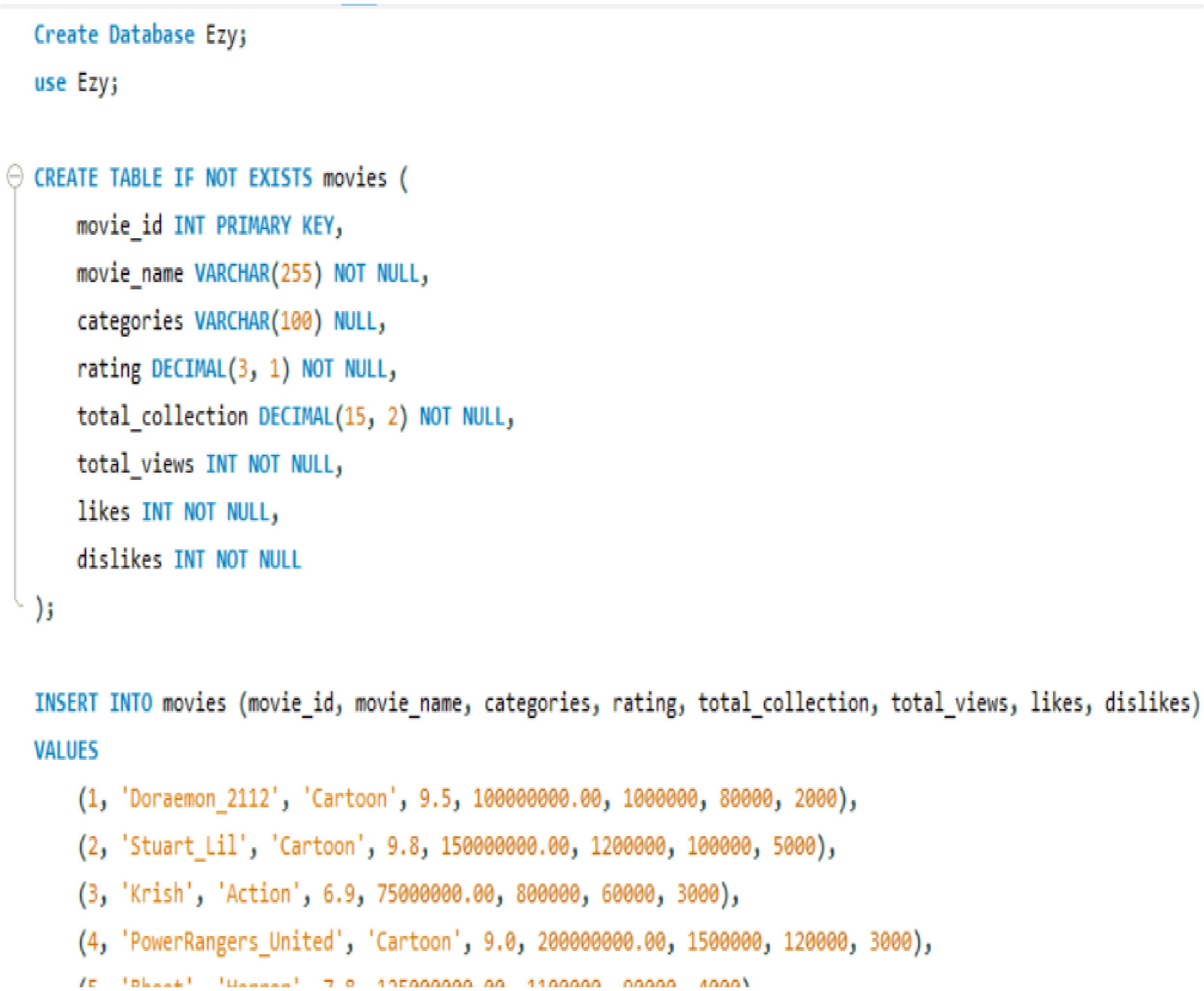
Name – Manish Devidas Thale

Project – Project of R Programming(Analysis and Visualization of Movies Data Set using Database Connectivity)

Steps 1:

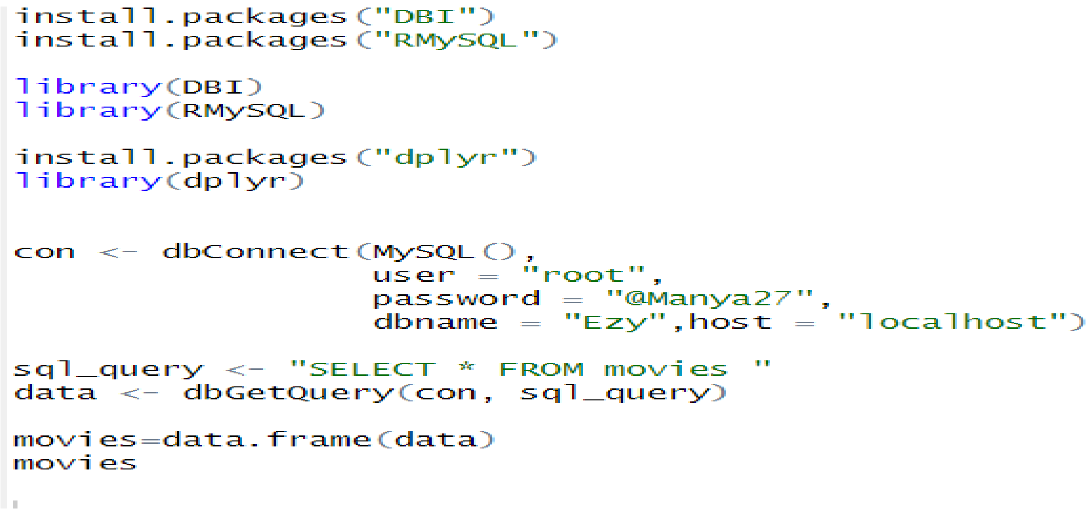
- Created a Database Ezy .
- Inside that database created table with columns like id , name , categories , view …etc.
- Inserted 20 rows inside that table.



