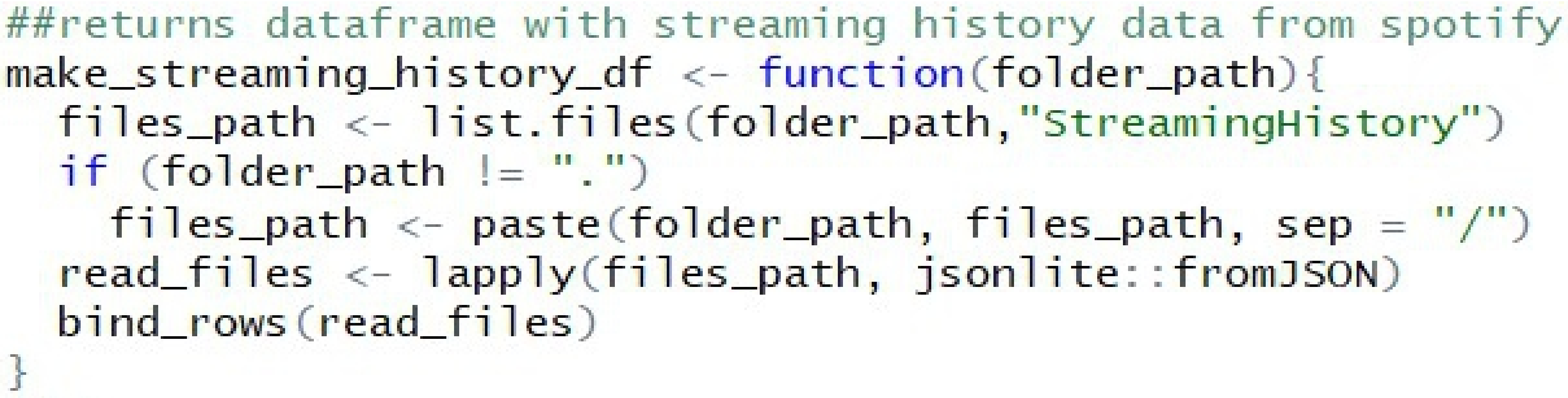
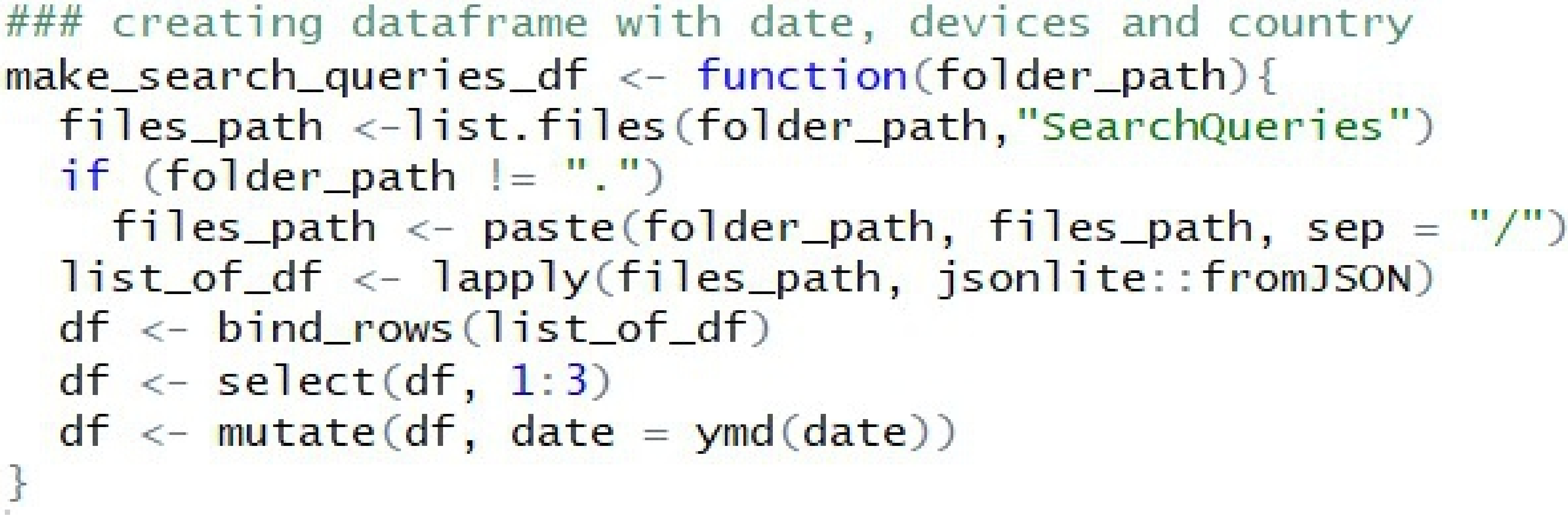
SpotifyVis

Magdalena Buszka
Wiktor Jacaszek
Natalia Jeszka

Wczytanie danych

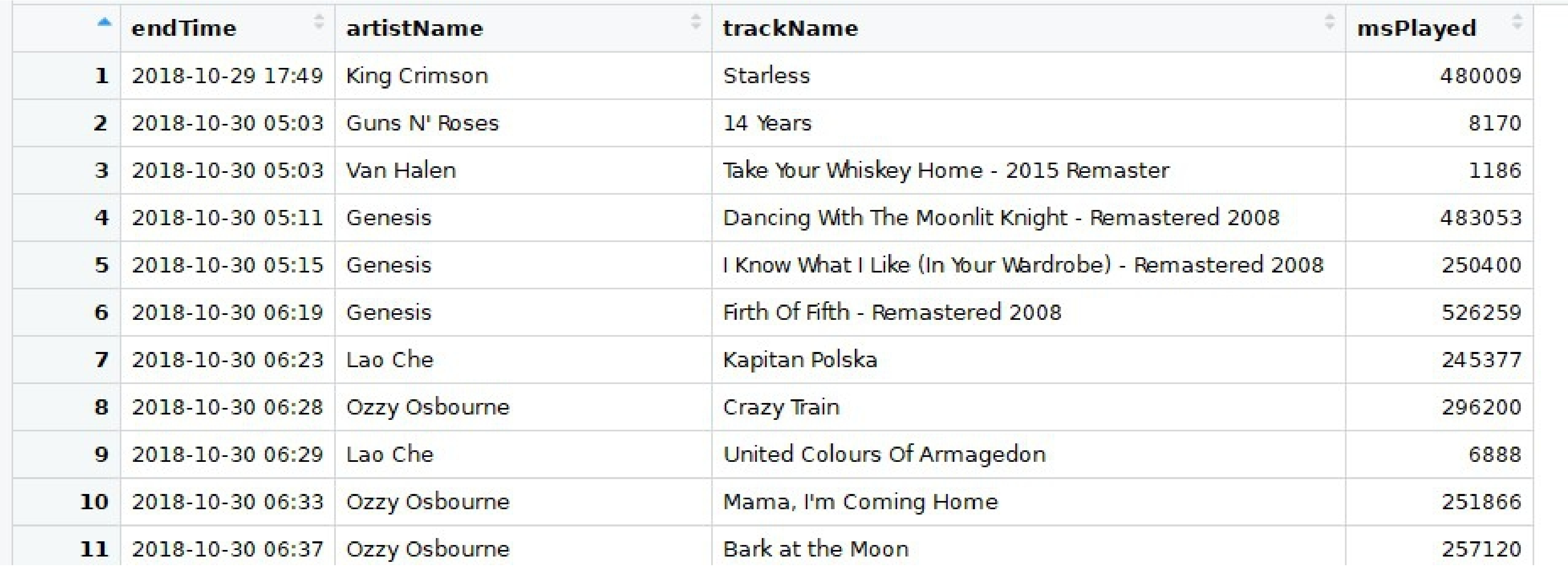


```r
##returns dataframe with streaming history data from spotify
make_streaming_history_df <- function(folder_path){
  files_path <- list.files(folder_path,"StreamingHistory")
  if (folder_path != ".")
    files_path <- paste(folder_path, files_path, sep = "/")
  read_files <- lapply(files_path, jsonlite::fromJSON)
  bind_rows(read_files)
}
```



