

# Sample on Movie Ratings

John Cruz

2023-02-02

## Introduction

I will be collecting movie ratings through an online survey on a scale of 1 to 5.

The overall steps of this project is the following:

1. Collect survey data
  2. Create a database to store this information
  3. Ability to query the data into R
  4. Analyze some descriptive measures.
- 

## Collecting Survey Data

The survey was conducted using a Google Forms named *Spoiled Corn*. This survey will collect ratings of six recent popular movies on a scale of 1 to 5, 1 being the lowest. If a person did not watch this movie, it will be categorized with an option of N/A. Once data has been collected, I exported the data via CSV, so that it can be imported into our PostgreSQL database.

[Online Survey](#)

---

## Building the Database

The database used was PostgreSQL for no particular reason but to learn how to use another database option outside of my previous ones such as SQL Server, duckDB and Google Cloud Storage.

Using PostgreSQL within pgAdmin 4, create a database named *movie\_survey*.

pic

Within the *movie\_survey* database, run the following code: [create\\_movie\\_survey\\_tables.sql](#)

This code is broken up into a few steps.

---

# Connecting to our Database

## Required Libraries

```
library('RPostgres')
library('DBI')
library('tidyverse')
```

## PostgreSQL Database

Using this connection we will be able to keep our username and password from being embedded into our code and instead have RStudio pop up a dialog box to input the information.

```
cnxn <- dbConnect(RPostgres::Postgres(),
  dbname = 'movie_survey',
  port = 5432,
  user = 'postgres',
  password = 'elliebelly968')
# user = rstudioapi::askForPassword("Database Username"),
# password = rstudioapi::askForPassword("Database Password"))
```

## Query the Database

Here we are able to directly query our PostgreSQL database and pull the information from the tables we created earlier. *dbFetch()* will allow us to move the results of the query into a readable DataFrame.

```
query <- dbSendQuery(cnxn,
  "SELECT
    r.datetimeid,
    r.user_id,
    f.name,
    r.rating,
    f.release_year
  FROM ratings r
  INNER JOIN films f
    ON r.movie_id = f.movie_id ")

df <- dbFetch(query)
```

## Close Connection

Since we have gathered the data we needed from the database, we will close the connection to PostgreSQL.

```
# Disconnect from database
dbDisconnect(cnxn)
```

```
## Warning in connection_release(conn@ptr): There is a result object still in use.
## The connection will be automatically released when it is closed
```