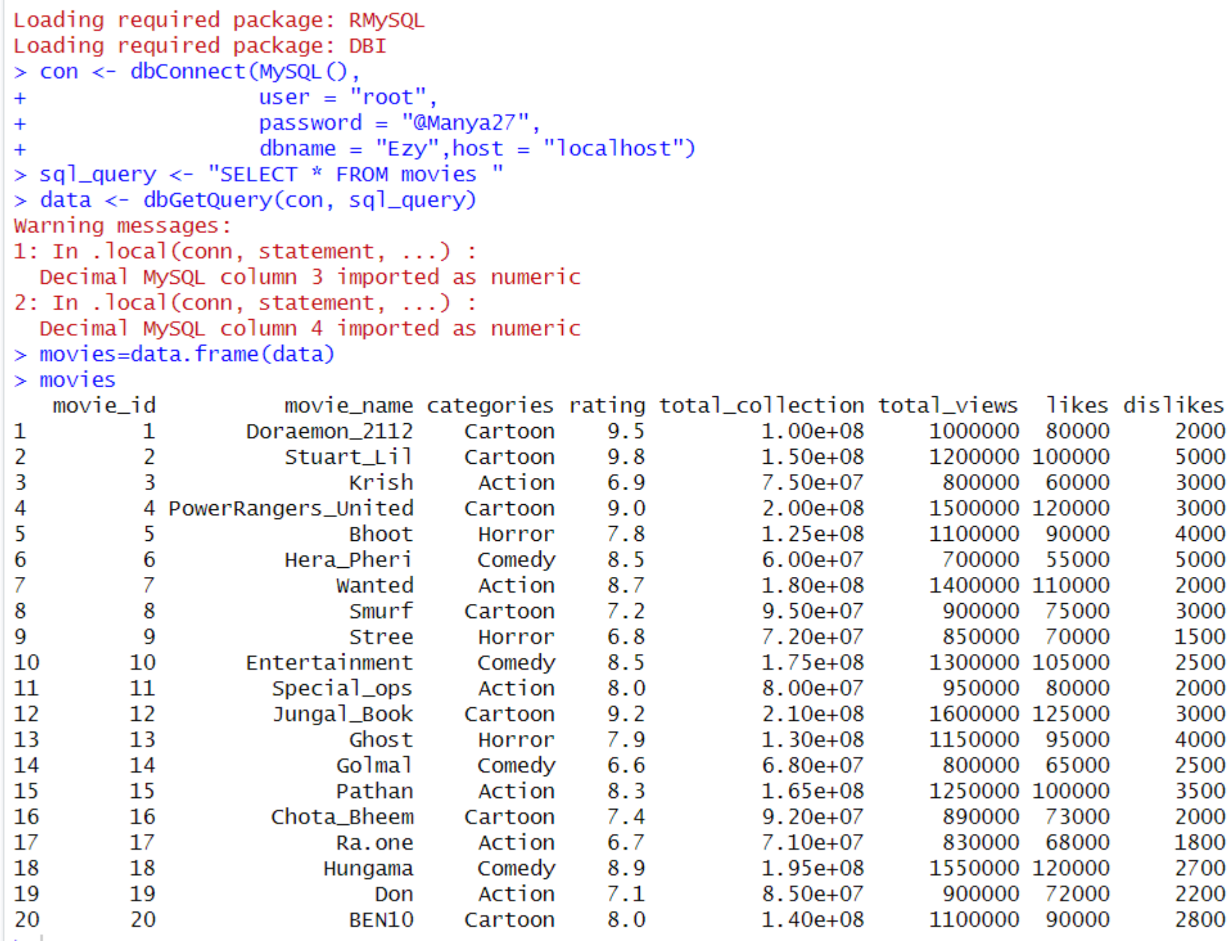
```
Create Database Ezy;
use Ezy;

CREATE TABLE IF NOT EXISTS movies (
    movie_id INT PRIMARY KEY,
    movie_name VARCHAR(255) NOT NULL,
    categories VARCHAR(100) NULL,
    rating DECIMAL(3, 1) NOT NULL,
    total_collection DECIMAL(15, 2) NOT NULL,
    total_views INT NOT NULL,
    likes INT NOT NULL,
    dislikes INT NOT NULL
);

INSERT INTO movies (movie_id, movie_name, categories, rating, total_collection, total_views, likes, dislikes)
VALUES
    (1, 'Doraemon_2112', 'Cartoon', 9.5, 100000000.00, 1000000, 80000, 2000),
    (2, 'Stuart_Lil', 'Cartoon', 9.8, 150000000.00, 1200000, 100000, 5000),
    (3, 'Krish', 'Action', 6.9, 75000000.00, 800000, 60000, 3000),
    (4, 'PowerRangers_United', 'Cartoon', 9.0, 200000000.00, 1500000, 120000, 3000),
    (5, 'Bhoot', 'Horror', 7.8, 125000000.00, 1100000, 90000, 4000)
```

Step 2 :

- In R studio Installed all Necessary Packages.
- Connect the database with R Using appropriate database name , password , username etc..
- Fetch the data and convert it to data frame .



```
install.packages("DBI")
install.packages("RMySQL")

library(DBI)
library(RMySQL)

install.packages("dplyr")
library(dplyr)


con <- dbConnect(MySQL(),
                 user = "root",
                 password = "@Manya27",
                 dbname = "Ezy",host = "localhost")

sql_query <- "SELECT * FROM movies "
data <- dbGetQuery(con, sql_query)

movies=data.frame(data)
movies
```