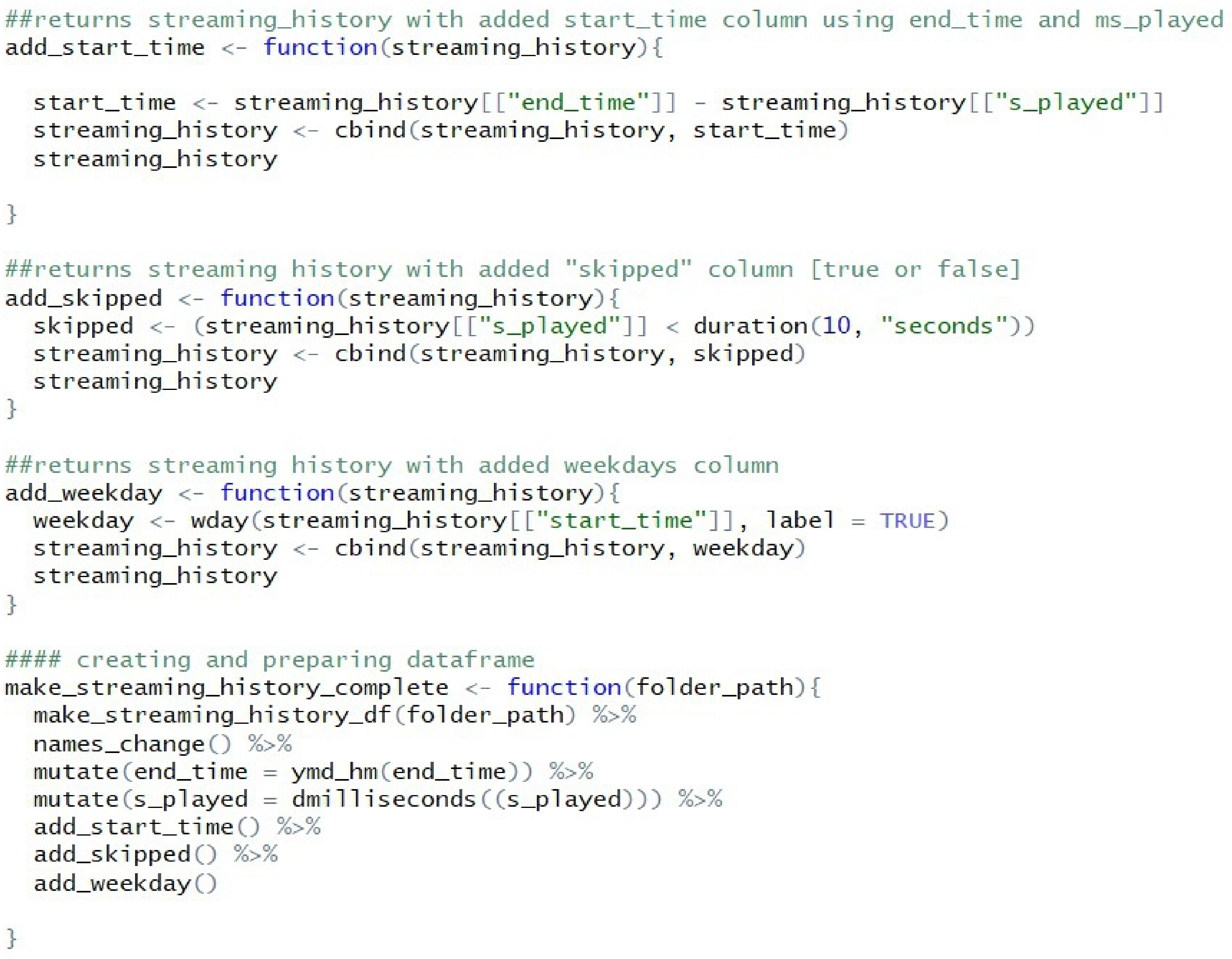
```r
### creating dataframe with date, devices and country
make_search_queries_df <- function(folder_path){
  files_path <-list.files(folder_path,"SearchQueries")
  if (folder_path != ".")
    files_path <- paste(folder_path, files_path, sep = "/")
  list_of_df <- lapply(files_path, jsonlite::fromJSON)
  df <- bind_rows(list_of_df)
  df <- select(df, 1:3)
  df <- mutate(df, date = ymd(date))
}
```

Ramka danych przed modyfikacją



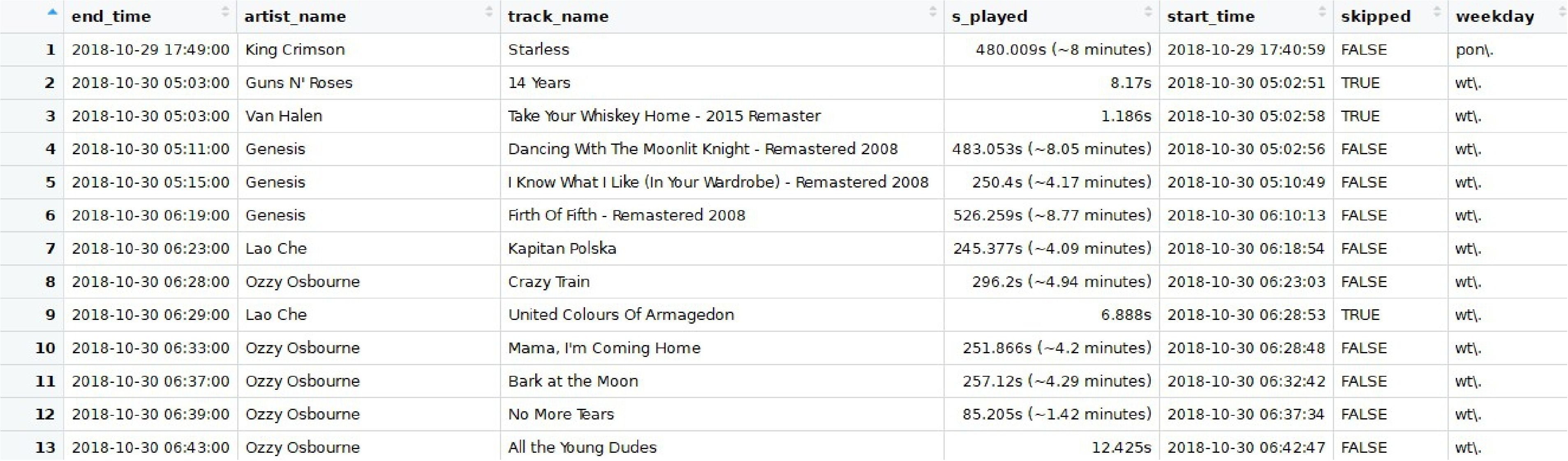
	endTime	artistName	trackName	msPlayed
1	2018-10-29 17:49	King Crimson	Starless	480009
2	2018-10-30 05:03	Guns N' Roses	14 Years	8170
3	2018-10-30 05:03	Van Halen	Take Your Whiskey Home - 2015 Remaster	1186
4	2018-10-30 05:11	Genesis	Dancing With The Moonlit Knight - Remastered 2008	483053
5	2018-10-30 05:15	Genesis	I Know What I Like (In Your Wardrobe) - Remastered 2008	250400
6	2018-10-30 06:19	Genesis	Firth Of Fifth - Remastered 2008	526259
7	2018-10-30 06:23	Lao Che	Kapitan Polska	245377
8	2018-10-30 06:28	Ozzy Osbourne	Crazy Train	296200
9	2018-10-30 06:29	Lao Che	United Colours Of Armagedon	6888
10	2018-10-30 06:33	Ozzy Osbourne	Mama, I'm Coming Home	251866
11	2018-10-30 06:37	Ozzy Osbourne	Bark at the Moon	257120

