

Trabalho de R - FIAP

Code ▾

Importando as bibliotecas necessárias

Hide

```
library(tidyverse)
library(readxl)
library(plotly)
library(gmodels)
library(sqldf)
```

Carregando o arquivo Cadastral para fazer as análises

Hide

```
ds <- read_excel("cadastral.xlsx")
ds
```

ID	DataNascimento	Sexo	TempodeServiço	EstadoCivil	NumerodeFilhos	TempodeRe
<dbl>	<S3: POSIXct>	<chr>	<dbl>	<dbl>	<dbl>	
1	1952-02-03	Masculino	98	3	0	
2	1958-05-23	Masculino	98	3	0	
3	1929-07-26	Feminino	98	1	0	
4	1947-04-15	Feminino	98	1	0	
5	1955-02-09	Masculino	98	2	3	
6	1958-08-22	Masculino	98	2	0	
7	1956-04-26	Masculino	98	3	0	
8	1966-05-06	Feminino	98	3	0	
9	1946-01-23	Feminino	98	2	6	
10	1946-02-13	Feminino	98	1	0	

1-10 of 946 rows | 1-7 of 9 columns

Previous123456...95Next

1. Tire uma tabela de frequência usando a função table na variável Sexo. Quantos homens e quantas mulheres têm no arquivo?

Hide

```
table(ds$Sexo)
```

Feminino	Masculino
432	514

2. Ordenar a variável ID.

```
ds <- ds[order(ds$ID),]  
ds
```

ID	DataNascimento	Sexo	TempodeServiço	EstadoCivil	NumerodeFilhos	TempodeRe						
<dbl>	<S3: POSIXct>	<chr>	<dbl>	<dbl>	<dbl>							
1	1952-02-03	Masculino	98	3	0							
1	1952-02-03	Masculino	98	3	0							
2	1958-05-23	Masculino	98	3	0							
2	1958-05-23	Masculino	98	3	0							
3	1929-07-26	Feminino	98	1	0							
3	1929-07-26	Feminino	98	1	0							
4	1947-04-15	Feminino	98	1	0							
4	1947-04-15	Feminino	98	1	0							
5	1955-02-09	Masculino	98	2	3							
5	1955-02-09	Masculino	98	2	3							
1-10 of 946 rows 1-7 of 9 columns			Previous	1	2	3	4	5	6	...	95	Next

3. Remova os ID duplicados. Coloque esse arquivo dentro de um objeto chamado A.

```
A <- unique(ds)  
A
```

ID	DataNascimento	Sexo	TempodeServiço	EstadoCivil	NumerodeFilhos					TempodeRe		
<dbl>	<S3: POSIXct>	<chr>	<dbl>	<dbl>	<dbl>							
1	1952-02-03	Masculino	98	3	0							
2	1958-05-23	Masculino	98	3	0							
3	1929-07-26	Feminino	98	1	0							
4	1947-04-15	Feminino	98	1	0							
5	1955-02-09	Masculino	98	2	3							
6	1958-08-22	Masculino	98	2	0							
7	1956-04-26	Masculino	98	3	0							
8	1966-05-06	Feminino	98	3	0							
9	1946-01-23	Feminino	98	2	6							
10	1946-02-13	Feminino	98	1	0							
1-10 of 473 rows 1-7 of 9 columns			Previous	1	2	3	4	5	6	...	48	Next

4. Já no objeto A. Tire uma tabela de frequência usando a função table na variável Sexo. Quantos homens e quantas mulheres têm no arquivo?

Hide

```
table(A$Sexo)
```

```
Feminino Masculino
      216       257
```

5. Crie uma variável data atual e acrescenta essa variável ao objeto/ arquivo A.

Hide

```
A$data_atual <- Sys.Date()
A
```

ID	DataNascimento	Sexo	TempodeServiço	EstadoCivil	NumerodeFilhos	TempodeRe
<dbl>	<S3: POSIXct>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
1	1952-02-03	Masculino	98	3	0	
2	1958-05-23	Masculino	98	3	0	
3	1929-07-26	Feminino	98	1	0	
4	1947-04-15	Feminino	98	1	0	
5	1955-02-09	Masculino	98	2	3	
6	1958-08-22	Masculino	98	2	0	
7	1956-04-26	Masculino	98	3	0	
8	1966-05-06	Feminino	98	3	0	
9	1946-01-23	Feminino	98	2	6	
10	1946-02-13	Feminino	98	1	0	

1-10 of 473 rows | 1-7 of 10 columns

Previous 1 2 3 4 5 6 ... 48 Next

6. Verifique se a variável salario é numérica?

Hide

```
is.numeric(A$salario)
```

```
[1] TRUE
```

7. Mostre o mínimo e o máximo da variável salario.

Hide

```
print(sprintf("Mínimo salário -> %0.2f", min(A$salario)))
```

```
[1] "Mínimo salário -> 1575.00"
```

