# Assignment - 2

Irene Jacob

01/09/2020

## ASSIGNMENT 2

Choose six recent popular movies. Ask at least five people that you know (friends, family, classmates, imaginary friends if necessary) to rate each of these movies that they have seen on a scale of 1 to 5. Take the results (observations) and store them in a SQL database of your choosing. Load the information from the SQL database into an R dataframe.

### Step 1: Collect the data and store in SQL database

To see the dataset click here.

### Step 2: Connecting R with MySQL

```
library(RMySQL)
library(RODBC)
library("dplyr")
library("dbplyr")

connection <- RODBC::odbcConnect("data607")
```

### Step 3: Load the dataset in R dataframe

```
library(sqldf)
library(DBI)

SQLtoR <- RODBC::sqlQuery(connection, "select * from `assign_2`")
print(SQLtoR)
```

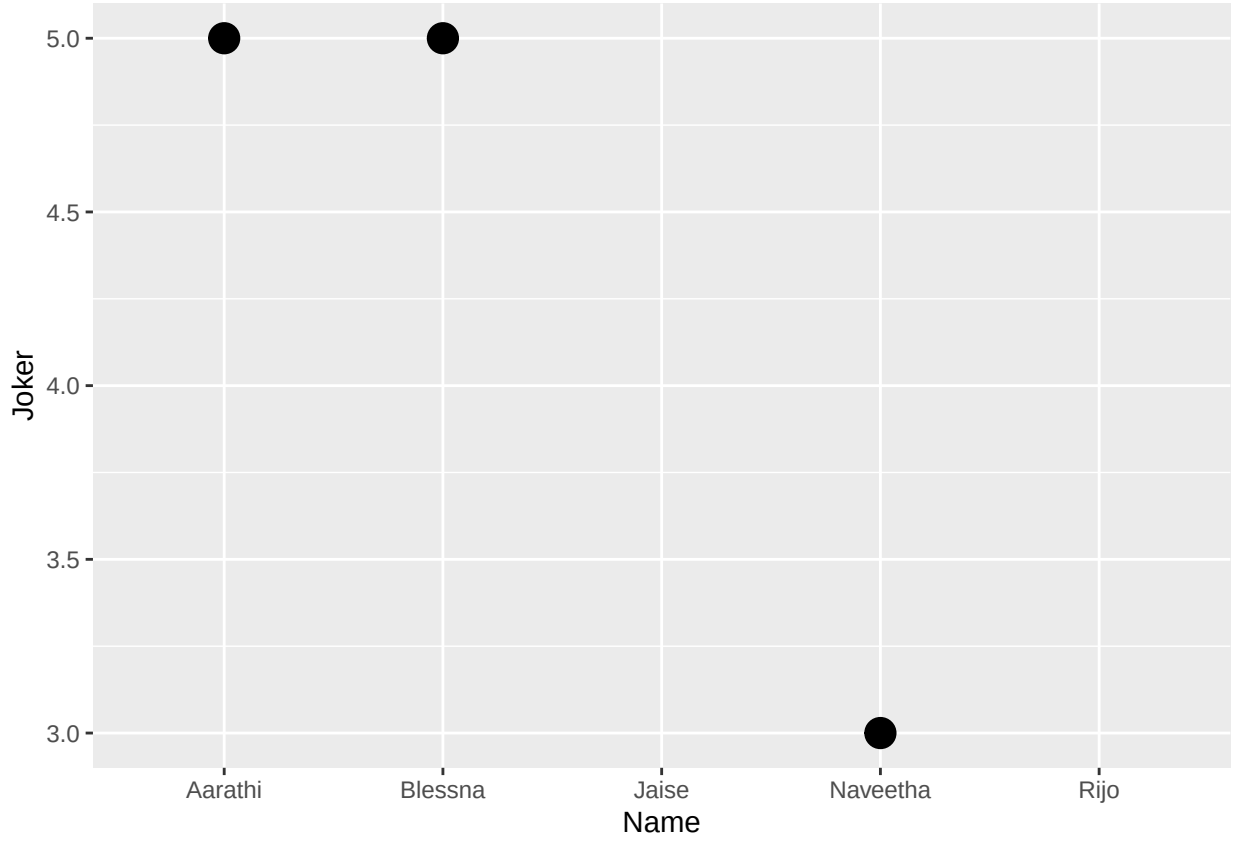
```
##       Name Joker Avengers_Endgame Lion_King Forrest_Gump Shutter_Island
## 1     Rijo    NA                3         3           NA             NA
## 2  Blessna     5                5        NA            5              5
## 3 Naveetha     3                5        NA           NA             NA
## 4  Aarathi     5                4         5            5              5
## 5    Jaise    NA               NA         4           NA             NA
##   Black_Panther
## 1             4
## 2             4
## 3             5
## 4             4
## 5            NA
```