Funkcje modyfikujące ramkę danych

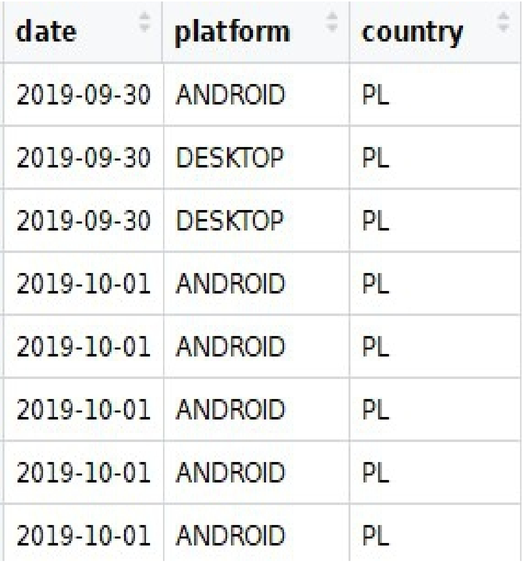


```r
##returns streaming_history with added start_time column using end_time and ms_played
add_start_time <- function(streaming_history){

  start_time <- streaming_history[["end_time"]] - streaming_history[["s_played"]]
  streaming_history <- cbind(streaming_history, start_time)
  streaming_history

}

##returns streaming history with added "skipped" column [true or false]
add_skipped <- function(streaming_history){
  skipped <- (streaming_history[["s_played"]] < duration(10, "seconds"))
  streaming_history <- cbind(streaming_history, skipped)
  streaming_history
}

##returns streaming history with added weekdays column
add_weekday <- function(streaming_history){
  weekday <- wday(streaming_history[["start_time"]], label = TRUE)
  streaming_history <- cbind(streaming_history, weekday)
  streaming_history
}

#### creating and preparing dataframe
make_streaming_history_complete <- function(folder_path){
  make_streaming_history_df(folder_path) %>%
  names_change() %>%
  mutate(end_time = ymd_hm(end_time)) %>%
  mutate(s_played = dmilliseconds((s_played))) %>%
  add_start_time() %>%
  add_skipped() %>%
  add_weekday()

}
```

