Project 8 --SEYI OGUNMODEDE

Instructor help: Dr. Mark

- Assisted with videos online to put through and Help with figuring out how to write a function.

TA: Ik Su Jeong

Question 1

In [1]:
```
tricy <- function (k) {
    # i can write other lines of R here
    k*k*k   # this is the value returned by the function at the end
}
```

In [2]:
```
tricy(20)
```

8000

In [3]:
```
library(data.table)
```

In [4]:
```
titles <- data.frame(fread("/anvil/projects/tdm/data/movies_and_tv/titles.csv"))
```

In [5]:
```
episodes <- data.frame(fread("/anvil/projects/tdm/data/movies_and_tv/episodes.csv"))
```

In [6]:
```
people <- data.frame(fread("/anvil/projects/tdm/data/movies_and_tv/people.csv"))
```

In [7]:
```
ratings <- data.frame(fread("/anvil/projects/tdm/data/movies_and_tv/ratings.csv"))
```

In [8]:
```
head(titles)
```

A data.frame: 6 x 9

	title_id	type	primary_title	original_title	is_adult	premiered	ended	runtime_minutes	
	<chr>	<chr>	<chr>	<chr>	<int>	<int>	<int>	<int>	
1	tt0000001	short	Carmencita	Carmencita	0	1894	NA	1	
2	tt0000002	short	Le clown et ses chiens	Le clown et ses chiens	0	1892	NA	5	
3	tt0000003	short	Pauvre Pierrot	Pauvre Pierrot	0	1892	NA	4	Anima
4	tt0000004	short	Un bon bock	Un bon bock	0	1892	NA	12	
5	tt0000005	short	Blacksmith Scene	Blacksmith Scene	0	1893	NA	1	
6	tt0000006	short	Chinese Opium Den	Chinese Opium Den	0	1894	NA	1	

In [9]:
```
table(titles$type)
```

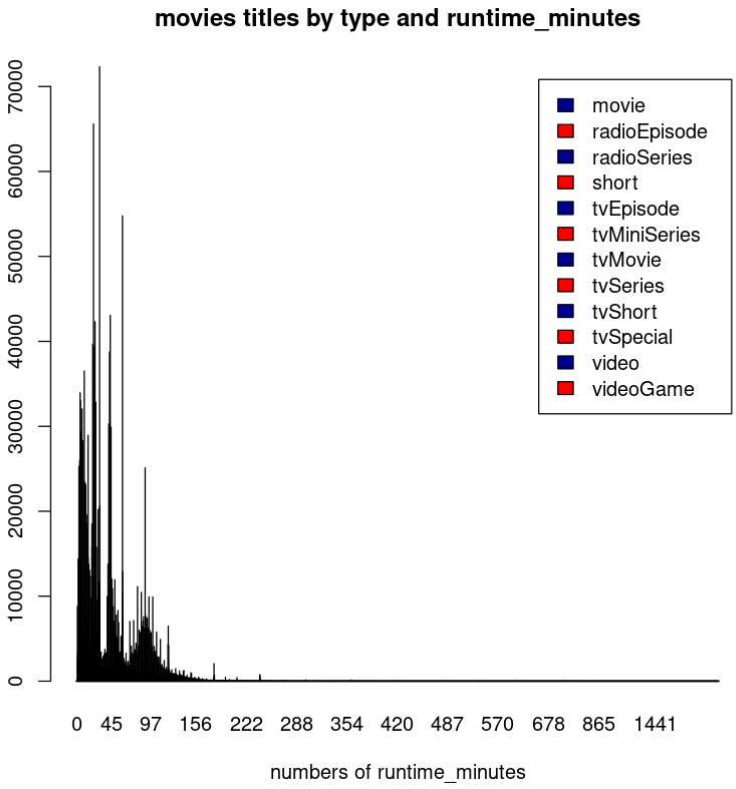
```
       movie radioEpisode  radioSeries        short    tvEpisode tvMiniSeries 
      581731            1            1       819518      5897385        38185 
    tvMovie     tvSeries      tvShort    tvSpecial        video    videoGame 
      132002       210273        10136        33255       313270        28502 
```

