desafio9

Luciano Floriano

2025-09-30

```
library(readr)
library(RSQLite)
library(DBI)
library(dplyr)
```

```
##
## Anexando pacote: 'dplyr'
```

```
## Os seguintes objetos são mascarados por 'package:stats':
##
##     filter, lag
```

```
## Os seguintes objetos são mascarados por 'package:base':
##
##     intersect, setdiff, setequal, union
```

1. Crie um arquivo de banco de dados em SQLite chamado voos.sqlite3. (Dica: o comando dbConnect() se conecta num banco de dados se o arquivo apontado existir ou cria um novo, caso o arquivo não exista.)

```
con <- dbConnect(RSQLite::SQLite(), "voos.sqlite3")
con
```

```
## <SQLiteConnection>
##   Path: C:\Users\luflo\OneDrive\Área de Trabalho\ME315\Desafio_09\voos.sqlite3
##   Extensions: TRUE
```

2. Leia os arquivos airlines.csv e airports.csv. Deposite o conteúdo de cada um destes arquivos nas tabelas, respectivamente, airlines e airports. Utilize o comando dbWriteTable() para isso.

```
airlines_file <- "airlines.csv"
airports_file <- "airports.csv"

# Ler os CSVs
airlines <- read.csv(airlines_file, stringsAsFactors = FALSE)
airports <- read.csv(airports_file, stringsAsFactors = FALSE)

# Depositar os dados no banco (criando tabelas airlines e airports)
dbWriteTable(con, "airlines", airlines, overwrite = TRUE)
```

```r
dbWriteTable(con, "airports", airports, overwrite = TRUE)

# Conferir as tabelas criadas
dbListTables(con)
```

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

```r
# Mostrar as primeiras linhas de cada tabela
head(dbReadTable(con, "airlines"))
```

```
##   IATA_CODE                AIRLINE
## 1        UA  United Air Lines Inc.
## 2        AA American Airlines Inc.
## 3        US        US Airways Inc.
## 4        F9 Frontier Airlines Inc.
## 5        B6        JetBlue Airways
## 6        OO  Skywest Airlines Inc.
```

```r
head(dbReadTable(con, "airports"))
```

```
##   IATA_CODE                              AIRPORT        CITY STATE COUNTRY
## 1       ABE Lehigh Valley International Airport   Allentown    PA     USA
## 2       ABI            Abilene Regional Airport     Abilene    TX     USA
## 3       ABQ   Albuquerque International Sunport Albuquerque    NM     USA
## 4       ABR           Aberdeen Regional Airport    Aberdeen    SD     USA
## 5       ABY  Southwest Georgia Regional Airport      Albany    GA     USA
## 6       ACK          Nantucket Memorial Airport   Nantucket    MA     USA
##   LATITUDE  LONGITUDE
## 1 40.65236  -75.44040
## 2 32.41132  -99.68190
## 3 35.04022 -106.60919
## 4 45.44906  -98.42183
## 5 31.53552  -84.19447
## 6 41.25305  -70.06018
```

