CS102 Activity 3

Vonkhar

2024-02-21

```
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(polite)
library(xml2)
library(magrittr)
library(rvest)
library(httr)
```

```
Keys <- c('g4w6ddbmqyzdo6ic4oxwjnrsr3smwajy3ykdd6hpaxb7yv35pjt6udc2oe4vxnbnb4dy3ik6g6vacybfohautcdhjuwl
          'g4w6ddbmqyzdo6ic4oxwjnzxrps44bzt2mnt777natepeud5pjt6uds6oa4vlnrnb4dwpyddjbidtthzjvknhdvuacda
          'g4w6ddbmqyzdo6ic4oxwjnzsrpsmybzz3motv6hfahep2tttfjrk4d2yoqyfjmbyods2dhi6gej2z6drga52tx4yjija
          'g4w6ddbmqyzdo6ic4oxwjnbzrhtmqbby3ektr6hmb3d74wt5pjt6udc7om2vjmjib4d3sbijuib73l2uugyiqamq4w7v
          'g4w6ddbmqyzdo6ic4oxwjnbsqlr4ubjy3ektd4poadb76tttfjrk4dcyouyvxpzyoakcnhar44ks4qjs7yrcpyxmb3ja
          'g4w6ddbmqyzdo6ic4oxwjnbtrlu4ucj634ndd6xeb7b7gtttfjrk4dssoa4vvnryoav7lvtwyifymnhf33vpv465m4lq
          'g4w6ddbmqyzdo6ic4oxwjnjyrtq4qab63qktx77ialapuv35pjt6ucs4ouyfvnrdb4dsn5kzynt3hwwniviy5dpiuepy
          'g4w6ddbmqyzdo6ic4oxwjnjwqpu4qbr53int56hmb7d6mqbdf4q26bk5peyvxjchswwsblrza5z6dhvoubu5gs7jym',
          'g4w6ddbmqyzdo6ic4oxwjnjxrhu42bz63imdv6ppapc7gu35pjt6udk2oizf7prob4dyfngiexvmcltazeibjono6uvh
          'g4w6ddbmqyzdo6ic4oxwjnjurxr42bz23eotd6hla7a7qvdgoq366ss5oyyflmrjdr4k4iy5dixztbdig3djhuzqdazd
          'g4w6ddbmqyzdo6ic4oxwjnjurlsmwbr43intz7xmape7gulhoq366ss3oezv5nzjdj4k5kyby542dlm332rbqvyxzrur
          'g4w6ddbmqyzdo6ic4oxwjnjvr3qm2bb33ent56pka7b74v3foq366ss3oezv5mjidb4k5kjwnydlc3r3nzm5kw7sx3ag
          ,'g4w6ddbmqyzdo6ic4oxwjnjuqprmyabt3eotr6peapapstttfjrk2ccspa4vxmryobs64u2jysdar2rm6qp5aykh3qj


movie_keys <- c('tt0167260', 'tt0110912', 'tt0109830','tt0137523', 'tt1375666', 'tt0088763', 'tt0245429
                'tt6751668', 'tt0172495')

titles <- character(0)
names <- character(0)
dates <- character(0)
ratings <- character(0)
reviews <- character(0)
updateTitle <- character(0)
```

```
updateName <- character(0)
updateDate <- character(0)
updateRating <- character(0)
updateReview <- character(0)
```

```
#Main Loop changes the movie after it loops through the inner loop and scrapes 300+ reviews
for (i in 1:length(movie_keys)) {
  movie_url <- paste('https://www.imdb.com/title/', movie_keys[i], '/reviews/_ajax?', sep = "")
#Inner loop grabs the pagination key and gets 300+ reviews
  for (j in 1:13) {


    url <- paste(movie_url, 'paginationKey=', Keys[j], sep = "")

    session <- bow(url,
                   user_agent = "Educational")
    session

#Parent Node
#Created a parent object to get the div container in order to also scrape N/A Values
    parent <- scrape(session) %>%
      html_elements('div.review-container')

    reviewTitle <- parent %>%
      html_node('.title') %>%
      html_text()

    reviewerName <- parent %>%
      html_node('.display-name-link') %>%
      html_text()

    dateUploaded <- parent %>%
      html_node('.review-date') %>%
      html_text()

    userRating <- parent %>%
      html_node('.ipl-ratings-bar') %>%
      html_text()

    reviewContent <- parent %>%
      html_node('.content') %>%
      html_text()

    titles <- c(titles, reviewTitle)
    names <- c(names, reviewerName)
    dates <- c(dates, dateUploaded)
    ratings <- c(ratings, userRating)
    reviews <- c(reviews, reviewContent)

  }
  #Only get the first 300 Reviews
  updateTitle <- c(updateTitle, titles[1:300])
  updateName <- c(updateName, names[1:300])
  updateDate <- c(updateDate, dates[1:300])
```

```
  updateRating <- c(updateRating, ratings[1:300])
  updateReview <- c(updateReview, reviews[1:300])
}
```

```
df <- data.frame(updateTitle, updateName, updateDate, updateReview, updateRating)
```