## 혹시나 모를 마우스오버 사태에 대비해 이렇게 미리 대피 준비합니다.

## 하... 이거 처음부터 다시 어떻게 하냐...

## 멜론 플레이리스트 // 제목에 따른 노래 가사 // 제목과 가사 상관관계 분석

library(RSelenium)

library(dplyr)

library(ggplot2)

library(KoNLP)

library(wordcloud)

library(wordcloud2)

library(htmlwidgets)

library(tm)

library(qgraph)

library(proxy)

# 데이터 스크래핑 순서

# 좋아요 개수 1000개 이상 클릭 -> 플레이리스트 제목 수집 -> 아래 가사 버튼 클릭 -> 가사 수집 -> csv 파일 저장

# 참고사항 : 전체 페이지는 1~35페이지

remDr <- remoteDriver(remoteServerAddr = "localhost" , port = 4445, browserName = "chrome")

remDr$open()

site <- "https://www.melon.com/genre/recmconts\_list.htm?gnrCode=GN0100&tabType=DJ#params%5BgnrCode%5D=GN0100&params%5BtabType%5D=DJ&po=pageObj&startIndex=81"

remDr$navigate(site)

# 페이지 넘어가기 셀렉터 / #pageObjNavgation > div > a

# 페이지 셀렉터 / #pageObjNavgation > div > span > a:nth-child(2)

# 좋아요 개수 selector / #djPlylstList > div > ul > li:nth-child(3) > div.entry > div.meta > button > span.cnt

# repeat 함수로 반복하기

# 10개씩 3번 넘어가기

for (x in 1:3) {

beyondpage <- remDr$findElement(using='css', '#pageObjNavgation > div > a.next')

beyondpage$clickElement()

Sys.sleep(2)

# 페이지 1~10, 11~20, 21~30, 31~35 넘어가기

for (i in 1:10) {

nextpage <- remDr$findElement(using='css', paste0('#pageObjNavgation > div > span > a:nth-child(', i, '),booksite'))

nextpage$clickElement()

Sys.sleep(2)

# 각 페이지 내 20개의 플레이리스트 훑어보기

for (b in 1:20){

likebtn<-remDr$findElement(using='css', paste0('#djPlylstList > div > ul > li:nth-child(', b,') > div.entry > div.meta > button > span.cnt'))

likebtn$getElementTagName() # 태그 // 리스트로 리턴

likebtn1 <- likebtn$getElementText()

x <- c(x, likebtn1)

checkcnt <- unlist(x)

checkcnt <- gsub("[[:punct:]]", "", checkcnt)

num <- as.numeric(checkcnt)

# 좋아요 1000개 이상 추출하기

# 좋아요 1000개 이상인 페이지 들어가기 // 플레이리스트 제목 클릭

like1000btn <- remDr$findElement(using='css', paste0('#djPlylstList > div > ul > li:nth-child(', 1000이상 원소 껴넣기,') > div.entry > div.info > a.ellipsis.album\_name)')

like1000btn\_title <- sapply(like1000btn, function(x) {x$getElementText()})

like1000btn$clickElement()

# 해시태그 추출하기 // 플레이리스트 내 해시태그 추출

hashtagnodes <- remDr$findElement(using='css', '#conts > div.section\_info.d\_djcol\_list > div > div.entry > div.tag\_list.type03 > a:nth-child() > span')

hastag <- sapply(hashtagnodes, function(x) {x$getElementText()})

hastag <- gsub("[[:cntrl:]]", "", hastag)

# 수록곡 만큼 반복하기 // 곡 정보 들어가기 & 가사 가져오기 // 뒤로 가기

songcntnodes <- remDr$findElement(using='css', '#conts > div.section\_contin > div.page\_header > h5 > span')

songcnt <- sapply(songcntnodes, function(x) {x$getElementText()})

songcnt <- gsub("[[:cntrl:]]", "", songcnt)

songcnt <- as.numeric(songcnt)

for (i in songcnt){

songclickbtn <- remDr$findElement(using='css', paste0('frm > div > table > tbody > tr:nth-child(', i,') > td:nth-child(4) > div > a'))

songclickbtn$clickElement()

songlyricnodes <- remDr$findElement(using='css', '#d\_video\_summary')

songlyric <- sapply(songlyric, function(x) {x$getElementText()})

songlyric <- gsub("[[:cntrl:][:cntrl:]]", "", songlyric)

}

#마지막에 remDr$goBack()

# 가사 저장하기

lyricnodes <- remDr$findElements(using='css', '')

lyric <- sapply(lyricnodes, function(x) {x$getElementText()})

lyric <- gsub("[[:cntrl:]]", "", lyric)

}

}

}

}