Question2B

Nabila Doctor

March 14, 2020

library(rvest)  
library(tidyverse)  
library(stringr)

# Modified Function

boxoffice <- function() {  
 movie\_html = read\_html("https://www.imdb.com/chart/boxoffice")  
 movie\_table = html\_table(html\_nodes(movie\_html, "table")[[1]])  
   
 movie\_table$Weekend = as.numeric(gsub("[$M]", "", movie\_table$Weekend))  
 movie\_table$Gross = as.numeric(gsub("[$M]", "", movie\_table$Gross))  
   
 boxoffice\_data = tibble(Name = movie\_table$Title,   
 Gross = movie\_table$Gross,   
 Weeks = as.numeric(movie\_table$Weeks),  
 RT = NA)  
   
 boxoffice\_data = boxoffice\_data %>%   
 mutate('PerWeek ($M)' = Gross / Weeks) %>%   
 select(Name, Gross, 'PerWeek ($M)',RT) %>%   
 select('BoxOffice ($M)' = Gross)   
   
 # will add to this url  
 rt\_url = 'https://www.rottentomatoes.com/m/'  
   
 # for each movie name in boxoffice\_data, this loop will run  
 for (i in 1:nrow(boxoffice\_data)) {  
 moviename = tolower(boxoffice\_data$Name[i])  
 moviename = gsub("[:.-]","", moviename)  
   
 # create a character list of the movie name  
 if (grepl(" ", moviename)) {  
 moviename = strsplit(moviename, '\\s+')[[1]] # split on spaces  
 }  
   
 if (length(moviename) > 1) {  
   
 url\_end = paste(moviename[1],moviename[2],sep = "\_")  
 j = 2  
   
 # Goal of while: adding each word to the url, one by one  
 # and stopping once we found a 'working' url  
 while(j<=length(moviename)){  
 rt\_movieurl = paste(rt\_url,url\_end,sep="")  
   
 rt\_html = try(read\_html(rt\_movieurl), silent = T) # testing the url  
   
 if ("try-error" %in% class(rt\_html)) { # if error  
   
 if (j == length(moviename)) {  
 url\_end = paste(url\_end,"2020",sep = "\_")  
 rt\_movieurl = paste(rt\_url,url\_end,sep="")  
   
 j = length(moviename)+1  
   
 } else {  
 url\_end = paste(url\_end,moviename[j+1],sep = "\_")  
   
 j = j+1  
 }  
 } else { # otherwise, no error  
 j = length(moviename)+1  
   
 # check date of movie is greater than or equal to year 2019  
 tmp = readLines(rt\_movieurl)  
   
 tmp = grep("<div class=\"meta-value\">",t)[5]  
 tp = trimws(t[tmp:tmp+1])  
 tp = gsub("<.\*?>","",tp)  
 tp = strsplit(gsub("<.\*?>","",tp), '\\s+')[[1]]  
   
 if (!(as.numeric(tp[3]) >= 2019)){  
 rt\_movieurl = paste(rt\_movieurl,"2020",sep="\_")  
 }  
 }  
 }  
   
 } else { # For length of one  
 rt\_movieurl = paste(rt\_url,moviename,sep="")  
   
 rt\_html = try(read\_html(rt\_movieurl), silent = T)  
   
 if ("try-error" %in% class(rt\_html)) {  
 rt\_movieurl = paste(rt\_movieurl,"2020",sep = "\_")  
 }  
 }  
   
 rt\_html = try(read\_html(rt\_movieurl), silent = T) # testing the url  
   
 if ("try-error" %in% class(rt\_html)) { # ratings for movies I couldn't match  
 rt\_rating = "NA"  
   
 } else {  
 rt\_html = readLines(rt\_movieurl)  
   
 a = grep('<span class=\"mop-ratings-wrap\_\_percentage\">', rt\_html)[1]  
   
 if (is.na(a)) {  
 rt\_movieurl = paste(rt\_movieurl,"2020",sep="\_")  
   
 rt\_html = readLines(rt\_movieurl)  
 a = grep('<span class=\"mop-ratings-wrap\_\_percentage\">', rt\_html)[1]  
 rt\_rating = trimws(rt\_html[a:a+1])  
   
 } else {  
 rt\_rating = trimws(rt\_html[a:a+1])  
 }  
 }  
   
 boxoffice\_data$RT[i] = rt\_rating  
 }  
   
 return(boxoffice\_data)  
}

**Calling Function and Providing the Database**

dat = boxoffice()

A picture containing screenshot

Description automatically generated