### Step 4: Graphs of each movie

Here the null values are eliminated

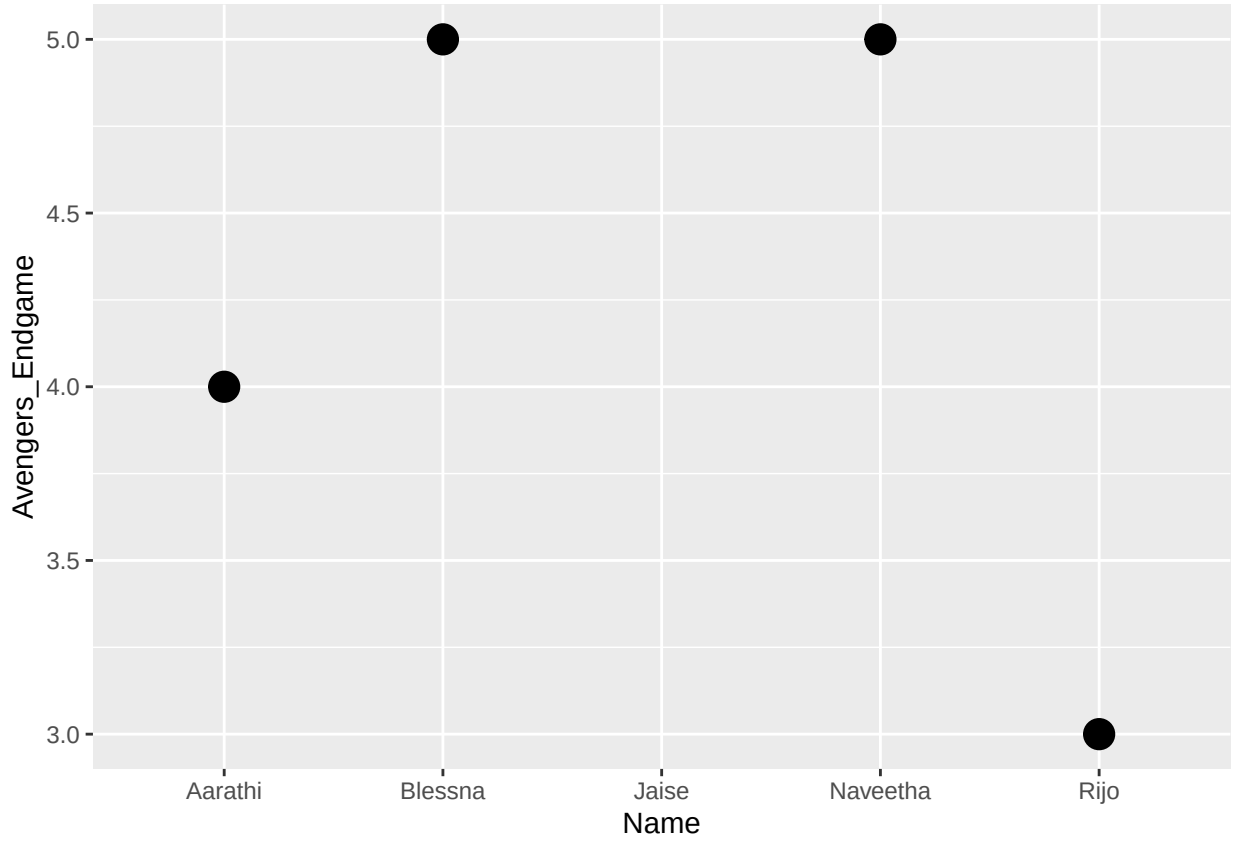
```
ggplot(data = SQLtoR, mapping = aes(x = Name, y = Joker)) +
  geom_point(size = 5)
```

```
## Warning: Removed 2 rows containing missing values (geom_point).
```