3. Crie uma função chamada lerDados contendo 2 argumentos, input e pos. A função deve apresentar ao usuário uma mensagem de progresso da leitura do arquivo flights.csv (utilize o comando message()), aos moldes do apresentado abaixo. A função deve salvar apenas os vôos que partiram ou chegaram aos seguintes aeroportos BWI, MIA, SEA, SFO e JFK, numa tabela chamada flights. Observe que a função não deve retornar nada para o usuário, deve apenas gravar a tabela obtida do chunk no banco de dados. (Dica: utilize o comando dbWriteTable() e estude como o argumento append deve ser utilizado para permitir que os chunks intermediários sejam adicionados ao fim da tabela.)

```r
# Função para processar cada chunk
lerDados <- function(input, pos) {
  # Mensagem de progresso
  message("Leitura atingiu a linha ", pos)

  # Aeroportos de interesse
  aeroportos <- c("BWI", "MIA", "SEA", "SFO", "JFK")
```

```r
  # Filtrar voos (usando os nomes corretos das colunas)
  filtrados <- input %>%
    filter(ORIGIN_AIRPORT %in% aeroportos | DESTINATION_AIRPORT %in% aeroportos)

  # Gravar no banco
  dbWriteTable(con,
               name = "flights",
               value = filtrados,
               append = TRUE,
               overwrite = FALSE)

  invisible(NULL)
}

head(dbReadTable(con, "flights"), 10)
```

```
##    YEAR MONTH DAY DAY_OF_WEEK AIRLINE FLIGHT_NUMBER TAIL_NUMBER ORIGIN_AIRPORT
## 1  2015     1   1           4      AS            98      N407AS            ANC
## 2  2015     1   1           4      US           840      N171US            SFO
## 3  2015     1   1           4      AA           258      N3HYAA            LAX
## 4  2015     1   1           4      AS           135      N527AS            SEA
## 5  2015     1   1           4      DL           806      N3730B            SFO
## 6  2015     1   1           4      AA          1112      N3LAAA            SFO
## 7  2015     1   1           4      AA          1674      N853AA            LAS
## 8  2015     1   1           4      DL          2440      N651DL            SEA
## 9  2015     1   1           4      AS           108      N309AS            ANC
## 10 2015     1   1           4      DL          1560      N3743H            ANC
##    DESTINATION_AIRPORT SCHEDULED_DEPARTURE DEPARTURE_TIME DEPARTURE_DELAY
## 1                  SEA                0005           2354            -11
## 2                  CLT                0020           0018             -2
## 3                  MIA                0020           0015             -5
## 4                  ANC                0025           0024             -1
## 5                  MSP                0025           0020             -5
## 6                  DFW                0030           0019            -11
## 7                  MIA                0035           0027             -8
## 8                  MSP                0040           0039             -1
## 9                  SEA                0045           0041             -4
## 10                 SEA                0045           0031            -14
##    TAXI_OUT WHEELS_OFF SCHEDULED_TIME ELAPSED_TIME AIR_TIME DISTANCE WHEELS_ON
## 1        21       0015            205          194      169     1448      0404
## 2        16       0034            286          293      266     2296      0800
## 3        15       0030            285          281      258     2342      0748
## 4        11       0035            235          215      199     1448      0254
## 5        18       0038            217          230      206     1589      0604
## 6        17       0036            195          193      173     1464      0529
## 7        21       0048            268          266      238     2174      0746
## 8        28       0107            189          198      166     1399      0553
## 9        17       0058            204          194      173     1448      0451
## 10       25       0056            210          200      171     1448      0447
##    TAXI_IN SCHEDULED_ARRIVAL ARRIVAL_TIME ARRIVAL_DELAY DIVERTED CANCELLED
## 1        4              0430         0408           -22        0         0
## 2       11              0806         0811             5        0         0
## 3        8              0805         0756            -9        0         0
```

```
## 4        5              0320         0259           -21        0         0
## 5        6              0602         0610             8        0         0
## 6        3              0545         0532           -13        0         0
## 7        7              0803         0753           -10        0         0
## 8        4              0549         0557             8        0         0
## 9        4              0509         0455           -14        0         0
## 10       4              0515         0451           -24        0         0
##    CANCELLATION_REASON AIR_SYSTEM_DELAY SECURITY_DELAY AIRLINE_DELAY
## 1                 <NA>               NA             NA            NA
## 2                 <NA>               NA             NA            NA
## 3                 <NA>               NA             NA            NA
## 4                 <NA>               NA             NA            NA
## 5                 <NA>               NA             NA            NA
## 6                 <NA>               NA             NA            NA
## 7                 <NA>               NA             NA            NA
## 8                 <NA>               NA             NA            NA
## 9                 <NA>               NA             NA            NA
## 10                <NA>               NA             NA            NA
##    LATE_AIRCRAFT_DELAY WEATHER_DELAY
## 1                   NA            NA
## 2                   NA            NA
## 3                   NA            NA
## 4                   NA            NA
## 5                   NA            NA
## 6                   NA            NA
## 7                   NA            NA
## 8                   NA            NA
## 9                   NA            NA
## 10                  NA            NA
```

```
# Exemplo de leitura a partir do ZIP
zip_file <- "flights.csv.zip"

read_csv_chunked(
  file = unz(zip_file, "flights.csv"),
  callback = SideEffectChunkCallback$new(lerDados),
  chunk_size = 100000
)
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_double(),
##   AIRLINE = col_character(),
##   TAIL_NUMBER = col_character(),
##   ORIGIN_AIRPORT = col_character(),
##   DESTINATION_AIRPORT = col_character(),
##   SCHEDULED_DEPARTURE = col_character(),
##   DEPARTURE_TIME = col_character(),
##   WHEELS_OFF = col_character(),
##   WHEELS_ON = col_character(),
##   SCHEDULED_ARRIVAL = col_character(),
##   ARRIVAL_TIME = col_character(),
##   CANCELLATION_REASON = col_character()
```

```
## )
## i Use ‘spec()‘ for the full column specifications.

## Leitura atingiu a linha 1

## Leitura atingiu a linha 100001

## Leitura atingiu a linha 200001

## Leitura atingiu a linha 300001

## Leitura atingiu a linha 400001

## Leitura atingiu a linha 500001

## Leitura atingiu a linha 600001

## Leitura atingiu a linha 700001

## Leitura atingiu a linha 800001

## Leitura atingiu a linha 900001

## Leitura atingiu a linha 1000001

## Leitura atingiu a linha 1100001

## Leitura atingiu a linha 1200001

## Leitura atingiu a linha 1300001

## Leitura atingiu a linha 1400001

## Leitura atingiu a linha 1500001

## Leitura atingiu a linha 1600001

## Leitura atingiu a linha 1700001

## Leitura atingiu a linha 1800001

## Leitura atingiu a linha 1900001

## Leitura atingiu a linha 2000001

## Leitura atingiu a linha 2100001

## Leitura atingiu a linha 2200001
```

```
## Leitura atingiu a linha 2300001

## Leitura atingiu a linha 2400001

## Leitura atingiu a linha 2500001

## Leitura atingiu a linha 2600001

## Leitura atingiu a linha 2700001

## Leitura atingiu a linha 2800001

## Leitura atingiu a linha 2900001

## Leitura atingiu a linha 3000001

## Leitura atingiu a linha 3100001

## Leitura atingiu a linha 3200001

## Leitura atingiu a linha 3300001

## Leitura atingiu a linha 3400001

## Leitura atingiu a linha 3500001

## Leitura atingiu a linha 3600001

## Leitura atingiu a linha 3700001

## Leitura atingiu a linha 3800001

## Leitura atingiu a linha 3900001

## Leitura atingiu a linha 4000001

## Leitura atingiu a linha 4100001

## Leitura atingiu a linha 4200001

## Leitura atingiu a linha 4300001

## Leitura atingiu a linha 4400001

## Leitura atingiu a linha 4500001

## Leitura atingiu a linha 4600001
```

```
## Leitura atingiu a linha 4700001

## Leitura atingiu a linha 4800001

## Leitura atingiu a linha 4900001

## Leitura atingiu a linha 5000001

## Leitura atingiu a linha 5100001

## Leitura atingiu a linha 5200001

## Leitura atingiu a linha 5300001

## Leitura atingiu a linha 5400001

## Leitura atingiu a linha 5500001

## Leitura atingiu a linha 5600001

## Leitura atingiu a linha 5700001

## Leitura atingiu a linha 5800001

## NULL
```