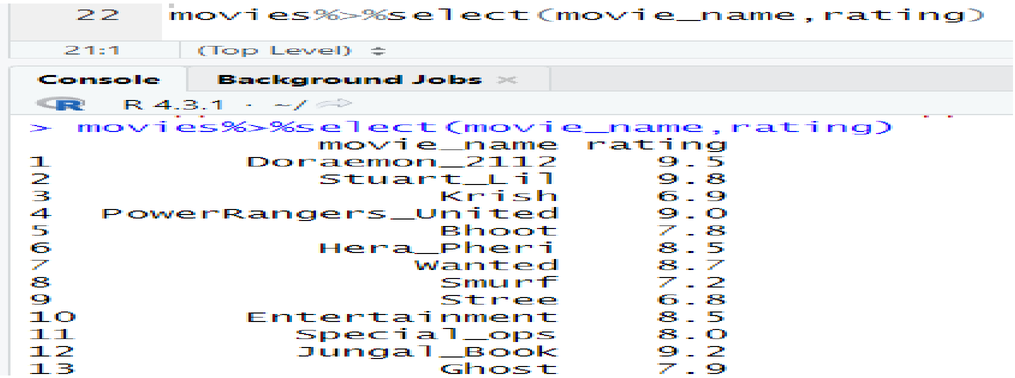
```
Loading required package: RMySQL
Loading required package: DBI
> con <- dbConnect(MySQL(),
+                  user = "root",
+                  password = "@Manya27",
+                  dbname = "Ezy",host = "localhost")
> sql_query <- "SELECT * FROM movies "
> data <- dbGetQuery(con, sql_query)
Warning messages:
1: In .local(conn, statement, ...) :
  Decimal MySQL column 3 imported as numeric
2: In .local(conn, statement, ...) :
  Decimal MySQL column 4 imported as numeric
> movies=data.frame(data)
> movies
   movie_id          movie_name categories rating total_collection total_views  likes dislikes
1         1       Doraemon_2112    Cartoon    9.5         1.00e+08     1000000  80000     2000
2         2          Stuart_Lil    Cartoon    9.8         1.50e+08     1200000 100000     5000
3         3               Krish     Action    6.9         7.50e+07      800000  60000     3000
4         4 PowerRangers_United    Cartoon    9.0         2.00e+08     1500000 120000     3000
5         5               Bhoot     Horror    7.8         1.25e+08     1100000  90000     4000
6         6          Hera_Pheri     Comedy    8.5         6.00e+07      700000  55000     5000
7         7              Wanted     Action    8.7         1.80e+08     1400000 110000     2000
8         8               Smurf    Cartoon    7.2         9.50e+07      900000  75000     3000
9         9               Stree     Horror    6.8         7.20e+07      850000  70000     1500
10       10       Entertainment     Comedy    8.5         1.75e+08     1300000 105000     2500
11       11         Special_ops     Action    8.0         8.00e+07      950000  80000     2000
12       12         Jungal_Book    Cartoon    9.2         2.10e+08     1600000 125000     3000
13       13               Ghost     Horror    7.9         1.30e+08     1150000  95000     4000
14       14              Golmal     Comedy    6.6         6.80e+07      800000  65000     2500
15       15              Pathan     Action    8.3         1.65e+08     1250000 100000     3500
16       16         Chota_Bheem    Cartoon    7.4         9.20e+07      890000  73000     2000
17       17              Ra.one     Action    6.7         7.10e+07      830000  68000     1800
18       18             Hungama     Comedy    8.9         1.95e+08     1550000 120000     2700
19       19                 Don     Action    7.1         8.50e+07      900000  72000     2200
20       20               BEN10    Cartoon    8.0         1.40e+08     1100000  90000     2800
```

Step3 :

- Data manipulation , Analyse and Visualize the data frame using different functions.
- Visualize data using different Visuals or charts or graphs.

➔ movies%>%select(movie_name,rating)

It gives only 2 columns movie name and rating

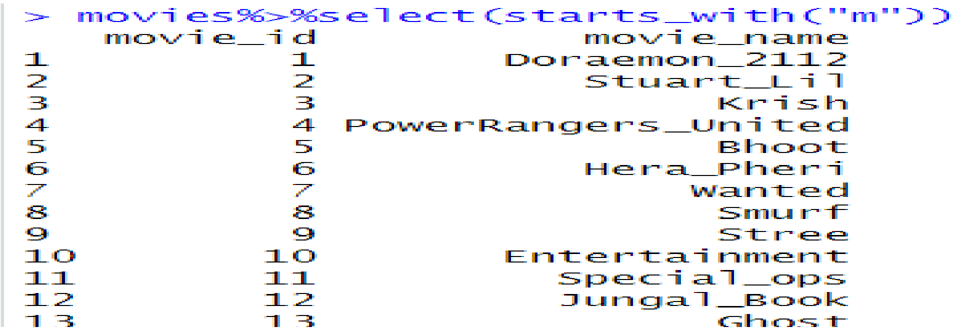


```
22   movies%>%select(movie_name,rating)
```

```
> movies%>%select(movie_name,rating)
              movie_name rating
1          Doraemon_2112    9.5
2            Stuart_Lil     9.8
3                 Krish     6.9
4   PowerRangers_United     9.0
5                 Bhoot     7.8
6            Hera_Pheri     8.5
7                Wanted     8.7
8                 Smurf     7.2
9                 Stree     6.8
10        Entertainment     8.5
11          Special_ops     8.0
12         Jungal_Book      9.2
13                Ghost     7.9
```

➔ data.frame%>%select(starts_with("m"))

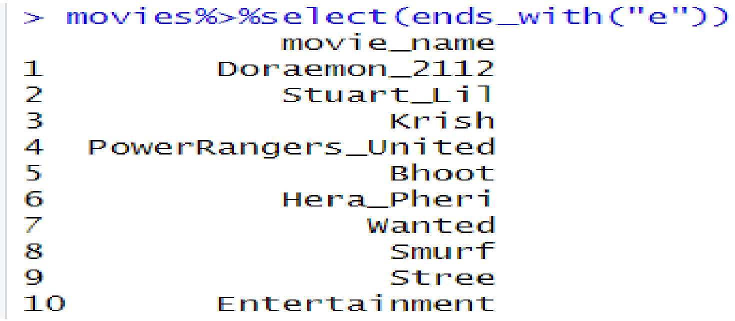
It gives Columns name starting with “m”



```
> movies%>%select(starts_with("m"))
   movie_id          movie_name
1         1       Doraemon_2112
2         2          Stuart_Lil
3         3               Krish
4         4 PowerRangers_United
5         5               Bhoot
6         6          Hera_Pheri
7         7              Wanted
8         8               Smurf
9         9               Stree
10       10       Entertainment
11       11         Special_ops
12       12         Jungal_Book
13       13               Ghost
```

➔ data.frame%>%select(ends_with("e"))