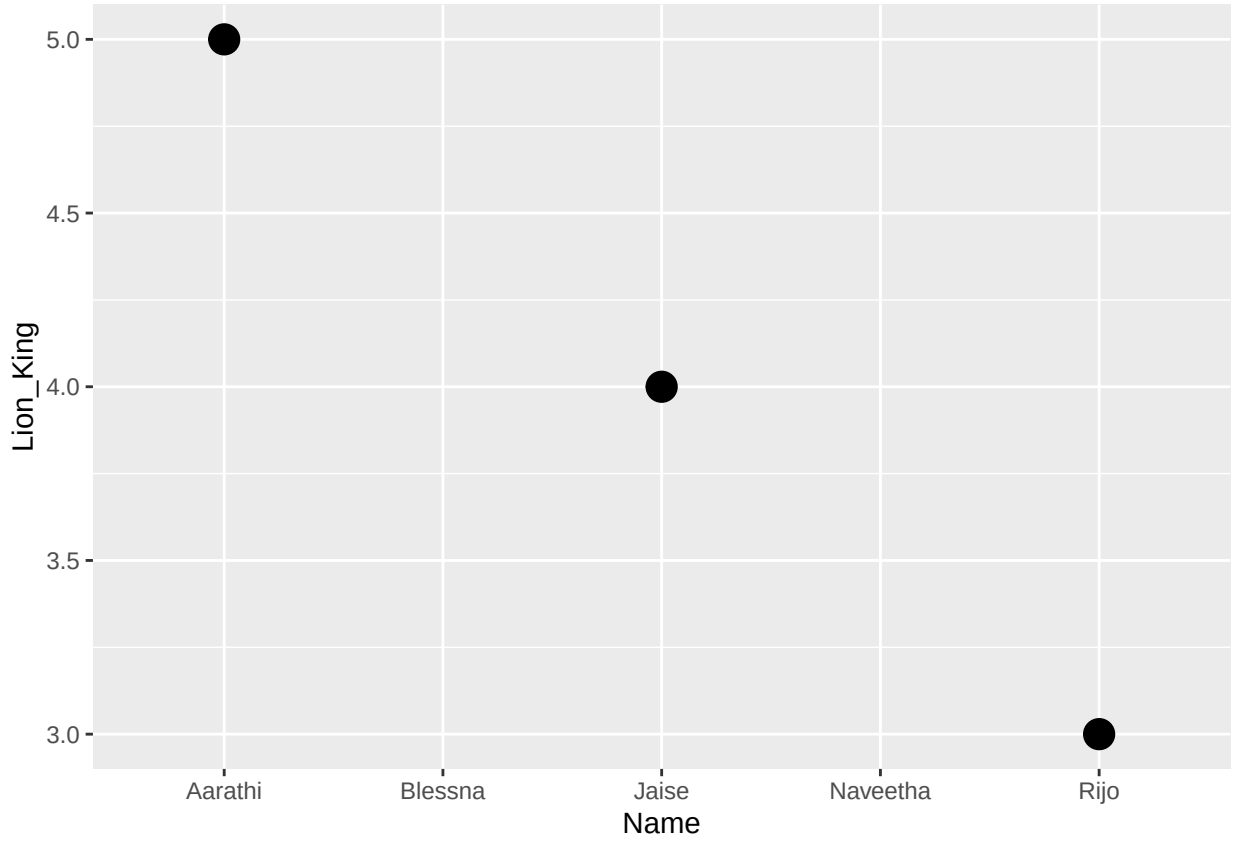

```
ggplot(data = SQLtoR, mapping = aes(x = Name, y = Avengers_Endgame)) +
  geom_point(size = 5)
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



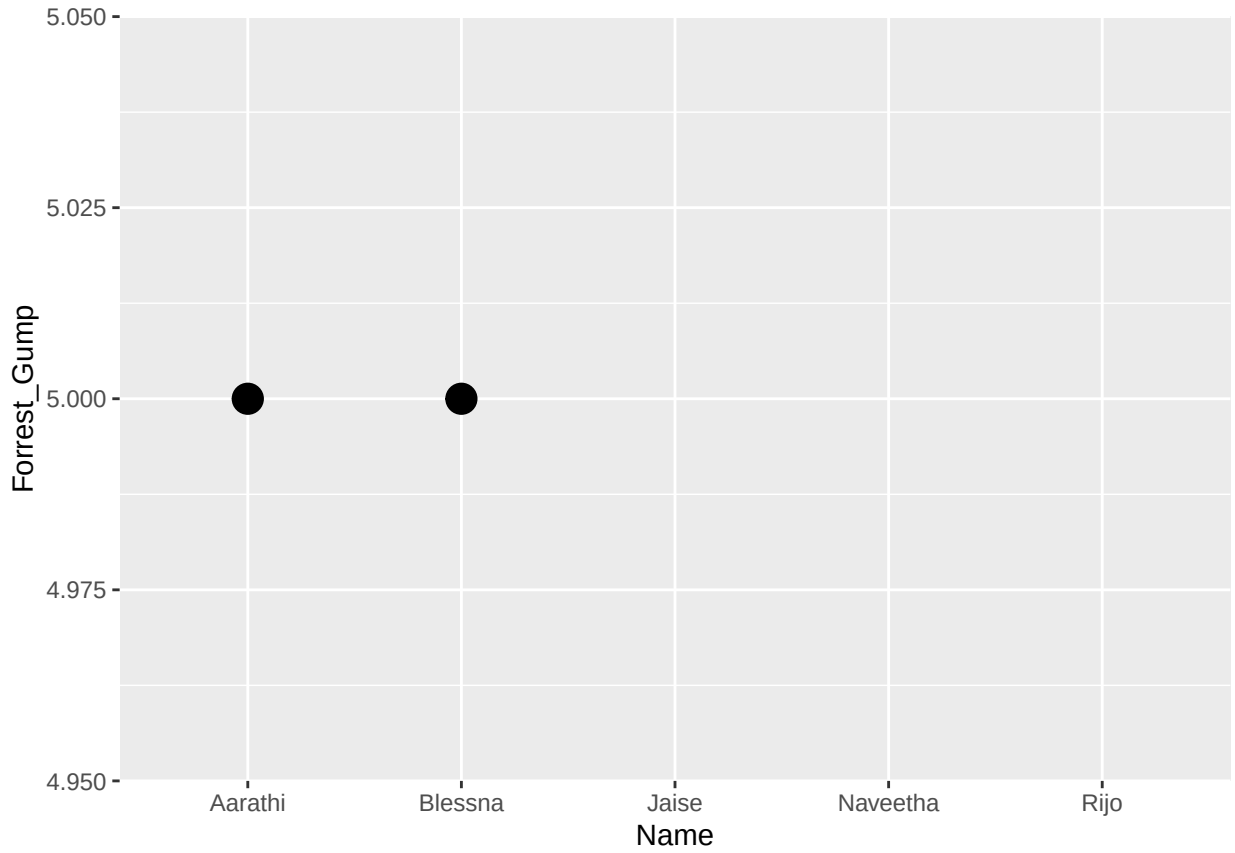