Ramki danych po modyfikacji

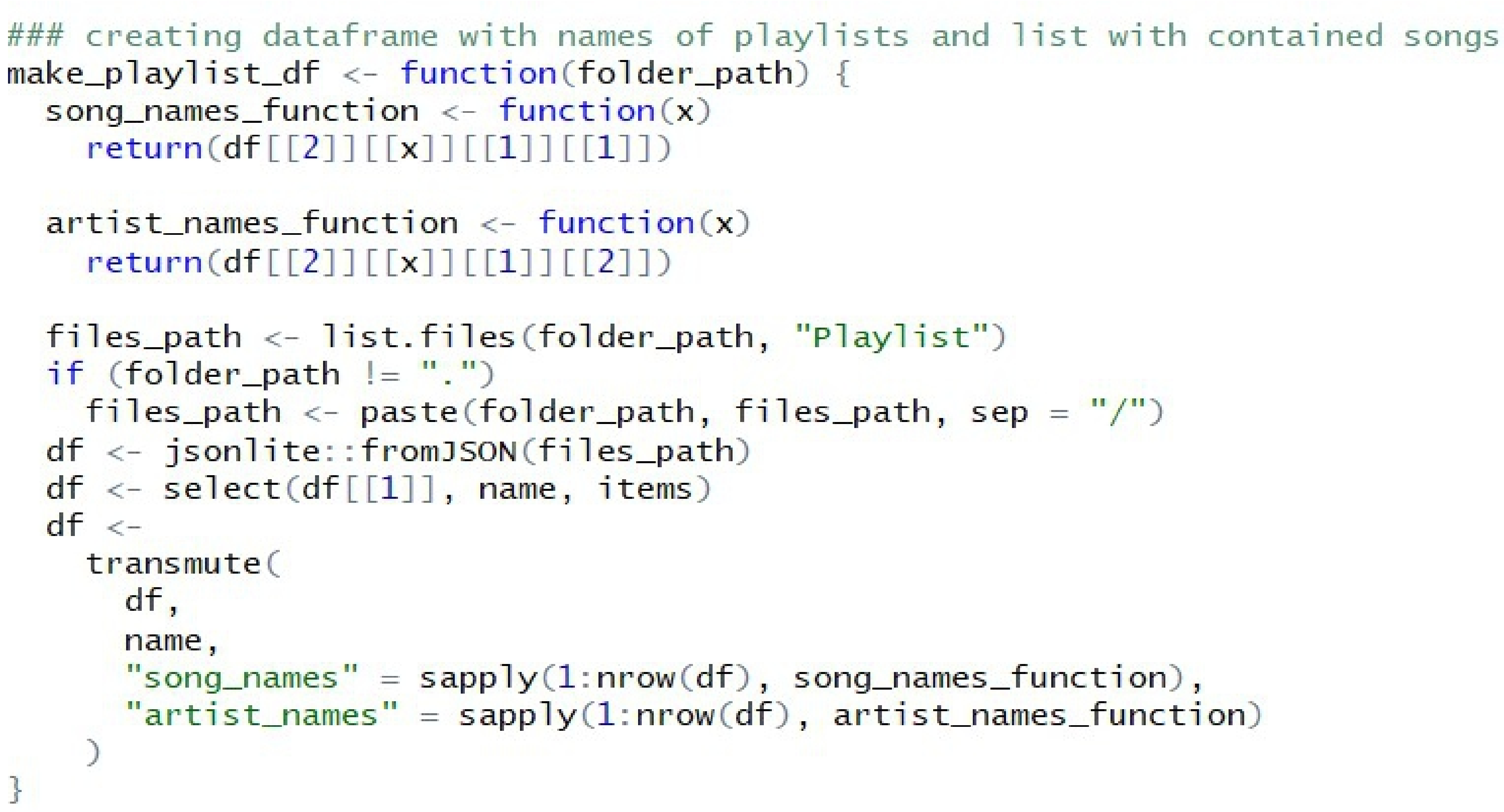


	end_time	artist_name	track_name	s_played	start_time	skipped	weekday
1	2018-10-29 17:49:00	King Crimson	Starless	480.009s (~8 minutes)	2018-10-29 17:40:59	FALSE	pon\.
2	2018-10-30 05:03:00	Guns N' Roses	14 Years	8.17s	2018-10-30 05:02:51	TRUE	wt\.
3	2018-10-30 05:03:00	Van Halen	Take Your Whiskey Home - 2015 Remaster	1.186s	2018-10-30 05:02:58	TRUE	wt\.
4	2018-10-30 05:11:00	Genesis	Dancing With The Moonlit Knight - Remastered 2008	483.053s (~8.05 minutes)	2018-10-30 05:02:56	FALSE	wt\.
5	2018-10-30 05:15:00	Genesis	I Know What I Like (In Your Wardrobe) - Remastered 2008	250.4s (~4.17 minutes)	2018-10-30 05:10:49	FALSE	wt\.
6	2018-10-30 06:19:00	Genesis	Firth Of Fifth - Remastered 2008	526.259s (~8.77 minutes)	2018-10-30 06:10:13	FALSE	wt\.
7	2018-10-30 06:23:00	Lao Che	Kapitan Polska	245.377s (~4.09 minutes)	2018-10-30 06:18:54	FALSE	wt\.
8	2018-10-30 06:28:00	Ozzy Osbourne	Crazy Train	296.2s (~4.94 minutes)	2018-10-30 06:23:03	FALSE	wt\.
9	2018-10-30 06:29:00	Lao Che	United Colours Of Armagedon	6.888s	2018-10-30 06:28:53	TRUE	wt\.
10	2018-10-30 06:33:00	Ozzy Osbourne	Mama, I'm Coming Home	251.866s (~4.2 minutes)	2018-10-30 06:28:48	FALSE	wt\.
11	2018-10-30 06:37:00	Ozzy Osbourne	Bark at the Moon	257.12s (~4.29 minutes)	2018-10-30 06:32:42	FALSE	wt\.
12	2018-10-30 06:39:00	Ozzy Osbourne	No More Tears	85.205s (~1.42 minutes)	2018-10-30 06:37:34	FALSE	wt\.
13	2018-10-30 06:43:00	Ozzy Osbourne	All the Young Dudes	12.425s	2018-10-30 06:42:47	FALSE	wt\.



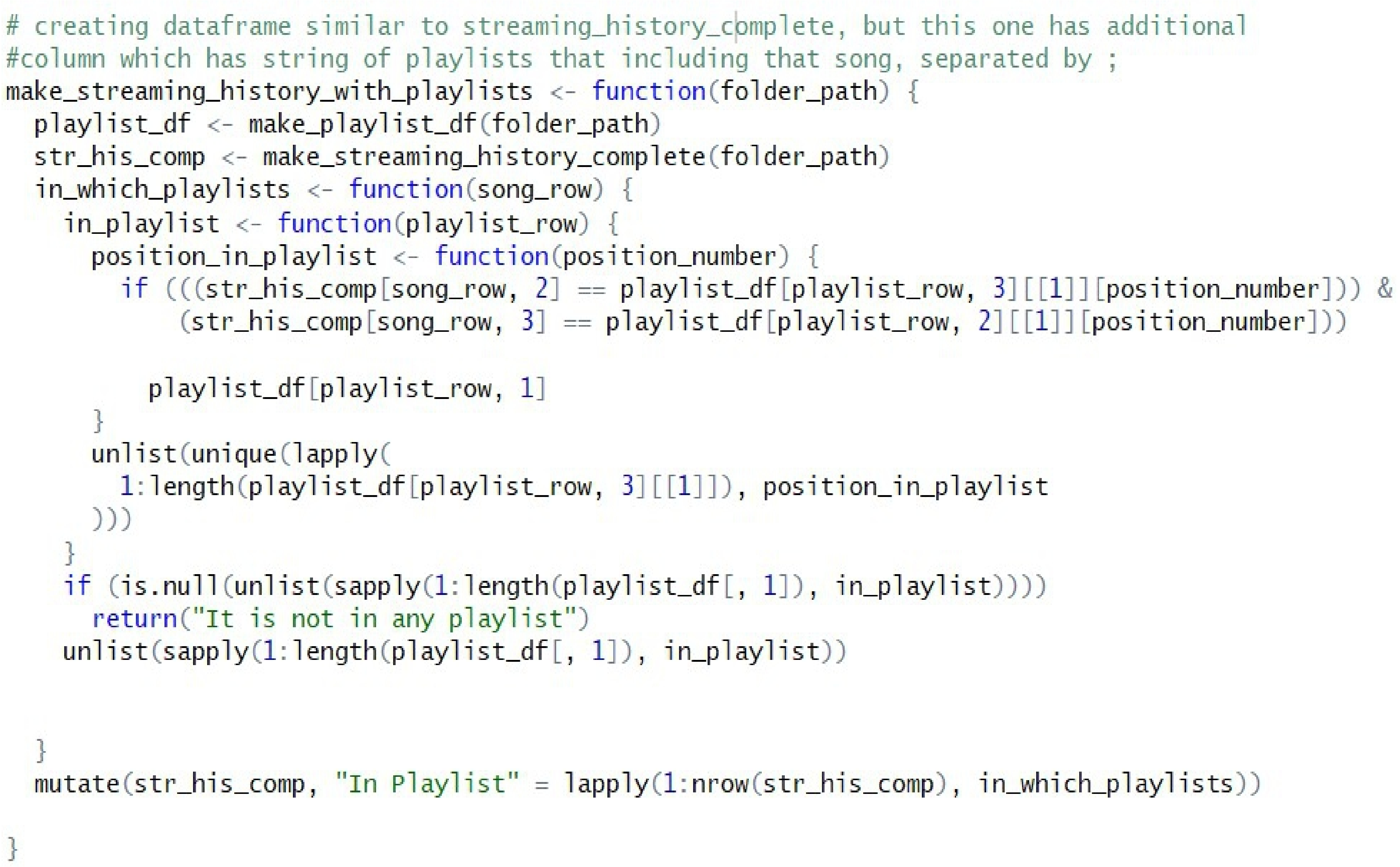
date	platform	country
2019-09-30	ANDROID	PL
2019-09-30	DESKTOP	PL
2019-09-30	DESKTOP	PL
2019-10-01	ANDROID	PL
2019-10-01	ANDROID	PL
2019-10-01	ANDROID	PL
2019-10-01	ANDROID	PL
2019-10-01	ANDROID	PL