In [10]:
```
head(table(titles$runtime_minutes))
```

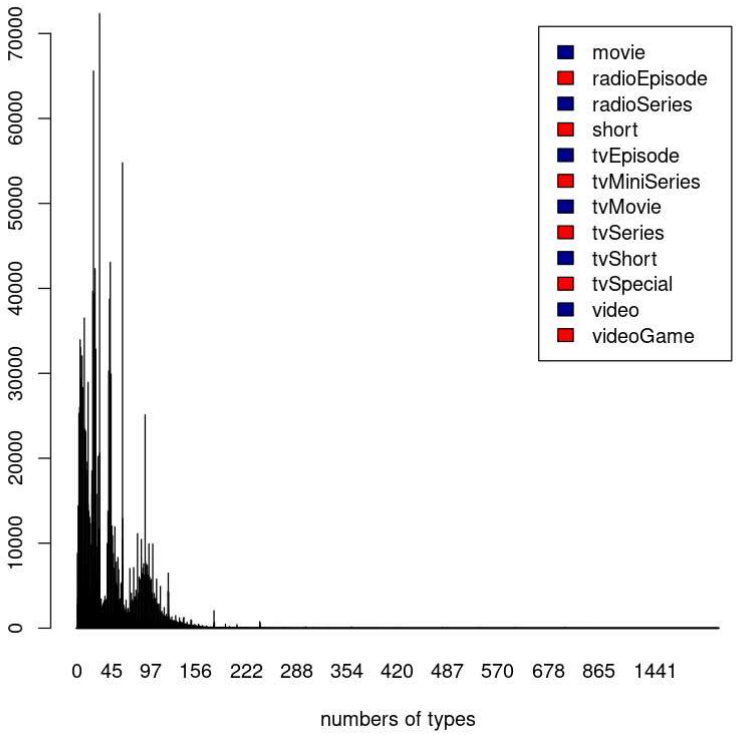
```
    0     1     2     3     4     5 
   10 16069 25790 56038 67054 57258 
```

In [11]:
```
counts <- table(titles$type, titles$runtime_minutes)
```

In [12]:
```
barplot(counts, main="movies titles by type and runtime_minutes",
        xlab= "numbers of runtime_minutes", col= c("darkblue","red"),
        legend = rownames(counts), beside=TRUE)
```



In [13]:
```
barplot(counts, main="movies titles by type and runtime_minutes",
        xlab= "numbers of types", col= c("darkblue","red"),
        legend = rownames(counts), beside=TRUE)
```



In [14]:
```
table(titles$type)
```

```
     movie radioEpisode  radioSeries        short    tvEpisode tvMiniSeries
    581731            1            1       819518      5897385        38185
   tvMovie     tvSeries      tvShort    tvSpecial        video    videoGame
    132002       210273        10136        33255       313270        28502
```

In [15]:
```
titlesub <- subset(titles, (titles$type=="tvMovie")
                  | (titles$type =="videoGame"))
```

In [16]:
```
dim(titlesub)
```

160504 · 9

In [17]:
```
head(titlesub)
```

A data.frame: 6 x 9

	title_id	type	primary_title	original_title	is_adult	premiered	ended	runtime_minutes
	<chr>	<chr>	<chr>	<chr>	<int>	<int>	<int>	<int>
29774	tt0030298	tvMovie	Julius Caesar	Julius Caesar	0	1938	NA	101
37643	tt0038309	tvMovie	As You Like It	As You Like It	0	1946	NA	NA
38066	tt0038738	tvMovie	A Midsummer Night's Dream	A Midsummer Night's Dream	0	1946	NA	150
38766	tt0039445	tvMovie	Hamlet Part 1	Hamlet Part 1	0	1947	NA	88
38939	tt0039618	tvMovie	The Merchant of Venice	The Merchant of Venice	0	1947	NA	90
38945	tt0039624	tvMovie	A Midsummer Night's Dream	A Midsummer Night's Dream	0	1947	NA	NA