4. Leia o arquivo flights.csv, restringindo-se às colunas YEAR, MONTH, DAY, AIRLINE, FLIGHT_NUMBER, ORIGIN_AIRPORT, DESTINATION_AIRPORT e ARRIVAL_DELAY, e aplique a função lerDados() criada acima. Observe, novamente, que a função lerDados() não retorna nada para o usuário. Por isso, a função de callback a ser utilizada é SideEffectChunkCallback$new(). Leia 100 mil registros por vez.

```
# Leitura em chunks de 100 mil registros, apenas colunas selecionadas
read_csv_chunked(
  file = unz(zip_file, "flights.csv"),
  callback = SideEffectChunkCallback$new(lerDados),
  chunk_size = 100000,
  col_types = cols_only(
    YEAR = col_integer(),
    MONTH = col_integer(),
    DAY = col_integer(),
    AIRLINE = col_character(),
    FLIGHT_NUMBER = col_integer(),
    ORIGIN_AIRPORT = col_character(),
    DESTINATION_AIRPORT = col_character(),
    ARRIVAL_DELAY = col_double()
  )
)
```

```
## Leitura atingiu a linha 1
```

```
## Leitura atingiu a linha 100001

## Leitura atingiu a linha 200001

## Leitura atingiu a linha 300001

## Leitura atingiu a linha 400001

## Leitura atingiu a linha 500001

## Leitura atingiu a linha 600001

## Leitura atingiu a linha 700001

## Leitura atingiu a linha 800001

## Leitura atingiu a linha 900001

## Leitura atingiu a linha 1000001

## Leitura atingiu a linha 1100001

## Leitura atingiu a linha 1200001

## Leitura atingiu a linha 1300001

## Leitura atingiu a linha 1400001

## Leitura atingiu a linha 1500001

## Leitura atingiu a linha 1600001

## Leitura atingiu a linha 1700001

## Leitura atingiu a linha 1800001

## Leitura atingiu a linha 1900001

## Leitura atingiu a linha 2000001

## Leitura atingiu a linha 2100001

## Leitura atingiu a linha 2200001

## Leitura atingiu a linha 2300001

## Leitura atingiu a linha 2400001
```

```
## Leitura atingiu a linha 2500001

## Leitura atingiu a linha 2600001

## Leitura atingiu a linha 2700001

## Leitura atingiu a linha 2800001

## Leitura atingiu a linha 2900001

## Leitura atingiu a linha 3000001

## Leitura atingiu a linha 3100001

## Leitura atingiu a linha 3200001

## Leitura atingiu a linha 3300001

## Leitura atingiu a linha 3400001

## Leitura atingiu a linha 3500001

## Leitura atingiu a linha 3600001

## Leitura atingiu a linha 3700001

## Leitura atingiu a linha 3800001

## Leitura atingiu a linha 3900001

## Leitura atingiu a linha 4000001

## Leitura atingiu a linha 4100001

## Leitura atingiu a linha 4200001

## Leitura atingiu a linha 4300001

## Leitura atingiu a linha 4400001

## Leitura atingiu a linha 4500001

## Leitura atingiu a linha 4600001

## Leitura atingiu a linha 4700001

## Leitura atingiu a linha 4800001
```

```
## Leitura atingiu a linha 4900001

## Leitura atingiu a linha 5000001

## Leitura atingiu a linha 5100001

## Leitura atingiu a linha 5200001

## Leitura atingiu a linha 5300001

## Leitura atingiu a linha 5400001

## Leitura atingiu a linha 5500001

## Leitura atingiu a linha 5600001

## Leitura atingiu a linha 5700001

## Leitura atingiu a linha 5800001

## NULL
```

