductPipelineConfig]

}

private val listTweetsProductPipelineConfig = productScope.let(ListTweetsProduct) {

injector.instance[ListTweetsProductPipelineConfig]

}

private val listRecommendedUsersProductPipelineConfig =

productScope.let(ListRecommendedUsersProduct) {

injector.instance[ListRecommendedUsersProductPipelineConfig]

}

private val subscribedProductPipelineConfig = productScope.let(SubscribedProduct) {

injector.instance[SubscribedProductPipelineConfig]

}

override val productPipelineConfigs = Seq(

followingProductPipelineConfig,

forYouProductPipelineConfig,

scoredTweetsProductPipelineConfig,

listTweetsProductPipelineConfig,

listRecommendedUsersProductPipelineConfig,

subscribedProductPipelineConfig,

)

}