In [18]:
```
tail(titlesub)
```

A data.frame: 6 x 9

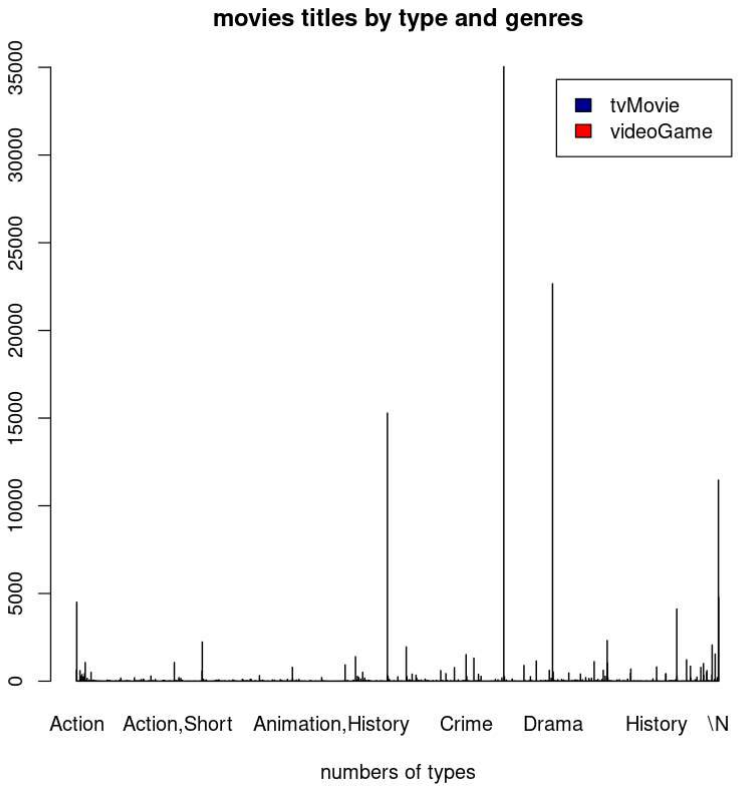
	title_id	type	primary_title	original_title	is_adult	premiered	ended	runtime_minu
	<chr>	<chr>	<chr>	<chr>	<int>	<int>	<int>	<i
8063880	tt9916064	videoGame	Aero Dancing i	Aero Dancing i	0	2001	NA	
8063940	tt9916192	tvMovie	Danielle Darrieux: Il est poli d'tre gai!	Danielle Darrieux: Il est poli d'tre gai!	0	2019	NA	
8063967	tt9916248	tvMovie	Mahiwaga	Mahiwaga	0	1996	NA	
8064071	tt9916460	tvMovie	Pink Taxi	Pink Taxi	0	2019	NA	
8064119	tt9916560	tvMovie	March of Dimes Presents: Once Upon a Dime	March of Dimes Presents: Once Upon a Dime	0	1963	NA	
8064181	tt9916692	tvMovie	Teatroteka: Czlowiek bez twarzy	Teatroteka: Czlowiek bez twarzy	0	2015	NA	

In [19]:
```
table(titlesub$type)
```

```
  tvMovie videoGame
   132002     28502
```

In [20]:
```
formms <- table(titlesub$ type, titlesub$genres)
```

In [21]:
```
barplot(formms, main="movies titles by type and genres",
        xlab= "numbers of types", col= c("darkblue","red"),
        legend = rownames(formms), beside=TRUE)
```



In [22]:
```
head(sort(table(titles$genres), decreasing=TRUE))
```

```
       Drama           \\N       Comedy    Talk-Show  Documentary 
      865049       629106       560246       445891       385909 
Drama,Romance 
      351919 
```

In [23]:
```
titlesuper <- subset(titlesub, (titlesub$genres=="Action")
                    | (titlesub$genres =="Sport"))
```

In [24]:
```
dim(titlesuper)
```

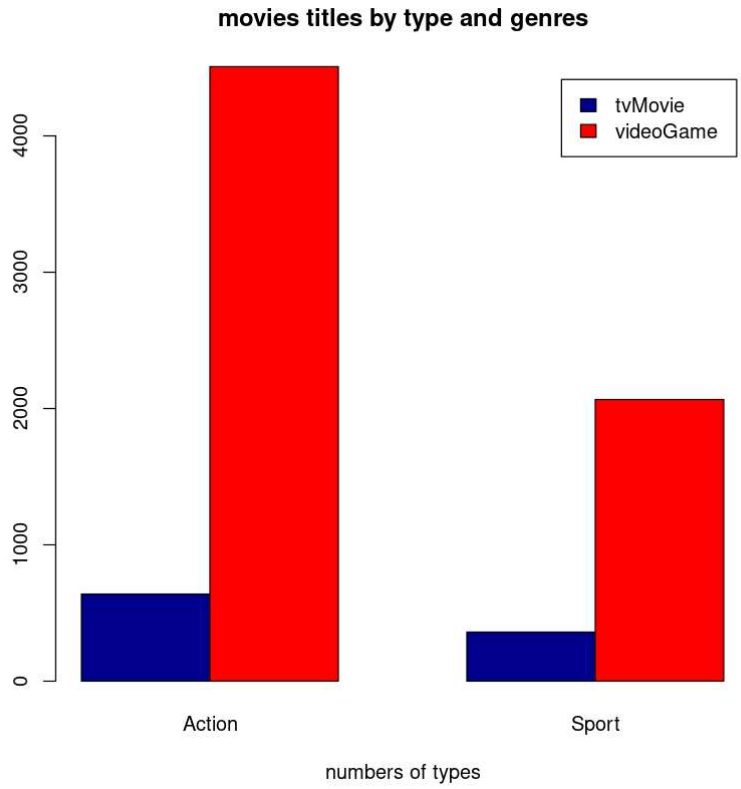
$7573 \cdot 9$

In [25]:
```
table(titlesuper$genres)
```

```
Action  Sport 
  5147   2426 
```

In [26]:
```
dormms <- table(titlesuper$ type, titlesuper$genres)
```

In [27]:
```
barplot(dormms, main="movies titles by type and genres",
        xlab= "numbers of types", col= c("darkblue","red"),
        legend = rownames(dormms), beside=TRUE)
```