Tworzenie ramki danych z playlistami



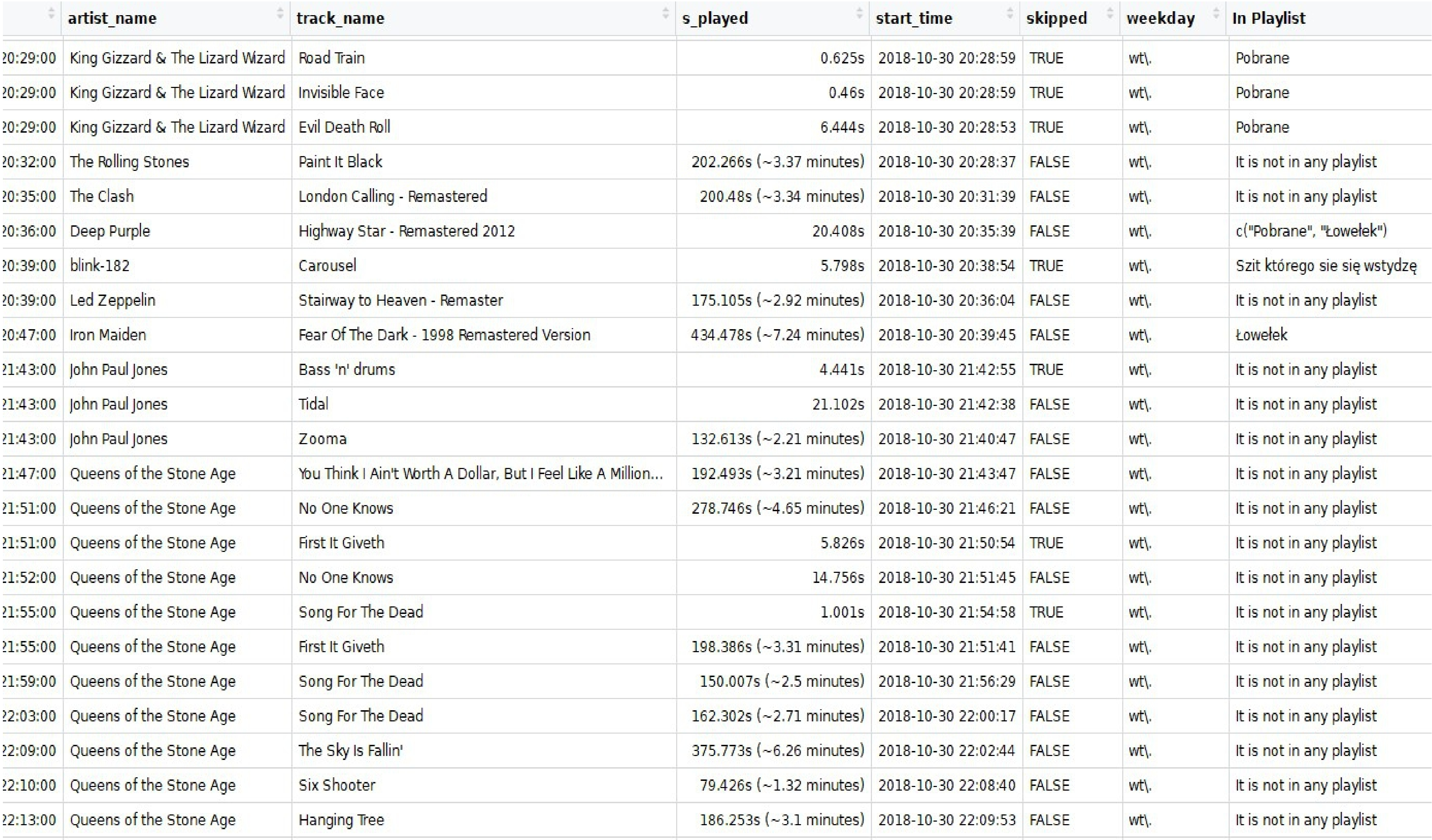
```
### creating dataframe with names of playlists and list with contained songs
make_playlist_df <- function(folder_path) {
  song_names_function <- function(x)
    return(df[[2]][[x]][[1]][[1]])

  artist_names_function <- function(x)
    return(df[[2]][[x]][[1]][[2]])

  files_path <- list.files(folder_path, "Playlist")
  if (folder_path != ".")
    files_path <- paste(folder_path, files_path, sep = "/")
  df <- jsonlite::fromJSON(files_path)
  df <- select(df[[1]], name, items)
  df <-
    transmute(
      df,
      name,
      "song_names" = sapply(1:nrow(df), song_names_function),
      "artist_names" = sapply(1:nrow(df), artist_names_function)
    )
}
```

Dodawanie informacji o playlistach



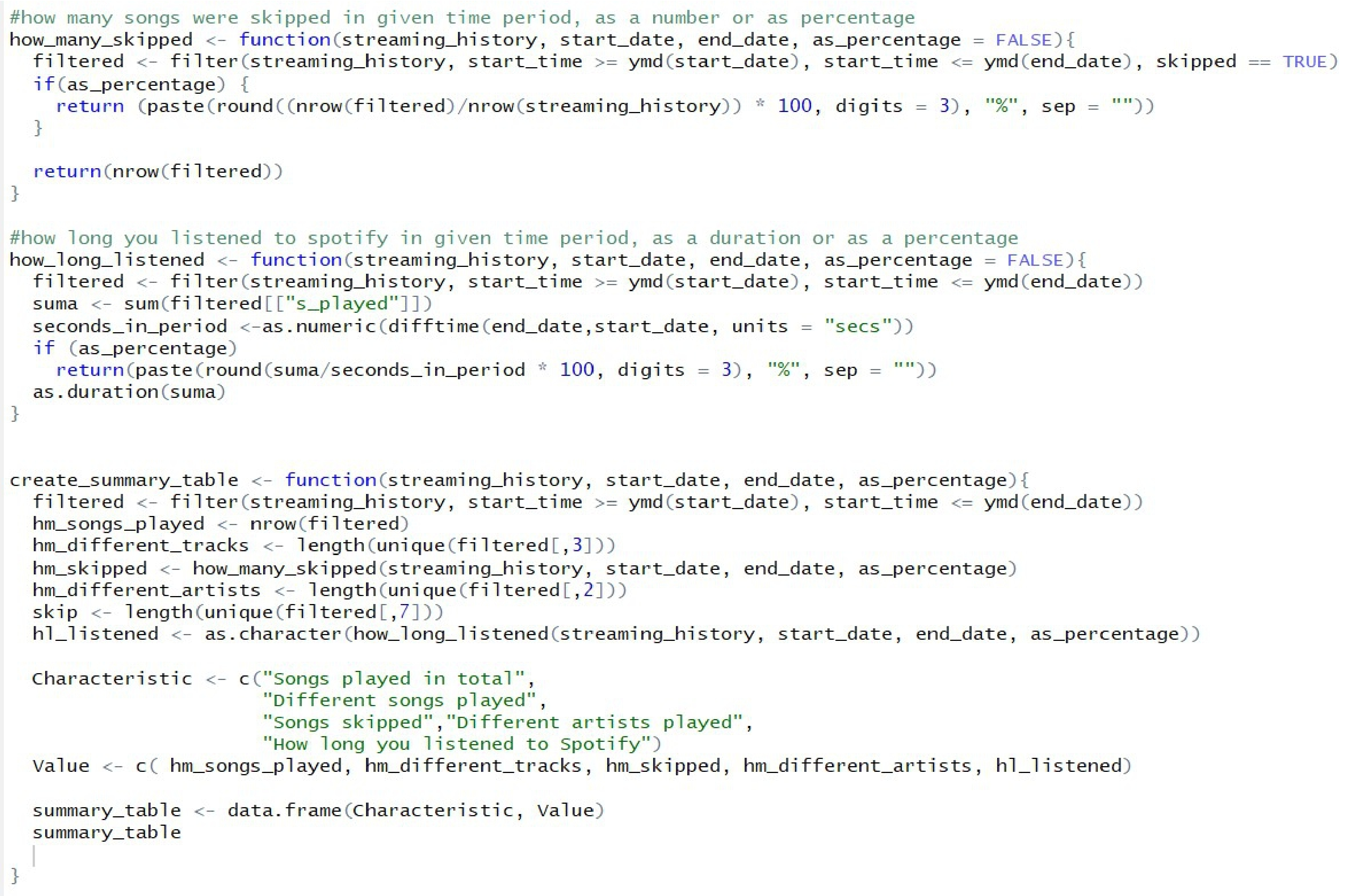
```r
# creating dataframe similar to streaming_history_complete, but this one has additional
#column which has string of playlists that including that song, separated by ;
make_streaming_history_with_playlists <- function(folder_path) {
  playlist_df <- make_playlist_df(folder_path)
  str_his_comp <- make_streaming_history_complete(folder_path)
  in_which_playlists <- function(song_row) {
    in_playlist <- function(playlist_row) {
      position_in_playlist <- function(position_number) {
        if (((str_his_comp[song_row, 2] == playlist_df[playlist_row, 3][[1]][position_number])) &
            (str_his_comp[song_row, 3] == playlist_df[playlist_row, 2][[1]][position_number]))

          playlist_df[playlist_row, 1]
      }
      unlist(unique(lapply(
        1:length(playlist_df[playlist_row, 3][[1]]), position_in_playlist
      )))
    }
    if (is.null(unlist(sapply(1:length(playlist_df[, 1]), in_playlist))))
      return("It is not in any playlist")
    unlist(sapply(1:length(playlist_df[, 1]), in_playlist))


  }
  mutate(str_his_comp, "In Playlist" = lapply(1:nrow(str_his_comp), in_which_playlists))

}
```

