Assignment Week-12

Student Name : Sachid Deshmukh

Date : 11/14/2018

- GitHub Location for rmd file
- GitHub Location for pdf file
- RPubs location of published file
- GitHub location for tb.sql file
- GitHub location for tb.csv file

1] Pre-requisites

Following are the pre-requisites for running this script

- Packages - mongolite, RMySQL, dplyr, reshape2, ggplot2
- Local installation for MySQL DB
- Local installation for Mongo DB
- Create database called movie catalogue in MySQL db
- create table tb and populate data. Please use tb.csv and tb.sql for data population

2] Library Initializations

Let's initialize necessary R library for this task

```
library(mongolite)
library(RMySQL)
```

```
## Loading required package: DBI
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(reshape2)
library(ggplot2)
```

3] Open connection to Mongo DB

Let's create database connection to Mongo DB

```
tb.mongo <- mongo("tb")
tb.mongo$drop()
```

4] Open connection to MySQL DB

Let's create database connection to MySql DB and load data from table tb to R dataframe.

```
con <- dbConnect(RMySQL::MySQL(), dbname = "moviecatalogue", username = "sachid", password="DatabasePas
tb.mysql <- dbGetQuery(con, "SELECT * FROM tb")
dim(tb.mysql)
```

```
## [1] 3800    6
```

5] Insert data into Mongo DB

Insert data from r dataframe to Mongo DB

```
tb.mongo.df = tb.mongo$insert(tb.mysql)
tb.mongo.df = tb.mongo$find()
dbDisconnect(con)
```

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

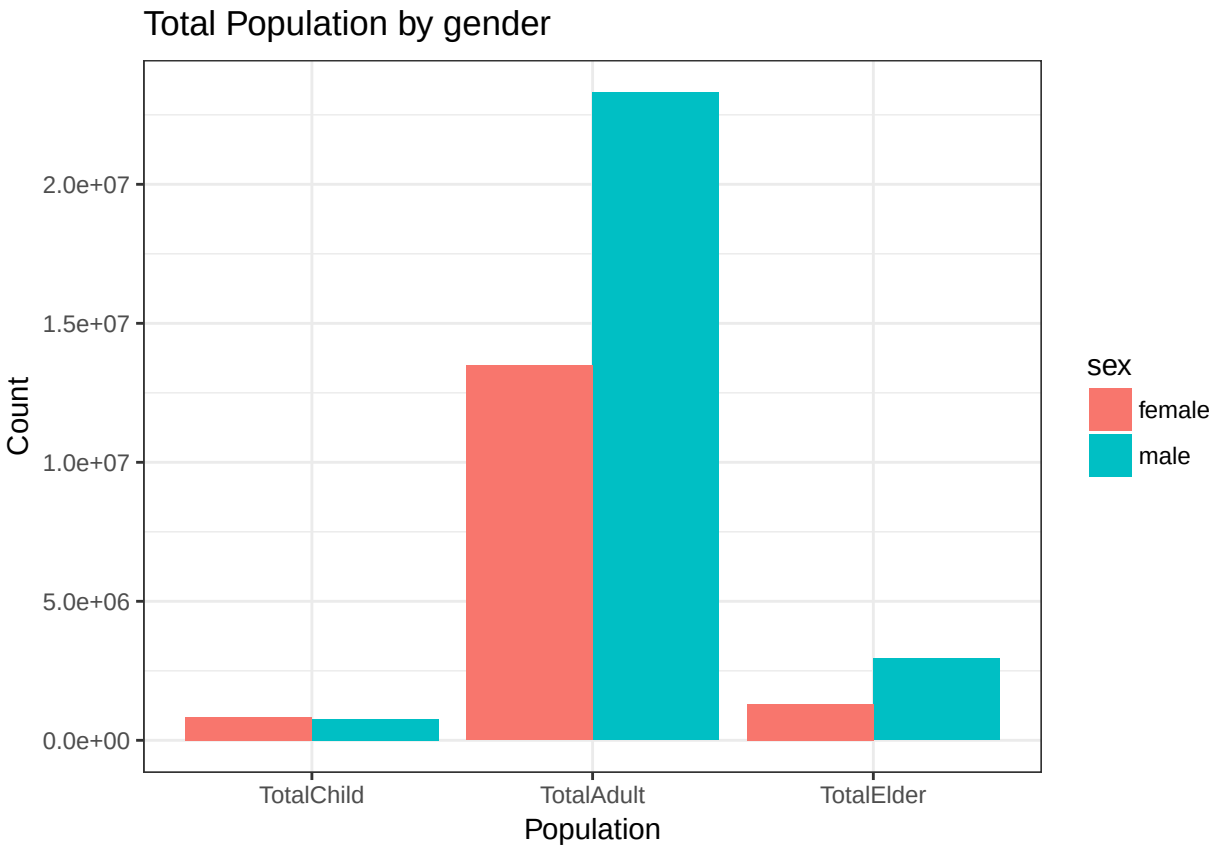
6] Plot bar chart for Child, Adult and Elderly population according to gender

Let's use the dataframe loaded form Mongo DB for plotting bar chart. We are plotting poulation count in three categories (Child, Adult and Elders) by gender

```
tb.mongo.df = na.omit(tb.mongo.df)
tb.mongo.agg = tb.mongo.df %>% group_by(sex) %>% summarize(TotalChild = sum(child), TotalAdult = sum(adu

tb.mongo.melt = melt(tb.mongo.agg, id=c("sex"))

p <-ggplot(tb.mongo.melt, aes(variable, value))
p +geom_bar(stat = "identity", aes(fill = sex), position = "dodge") +
  xlab("Population") + ylab("Count") +
  ggtitle("Total Population by gender") +
  theme_bw()
```



From the bar chart we can see that we were successfully able to load migrated data from Mongo DB and visualize it in the Bar Plot

7] Advantages and disadvantages of storing the data in a relational database vs. your NoSQL database.

- SQL databases are relational databases while NoSQL databases are graphical or network datases.
- NoSQL databases strength lie on data that are naturally heirarchical (such as organization charts) while SQL databases are better in performing aggregate functions such as summation and averaging of column values.
- SQL databases are vertically scalable while NoSQL databases are horizontally scalable. This means that we can increase the scale of SQL databases by increasing CPU speed while we can increase the scale of NoSQL databases by increasing the number of servers.
- SQL databases are the preferred databases for applications with complex queries while NoSQL databases are the preferred databases for big data.
- SQL databases emphasizes on ACID properties (Atomicity, Consistency, Isolation and Durability) whereas the NoSQL database follows the Brewers CAP theorem (Consistency, Availability and Partition tolerance).
- SQL databases are better for applications with heavy transaction processing because it provides better data integrity than NoSQL databases.