5. Acesse o banco de dados e, por meio de uma chamada em SQL, apresente o tempo médio de atraso de chegada por aeroporto de destino, a sigla do aeroporto, o nome completo do aeroporto e o nome completo da companhia aérea. Ordene o resultado (na mesma chamada de SQL) por ordem decrescente deste atraso médio (i.e., o primeiro registro deve ser o aeroporto que tem o maior tempo de atraso na chegada). Atente para o fato de que o mesmo nome de coluna pode acontecer em diferentes tabelas.

```
resultado <- dbGetQuery(con, "
  SELECT
      f.DESTINATION_AIRPORT AS destino,
      a.AIRPORT AS nome_aeroporto,
      al.AIRLINE AS nome_companhia,
      AVG(f.ARRIVAL_DELAY) AS atraso_medio
  FROM flights f
  JOIN airports a
      ON f.DESTINATION_AIRPORT = a.IATA_CODE
  JOIN airlines al
      ON f.AIRLINE = al.IATA_CODE
  GROUP BY f.DESTINATION_AIRPORT, a.AIRPORT, al.AIRLINE
  ORDER BY atraso_medio DESC
")

head(resultado, 10)  # mostra os 10 primeiros
```

```
##    destino                                  nome_aeroporto
## 1      DTW                    Detroit Metropolitan Airport
## 2      JAX              Jacksonville International Airport
## 3      IAH            George Bush Intercontinental Airport
## 4      MSN                    Dane County Regional Airport
```

```
## 5      IAH             George Bush Intercontinental Airport
## 6      ATL Hartsfield-Jackson Atlanta International Airport
## 7      LGA          LaGuardia Airport (Marine Air Terminal)
## 8      ORD             Chicago O'Hare International Airport
## 9      MTJ                          Montrose Regional Airport
## 10     MSP     Minneapolis-Saint Paul International Airport
##            nome_companhia atraso_medio
## 1   Skywest Airlines Inc.    143.00000
## 2    Delta Air Lines Inc.     41.00000
## 3         US Airways Inc.     36.20000
## 4   United Air Lines Inc.     34.50000
## 5   Skywest Airlines Inc.     27.71429
## 6        Spirit Air Lines     25.79268
## 7  Frontier Airlines Inc.     24.02069
## 8        Spirit Air Lines     22.45725
## 9   Skywest Airlines Inc.     22.44828
## 10       Spirit Air Lines     22.17614
```