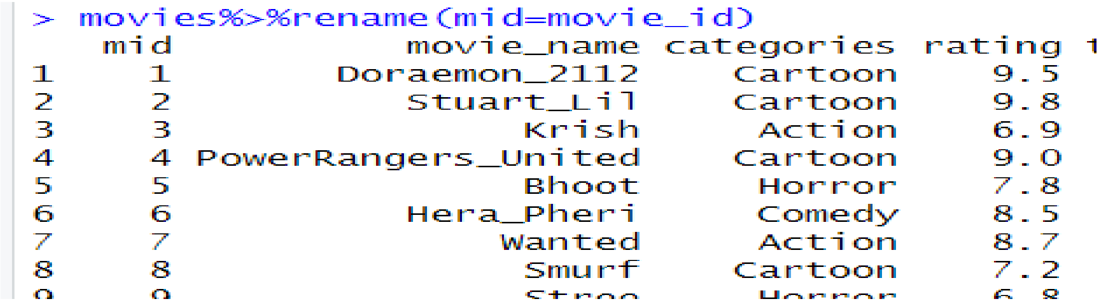
It gives Columns name ends with “ E “



```
> movies%>%select(ends_with("e"))
            movie_name
1        Doraemon_2112
2           Stuart_Lil
3                Krish
4  PowerRangers_United
5                Bhoot
6           Hera_Pheri
7               Wanted
8                Smurf
9                Stree
10       Entertainment
```

➔ movies%>%rename(mid=movie_id)

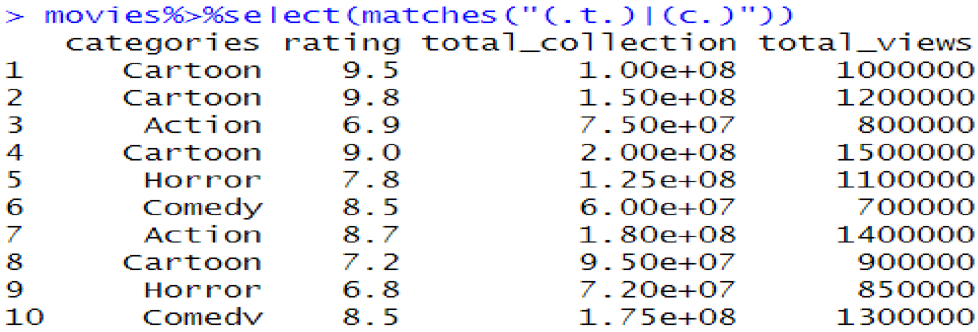
It will change the column name from movie_id to mid



```
> movies%>%rename(mid=movie_id)
  mid            movie_name categories rating 
1   1         Doraemon_2112    Cartoon    9.5
2   2           Stuart_Lil    Cartoon    9.8
3   3                Krish     Action    6.9
4   4 PowerRangers_United    Cartoon    9.0
5   5                Bhoot     Horror    7.8
6   6           Hera_Pheri     Comedy    8.5
7   7               Wanted     Action    8.7
8   8                Smurf    Cartoon    7.2
9   9                Stree     Horror    6.8
```

➔ movies%>%select(matches("(.t.)|(c.)"))

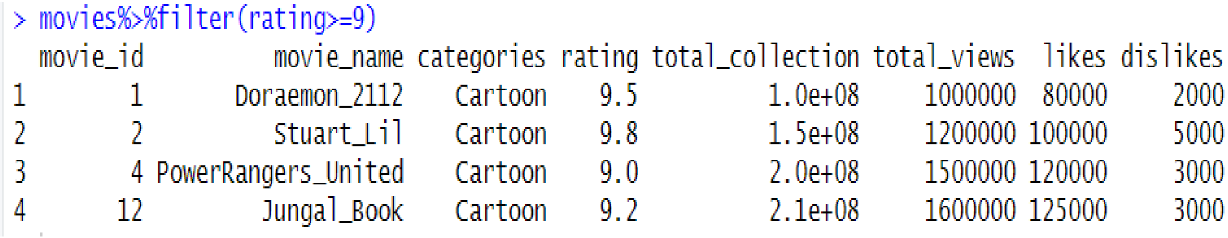
It gives Columns name that matches the condition written inside ( ) brackets .



```
> movies%>%select(matches("(.t.)|(c.)"))
   categories rating total_collection total_views
1     Cartoon    9.5         1.00e+08     1000000
2     Cartoon    9.8         1.50e+08     1200000
3      Action    6.9         7.50e+07      800000
4     Cartoon    9.0         2.00e+08     1500000
5      Horror    7.8         1.25e+08     1100000
6      Comedy    8.5         6.00e+07      700000
7      Action    8.7         1.80e+08     1400000
8     Cartoon    7.2         9.50e+07      900000
9      Horror    6.8         7.20e+07      850000
10     Comedy    8.5         1.75e+08     1300000
```

➔ movies%>%filter(rating>=9)

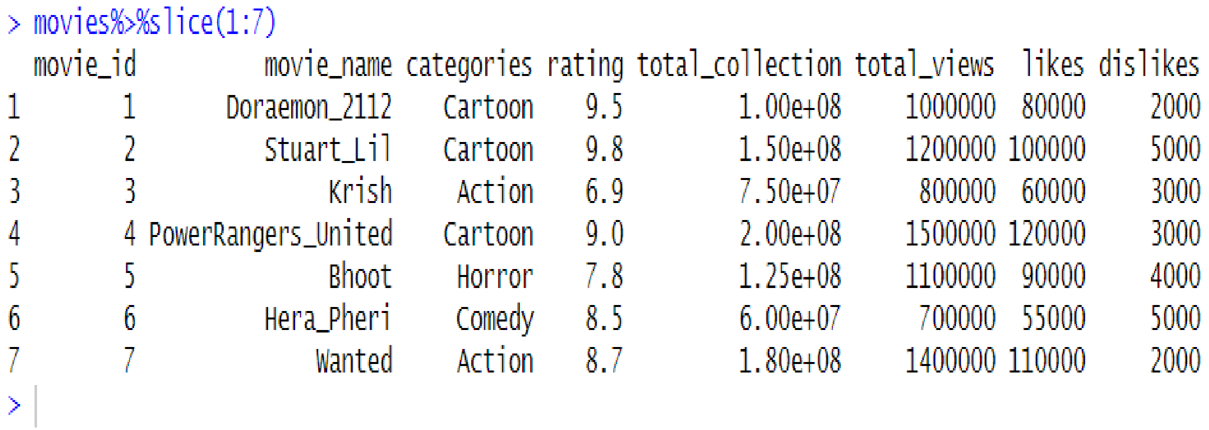
It will give all the movies data with rating 9 or greater than 9



```
> movies%>%filter(rating>=9)
  movie_id          movie_name categories rating total_collection total_views  likes dislikes
1        1       Doraemon_2112    Cartoon    9.5           1.0e+08     1000000  80000     2000
2        2          Stuart_Lil    Cartoon    9.8           1.5e+08     1200000 100000     5000
3        4 PowerRangers_United    Cartoon    9.0           2.0e+08     1500000 120000     3000
4       12          Jungal_Book    Cartoon    9.2           2.1e+08     1600000 125000     3000
```