```
ggplot(data = SQLtoR, mapping = aes(x = Name, y = Lion_King)) +
  geom_point(size = 5)
```

```
## Warning: Removed 2 rows containing missing values (geom_point).
```



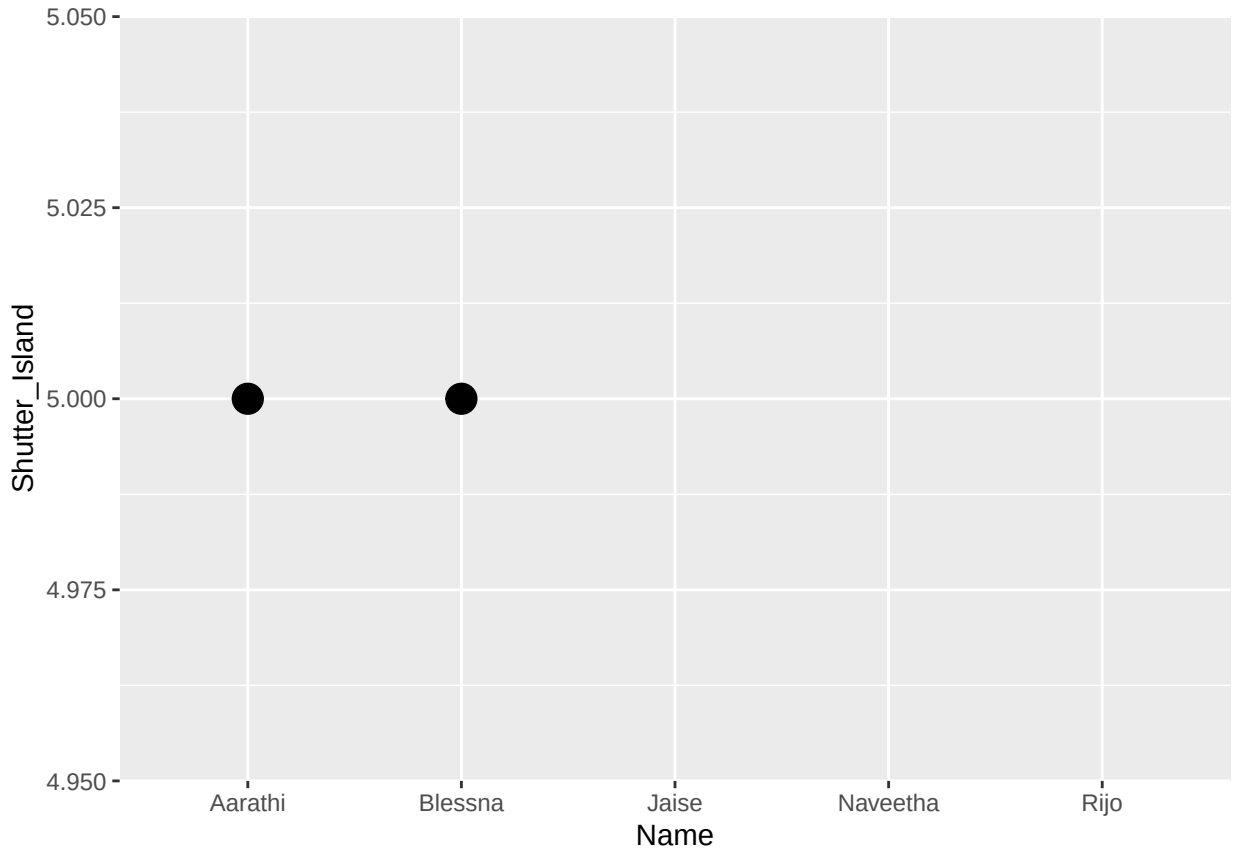