Ramka danych z playlistami



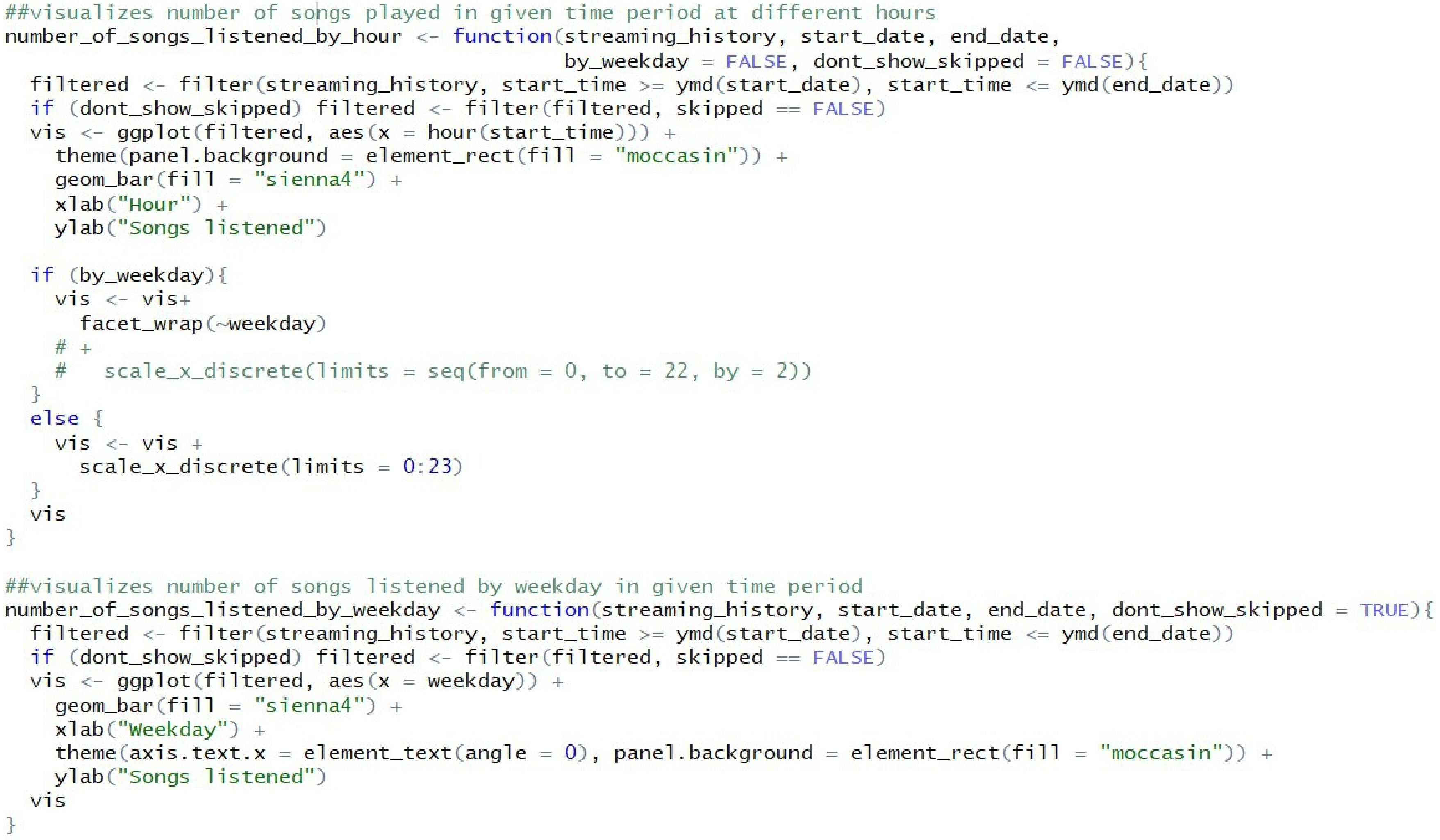
	artist_name	track_name	s_played	start_time	skipped	weekday	In Playlist
20:29:00	King Gizzard & The Lizard Wizard	Road Train	0.625s	2018-10-30 20:28:59	TRUE	wt\.	Pobrane
20:29:00	King Gizzard & The Lizard Wizard	Invisible Face	0.46s	2018-10-30 20:28:59	TRUE	wt\.	Pobrane
20:29:00	King Gizzard & The Lizard Wizard	Evil Death Roll	6.444s	2018-10-30 20:28:53	TRUE	wt\.	Pobrane
20:32:00	The Rolling Stones	Paint It Black	202.266s (~3.37 minutes)	2018-10-30 20:28:37	FALSE	wt\.	It is not in any playlist
20:35:00	The Clash	London Calling - Remastered	200.48s (~3.34 minutes)	2018-10-30 20:31:39	FALSE	wt\.	It is not in any playlist
20:36:00	Deep Purple	Highway Star - Remastered 2012	20.408s	2018-10-30 20:35:39	FALSE	wt\.	c("Pobrane", "Łowełek")
20:39:00	blink-182	Carousel	5.798s	2018-10-30 20:38:54	TRUE	wt\.	Szit którego sie się wstydzę
20:39:00	Led Zeppelin	Stairway to Heaven - Remaster	175.105s (~2.92 minutes)	2018-10-30 20:36:04	FALSE	wt\.	It is not in any playlist
20:47:00	Iron Maiden	Fear Of The Dark - 1998 Remastered Version	434.478s (~7.24 minutes)	2018-10-30 20:39:45	FALSE	wt\.	Łowełek
21:43:00	John Paul Jones	Bass 'n' drums	4.441s	2018-10-30 21:42:55	TRUE	wt\.	It is not in any playlist
21:43:00	John Paul Jones	Tidal	21.102s	2018-10-30 21:42:38	FALSE	wt\.	It is not in any playlist
21:43:00	John Paul Jones	Zooma	132.613s (~2.21 minutes)	2018-10-30 21:40:47	FALSE	wt\.	It is not in any playlist
21:47:00	Queens of the Stone Age	You Think I Ain't Worth A Dollar, But I Feel Like A Million...	192.493s (~3.21 minutes)	2018-10-30 21:43:47	FALSE	wt\.	It is not in any playlist
21:51:00	Queens of the Stone Age	No One Knows	278.746s (~4.65 minutes)	2018-10-30 21:46:21	FALSE	wt\.	It is not in any playlist
21:51:00	Queens of the Stone Age	First It Giveth	5.826s	2018-10-30 21:50:54	TRUE	wt\.	It is not in any playlist
21:52:00	Queens of the Stone Age	No One Knows	14.756s	2018-10-30 21:51:45	FALSE	wt\.	It is not in any playlist
21:55:00	Queens of the Stone Age	Song For The Dead	1.001s	2018-10-30 21:54:58	TRUE	wt\.	It is not in any playlist
21:55:00	Queens of the Stone Age	First It Giveth	198.386s (~3.31 minutes)	2018-10-30 21:51:41	FALSE	wt\.	It is not in any playlist
21:59:00	Queens of the Stone Age	Song For The Dead	150.007s (~2.5 minutes)	2018-10-30 21:56:29	FALSE	wt\.	It is not in any playlist
22:03:00	Queens of the Stone Age	Song For The Dead	162.302s (~2.71 minutes)	2018-10-30 22:00:17	FALSE	wt\.	It is not in any playlist
22:09:00	Queens of the Stone Age	The Sky Is Fallin'	375.773s (~6.26 minutes)	2018-10-30 22:02:44	FALSE	wt\.	It is not in any playlist
22:10:00	Queens of the Stone Age	Six Shooter	79.426s (~1.32 minutes)	2018-10-30 22:08:40	FALSE	wt\.	It is not in any playlist
22:13:00	Queens of the Stone Age	Hanging Tree	186.253s (~3.1 minutes)	2018-10-30 22:09:53	FALSE	wt\.	It is not in any playlist