- ➔ movies%>%slice(1:7)

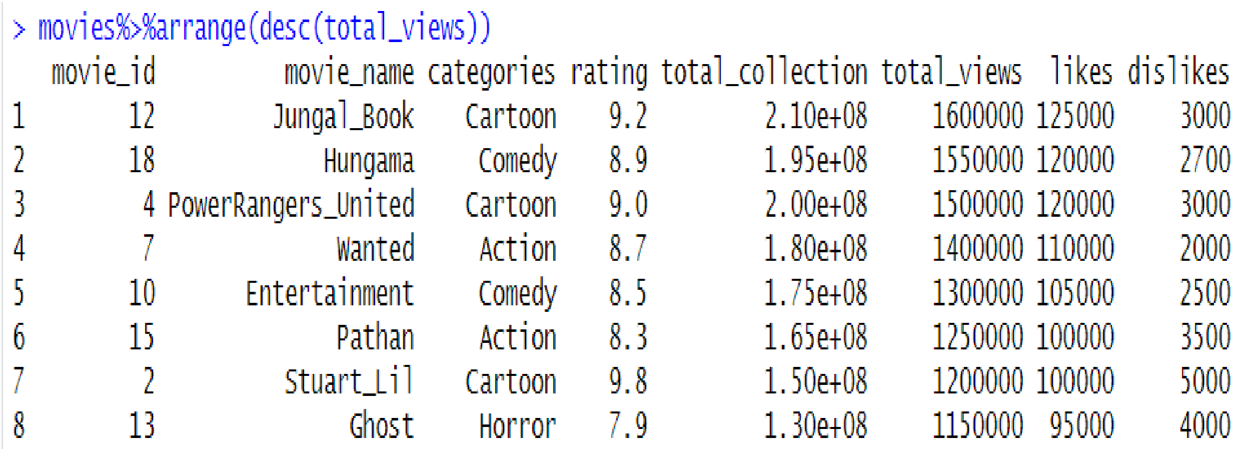
  It will gives data frame first 7 rows



```
> movies%>%slice(1:7)
  movie_id         movie_name categories rating total_collection total_views  likes dislikes
1        1       Doraemon_2112    Cartoon    9.5         1.00e+08     1000000  80000     2000
2        2          Stuart_Lil    Cartoon    9.8         1.50e+08     1200000 100000     5000
3        3               Krish     Action    6.9         7.50e+07      800000  60000     3000
4        4 PowerRangers_United    Cartoon    9.0         2.00e+08     1500000 120000     3000
5        5               Bhoot     Horror    7.8         1.25e+08     1100000  90000     4000
6        6          Hera_Pheri     Comedy    8.5         6.00e+07      700000  55000     5000
7        7              Wanted     Action    8.7         1.80e+08     1400000 110000     2000
> 
```

- ➔ movies%>%arrange(desc(total_views))

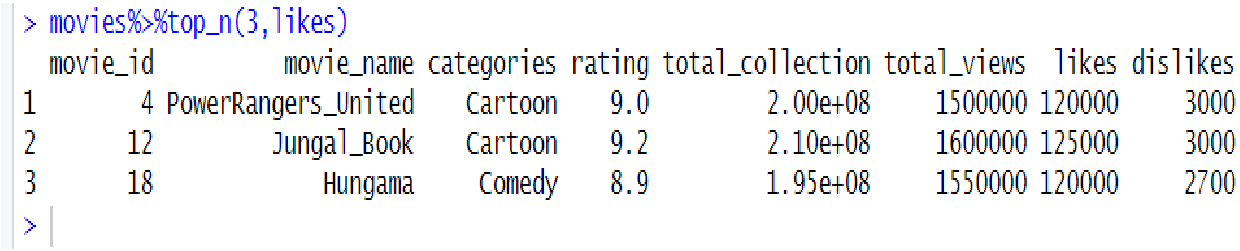
  It will arrange data frame in descending order according to the total views



```
> movies%>%arrange(desc(total_views))
  movie_id         movie_name categories rating total_collection total_views  likes dislikes
1       12          Jungal_Book    Cartoon    9.2         2.10e+08     1600000 125000     3000
2       18             Hungama     Comedy    8.9         1.95e+08     1550000 120000     2700
3        4 PowerRangers_United    Cartoon    9.0         2.00e+08     1500000 120000     3000
4        7              Wanted     Action    8.7         1.80e+08     1400000 110000     2000
5       10       Entertainment     Comedy    8.5         1.75e+08     1300000 105000     2500
6       15              Pathan     Action    8.3         1.65e+08     1250000 100000     3500
7        2          Stuart_Lil    Cartoon    9.8         1.50e+08     1200000 100000     5000
8       13               Ghost     Horror    7.9         1.30e+08     1150000  95000     4000
```

- ➔ movies%>%top_n(3,likes)