In [28]:
```
head(titles)
```

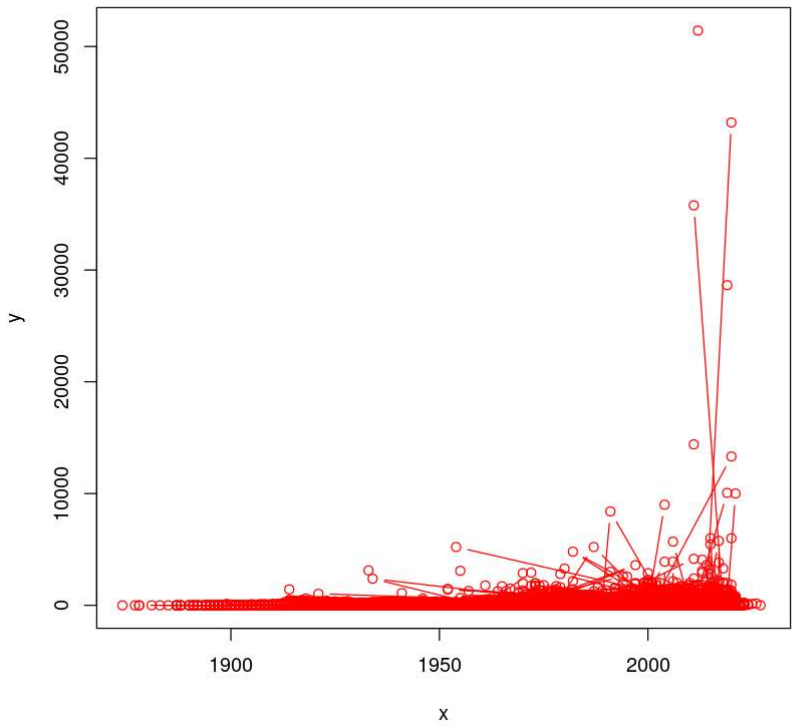
A data.frame: 6 x 9

	title_id	type	primary_title	original_title	is_adult	premiered	ended	runtime_minutes	
	<chr>	<chr>	<chr>	<chr>	<int>	<int>	<int>	<int>	
1	tt0000001	short	Carmencita	Carmencita	0	1894	NA	1	
2	tt0000002	short	Le clown et ses chiens	Le clown et ses chiens	0	1892	NA	5	
3	tt0000003	short	Pauvre Pierrot	Pauvre Pierrot	0	1892	NA	4	Anima
4	tt0000004	short	Un bon bock	Un bon bock	0	1892	NA	12	
5	tt0000005	short	Blacksmith Scene	Blacksmith Scene	0	1893	NA	1	
6	tt0000006	short	Chinese Opium Den	Chinese Opium Den	0	1894	NA	1	