Funkcje do stworzenia tabelki z podsumowaniem



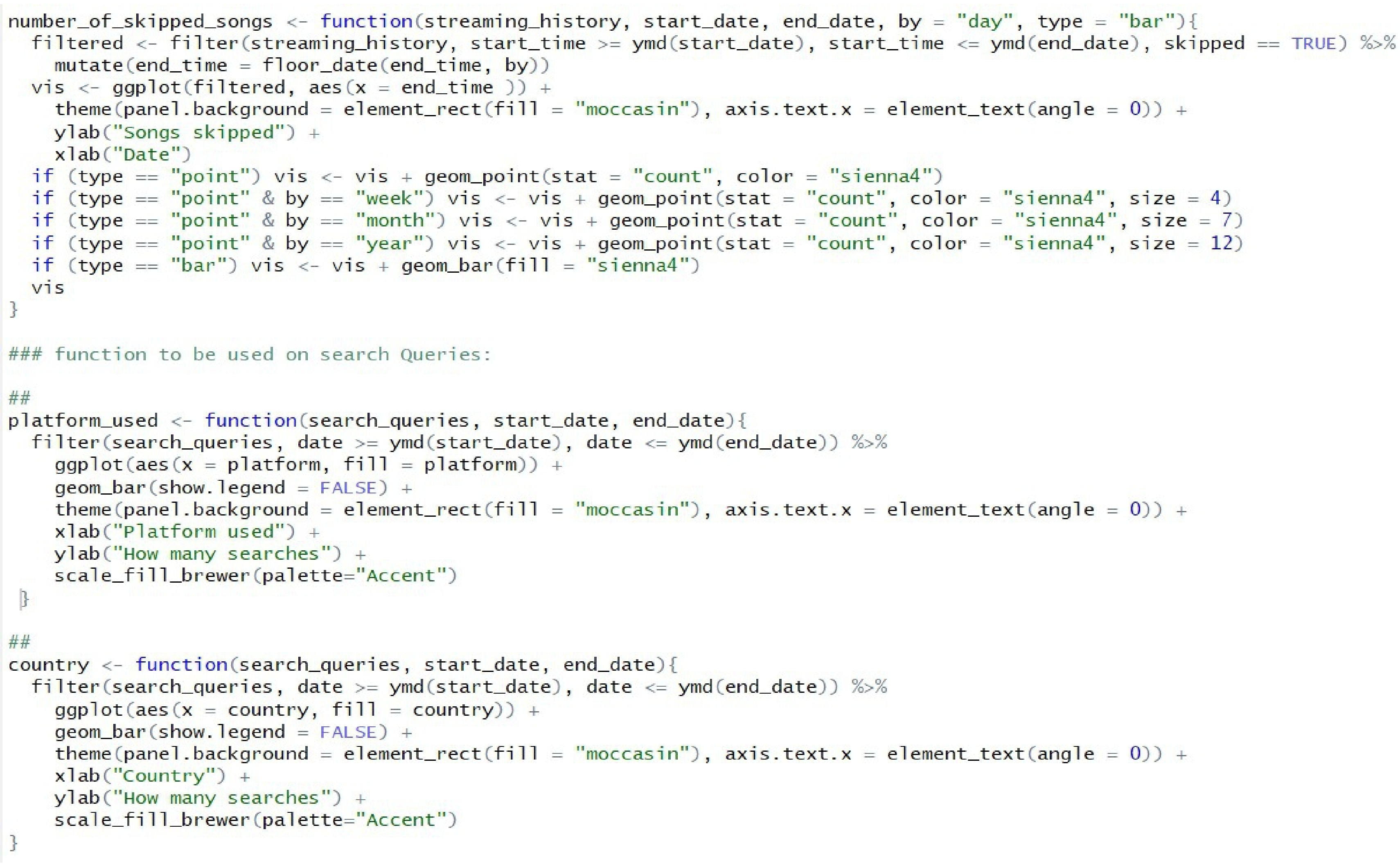
```r
#how many songs were skipped in given time period, as a number or as percentage
how_many_skipped <- function(streaming_history, start_date, end_date, as_percentage = FALSE){
  filtered <- filter(streaming_history, start_time >= ymd(start_date), start_time <= ymd(end_date), skipped == TRUE)
  if(as_percentage) {
    return (paste(round((nrow(filtered)/nrow(streaming_history)) * 100, digits = 3), "%", sep = ""))
  }

  return(nrow(filtered))
}

#how long you listened to spotify in given time period, as a duration or as a percentage
how_long_listened <- function(streaming_history, start_date, end_date, as_percentage = FALSE){
  filtered <- filter(streaming_history, start_time >= ymd(start_date), start_time <= ymd(end_date))
  suma <- sum(filtered[["s_played"]])
  seconds_in_period <-as.numeric(difftime(end_date,start_date, units = "secs"))
  if (as_percentage)
    return(paste(round(suma/seconds_in_period * 100, digits = 3), "%", sep = ""))
  as.duration(suma)
}


create_summary_table <- function(streaming_history, start_date, end_date, as_percentage){
  filtered <- filter(streaming_history, start_time >= ymd(start_date), start_time <= ymd(end_date))
  hm_songs_played <- nrow(filtered)
  hm_different_tracks <- length(unique(filtered[,3]))
  hm_skipped <- how_many_skipped(streaming_history, start_date, end_date, as_percentage)
  hm_different_artists <- length(unique(filtered[,2]))
  skip <- length(unique(filtered[,7]))
  hl_listened <- as.character(how_long_listened(streaming_history, start_date, end_date, as_percentage))

  Characteristic <- c("Songs played in total",
                      "Different songs played",
                      "Songs skipped","Different artists played",
                      "How long you listened to Spotify")
  Value <- c( hm_songs_played, hm_different_tracks, hm_skipped, hm_different_artists, hl_listened)

  summary_table <- data.frame(Characteristic, Value)
  summary_table

}
```