Hide

```
print(sprintf("Máximo salário -> %0.2f", max(A$salario)))
```

```
[1] "Máximo salário -> 13500.00"
```

8. Crie uma variável faixa de salario com as seguintes quebras: 1574, 3000, 5000, 7000, 13500.

Hide

```
A$faixa_salario <- cut(A$salario, c(1574, 3000, 5000, 7000, 13500),label=c("A","B","C","D"))
A
```

ID	DataNascimento	Sexo	TempodeServiço	EstadoCivil	NumerodeFilhos	TempodeRe
<dbl>	<S3: POSIXct>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
1	1952-02-03	Masculino	98	3	0	
2	1958-05-23	Masculino	98	3	0	
3	1929-07-26	Feminino	98	1	0	
4	1947-04-15	Feminino	98	1	0	
5	1955-02-09	Masculino	98	2	3	
6	1958-08-22	Masculino	98	2	0	
7	1956-04-26	Masculino	98	3	0	
8	1966-05-06	Feminino	98	3	0	
9	1946-01-23	Feminino	98	2	6	
10	1946-02-13	Feminino	98	1	0	

1-10 of 473 rows | 1-7 of 11 columns

Previous 1 2 3 4 5 6 ... 48 Next

9. Crie um visualizador/ matriz usando a função View(A). Exatamente esse comando.

Hide

```
View(A)
```

10. Atribua o arquivo Transacional ao objeto B. E crie um visualizador/ matriz usando a função View(B).

Hide

```
B <- read_excel("transacional.xlsx")
View(B)
```

11. Crie um objeto chamado consolidado e faça uma união dos arquivos A e B através do Left join. Usando a função do R.

Hide

```
consolidado <- merge(A,B, by='ID', all.x=T)
consolidado
```

ID	DataNascimento	Sexo	TempodeServiço	EstadoCivil	NumerodeFilhos	TempodeRe						
<dbl>	<S3: POSIXct>	<chr>	<dbl>	<dbl>	<dbl>							
1	1952-02-03	Masculino	98	3	0							
2	1958-05-23	Masculino	98	3	0							
3	1929-07-26	Feminino	98	1	0							
4	1947-04-15	Feminino	98	1	0							
5	1955-02-09	Masculino	98	2	3							
6	1958-08-22	Masculino	98	2	0							
7	1956-04-26	Masculino	98	3	0							
8	1966-05-06	Feminino	98	3	0							
9	1946-01-23	Feminino	98	2	6							
10	1946-02-13	Feminino	98	1	0							
1-10 of 473 rows 1-7 of 17 columns			Previous	1	2	3	4	5	6	...	48	Next

12. Crie uma variável comprometimento de renda usando as variáveis ValorEmprestimo e Salario. Para isso utilize a expressão. (ValorEmprestimo / salario). Quantas variáveis ficaram no arquivo?

Hide

```
consolidado$comprometido_de_renda <- (consolidado$ValorEmprestimo / consolidado$salario)
print(sprintf("Número de colunas -> %0.0f", ncol(consolidado)))
```

```
[1] "Número de colunas -> 18"
```

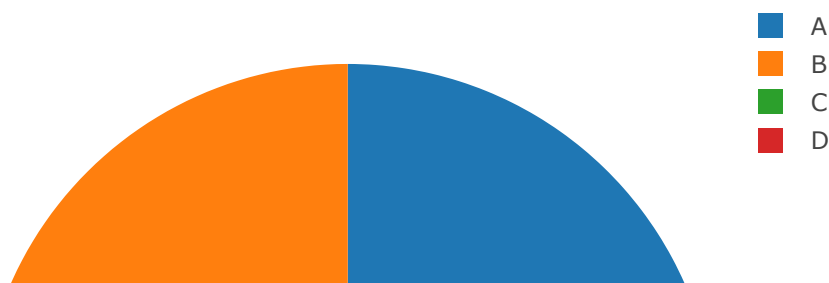
13. Faça um gráfico de pie usando o pacote plotly. Pode usar qualquer variáveis Explique o gráfico.

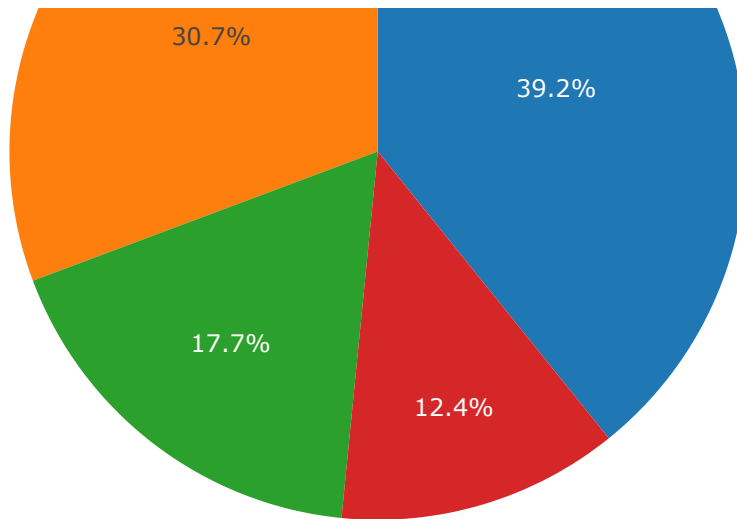
Hide

```
fig <- plot_ly(consolidado, labels = ~faixa_salario, values = ~salario, type = 'pie')
fig <- fig %>% layout(title = 'Faixa salarial',
  xaxis = list(showgrid = FALSE, zeroline = FALSE, showticklabels = FALSE),
  yaxis = list(showgrid = FALSE, zeroline = FALSE, showticklabels = FALSE))
```

```
fig
```