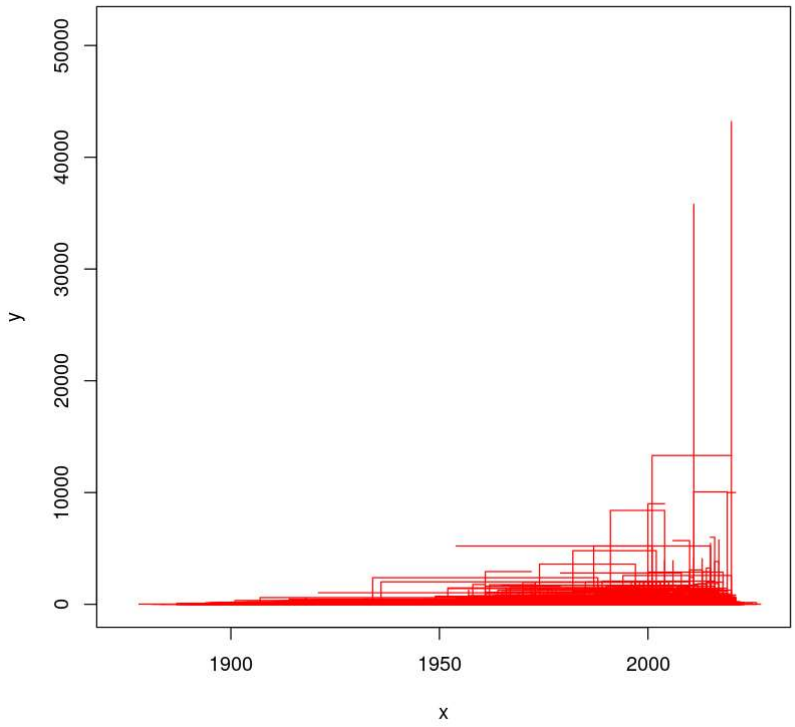
In [29]:
```
x<- titles$premiered
```

In [30]:
```
y<- titles$runtime_minutes
```

In [31]:
```
plot(x, y, type = "b",
     col = "red", xlab = "x", ylab = "y")
```



In [32]:
```
plot(x, y, type = "s",
     col = "red", xlab = "x", ylab = "y")
```



A grouped bar

A line plot

What information are you gaining from either of these graphs?

In the bar plot the values determine the heights of the bars, also known as side by side bar plot or clustered bar chart. While the line plot is, you will have the cordinate vectors of points to join, and different types of plotting to allow/to choose from.

Question 2

In [33]:
```
find_movie_with_at_least_rating <- function(titles_df, ratings_df,
                                            ratings_of_at_least) {
    results <- merge(ratings_df, titles_df,
                     by.x = "title_id", by.y = "title_id")
    popular_movie_results <- results[results$type == "movie" &
                                     results$rating >= ratings_of_at_least, ]
    return(popular_movie_results)
}
```

In [34]:
```
newDF <- find_movie_with_at_least_rating(titles, ratings, 9)
# outside the function you can pass in anything you like.
```

In [35]:
```
dim(titles)
```

8064259 · 9

In [36]:
```
dim (ratings)
```

1170056 · 3

In [37]:
```
dim (newDF)
```

2249 · 11

In [38]:
```
head (newDF)
```

A data.frame: 6 x 11

	title_id	rating	votes	type	primary_title	original_title	is_adult	premiered	ended	runti
	<chr>	<dbl>	<int>	<chr>	<chr>	<chr>	<int>	<int>	<int>	
1867	tt0006371	9.2	5	movie	Arms and the Woman	Arms and the Woman	0	1916	NA	
2071	tt0007326	9.0	5	movie	Seventeen	Seventeen	0	1916	NA	
2591	tt0009851	9.4	7	movie	Masked Ball	<81>larcosbl	0	1917	NA	
3230	tt0012297	9.4	12	movie	Hunger... Hunger... Hunger	Golod... golod... golod	0	1921	NA	
3314	tt0012623	9.6	5	movie	The Rider of the King Log	The Rider of the King Log	0	1921	NA	
5320	tt0018359	9.2	5	movie	The Satin Woman	The Satin Woman	0	1927	NA	

In [39]:
```
tail (newDF)
```

A data.frame: 6 x 11

	title_id	rating	votes	type	primary_title	original_title	is_adult	premiered	ended	ru
	<chr>	<dbl>	<int>	<chr>	<chr>	<chr>	<int>	<int>	<int>	
1167663	tt9839008	9.0	6	movie	Man in Manchester of Maharashtra	Man in Manchester of Maharashtra	0	2018	NA	
1168463	tt9866712	9.2	39	movie	Aadim Vichar	Aadim Vichar	0	2014	NA	
1168745	tt9876582	9.6	7	movie	Route 66	Route 66	0	2018	NA	
1168881	tt9880724	9.1	58	movie	Verdict 19	Verdict 19	0	2019	NA	
1169168	tt9890120	9.2	5	movie	Resurrection Corporation	Resurrection Corporation	0	2021	NA	
1169637	tt9905476	9.0	22	movie	Hand Rolled	Hand Rolled	0	2019	NA	

Markdown notes and sentences and analysis written here.

Question 3

In [40]:
```
find_movie_with_at_least_rating <- function(titles_df, ratings_df,
                                            ratings_of_at_least) {
    # the function contain the different data frame for titles,
    ratings as i have them above in question 1 folder and the least rating
     results <- merge(ratings_df, titles_df, by.x = "title_id", by.y = "title_id")
    # create another folder for merging
    # use the command for merging to create merging folder
    popular_movie_results <- results[results$type == "movie" &
```

```
                                        results$rating >= ratings_of_at_least, ]
    # after merging you run in the result folder,
    for type and ratings and the least in consideration
    return(popular_movie_results)
    # it will return the popular movie result
}
```

Markdown notes and sentences and analysis written here.

