Final Project Webscraping and SQL

B. Taczy, J. Wilcox J. Zimmerman

3/1/2023

```
require(rvest)

## Loading required package: rvest

require(stringr)

## Loading required package: stringr

#Read the html code from the metacritic webpage, and store it in the webpage
variable
webpage=read_html("https://www.metacritic.com/browse/games/score/metascore/al
l/all/filtered")

#select the ".numbered" node in the webpage to get the HTML that contains the
rank of the games from the website HTML
rankH=html_nodes(webpage,".numbered")


#Take the HTML code that is stored in the rankH variable and use html_text to
extract the rank value
rank_r=html_text(rankH)

#Clean rank_r data by storing the rank data in the rank variable by selecting
a portion of the rank_r string
rank=substring(rank_r,70,73)


#remove the ./n from each value of rank, and then remove the extra . from the
100th value of rank
rank_new=gsub(".\n","",rank)
rank_new[100] = substring(rank_new[100],1,3)
rank_new = str_trim(rank_new)
rank_new = as.numeric(rank_new)

#select the ".clamp-metascore" node in the webpage to get the HTML that
contains the metascore of the games from the website HTML
metascore_rankH=html_nodes(webpage, ".clamp-metascore")

#Take the HTML code that is stored in the metascore variable and use
html_text to extract the metascore value
metascore_r = html_text(metascore_rankH)
```

```
#Clean metascore_r data by selecting a portion of the metascore_r string and
saving it to the metascore_rank variable
metascore_rank=substring(metascore_r,93,94)
metascore_rank =as.numeric(metascore_rank)

#select the ".clamp-userscore" node in the webpage to get the HTML that
contains the userscore of the games from the website HTML
user_rankH=html_nodes(webpage, ".clamp-userscore")

#Take the HTML code that is stored in the user_r variable and use html_text
to extract the userscore value
user_r = html_text(user_rankH)

#Store the 92nd through the 94th character in the user_rank variable
user_rank=substring(user_r,92,94)
userrank = as.numeric(user_rank)

#select the ".platform" node in the webpage to get the HTML that contains the
platform of the games from the website HTML
platformH=html_nodes(webpage, ".platform")

#Take the HTML code that is stored in the platformH variable and use
html_text to extract the platform value
platform_r = html_text(platformH)


#Take the substring from 125th character to the 138th character of platform_r
and store in the platform variable
platform=substring(platform_r,125,138)


#Remove the \n from each of the strings in platform using gsub
platform_sub=gsub("\n"," ",platform)



#first element not apart of actual list, so remove it from platform_sub
platform_new=platform_sub[-1]


#Use str_trim to remove all whitespace from the strings in platform_new
platform_new=str_trim(platform_new)


#select the ".clamp-details" node in the webpage to get the HTML that
contains the release date of the games from the website HTML
release_dateH=html_nodes(webpage,".clamp-details")
```

```
#Take the HTML code that is stored in the release_dateH variable and use
html_text to extract the date value
date_r=html_text(release_dateH)


#Take the substring of date_r that contains the date value and remove the
whitespace
date=substring(date_r,451,501)
date=str_trim(date)


#take the substring of date from the first element up until the 8th element
from the end to get the month and then remove the whitespace
month=substring(date,1,nchar(date)-8)
month=str_trim(month)


#year of release
#take the substring of date that contains the year value and save it in the
year variable
year=substring(date,nchar(date)-3,nchar(date)+3)


#select the ".title h3" node in the webpage to get the HTML that contains the
name of the games from the website HTML
nameH=html_nodes(webpage,".title h3")


#Take the HTML code that is stored in the nameH variable and use html_text to
extract the name value
name=html_text(nameH)


#Add all of these colunns together into a data frame called data, name the
columns and then write that datable to a file called Videogames.txt
data=cbind(rank_new,name,metascore_rank,user_rank,platform_new,month,year)
colnames(data)=c("Overall Rank","Title","Metascore","User Rank",
"Platform","Release Month","Release Year")
head(data)
```

```
##      Overall Rank Title                                Metascore User
Rank
## [1,] "1"          "The Legend of Zelda: Ocarina of Time" "99"      "9.1"
## [2,] "2"          "Tony Hawk's Pro Skater 2"             "98"      "7.5"
## [3,] "3"          "Grand Theft Auto IV"                  "98"      "7.8"
## [4,] "4"          "SoulCalibur"                          "98"      "8.5"
## [5,] "5"          "Grand Theft Auto IV"                  "98"      "8.0"
```

```
## [6,] "6"          "Super Mario Galaxy"                    "97"      "9.1"
##      Platform        Release Month Release Year
## [1,] "Nintendo 64"   "November"    "1998"
## [2,] "PlayStation"   "September"   "2000"
## [3,] "PlayStation 3" "April"       "2008"
## [4,] "Dreamcast"     "September"   "1999"
## [5,] "Xbox 360"      "April"       "2008"
## [6,] "Wii"           "November"    "2007"

write.table(data,"Videogames.txt", row.names = FALSE)
```

Table Creation in MySQL:

```
Create Table VideoGames(
 Overall_Rank  Int not null primary key,
 Title  Char(125),
 Metascore  Int,
 User_Rank float,
 Platform  Char(25),
 Release_Month  Char(25),
 Release_Year  Int
 );
```

SQL Queries:

1. Which years had an average score of 9.0 or higher for games released?

```
Select Release_Year
from VideoGames
group by Release_Year
having avg(User_Rank) > 9.0
order by Release_Year asc limit 5;
```

2. How many games were released in each month? Show in decreasing order.

```
Select Release_Month, count(Overall_Rank) as Number_Released
from Video_Games
group by Release_Month
order by Number_Released desc limit 5;
```

3. How many games were released in each year? Show in decreasing order.

```
Select Release_Year, count(Overall_Rank) as Number_Released
from VideoGames
```

```
group by Release_Year
order by Number_Released desc;
```

4. What is the average User_Rank of Grand Theft Auto IV across all platforms?

```
Select Title, avg(User_Rank) as Average_Rank
from VideoGames
where Title = "Grand Theft Auto IV"
group by Title
order by Average_Rank desc;
```

5. Order the games from highest UserRank to lowest based on average UserRank across all platforms.

```
Select Title, avg(User_Rank) as Average_Rank
from VideoGames
group by Title
order by Average_Rank desc;
```