

desafio 5

Desafio 5

```
#install.packages("RSQLite")
library(RSQLite)
```

Warning: pacote 'RSQLite' foi compilado no R versão 4.4.3

```
db = dbConnect(SQLite(), '//smb/ra277230/Documentos/me315/disco.db')
db
```

```
<SQLiteConnection>
  Path: \\smb\ra277230\Documentos\me315\disco.db
  Extensions: TRUE
dbListFields(db, 'albums')
```

```
dbListTables(db)
```

```
[1] "albums"           "artists"           "customers"          "employees"
[5] "genres"           "invoice_items"     "invoices"
"media_types"
[9] "playlist_track"   "playlists"         "sqlite_sequence"
"sqlite_stat1"
[13] "tracks"
```

```
dbListFields(db, 'albums')
```

```
[1] "AlbumId" "Title" "ArtistId"
```

- O comando `dbGetQuery` consulta o banco de dados, extrai os resultados solicitado e os retorna ao R;

- A sintaxe é `dbGetQuery(<conexão>, <chamada sql>);`
- Extraia da tabela `albums` todas as colunas e todas as linhas e armazene-os em um objeto chamado `album_db`:

```
album_db = dbGetQuery(db, 'SELECT * FROM albums')
head(album_db)
```

	AlbumId	Title	ArtistId
1	1	For Those About To Rock We Salute You	1
2	2	Balls to the Wall	2
3	3	Restless and Wild	2
4	4	Let There Be Rock	1
5	5	Big Ones	3
6	6	Jagged Little Pill	4

```
tracks = dbGetQuery(db, 'SELECT * FROM tracks')
```

```
dim(album_db)
```

```
[1] 347 3
```

O comando `SELECT` é o comando mais utilizado em SQL; É, também, um dos comandos mais complexos; Pode ser combinado com uma série de argumentos:

- `ORDER BY`: ordenar o resultado;
- `DISTINCT`: pesquisar por linhas únicas;
- `WHERE`: filtrar linhas;
- `LIMIT`: restringir o número de linhas do resultado;
- `INNER JOIN/LEFT JOIN`: consultar múltiplas tabelas;
- `GROUP BY`: agrupar e aplicar funções para agregação nos grupos;
- `HAVING`: filtrar em grupos;

Selecione as colunas `trackid`, `name`, `composer` e `unitprice`. Ordene o objeto resultante por `unitprice`.

```
sql = 'SELECT trackid, name, composer, unitprice FROM tracks ORDER BY
unitprice'
res = dbGetQuery(db, sql)
```

```
head(res)
```

	TrackId	Name
1	1	For Those About To Rock (We Salute You)
2	2	Balls to the Wall
3	3	Fast As a Shark
4	4	Restless and Wild
5	5	Princess of the Dawn
6	6	Put The Finger On You

```
Composer
```

1	Angus Young, Malcolm Young, Brian Johnson
2	<NA>
3	F. Baltes, S. Kaufman, U. Dirkschneider & W. Hoffman
4	F. Baltes, R.A. Smith-Diesel, S. Kaufman, U. Dirkschneider & W. Hoffman
5	Deaffy & R.A. Smith-Diesel
6	Angus Young, Malcolm Young, Brian Johnson

```
UnitPrice
```

1	0.99
2	0.99
3	0.99
4	0.99
5	0.99
6	0.99

Quais são as cidades de todos os clientes, ordenadas por nome de cidade?

```
sql = 'SELECT city FROM customers ORDER BY city'
ex3a = dbGetQuery(db, sql)
head(ex3a)
```

	City
1	Amsterdam
2	Bangalore
3	Berlin
4	Berlin
5	Bordeaux
6	Boston

```
dim(ex3a)
```

```
[1] 59 1
```

Quais são as cidades de todos os clientes, ordenadas por nome de cidade?

```
sql = 'SELECT DISTINCT city FROM customers ORDER BY city'
ex3b = dbGetQuery(db, sql)
head(ex3b)
```

	City
1	Amsterdam
2	Bangalore
3	Berlin
4	Bordeaux
5	Boston
6	Brasília

```
dim(ex3b)
```

```
[1] 53 1
```

Quais são todas as músicas do álbum 1?

```
ex4 = dbGetQuery(db, 'SELECT name, albumid FROM tracks WHERE
albumid=1')
head(ex4)
```

	Name	AlbumId
1	For Those About To Rock (We Salute You)	1
2	Put The Finger On You	1
3	Let's Get It Up	1
4	Inject The Venom	1
5	Snowballed	1
6	Evil Walks	1

```
dim(ex4)
```

```
[1] 10 2
```