Question 4

In [41]:
```
find_movie_with_at_least_rating <- function(titles_df, ratings_df,
                                            ratings_of_at_least) {
    results <- merge(ratings_df, titles_df,
                     by.x = "title_id", by.y = "title_id")
    popular_movie_results <- results[results$type == "movie" &
                                        results$rating >= ratings_of_at_least, ]
    return(popular_movie_results)
}
```

In [42]:
```
my_selection <- find_movie_with_at_least_rating(titles, ratings, 7.6)
```

In [43]:
```
dim(my_selection)
```

34340 · 11

In [44]:
```
head(my_selection)
```

A data.frame: 6 x 11

	title_id	rating	votes	type	primary_title	original_title	is_adult	premiered	ended	runti
	<chr>	<dbl>	<int>	<chr>	<chr>	<chr>	<int>	<int>	<int>	
834	tt0001498	7.7	6	movie	The Battle of Trafalgar	The Battle of Trafalgar	0	1911	NA	
1057	tt0002305	7.6	17	movie	Life of Villa	Life of Villa	0	1912	NA	
1153	tt0002637	7.7	6	movie	Arizona	Arizona	0	1913	NA	
1188	tt0002832	7.8	10	movie	Evangeline	Evangeline	0	1914	NA	
1282	tt0003386	7.7	9	movie	Sodoms Ende	Sodoms Ende	0	1913	NA	
1362	tt0003748	8.3	6	movie	Captain Alvarez	Captain Alvarez	0	1914	NA	

In [45]:
```
tail(my_selection)
```

A data.frame: 6 x 11

	title_id	rating	votes	type	primary_title	original_title	is_adult	premiered	enc
	<chr>	<dbl>	<int>	<chr>	<chr>	<chr>	<int>	<int>	<ir
1169830	tt9911774	7.7	267	movie	Padmavyuhathile Abhimanyu	Padmavyuhathile Abhimanyu	0	2019	
1169969	tt9914254	7.9	7	movie	A Cherry Tale	Kirsebreventyret	0	2019	
1169981	tt9914642	8.1	32	movie	Albatross	Albatross	0	2017	
1169982	tt9914644	8.3	69	movie	9/11: Escape from the Towers	9/11: Escape from the Towers	0	2018	
1169991	tt9914972	7.9	9	movie	Blind Ambition	Blind Ambition	0	2021	
1170046	tt9916538	8.3	6	movie	Kuambil Lagi Hatiku	Kuambil Lagi Hatiku	0	2019	

In [46]:
```
find_movie_with_at_most_rating <- function(titles_df, ratings_df,
                                           ratings_of_at_most) {
    results <- merge(ratings_df, titles_df,
                     by.x = "title_id", by.y = "title_id")
    popular_movie_results <- results[results$type == "movie" &
                                     results$rating <= ratings_of_at_most, ]
    return(popular_movie_results)
}
```

In [47]:
```
my_view <- find_movie_with_at_most_rating(titles, ratings, 6.0)
```

In [48]:
```
dim(my_view)
```

114707 · 11

In [49]:
```
head(my_view)
```

A data.frame: 6 x 11

	title_id	rating	votes	type	primary_title	original_title	is_adult	premiered	ended	runtim
	<chr>	<dbl>	<int>	<chr>	<chr>	<chr>	<int>	<int>	<int>	
341	tt0000502	4.5	8	movie	Bohemios	Bohemios	0	1905	NA	
384	tt0000591	5.2	6	movie	The Prodigal Son	L'enfant prodigue	0	1907	NA	
399	tt0000615	4.5	14	movie	Robbery Under Arms	Robbery Under Arms	0	1907	NA	
407	tt0000630	3.8	12	movie	Hamlet	Amleto	0	1908	NA	
437	tt0000675	4.9	9	movie	Don Quijote	Don Quijote	0	1908	NA	
440	tt0000679	5.2	35	movie	The Fairylogue and Radio-Plays	The Fairylogue and Radio-Plays	0	1908	NA	

In [50]:
```
tail(my_view)
```

A data.frame: 6 x 11

	title_id	rating	votes	type	primary_title	original_title	is_adult	premiered	ended	ru
	<chr>	<dbl>	<int>	<chr>	<chr>	<chr>	<int>	<int>	<int>	
1169963	tt9913872	3.9	11	movie	De la piel del Diablo	De la piel del Diablo	0	2019	NA	
1169968	tt9914192	5.6	191	movie	No Gog do Paulinho	No Gog do Paulinho	0	2020	NA	
1170028	tt9916132	4.0	6	movie	The Mystery of a Buryat Lama	The Mystery of a Buryat Lama	0	2018	NA	
1170031	tt9916190	3.6	139	movie	Safeguard	Safeguard	0	2020	NA	
1170037	tt9916270	5.8	1085	movie	Il talento del calabrone	Il talento del calabrone	0	2020	NA	
1170043	tt9916428	3.8	12	movie	The Secret of China	Hong xing zhao yao Zhong guo	0	2019	NA	