```
ggplot(data = SQLtoR, mapping = aes(x = Name, y = Forrest_Gump)) +
  geom_point(size = 5)
```

```
## Warning: Removed 3 rows containing missing values (geom_point).
```



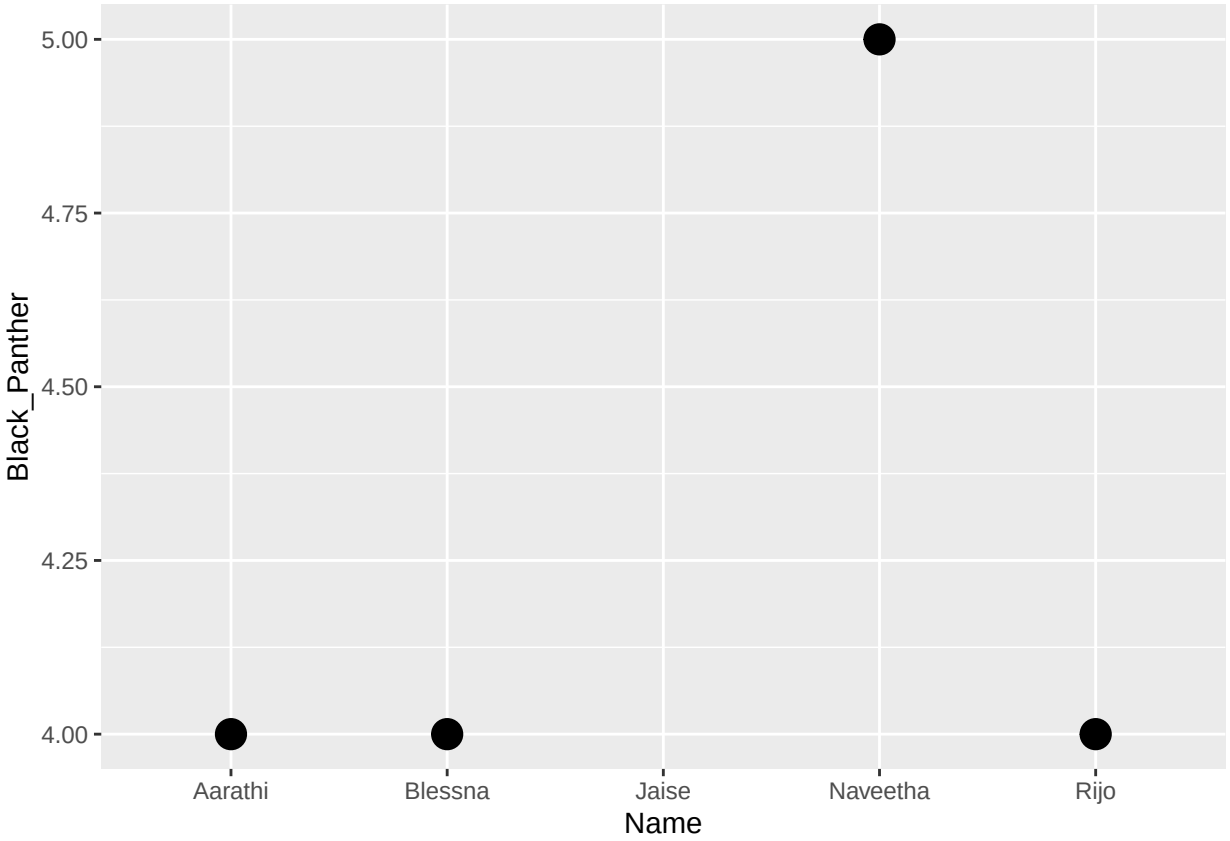

