library(tidyverse)

library(dplyr)

library(googlesheets)

library(forcats)

library(lubridate)

# Importando a planilha do google sheets

gastos\_simplif <- gs\_title("planilha simplificada de divisão de gastos")

# Listando as abas da planilha. No caso tem "Página1" "Página2".

gs\_ws\_ls(gastos\_simplif)

# Criando o objeto para depois fazer o data frame.

wsret <- gs\_read(ss = gastos\_simplif, ws = "Página1")

# Transformando em data frame.

df\_gastos\_simplif <- as.data.frame(wsret) %>%

mutate\_if(is.character, as.factor) %>% # transformando em categoria todas colunas que são character.

mutate(mes = fct\_relevel(mes, "Janeiro", "Fevereiro", "Março", "Abril", "Maio", "Junho", "Julho", "Agosto", "Setembro",

"Outubro", "Novembro", "Dezembro")) # ordenando o factor mes

# Criando o df resultado onde será mostrado quanto um tem que pagar ao outro e quanto eu gastei no total.

result <- df\_gastos\_simplif %>% group\_by(ano, mes, nome, tipo\_desp) %>% # Agrupando para somar por grupo

summarise(total = sum(valor)) %>% # Somando

ungroup() %>% # Sem desagrupar as funções lag lead não funcionam

mutate(dif\_a\_pagar = (lead(total, 2) - total)/2) %>% # Calculando a diferença a pagar

mutate(total\_mario = total/2 + lead(total,2)/2 + lead(total,3)) %>% #Fazendo a conta (comum + comum)/2 + próprio

filter(nome == "a\_Maria" & tipo\_desp == "comum") %>% # Só estas linhas me interessam.

mutate\_if(is.character, as.factor) # transformando em categoria todas colunas que são character.

saveRDS(df\_gastos\_simplif, "df\_gastos\_simplif.rds")

saveRDS(result, "result.rds")

# GRÁFICO 1.

# Preparando para fazer o gráfico.

# Agrupando para somar. Saber to total de cada despesa por ano, mês, nome e tipo.

df\_gastos\_simplif %>% group\_by(ano, mes, nome, tipo\_desp, categ\_desp) %>%

summarise(total = sum(valor)) %>% ungroup() %>%

# filtrando as linhas que me competam: todas as minhas e as comuns da Maria.

filter((nome == 'a\_Maria' & tipo\_desp == 'comum') | (nome == 'o\_Mário')) %>%

# Não posso somar direto os valores das despesas comuns com as próprias,

# pois nas comuns só arco com a metade,

# assim tenho que dividí-las por 2 para depois agrupar.

mutate(total = ifelse(tipo\_desp == "comum", total/2, total)) %>%

# agrupando por categoria despesa.

group\_by(mes, categ\_desp) %>%

# summarizando os totais por despesa.

summarise(total = round (sum(total), 0)) %>%

# ordenando a categ\_desp do maior para o menor.

# mutate(categ\_desp = fct\_rev(fct\_reorder(categ\_desp, total))) %>%

# criando a coluna total\_percentual

mutate(total\_perc = round (total/sum(total),3)) %>%

# Agora fazendo o gráfico.

ggplot(aes(x = categ\_desp, y = total)) +

geom\_col(aes(fill = categ\_desp), color = 'black') +

# Pode haver diversos geom\_text, cada um com alguma coisa escrita

geom\_text(aes(label = paste(100\*total\_perc,"%")), vjust = 1.3, size = 3.5) +

geom\_text(aes(label = paste("R$", total)), vjust = - 0.3, size = 3.5) +

# Aumentando olimite do eixo y para o label não ficar fora do gráfico.

scale\_y\_continuous(limits=c(0, 6500)) +

#scale\_fill\_manual(values = c("red", "grey", "seagreen3", "blue", "yellow", "orange", "violet", "gold", "aliceblue", "green")) +

scale\_fill\_brewer(palette = "Set1") +

facet\_wrap(~ mes, ncol = 2) +

theme(legend.position = "none",

axis.title = element\_blank(), # tirando os títulos dos 2 eixos

axis.text.x = element\_text(face = "bold", color = "black", size = 8, angle = 0),

axis.text.y = element\_blank(), # tirando texto do eixo y

axis.ticks.x = element\_blank(), # tirando os traços do eixo x

axis.ticks.y = element\_blank(), # tirando os traços do eixo y

panel.grid = element\_blank(), # tirando todas linhas de grade

# trabalhando elementos do fundo e contorno do gráfico.

panel.background = element\_rect(fill = "cornsilk",

colour = "black",

size = 1.5, linetype = "solid"),

# modificando o quadro acima do gráfico

strip.background = element\_rect(colour = "black", size = 1.5, fill = alpha("cornsilk",0.5)),