Wizualizacja ilości słuchanych utworów



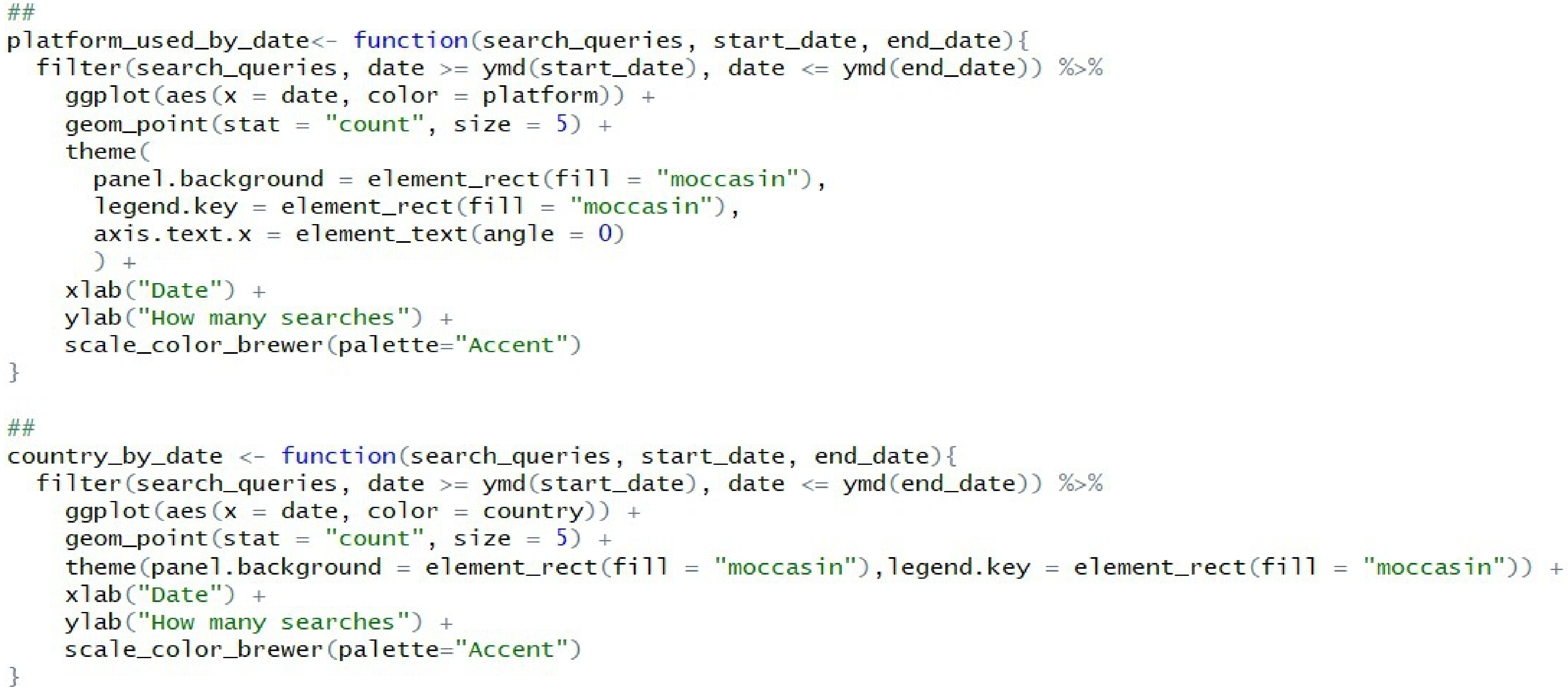
```
##visualizes number of songs played in given time period at different hours
number_of_songs_listened_by_hour <- function(streaming_history, start_date, end_date,
                                             by_weekday = FALSE, dont_show_skipped = FALSE){
  filtered <- filter(streaming_history, start_time >= ymd(start_date), start_time <= ymd(end_date))
  if (dont_show_skipped) filtered <- filter(filtered, skipped == FALSE)
  vis <- ggplot(filtered, aes(x = hour(start_time))) +
    theme(panel.background = element_rect(fill = "moccasin")) +
    geom_bar(fill = "sienna4") +
    xlab("Hour") +
    ylab("Songs listened")

  if (by_weekday){
    vis <- vis+
      facet_wrap(~weekday)
    # +
    #   scale_x_discrete(limits = seq(from = 0, to = 22, by = 2))
  }
  else {
    vis <- vis +
      scale_x_discrete(limits = 0:23)
  }
  vis
}

##visualizes number of songs listened by weekday in given time period
number_of_songs_listened_by_weekday <- function(streaming_history, start_date, end_date, dont_show_skipped = TRUE){
  filtered <- filter(streaming_history, start_time >= ymd(start_date), start_time <= ymd(end_date))
  if (dont_show_skipped) filtered <- filter(filtered, skipped == FALSE)
  vis <- ggplot(filtered, aes(x = weekday)) +
    geom_bar(fill = "sienna4") +
    xlab("Weekday") +
    theme(axis.text.x = element_text(angle = 0), panel.background = element_rect(fill = "moccasin")) +
    ylab("Songs listened")
  vis
}
```

Wizualizacja ilości pomijanych utworów oraz ilości wyszukiwań z podziałem na kraj i rodzaj urządzenia



```
number_of_skipped_songs <- function(streaming_history, start_date, end_date, by = "day", type = "bar"){
  filtered <- filter(streaming_history, start_time >= ymd(start_date), start_time <= ymd(end_date), skipped == TRUE) %>%
    mutate(end_time = floor_date(end_time, by))
  vis <- ggplot(filtered, aes(x = end_time )) +
    theme(panel.background = element_rect(fill = "moccasin"), axis.text.x = element_text(angle = 0)) +
    ylab("Songs skipped") +
    xlab("Date")
  if (type == "point") vis <- vis + geom_point(stat = "count", color = "sienna4")
  if (type == "point" & by == "week") vis <- vis + geom_point(stat = "count", color = "sienna4", size = 4)
  if (type == "point" & by == "month") vis <- vis + geom_point(stat = "count", color = "sienna4", size = 7)
  if (type == "point" & by == "year") vis <- vis + geom_point(stat = "count", color = "sienna4", size = 12)
  if (type == "bar") vis <- vis + geom_bar(fill = "sienna4")
  vis
}

### function to be used on search Queries:

##
platform_used <- function(search_queries, start_date, end_date){
  filter(search_queries, date >= ymd(start_date), date <= ymd(end_date)) %>%
    ggplot(aes(x = platform, fill = platform)) +
    geom_bar(show.legend = FALSE) +
    theme(panel.background = element_rect(fill = "moccasin"), axis.text.x = element_text(angle = 0)) +
    xlab("Platform used") +
    ylab("How many searches") +
    scale_fill_brewer(palette="Accent")
 }

##
country <- function(search_queries, start_date, end_date){
  filter(search_queries, date >= ymd(start_date), date <= ymd(end_date)) %>%
    ggplot(aes(x = country, fill = country)) +
    geom_bar(show.legend = FALSE) +
    theme(panel.background = element_rect(fill = "moccasin"), axis.text.x = element_text(angle = 0)) +
    xlab("Country") +
    ylab("How many searches") +
    scale_fill_brewer(palette="Accent")
}
```

Wizualizacja ilości wyszukiwań w różnych krajach i na różnych urządzeniach w czasie



```
##
platform_used_by_date<- function(search_queries, start_date, end_date){
  filter(search_queries, date >= ymd(start_date), date <= ymd(end_date)) %>%
    ggplot(aes(x = date, color = platform)) +
    geom_point(stat = "count", size = 5) +
    theme(
      panel.background = element_rect(fill = "moccasin"),
      legend.key = element_rect(fill = "moccasin"),
      axis.text.x = element_text(angle = 0)
      ) +
    xlab("Date") +
    ylab("How many searches") +
    scale_color_brewer(palette="Accent")
}

##
country_by_date <- function(search_queries, start_date, end_date){
  filter(search_queries, date >= ymd(start_date), date <= ymd(end_date)) %>%
    ggplot(aes(x = date, color = country)) +
    geom_point(stat = "count", size = 5) +
    theme(panel.background = element_rect(fill = "moccasin"),legend.key = element_rect(fill = "moccasin")) +
    xlab("Date") +
    ylab("How many searches") +
    scale_color_brewer(palette="Accent")
}
```