```
ggplot(data = SQLtoR, mapping = aes(x = Name, y = Shutter_Island)) +
  geom_point(size = 5)
```

```
## Warning: Removed 3 rows containing missing values (geom_point).
```




```
ggplot(data = SQLtoR, mapping = aes(x = Name, y = Black_Panther)) +
  geom_point(size = 5)
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```


5.00
4.75
4.50
4.25
4.00
Black_Panther
Aarathi
Blessna
Jaise
Naveetha
Rijo
Name

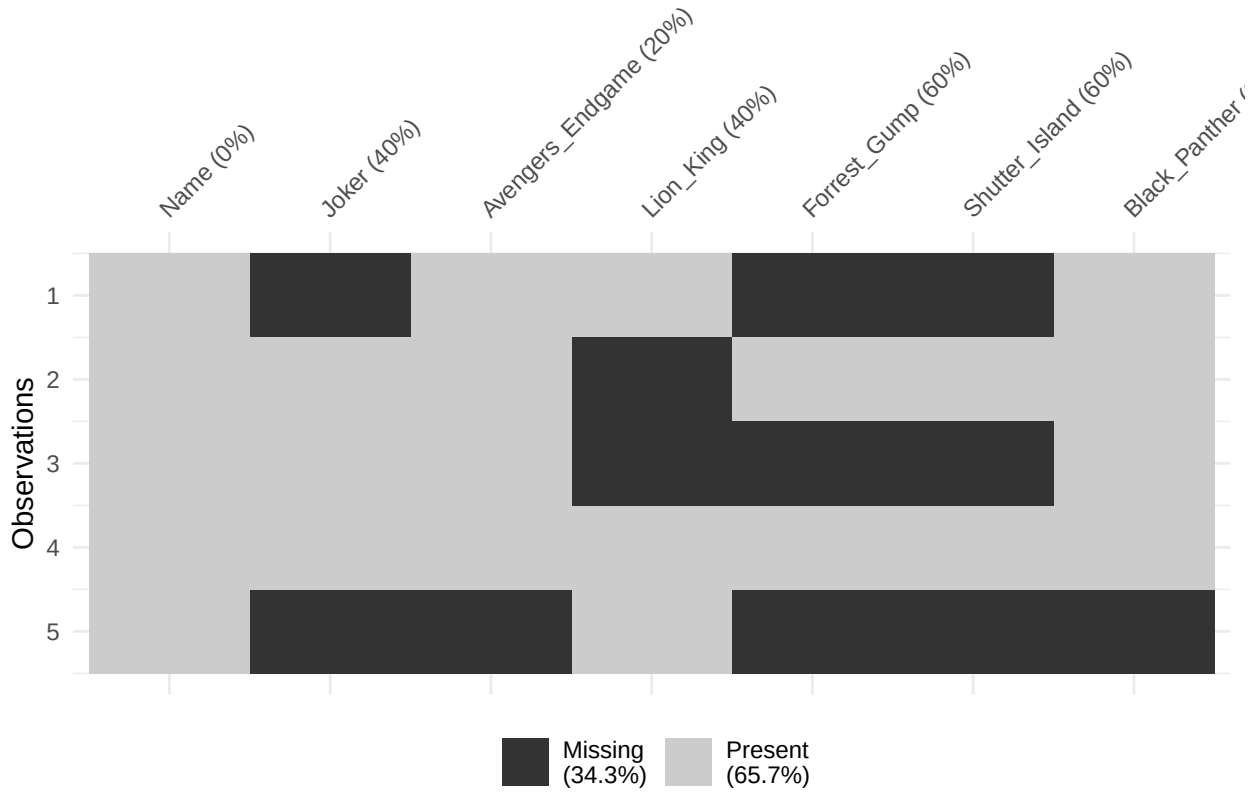
### Step 5: Handling missing values

The 'naniar' package provides many functions to identify and deal with missing values.