# modificando o texto

strip.text.x = element\_text(colour = "black", face = "bold", size=12, angle=0))

# GRÁFICO 2.

# Agora fazer outro gráfico de colunas, com os mesmos valores, porém ao invés

# de agrupar várias categorias em cada mês, vou agrupar a mesma categoria

# em diversos meses.

df\_gastos\_simplif %>% group\_by(ano, mes, nome, tipo\_desp, categ\_desp) %>%

summarise(total = sum(valor)) %>% ungroup() %>%

# filtrando as linhas que me competam: todas as minhas e as comuns da Maria.

filter((nome == 'a\_Maria' & tipo\_desp == 'comum') | (nome == 'o\_Mário')) %>%

# Não posso somar direto os valores das despesas comuns com as próprias,

# pois nas comuns só arco com a metade,

# assim tenho que dividí-las por 2 para depois agrupar.

mutate(total = ifelse(tipo\_desp == "comum", total/2, total)) %>%

# agrupando por categoria despesa.

group\_by(mes, categ\_desp) %>%

# summarizando os totais por despesa.

summarise(total = round (sum(total), 0)) %>%

# ordenando a categ\_desp do maior para o menor.

# mutate(categ\_desp = fct\_rev(fct\_reorder(categ\_desp, total))) %>%

# criando a coluna total\_percentual

mutate(total\_perc = round (total/sum(total),3)) %>%

# Agora fazendo o gráfico propriamente dito.

ggplot(aes(x = mes, y = total)) +

geom\_col(aes(fill = categ\_desp), color = 'black') +

# Pode haver diversos geom\_text, cada um com alguma coisa escrita

geom\_text(aes(label = paste(100\*total\_perc,"%")), vjust = 1.3, size = 3.5) +

geom\_text(aes(label = paste("R$", total)), vjust = - 0.3, size = 3.5) +

# Abreviando o nome dos meses.

scale\_x\_discrete(labels = c("Janeiro" = "Jan","Fevereiro" = "Fev",

"Março" = "Mar", "Abril" = "Abr", "Maio" = "Mai", "Junho" = "Jun", "Julho" = "Jul",

"Agosto" = "Ago", "Setembro" = "Set", "Outubro" = "Out", "Novembro" = "Nov",

"Dezembro" = "Dez")) +

# Aumentando o limite do eixo y para o label não ficar fora do gráfico.

scale\_y\_continuous(limits=c(0, 6500)) +

#scale\_fill\_manual(values = c("red", "grey", "seagreen3", "blue", "yellow", "orange", "violet", "gold", "aliceblue", "green")) +

scale\_fill\_brewer(palette = "Set1") +

facet\_wrap(~ categ\_desp, ncol = 4) +

theme\_bw() +

theme(legend.position = "none",

axis.title = element\_blank(), # tirando os títulos dos 2 eixos

axis.text.x = element\_text(face = "bold", color = "black", size = 8, angle = 0),

axis.text.y = element\_blank(), # tirando texto do eixo y

axis.ticks.x = element\_blank(), # tirando os traços do eixo x

axis.ticks.y = element\_blank(), # tirando os traços do eixo y

panel.grid = element\_blank(), # tirando todas linhas de grade

# Aumentando espaço horizontal entre os gráficos

panel.spacing.y=unit(1.5, "lines"),

# trabalhando elementos do fundo e contorno do gráfico.

panel.background = element\_rect(fill = "cornsilk",

colour = "black",

size = 1.5, linetype = "solid"),

# modificando o quadro acima do gráfico

strip.background = element\_rect(colour = "black", size = 1.5, fill = "cornsilk"),

# modificando o texto

strip.text.x = element\_text(colour = "black", face = "bold", size=12, angle=0))