

Manipulação de Banco de Dados - Julia

Gabriella de Oliveira Argenton 255677

Criando um banco de dados

```
using Statistics
using DataFrames
using CSV

# Banco de dados principal
df = DataFrame(
    id      = 1:8,
    nome    = ["Ana", "Bruno", "Carla", "Diego", "Eva", "Felipe", "Gabi", "Hugo"],
    grupo   = ["A", "A", "A", "B", "B", "B", "B", "A"],
    idade   = [23, 35, 29, 41, 19, 33, 27, 30],
    sexo    = ["F", "M", "F", "M", "F", "M", "F", "M"],
    renda_2023 = [2500, 4800, 3200, 6000, 1800, 4200, 3100, 3500],
    renda_2024 = [2700, 5000, 3400, 6200, 2000, 4400, 3300, 3800]
)

# Tabela de bônus para o exemplo de left_join
df_bonus = DataFrame(
    id     = [2, 4, 8],
    bonus = [500, 1000, 700]
)
```

	id	bonus
	Int64	Int64
1	2	500
2	4	1000
3	8	700

Filter

```
# Pessoas do grupo A com renda_2024 > 3500
df_filtrado = filter(row -> row.grupo == "A" && row.renda_2024 > 3500, df)
```

	id	nome	grupo	idade	sexo	renda_2023	renda_2024
	Int64	String	String	Int64	String	Int64	Int64
1	2	Bruno	A	35	M	4800	5000
2	8	Hugo	A	30	M	3500	3800

Select

```
# Manter apenas id, nome, grupo e renda_2024
df_select = select(df, :id, :nome, :grupo, :renda_2024)
```

	id	nome	grupo	renda_2024
	Int64	String	String	Int64
1	1	Ana	A	2700
2	2	Bruno	A	5000
3	3	Carla	A	3400
4	4	Diego	B	6200
5	5	Eva	B	2000
6	6	Felipe	B	4400
7	7	Gabi	B	3300
8	8	Hugo	A	3800

Rename

```
# Renomear renda_2024 para renda_atual (sem alterar o original)
df_renomeado = rename(df, :renda_2024 => :renda_atual)

# Ou alterando o próprio df
# rename!(df, :renda_2024 => :renda_atual)
```

	id	nome	grupo	idade	sexo	renda_2023	renda_atual
	Int64	String	String	Int64	String	Int64	Int64
1	1	Ana	A	23	F	2500	2700
2	2	Bruno	A	35	M	4800	5000
3	3	Carla	A	29	F	3200	3400
4	4	Diego	B	41	M	6000	6200
5	5	Eva	B	19	F	1800	2000
6	6	Felipe	B	33	M	4200	4400
7	7	Gabi	B	27	F	3100	3300
8	8	Hugo	A	30	M	3500	3800

Mutate

```
# Criar renda_anual_2024 e faixa_idade
df_mut = transform(
  df,
  :renda_2024 => (x -> x .* 12) => :renda_anual_2024,
  :idade       => ByRow(x -> x < 30 ? "jovem" : "adulto") => :faixa_idade
)
```

	id	nome	grupo	idade	sexo	renda_2023	renda_2024	renda_anual_2024	
	Int64	String	String	Int64	String	Int64	Int64	Int64	
1	1	Ana	A	23	F	2500	2700	32400	...
2	2	Bruno	A	35	M	4800	5000	60000	...
3	3	Carla	A	29	F	3200	3400	40800	...
4	4	Diego	B	41	M	6000	6200	74400	...
5	5	Eva	B	19	F	1800	2000	24000	...
6	6	Felipe	B	33	M	4200	4400	52800	...
7	7	Gabi	B	27	F	3100	3300	39600	...
8	8	Hugo	A	30	M	3500	3800	45600	...

Arrange

```
# Ordenar pela renda_2024 em ordem decrescente
df_ord = sort(df, :renda_2024; rev = true)

# Ordenar por grupo e idade crescente
df_ord2 = sort(df, [:grupo, :idade])
```

	id	nome	grupo	idade	sexo	renda_2023	renda_2024	
	Int64	String	String	Int64	String	Int64	Int64	
1	1	Ana	A	23	F	2500	2700	
2	3	Carla	A	29	F	3200	3400	
3	8	Hugo	A	30	M	3500	3800	
4	2	Bruno	A	35	M	4800	5000	
5	5	Eva	B	19	F	1800	2000	
6	7	Gabi	B	27	F	3100	3300	
7	6	Felipe	B	33	M	4200	4400	
8	4	Diego	B	41	M	6000	6200	

Group_by + Summarise

```
g = groupby(df, :grupo)

resumo_grupo = combine(
  g,
```

```

:renda_2024 => mean => :renda_media,
:idade      => mean => :idade_media,
nrow        => :n_pessoas
)

```

	grupo	renda_media	idade_media	n_pessoas
	String	Float64	Float64	Int64
1	A	3725.0	29.25	4
2	B	3975.0	30.0	4

pivot_longer (Wide → Long)

```

# Deixar rendas em formato long
df_long = stack(
  df,
  [:renda_2023, :renda_2024],
  variable_name = :ano,
  value_name    = :renda
)

```

	id	nome	grupo	idade	sexo	ano	renda
	Int64	String	String	Int64	String	String	Int64
1	1	Ana	A	23	F	renda_2023	2500
2	2	Bruno	A	35	M	renda_2023	4800
3	3	Carla	A	29	F	renda_2023	3200
4	4	Diego	B	41	M	renda_2023	6000
5	5	Eva	B	19	F	renda_2023	1800
6	6	Felipe	B	33	M	renda_2023	4200
7	7	Gabi	B	27	F	renda_2023	3100
8	8	Hugo	A	30	M	renda_2023	3500
9	1	Ana	A	23	F	renda_2024	2700
10	2	Bruno	A	35	M	renda_2024	5000
11	3	Carla	A	29	F	renda_2024	3400
12	4	Diego	B	41	M	renda_2024	6200
13	5	Eva	B	19	F	renda_2024	2000
14	6	Felipe	B	33	M	renda_2024	4400
15	7	Gabi	B	27	F	renda_2024	3300
16	8	Hugo	A	30	M	renda_2024	3800

pivot_wider (Long → Wide)

```

# :id → permanece como identificador das linhas
# :ano → vira o nome das novas colunas (ex.: renda_2023, renda_2024)
# :renda → são os valores preenchidos nessas colunas
# Cada ano que estava em linhas vira uma coluna nova, reorganizando a tabela.

df_wide = unstack(df_long, :id, :ano, :renda)

```

	id	renda_2023	renda_2024
	Int64	Int64?	Int64?
1	1	2500	2700
2	2	4800	5000
3	3	3200	3400
4	4	6000	6200
5	5	1800	2000
6	6	4200	4400
7	7	3100	3300
8	8	3500	3800

left_join

```

# Juntar df com df_bonus pela coluna id
df_left = leftjoin(df, df_bonus, on = :id)

```

	id	nome	grupo	idade	sexo	renda_2023	renda_2024	bonus
	Int64	String	String	Int64	String	Int64	Int64	Int64?
1	2	Bruno	A	35	M	4800	5000	500
2	4	Diego	B	41	M	6000	6200	1000
3	8	Hugo	A	30	M	3500	3800	700
4	1	Ana	A	23	F	2500	2700	missing
5	3	Carla	A	29	F	3200	3400	missing
6	5	Eva	B	19	F	1800	2000	missing
7	6	Felipe	B	33	M	4200	4400	missing
8	7	Gabi	B	27	F	3100	3300	missing