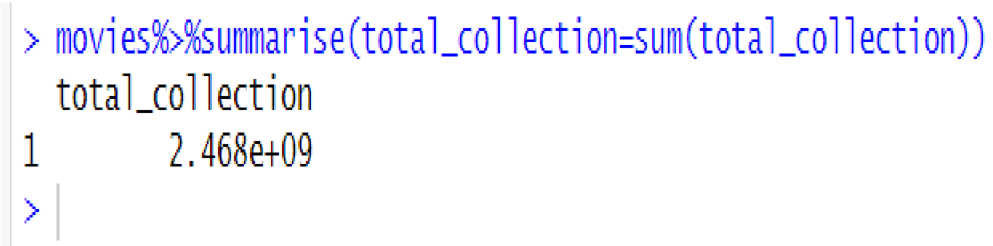
  It will gives 3 rows from dataframe with highest likes



```
> movies%>%top_n(3,likes)
  movie_id         movie_name categories rating total_collection total_views  likes dislikes
1        4 PowerRangers_United    Cartoon    9.0         2.00e+08     1500000 120000     3000
2       12          Jungal_Book    Cartoon    9.2         2.10e+08     1600000 125000     3000
3       18             Hungama     Comedy    8.9         1.95e+08     1550000 120000     2700
> 
```

➔ movies%>%summarise(total_collection=sum(total_collection))

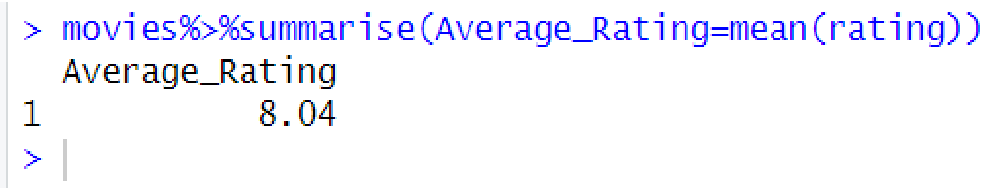
It will gives the total collection earn by all the movies from the data frame



```
> movies%>%summarise(total_collection=sum(total_collection))
  total_collection
1        2.468e+09
> 
```

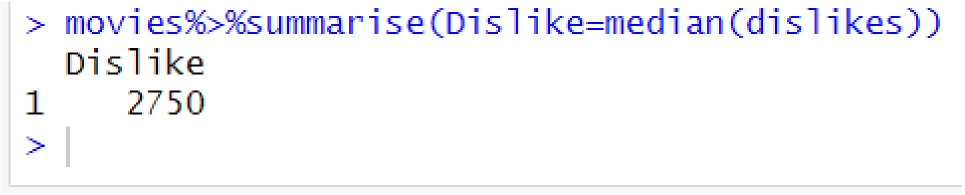
➔ movies%>%summarise(Average_Rating=mean(rating))

This code will give the average rating received by movies from audience



```
> movies%>%summarise(Average_Rating=mean(rating))
  Average_Rating
1           8.04
> 
```

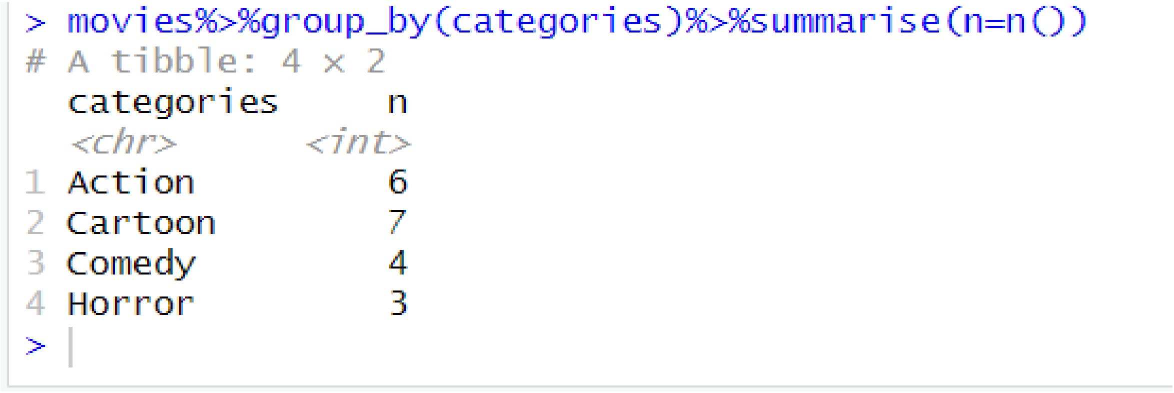
➔ movies%>%summarise(Dislike=median(dislikes))



```
> movies%>%summarise(Dislike=median(dislikes))
  Dislike
1    2750
> 
```

➔ movies%>%group_by(categories)%>%summarise(n=n())

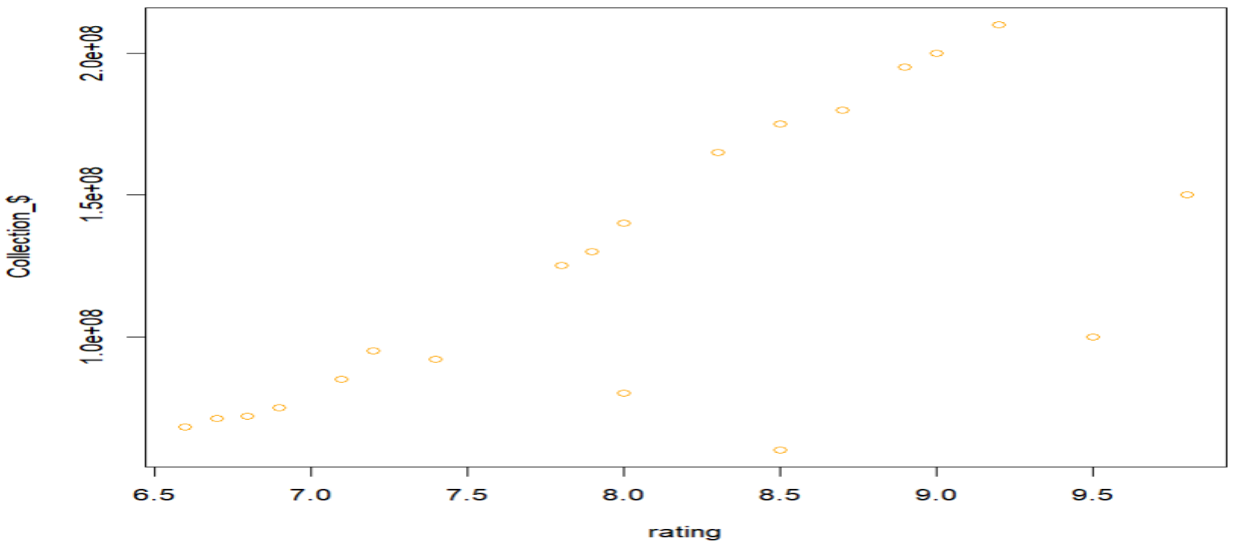
It will give the total count of movies in particular category .