Limitando-se a 5 registros, após ordenação por nome, quais são os nomes, ID de álbum e ID de mídia de músicas com mídias de tipo 1 ou 2?

```
sql = paste('SELECT name, albumid, mediatypeid FROM tracks',  
            'WHERE mediatypeid IN (1, 2)',  
            'ORDER BY name LIMIT 5')  
ex4b = dbGetQuery(db, sql)  
head(ex4b)
```

	Name	AlbumId
1	"40"	239
2	"Eine Kleine Nachtmusik" Serenade In G, K. 525: I. Allegro	281
3	#1 Zero	11
4	#9 Dream	255
5	'Round Midnight	48

	MediaTypeId
1	1
2	2
3	1
4	2
5	1

```
dim(ex4b)
```

```
[1] 5 3
```

SelecãodeSelecão Quais são as músicas e identificadores de faixa e álbum produzidas pelo artista que identificador 12?

```
sql = paste('SELECT trackid, name, albumid FROM tracks',  
            'WHERE albumid IN',  
            '(SELECT albumid FROM albums WHERE artistid==12)')  
ex5 = dbGetQuery(db, sql)  
head(ex5)
```

	TrackId	Name	AlbumId
1	149	Black Sabbath	16
2	150	The Wizard	16
3	151	Behind The Wall Of Sleep	16
4	152	N.I.B.	16
5	153	Evil Woman	16
6	154	Sleeping Village	16

```
dim(ex5)
```

```
[1] 17 3
```

Quais são as faixas cujos nomes começam com qualquer caracter seguido de 'ere' e terminam com qualquer expressão?

```
sql = "SELECT trackid, name FROM tracks WHERE name GLOB '?ere*'"
ex5 = dbGetQuery(db, sql)[1:5,]
head(ex5)
```

	TrackId	Name
1	324	Pererê
2	1132	Serenity
3	1452	Were Do We Go From Here
4	1740	Sereia
5	2198	Jeremy

```
dim(ex5)
```

```
[1] 5 2
```

Quais são as faixas cujos nomes possuem algum dígito?

```
sql = "SELECT trackid, name FROM tracks WHERE name GLOB '*[0-9]*'"
ex6 = dbGetQuery(db, sql)[1:5,]
head(ex6)
```

	TrackId	Name
1	109	#1 Zero

2	122	20 Flight Rock
3	132	13 Years Of Grief
4	343	Communication Breakdown(2)
5	347	Communication Breakdown(3)

```
dim(ex6)
```

```
[1] 5 2
```

Quantas faixas por disco?

```
sql = 'SELECT albumid, COUNT(trackid) FROM tracks GROUP BY albumid'
ex7 = dbGetQuery(db, sql)[1:5,]
head(ex7)
```

	AlbumId	COUNT(trackid)
1	1	10
2	2	1
3	3	3
4	4	8
5	5	15

```
dim(ex7)
```

```
[1] 5 2
```

Quantas faixas por disco para o album 1?

```
sql = paste('SELECT albumid, COUNT(trackid)',
            'FROM tracks GROUP BY albumid',
            'HAVING albumid=1')
ex8 = dbGetQuery(db, sql)
head(ex8)
```

	AlbumId	COUNT(trackid)
1	1	10

Quais são os nomes de cada faixa com os respectivos títulos dos álbuns?

```
sql = paste('SELECT trackid, name, title FROM tracks',
            'INNER JOIN albums ON albums.albumid=tracks.albumid')
ex9 = dbGetQuery(db, sql)
head(ex9)
```

	TrackId	Name
1	1	For Those About To Rock (We Salute You)
2	6	Put The Finger On You
3	7	Let's Get It Up
4	8	Inject The Venom
5	9	Snowballed
6	10	Evil Walks

	Title
1	For Those About To Rock We Salute You
2	For Those About To Rock We Salute You
3	For Those About To Rock We Salute You
4	For Those About To Rock We Salute You
5	For Those About To Rock We Salute You
6	For Those About To Rock We Salute You

Funções de Agregação

- AVG: AVG([ALL | DISTINCT] expressao) calcula a média de todos os valores não-nulos ou dos valores distintos;
- COUNT: COUNT([ALL | DISTINCT] expressao) realiza a contagem de todos registros;
- MAX, MIN, SUM funcionam de maneira análoga às funções anteriores;

undefinedundefined