cocoapods 을 활용한 코드

//

// ViewController.swift

// Movie\_BoxOffice

//

// Created by 신현호 on 2023/05/21.

//

import UIKit

import Foundation

import Alamofire // For network requests

struct NaverMovie {

let userRating: String

let link: String

let image: String

let title: String

}

struct MovieData: Codable {

let boxOfficeResult: BoxOfficeResult

}

struct BoxOfficeResult: Codable {

let dailyBoxOfficeList: [DailyBoxOfficeList]

}

struct DailyBoxOfficeList: Codable {

let movieNm : String

let audiAcc: String

let audiCnt: String

let openDt: String

let movieCd: String

}

class ViewController: UIViewController,UITableViewDelegate,UITableViewDataSource {

var moviedata : MovieData?

var movieURL = "https://kobis.or.kr/kobisopenapi/webservice/rest/boxoffice/searchDailyBoxOfficeList.json?key=30e77e7642a595f8537d082d2de83fc2&targetDt=" //20230510

func tableView(\_ tableView: UITableView, numberOfRowsInSection section: Int) -> Int {

return 10

}

func tableView(\_ tableView: UITableView, titleForHeaderInSection section: Int) -> String? {

return "박스오피스(영화진흥위원회제공: " + getYesterdayDate() + ")"

}

func tableView(\_ tableView: UITableView, cellForRowAt indexPath: IndexPath) -> UITableViewCell {

let cell = tableView.dequeueReusableCell(withIdentifier: "myCell", for: indexPath)

as! MyTableViewCell

guard let movieCode = moviedata?.boxOfficeResult.dailyBoxOfficeList[indexPath.row].movieCd else { return cell }

print(movieCode)

cell.Moviename.text = moviedata?.boxOfficeResult.dailyBoxOfficeList[indexPath.row].movieNm

guard let Acc = moviedata?.boxOfficeResult.dailyBoxOfficeList[indexPath.row].audiAcc else {

return cell

}

let numS = NumberFormatter()

numS.numberStyle = .decimal

guard let Accmulate = Int(Acc), let result = numS.string(for: Accmulate) else {

return cell

}

cell.MovieAcc.text = "누적: \(result) 명"

guard let openDt = moviedata?.boxOfficeResult.dailyBoxOfficeList[indexPath.row].openDt else {

cell.OpenDt.text = nil

return cell

}

let formattedOpenDate = "개봉일: " + openDt

cell.OpenDt.text = formattedOpenDate

fetchMovieInformation(query: "범죄도시3") { result in

switch result {

case .success(let movie):

cell.MovieRt.text = movie.userRating

DispatchQueue.main.async {

tableView.reloadRows(at: [indexPath], with: .automatic)

}

print(movie.userRating)

print(movie.title)

// Handle other properties as needed

case .failure(let error):

print("Error: \(error.localizedDescription)")

}

}

return cell

}

func fetchMovieInformation(query: String, completion: @escaping (Result<NaverMovie, Error>) -> Void) {

let naverId = "ctwIeRpCU4xcbGeFbhPP"

let apiKey = "bYHTkauZBG"

let urlString = "https://openapi.naver.com/v1/search/movie.json?query=\(query)"

let headers: HTTPHeaders = [

"X-Naver-Client-Id": naverId,

"X-Naver-Client-Secret": apiKey

]

AF.request(urlString, headers: headers).responseJSON { response in

switch response.result {

case .success(let data):

if let json = data as? [String: Any],

let items = json["items"] as? [[String: Any]],

let item = items.first {

let userRating = item["userRating"] as? String ?? ""

let link = item["link"] as? String ?? ""

let image = item["image"] as? String ?? ""

let title = item["title"] as? String ?? ""

// Create a movie object

let movie = NaverMovie(userRating: userRating, link: link, image: image, title: title)

completion(.success(movie))

} else {

let error = NSError(domain: "InvalidResponse", code: 0, userInfo: nil)

completion(.failure(error))

}

case .failure(let error):

completion(.failure(error))

}

}

}

func numberOfSections(in tableView: UITableView) -> Int {

return 1

}

@IBOutlet weak var table: UITableView!

override func viewDidLoad() {

super.viewDidLoad()

table.dataSource = self

table.delegate = self

movieURL += getYesterdayDate()

getData()

}

func getYesterdayDate() -> String {

let yesterday = Calendar.current.date(byAdding: .day, value: -1, to: Date()) // 어제의 날짜 가져오기

let dateFormatter = DateFormatter()

dateFormatter.dateFormat = "yyyyMMdd" // 원하는 날짜 형식 설정

return dateFormatter.string(from: yesterday!) // 날짜를 지정된 형식으로 문자열로 변환하여 반환

}

func getData() {

guard let url = URL(string: movieURL) else {

print("잘못된 URL")

return

}

let session = URLSession(configuration: .default)

let task = session.dataTask(with: url) { data, response, error in

if let error = error {

print("오류: \(error.localizedDescription)")

return

}

guard let JSONdata = data else {

print("데이터를 받지 못했습니다")

return

}

let dataString = String(data: JSONdata, encoding: .utf8)

let decoder = JSONDecoder()

do {

let decodedData = try decoder.decode(MovieData.self, from: JSONdata)

self.moviedata = decodedData

DispatchQueue.main.async {

self.table.reloadData()

}

} catch {

print("해독 오류: \(error)")

}

}

task.resume()

}

}