In [51]:
```
find_movie_with_at_most_rating <- function(titles_df, ratings_df,
                                           ratings_of_at_most) {
    results <- merge(ratings_df, titles_df,
                     by.x = "title_id", by.y = "title_id")
    popular_movie_results <- results[results$type == "movie"
                                     & results$rating < ratings_of_at_most, ]
    return(popular_movie_results)
}
```

In [52]:
```
my_consent <- find_movie_with_at_most_rating(titles, ratings, 6.0)
```

In [53]:
```
dim(my_consent)
```

106471 · 11

In [54]:
```
head(my_consent)
```

A data.frame: 6 x 11

	title_id	rating	votes	type	primary_title	original_title	is_adult	premiered	ended	runtim
	<chr>	<dbl>	<int>	<chr>	<chr>	<chr>	<int>	<int>	<int>	
341	tt0000502	4.5	8	movie	Bohemios	Bohemios	0	1905	NA	
384	tt0000591	5.2	6	movie	The Prodigal Son	L'enfant prodigue	0	1907	NA	
399	tt0000615	4.5	14	movie	Robbery Under Arms	Robbery Under Arms	0	1907	NA	
407	tt0000630	3.8	12	movie	Hamlet	Amleto	0	1908	NA	
437	tt0000675	4.9	9	movie	Don Quijote	Don Quijote	0	1908	NA	
440	tt0000679	5.2	35	movie	The Fairylogue and Radio-Plays	The Fairylogue and Radio-Plays	0	1908	NA	

In [55]:
```
tail(my_consent)
```

A data.frame: 6 x 11

	title_id	rating	votes	type	primary_title	original_title	is_adult	premiered	ended	ru
	<chr>	<dbl>	<int>	<chr>	<chr>	<chr>	<int>	<int>	<int>	
1169963	tt9913872	3.9	11	movie	De la piel del Diablo	De la piel del Diablo	0	2019	NA	
1169968	tt9914192	5.6	191	movie	No Gog do Paulinho	No Gog do Paulinho	0	2020	NA	
1170028	tt9916132	4.0	6	movie	The Mystery of a Buryat Lama	The Mystery of a Buryat Lama	0	2018	NA	
1170031	tt9916190	3.6	139	movie	Safeguard	Safeguard	0	2020	NA	
1170037	tt9916270	5.8	1085	movie	Il talento del calabrone	Il talento del calabrone	0	2020	NA	
1170043	tt9916428	3.8	12	movie	The Secret of China	Hong xing zhao yao Zhong guo	0	2019	NA	

How many movies in total are there, which are above that limit? 34340.11

Change the limits in the function from "at least 6.0" to "lower than 6.0". 114707.11

How many movies have ratings lower than 6.0? 106471.11

Question 5

In [56]:
```
my_most_popular_movie <-
# send the titles data frame, ratings data frame, and the name of the genres
#  merge the data frames, just like we did above
#  make a similar data frame, only contain movies
# which are only from the genre you picked
# make an even smaller data frame, with rows that
# have the property myDF$rating = max(myDF$rating)
# return that even smaller data frame
```

```
Error in parse(text = x, srcfile = src): <text>:6:0: unexpected end of input
4:                          # make an even smaller data frame, with rows that have the
property myDF$rating = max(myDF$rating)
5:                          # return that even smaller data frame
  ^
Traceback:
```

In [57]:
```
my_best_movie <- function(titles_df, ratings_df, genre) {
    results <- merge(ratings_df, titles_df,
                     by.x = "title_id", by.y = "title_id")
    my_genres_output <- results[results$type == "movie" &
                                results$genres == genre, ]
    famous <-  my_genres_output[ my_genres_output$rating==
                                max( my_genres_output$rating), ]
     return(famous)
}
```

In [58]:
```
results <- merge(ratings, titles, by.x = "title_id", by.y = "title_id")
```

In [59]:
```
dim(results)
```

1170056 · 11

In [60]:
```
genre<- "Thriller"
```

In [61]:
```
my_genres_output <- results[results$type == "movie"
                            & results$genres == genre, ]
```

In [62]:
```
famous <-  my_genres_output[ my_genres_output$rating==
                            max( my_genres_output$rating), ]
```