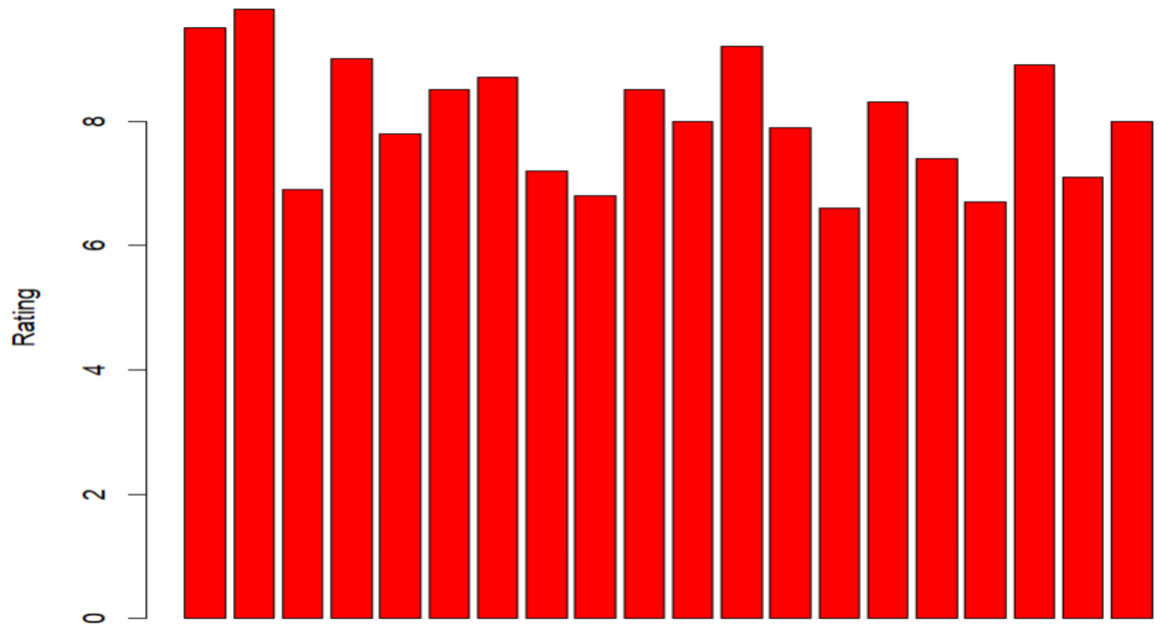
```
> movies%>%group_by(categories)%>%summarise(n=n())
# A tibble: 4 x 2
  categories     n
  <chr>      <int>
1 Action         6
2 Cartoon        7
3 Comedy         4
4 Horror         3
> 
```

# Data Visualization

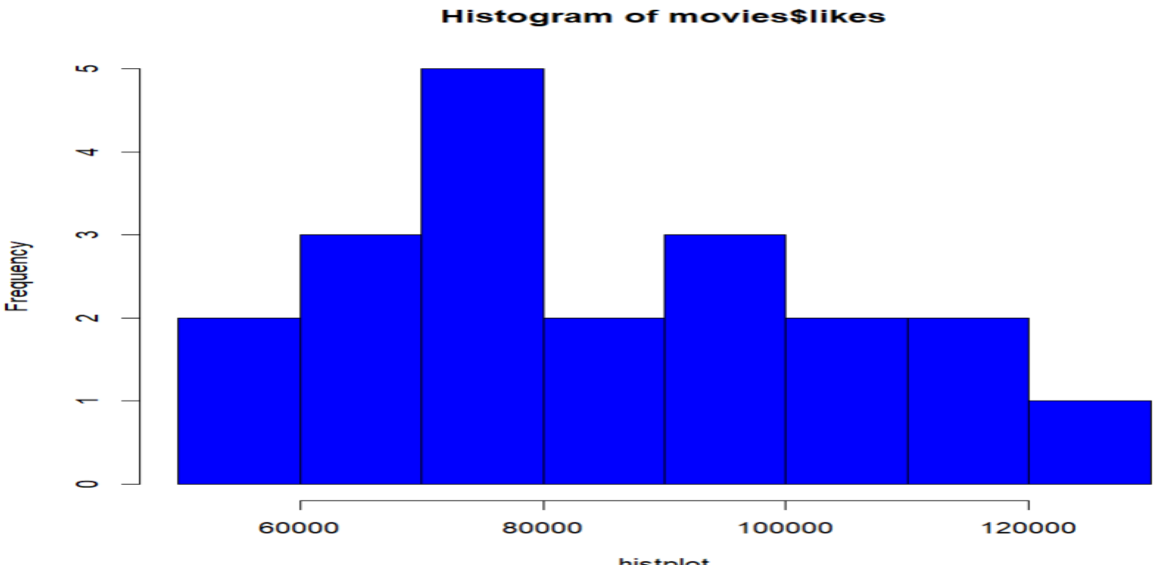
```
#scatter_plot

plot(movies$rating,movies$total_collection,xlab = "rating",ylab = "Collection_$",col='orange')
```



