Wk2: SQL and R

Dirk Hartog

2023-09-09

SQL and R

STEP 1:

For this problem I created a new database 'movies' in MySQL using the DBMS MySQLWorkbench. The sql code file can be found here SQL CODE

STEP 2:

Since I wanted to do most of the data transfromation and wrangling in R studio I connected to the local server in MySQL and loaded in the

I first needed to load in all the packages I intended to use. You will also need to hav the keyring.r file in the same folder to access the database username and password, which can be found in the link above

```
library(DBI)
library(RMySQL)
library(dplyr)
library(keyring)
```

Connecting to the database and loading the table into R was successful

```
mydb <-  dbConnect(dbDriver("MySQL"),
                   user = key_get("un"),
                   password = key_get("pw"),
                   dbname = 'movies',
                   host = 'Dirks-MacBook-Air.local',
                   port = 3306)
dbListTables(mydb)
```

```
## [1] "movie_ratings"
```

```
movie_ratings <- dbReadTable(mydb, "movie_ratings")
movie_ratings
```

```
##   user_id barbie oppenheimer the_little_mermaid guardians_of_the_galaxy
## 1       1      4           4                  3                       5
## 2       2      5          NA                 NA                       4
## 3       3     NA          NA                  1                       2
## 4       4      4           4                  3                       4
## 5       5      4          NA                  5                       3
##   spiderman_across_the_spiderverse the_super_mario_bros_movie
## 1                                4                          2
## 2                               NA                         NA
## 3                                2                          1
## 4                                4                          4
## 5                                3                          2
```

STEP 3:

As this was a small data set we could view a lot of information of the data frame by calling glimpse. The data types, column names and getting an idea of, any missing values as well as the values in each variable or column.

```
## Rows: 5
## Columns: 7
## $ user_id                          <int> 1, 2, 3, 4, 5
## $ barbie                           <int> 4, 5, NA, 4, 4
## $ oppenheimer                      <int> 4, NA, NA, 4, NA
## $ the_little_mermaid               <int> 3, NA, 1, 3, 5
## $ guardians_of_the_galaxy          <int> 5, 4, 2, 4, 3
## $ spiderman_across_the_spiderverse <int> 4, NA, 2, 4, 3
## $ the_super_mario_bros_movie       <int> 2, NA, 1, 4, 2
```

STEP 4:

I also wanted to get some summary statistics about the ratings of each movie.

```
summary(movie_ratings)
```

```
##     user_id      barbie      oppenheimer the_little_mermaid
##  Min.   :1   Min.   :4.00   Min.   :4    Min.   :1.0
##  1st Qu.:2   1st Qu.:4.00   1st Qu.:4    1st Qu.:2.5
##  Median :3   Median :4.00   Median :4    Median :3.0
##  Mean   :3   Mean   :4.25   Mean   :4    Mean   :3.0
##  3rd Qu.:4   3rd Qu.:4.25   3rd Qu.:4    3rd Qu.:3.5
##  Max.   :5   Max.   :5.00   Max.   :4    Max.   :5.0
##              NA's   :1      NA's   :3    NA's   :1
##  guardians_of_the_galaxy spiderman_across_the_spiderverse
##  Min.   :2.0             Min.   :2.00
##  1st Qu.:3.0             1st Qu.:2.75
##  Median :4.0             Median :3.50
##  Mean   :3.6             Mean   :3.25
##  3rd Qu.:4.0             3rd Qu.:4.00
##  Max.   :5.0             Max.   :4.00
```

```
##                          NA's   :1
##  the_super_mario_bros_movie
##  Min.   :1.00
##  1st Qu.:1.75
##  Median :2.00
##  Mean   :2.25
##  3rd Qu.:2.50
##  Max.   :4.00
##  NA's   :1
```

STEP 5:

Next was deciding how to handle missing values. Looking at the mean and median values in this data set were similar. For this particular data set I decided to use the mean ratings for each movie to fill in the NA values.

```
movie_ratings$barbie[is.na(movie_ratings$barbie)] <- mean(movie_ratings$barbie, na.rm = TRUE)

movie_ratings$oppenheimer[is.na(movie_ratings$oppenheimer)] <- mean(movie_ratings$oppenheimer, na.rm = 

movie_ratings$the_little_mermaid[is.na(movie_ratings$the_little_mermaid)] <- mean(movie_ratings$the_lit

movie_ratings$spiderman_across_the_spiderverse[is.na(movie_ratings$spiderman_across_the_spiderverse)] <

movie_ratings$the_super_mario_bros_movie[is.na(movie_ratings$the_super_mario_bros_movie)] <- mean(movie
```

As we see from the summary statistics the mean ratings of each movie did not change when we filled in the NA values with the movies respective mean.

```
summary(movie_ratings)
```

```
##     user_id      barbie      oppenheimer the_little_mermaid
##  Min.   :1   Min.   :4.00   Min.   :4    Min.   :1
##  1st Qu.:2   1st Qu.:4.00   1st Qu.:4    1st Qu.:3
##  Median :3   Median :4.00   Median :4    Median :3
##  Mean   :3   Mean   :4.25   Mean   :4    Mean   :3
##  3rd Qu.:4   3rd Qu.:4.25   3rd Qu.:4    3rd Qu.:3
##  Max.   :5   Max.   :5.00   Max.   :4    Max.   :5
##  guardians_of_the_galaxy spiderman_across_the_spiderverse
##  Min.   :2.0             Min.   :2.00
##  1st Qu.:3.0             1st Qu.:3.00
##  Median :4.0             Median :3.25
##  Mean   :3.6             Mean   :3.25
##  3rd Qu.:4.0             3rd Qu.:4.00
##  Max.   :5.0             Max.   :4.00
##  the_super_mario_bros_movie
##  Min.   :1.00
##  1st Qu.:2.00
##  Median :2.00
##  Mean   :2.25
##  3rd Qu.:2.25
##  Max.   :4.00
```

CONCLUSION:

Other solutions to this problem include the following

1. To make gathering data easier and efficient would be to create a google form survey. This collects the responses and can populates them into an excel spreadsheet. This can then be easily read into an R data frame.

2. Finding other techniques to encrypt the username and password to connect to MySQL would also be something to explore in the future. While this program does not send the username and password directly there are still ways of accessing it. There are many packages in R that are used for creating a secure flow of information that need to be explored.