Faixa salarial



[Hide](#)

```
#Criei um gráfico de pizza com a variável de faixa salarial:  
#      39,2% das pessoas pertecem ao grupo A ([1500 - 3000[)  
#      30,7% das pessoas pertecem ao grupo B ([3000 - 5000[)  
#      17,7% das pessoas pertecem ao grupo C ([5000 - 7000[)  
#      12,4% das pessoas pertecem ao grupo D ([7000 - 13500[)  
# Temos praticamente 70% das pessoas com renda salarial de até 5000,00 reais formados pela cl  
ass A e # B
```

14. Faça um tabela cruzada usando a função Crosstable do package gmodels das variáveis sexo e default. Olhando para percentual coluna, quem tem o maior % de inadimplência H ou M? e qual é esse valor.

[Hide](#)

```
Cross <- CrossTable(consolidado$Sexo, consolidado$default)
```

Cell Contents

N
Chi-square contribution
N / Row Total
N / Col Total
N / Table Total

Total Observations in Table: 473

	consolidado\$default		
consolidado\$Sexo	Adimplente	Inadimplente	Row Total
-----	-----	-----	-----
Feminino	49	167	216
	1.810	0.686	
	0.227	0.773	0.457
	0.377	0.487	
	0.104	0.353	
-----	-----	-----	-----
Masculino	81	176	257
	1.521	0.577	
	0.315	0.685	0.543
	0.623	0.513	
	0.171	0.372	
-----	-----	-----	-----
Column Total	130	343	473
	0.275	0.725	
-----	-----	-----	-----

[Hide](#)

```
print("Quem tem o maior percentual de inadimplência são os homens com 51,30%")
```

```
[1] "Quem tem o maior percentual de inadimplência são os homens com 51,30%"
```

15. Utilizando a biblioteca SQLDF pede-se:

- Selecione da tabela consolidado os registros do sexo masculino e que possui um conta particular.
- Tira a média de QtdaPagas pela variável Atraso .

[Hide](#)

```
sexo_masculino_particular <- sqldf("select * from consolidado where sexo = 'Masculino' and conta = 'Particular'")
sexo_masculino_particular
```

ID	DataNascimento	Sexo	TempodeServiço	EstadoCivil	NumerodeFilhos	Temp
<dbl>	<S3: POSIXct>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>

ID <dbl>	DataNascimento <S3: POSIXct>	Sexo <chr>	TempodeServiço <dbl>	EstadoCivil <dbl>	NumerodeFilhos <dbl>	Temp
1	1952-02-02 22:00:00	Masculino	98	3	0	
2	1958-05-22 21:00:00	Masculino	98	3	0	
5	1955-02-08 21:00:00	Masculino	98	2	3	
6	1958-08-21 21:00:00	Masculino	98	2	0	
7	1956-04-25 21:00:00	Masculino	98	3	0	
12	1966-01-10 22:00:00	Masculino	98	3	0	
13	1960-07-16 21:00:00	Masculino	98	1	0	
15	1962-08-28 21:00:00	Masculino	97	2	0	
16	1964-11-16 21:00:00	Masculino	97	4	3	
17	1962-07-17 21:00:00	Masculino	97	1	3	

1-10 of 257 rows | 1-7 of 18 columns

Previous 1 2 3 4 5 6 ... 26 Next

Hide

```
QtdaPagas_vs_Atraso <- sqldf("select Atraso, avg(QtdaPagas) as media_QtdaPagas from consolidado group by Atraso")
QtdaPagas_vs_Atraso
```

Atraso <chr>	media_QtdaPagas <dbl>
Não	7.641221
Sim	6.937500

2 rows

16. Calcule a expressão numérica: $\frac{((\sqrt{16})/2)3^2}{2(9-2^3)}$,

Hide

```
calculo1 = ((sqrt(16)/2)*3^2)/2*(9-2^3)
calculo1
```

```
[1] 9
```

17. Calcule a expressão numérica: $-(-2^3)^{-\lceil 1 \rceil} 0 - (\sqrt{(25-3^2)}) - 5^3/25$

Hide

```
calculo2 <- -(-2^3) + (-1)^0 - sqrt(25 - ( 3 ^ 2)) - ( ( 5 ^ 3 ) / 25 )
calculo2
```

```
[1] 0
```