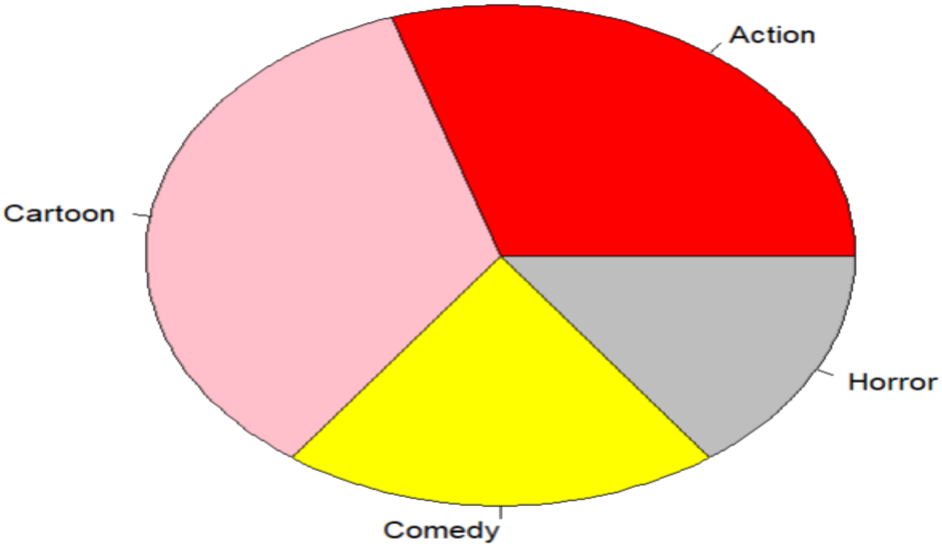
```
#Bar plot

barplot(movies$rating,xlab = "names",ylab = "Rating",col='red')
```



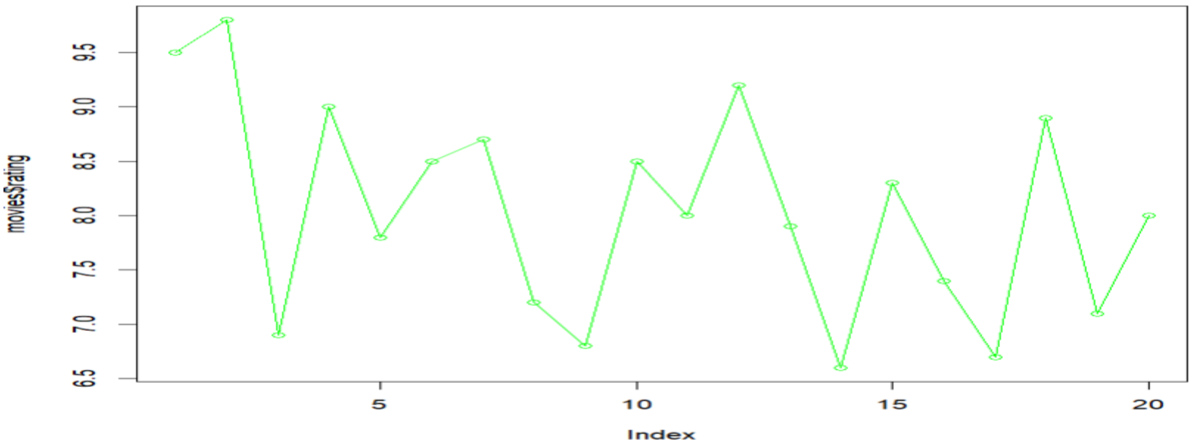
```
#Histogram

hist(movies$likes,xlab = "histplot",col = 'blue',border = 'black')
```



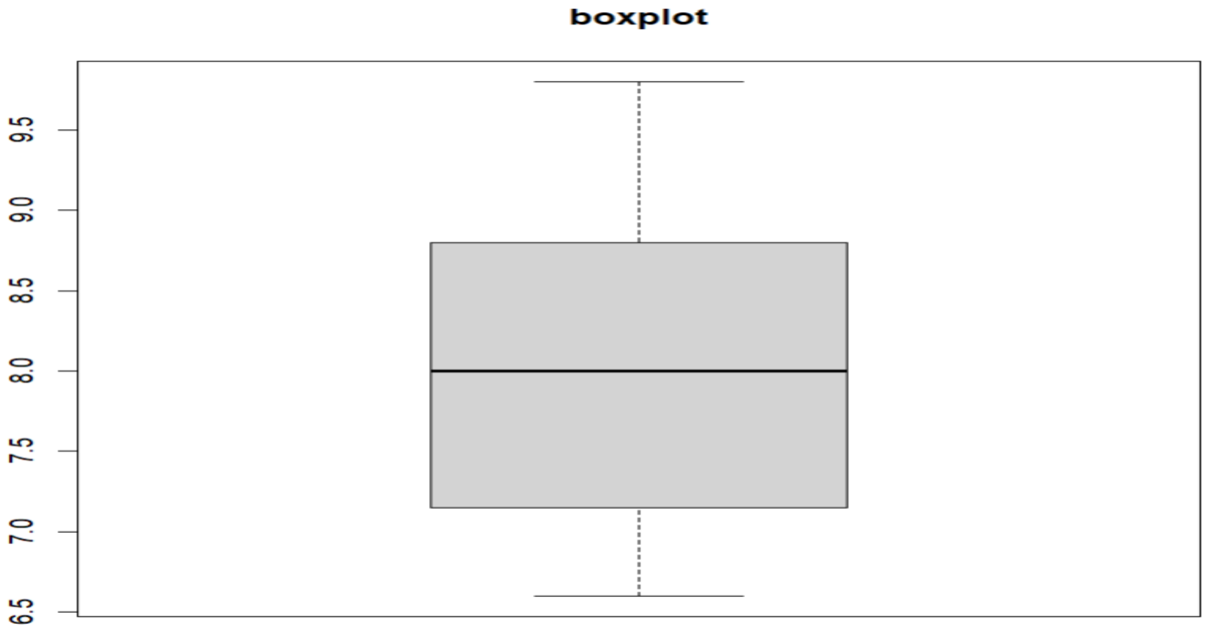
```
#Pie Chart

df2=movies%>%group_by(categories)%>%summarise(n=n())

df2

v=c("red","pink","yellow","gray")

pie(df2$n,labels=df2$categories,col=v)
```



```
#Line chart
plot(movies$rating,type = "o",col='green')
```



```
#Box plot
boxplot(movies$rating,main='boxplot')
```


boxplot