package com.twitter.tweetypie

package repository

import com.twitter.stitch.Stitch

import com.twitter.takedown.util.TakedownReasons

import com.twitter.tseng.withholding.thriftscala.TakedownReason

/\*\*

\* Query TakedownReason objects from gizmoduck

\*

\* No backfill job has been completed so there may exist users that have a takedown

\* country\_code without a corresponding UnspecifiedReason takedown\_reason. Therefore,

\* read from both fields and merge into a set of TakedownReason, translating raw takedown

\* country\_code into TakedownReason.UnspecifiedReason(country\_code).

\*/

object UserTakedownRepository {

type Type = UserId => Stitch[Set[TakedownReason]]

val userQueryOptions: UserQueryOptions =

UserQueryOptions(Set(UserField.Takedowns), UserVisibility.All)

def apply(userRepo: UserRepository.Type): UserTakedownRepository.Type =

userId =>

userRepo(UserKey(userId = userId), userQueryOptions)

.map(\_.takedowns.map(TakedownReasons.userTakedownsToReasons).getOrElse(Set.empty))

}