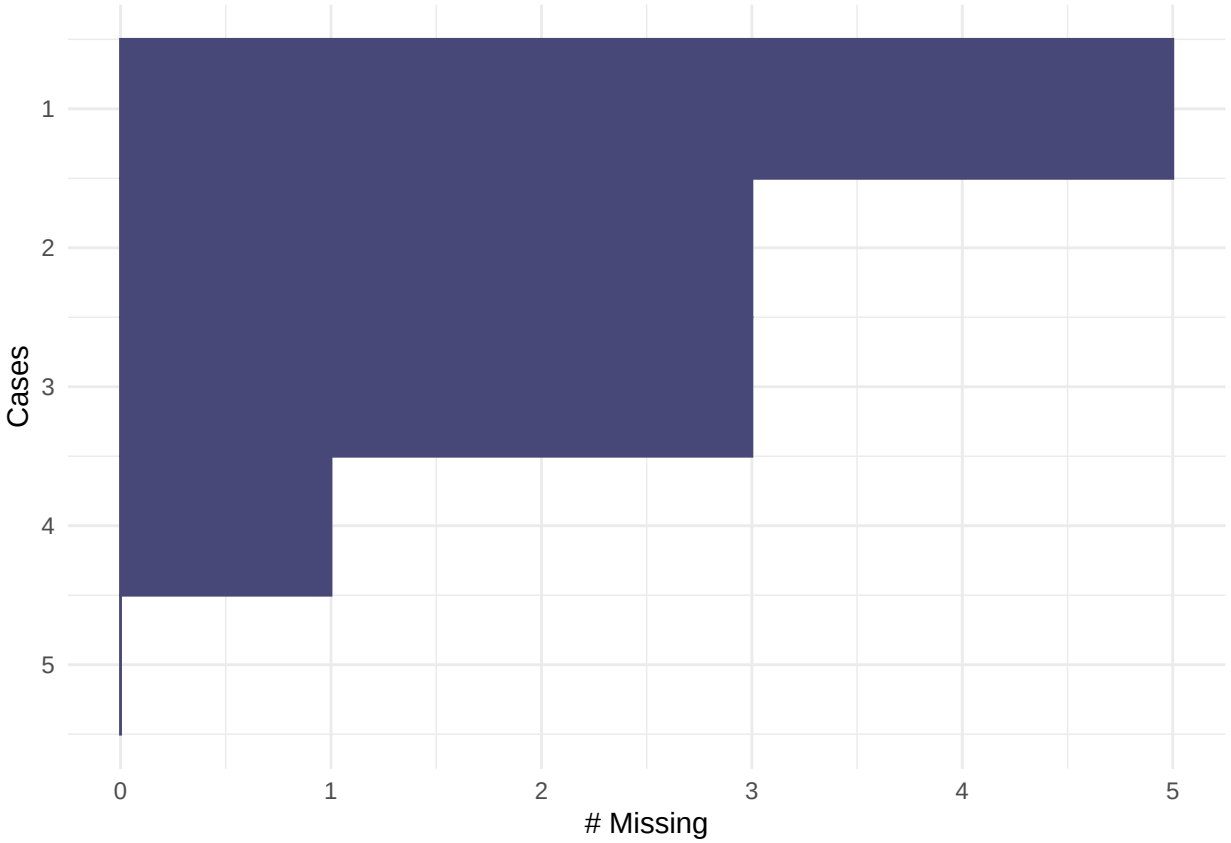
```
library(naniar)

any_na(SQLtoR)          #will tell if there are missing values in the data frame
```

```
## [1] TRUE
```

```
vis_miss(SQLtoR)        #will give a visual on the missing data
```




```
gg_miss_case(SQLtoR)    #will give a visual on the missing data at the row level
```


Cases
1
2
3
4
5
0
1
2
3
4
5
# Missing