Assignment 12

Samuel I Kigamba

November 25, 2019

Instructions

For this assignment, you should take information from a relational database and migrate it to a NoSQL database of your own choosing.

For the relational database, you might use the flights database, the tb database, the "data skills" database your team created for Project 3, or another database of your own choosing or creation.

For the NoSQL database, you may use MongoDB (which we introduced in week 7), Neo4j, or another NoSQL database of your choosing.

Your migration process needs to be reproducible. R code is encouraged, but not required. You should also briefly describe the advantages and disadvantages of storing the data in a relational database vs. your NoSQL database.

```r
#install.packages("RMySQL")
#install.packages('ggmap')
```

```r
library(RMySQL)
```

```
## Warning: package 'RMySQL' was built under R version 3.5.3
```

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

```
## Warning: package 'DBI' was built under R version 3.5.3
```

```r
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.5.3
```

```r
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.5.3
```

```
##
## 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
```

```r
library(dbplyr)
```

```
## Warning: package 'dbplyr' was built under R version 3.5.3
```

```
##
## Attaching package: 'dbplyr'
```

```
## The following objects are masked from 'package:dplyr':
##
##     ident, sql
```

```r
library(knitr)
```

```
## Warning: package 'knitr' was built under R version 3.5.3
```

```r
library(maps)
```

```
## Warning: package 'maps' was built under R version 3.5.3
```

```r
library(ggmap)
```

```
## Warning: package 'ggmap' was built under R version 3.5.3
```

```
## Google's Terms of Service: https://cloud.google.com/maps-platform/terms/.
```

```
## Please cite ggmap if you use it! See citation("ggmap") for details.
```

```r
library(mongolite)
```

```
## Warning: package 'mongolite' was built under R version 3.5.3
```

```r
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:base':
##
##     date
```

```r
library(gridExtra)
```

```
## Warning: package 'gridExtra' was built under R version 3.5.3
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

Connecting to flights database through RMysql

```
password = 'Sammiggy2030^'

#connect RMySQL flights database

mydb = dbConnect(MySQL(), user='root', password = password, dbname='flights', host='localhost') #Use th

dbSendQuery(mydb,'USE flights;')
```

```
## <MySQLResult:256,0,0>
```

There are five tables under the flights database and they include airlines, airports, flights, planes, weather.

```
dbListTables(mydb)
```

```
## [1] "airlines" "airports" "flights"  "planes"   "weather"
```

[1] "airlines" "airports" "flights" "planes" "weather"

Load the database into R

```
flightdetails <- dbGetQuery(mydb, "SELECT * FROM flights")
planes <- dbGetQuery(mydb, "SELECT * FROM planes")
weather <- dbGetQuery(mydb, "SELECT * FROM weather")
airlines <- dbGetQuery(mydb, "SELECT * FROM airlines")
airports <- dbGetQuery(mydb, "SELECT * FROM airports")
```

```
head(airports, 2)
```

```
##   faa                        name      lat       lon  alt tz dst
## 1 04G            Lansdowne Airport 41.13047 -80.61958 1044 -5   A
## 2 06A Moton Field Municipal Airport 32.46057 -85.68003  264 -5   A
```

```
head(airlines, 2)
```

```
##   carrier                  name
## 1      9E       Endeavor Air Inc.\r
## 2      AA American Airlines Inc.\r
```

```
head(flightdetails, 2)
```

```
##   year month day dep_time dep_delay arr_time arr_delay carrier tailnum
## 1 2013     1   1      517         2      830        11      UA  N14228
## 2 2013     1   1      533         4      850        20      UA  N24211
##   flight origin dest air_time distance hour minute
## 1   1545    EWR  IAH      227     1400    5     17
## 2   1714    LGA  IAH      227     1416    5     33
```

```r
head(planes, 2)
```

```
##   tailnum year                          type       manufacturer     model engines
## 1  N10156 2004 Fixed wing multi engine          EMBRAER EMB-145XR        2
## 2  N102UW 1998 Fixed wing multi engine AIRBUS INDUSTRIE  A320-214        2
##   seats speed     engine
## 1    55    NA Turbo-fan
## 2   182    NA Turbo-fan
```

```r
head(weather, 2)
```

```
##   origin year month day hour  temp  dewp humid wind_dir wind_speed
## 1    EWR 2013     1   1    0 37.04 21.92 53.97      230   10.35702
## 2    EWR 2013     1   1    1 37.04 21.92 53.97      230   13.80936
##   wind_gust precip pressure visib
## 1  11.91865      0   1013.9    10
## 2  15.89154      0   1013.0    10
```

```r
#Close the connection to MySQL.
#dbDisconnect(mydb)
```

Loading the data into my non relational (nonsql) database - MongoDB

```r
library(devtools)
```

```
## Warning: package 'devtools' was built under R version 3.5.3
```

```
## Loading required package: usethis
```

```
## Warning: package 'usethis' was built under R version 3.5.3
```

```r
install_github(repo = "mongosoup/rmongodb")
```

```
## Skipping install of 'rmongodb' from a github remote, the SHA1 (8eb2bca2) has not changed since last
##   Use `force = TRUE` to force installation
```

```r
library(rmongodb)
# connect to MongoDB
mongo = mongo.create(host = "localhost")
mongo.is.connected(mongo)
```

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

```r
library(mongolite)
```

Create collections for all the tables and load the various tables into their respective collection

```r
flights <- mongo(collection = "Airlines", db = "flights")
flights$insert(airlines)
```

```
## List of 5
##  $ nInserted  : num 16
##  $ nMatched   : num 0
##  $ nRemoved   : num 0
##  $ nUpserted  : num 0
##  $ writeErrors: list()
```

```r
flights <- mongo(collection = "Airports", db = "flights")
flights$insert(airports)
```

```
## List of 5
##  $ nInserted  : num 1397
##  $ nMatched   : num 0
##  $ nRemoved   : num 0
##  $ nUpserted  : num 0
##  $ writeErrors: list()
```

```r
flights <- mongo(collection = "Flightdetails", db = "flights")
flights$insert(flightdetails)
```

```
## List of 5
##  $ nInserted  : num 336776
##  $ nMatched   : num 0
##  $ nRemoved   : num 0
##  $ nUpserted  : num 0
##  $ writeErrors: list()
```

```r
flights <- mongo(collection = "Planes", db = "flights")
flights$insert(planes)
```

```
## List of 5
##  $ nInserted  : num 3322
##  $ nMatched   : num 0
##  $ nRemoved   : num 0
##  $ nUpserted  : num 0
##  $ writeErrors: list()
```

```r
flights <- mongo(collection = "Weather", db = "flights")
flights$insert(weather)
```

```
## List of 5
##  $ nInserted  : num 8719
##  $ nMatched   : num 0
##  $ nRemoved   : num 0
##  $ nUpserted  : num 0
##  $ writeErrors: list()
```

Querying the data

```
flights$count()
```

```
## [1] 43595
```

Relational vs Non relational Reasons to use a Relational database

The main advantages of non relational database vs relational databases are:

a) The use of cloud computing and storage.
b) The rapid technological developments and the ease of evolution.
c) Large storage spaces for unstructured data.

The greatest disadvantages include:

a) non standardization.
b) non consistent query language unlike relational databases.