In [63]:
```
thrillerDF<-my_best_movie(titles,ratings,"Thriller")
```

In [64]:
```
dim(thrillerDF)
```

1 · 11

In [65]:
```
thrillerDF
```

A data.frame: 1 x 11

	title_id	rating	votes	type	primary_title	original_title	is_adult	premiered	ended	ru
	<chr>	<dbl>	<int>	<chr>	<chr>	<chr>	<int>	<int>	<int>	
616603	tt13992188	9.4	31	movie	House of Cards - Rust	House of Cards - Rust	0	2021	NA	

In [66]:
```
dim(my_genres_output)
```

3772 · 11

In [67]:
```
dim(famous)
```

1 · 11

In [68]:
```
thrillerDF<-my_best_movie(titles,ratings,"Romance")
```

In [69]:
```
dim(thrillerDF)
```

1 · 11

In [70]:
```
thrillerDF
```

A data.frame: 1 x 11

	title_id	rating	votes	type	primary_title	original_title	is_adult	premiered	ended	ru
	<chr>	<dbl>	<int>	<chr>	<chr>	<chr>	<int>	<int>	<int>	
135262	tt0211363	9.4	5	movie	The Feather	The Feather	0	1929	NA	

In [71]:
```
thrillerDF<-my_best_movie(titles,ratings,"Action")
```

In [72]:
```
dim(thrillerDF)
```

2 · 11

In [73]:
```
thrillerDF
```

A data.frame: 2 x 11

	title_id	rating	votes	type	primary_title	original_title	is_adult	premiered	ended	r
	<chr>	<dbl>	<int>	<chr>	<chr>	<chr>	<int>	<int>	<int>	
487348	tt10915200	9.4	7	movie	Notcias do fim do mundo	Notcias do fim do mundo	0	2019	NA	
1158637	tt9526156	9.4	8	movie	College Diary	College Diary	0	2019	NA	

In [74]:
```
head(results)
```

A data.frame: 6 x 11

	title_id	rating	votes	type	primary_title	original_title	is_adult	premiered	ended	runtime_
	<chr>	<dbl>	<int>	<chr>	<chr>	<chr>	<int>	<int>	<int>	
1	tt0000001	5.7	1718	short	Carmencita	Carmencita	0	1894	NA	
2	tt0000002	6.0	211	short	Le clown et ses chiens	Le clown et ses chiens	0	1892	NA	
3	tt0000003	6.5	1480	short	Pauvre Pierrot	Pauvre Pierrot	0	1892	NA	
4	tt0000004	6.1	124	short	Un bon bock	Un bon bock	0	1892	NA	
5	tt0000005	6.2	2283	short	Blacksmith Scene	Blacksmith Scene	0	1893	NA	
6	tt0000006	5.1	127	short	Chinese Opium Den	Chinese Opium Den	0	1894	NA	

In [75]:
```
my_best_movie_by_vote <- function(titles_df, ratings_df, genre) {
    results <- merge(ratings_df, titles_df,
                     by.x = "title_id", by.y = "title_id")
    my_genres_output <- results[results$type == "movie" &
                                results$genres == genre, ]
# to look for best movie by vote, you will replace votes for rating
    famous <-  my_genres_output[ my_genres_output$votes==
                                max( my_genres_output$votes), ]
     return(famous)
}
```

In [76]:
```
thrillerDF<-my_best_movie_by_vote(titles,ratings,"Action")
```

In [77]:
```
thrillerDF
```

A data.frame: 1 x 11

	title_id	rating	votes	type	primary_title	original_title	is_adult	premiered	ended	ru
	<chr>	<dbl>	<int>	<chr>	<chr>	<chr>	<int>	<int>	<int>	
562664	tt1258972	5.4	61150	movie	The Man with the Iron Fists	The Man with the Iron Fists	0	2012	NA	

In [78]:
```
thrillerDF<-my_best_movie_by_vote(titles,ratings,"Romance")
```

In [79]:
```
thrillerDF
```

A data.frame: 1 x 11

	title_id	rating	votes	type	primary_title	original_title	is_adult	premiered	ended	ru
	<chr>	<dbl>	<int>	<chr>	<chr>	<chr>	<int>	<int>	<int>	
998547	tt5934922	6.1	6477	movie	Ikimizin Yerine	Ikimizin Yerine	0	2016	NA	

the movie from that genre that has the largest number of votes, OR

the movie from that genre that has the highest rating.

Pledge

By submitting this work I hereby pledge that this is my own, personal work. I've acknowledged in the designated place at the top of this file all sources that I used to complete said work, including but not limited to: online resources, books, and electronic communications. I've noted all collaboration with fellow students and/or TA's. I did not copy or plagiarize another's work.

> As a Boilermaker pursuing academic excellence, I pledge to be honest and true in all that I do. Accountable together – We are Purdue.