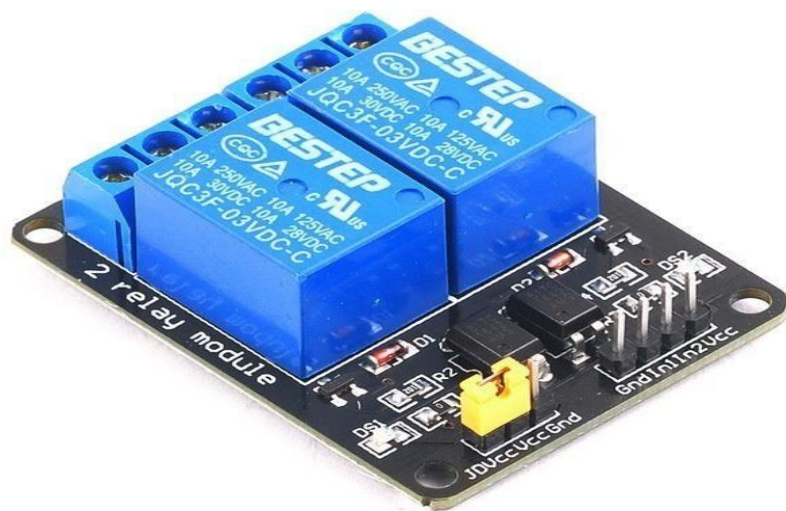
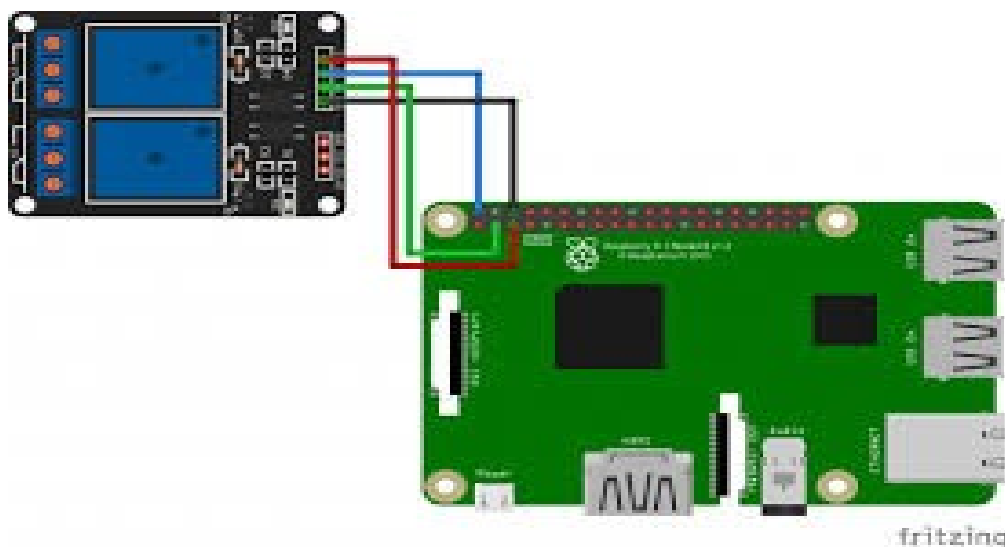


# Configurando o Raspberry para controle de Relés

Configurando o Raspberry pi no controle de equipamentos através da programação das portas GPIO para a programação dos relés de controle. Na Figura abaixo temos um módulo relé de duas entradas, podendo assim termos o controle de dois equipamentos simultaneamente. É composto por 04 pinos, sendo dois de alimentação (Vcc e Gnd) e 02 pinos de entrada (In1 e In2), os quais estarão ligados à placa \ Raspberry através da GPIO.



A seguir temos o esquema de ligação entre a placa Raspberry e o módulo relé, com as devidas ligações necessárias.



Como visto na Figura acima, selecionamos duas portas GPIO para controlar as entradas In1 e In2 do módulo relé, e também conectamos os pinos de alimentação da placa Raspberry ao módulo relé. A seguir vamos fazer a programação para acionarmos os relés cujas entradas são In1 e In2 através de dois botões B1 e B2, respectivamente.

## Código Python

```
import Rpi.GPIO as gpio
import time

gpio.setmode(gpio.BCM)
gpio.setup(2, gpio.OUT) # relé 1
gpio.setup(3, gpio.OUT) # relé 2
gpio.setup(5, gpio.IN)   # Botão B1
gpio.setup(6, gpio.IN)   # Botão B2

gpio.output(2,False) # desliga os relés inicialmente
gpio.output(3,False)

while (True):
    try:
        if gpio.input(5) == True:
            gpio.output(2,1)
            time.sleep(2)
            gpio.output(2,0)
        if gpio.input(6) == True:
            gpio.output(3,1)
            time.sleep(2)
            gpio.output(3,0)

    except KeyboardInterrupt:
        gpio.output(2,False) # desliga os relés ao término
        gpio.output(3,False)
```

# Interface Gráfica

Uma Interface Gráfica do Usuário (GUI), nada mais é que um container no qual podemos inserir componentes, também chamados de *widgets*, os quais permitem que nosso programa tenha uma apresentação visual mais bonita e amigável, onde o usuário pode interagir através do mouse, ou mesmo do teclado, de forma mais rápida.

Quando instalamos o sistema Pi OS, a linguagem Python é instalada de forma automática, assim como algumas bibliotecas da linguagem, entre elas temos o ***tkinter*** que é uma biblioteca de componentes gráficos que nos permitem criar programas de forma visual. Dentre esses componentes podemos citar alguns que são bastante úteis quando estamos trabalhando com IoT, por exemplo: Button, Label, Slider, dentre outros que veremos.

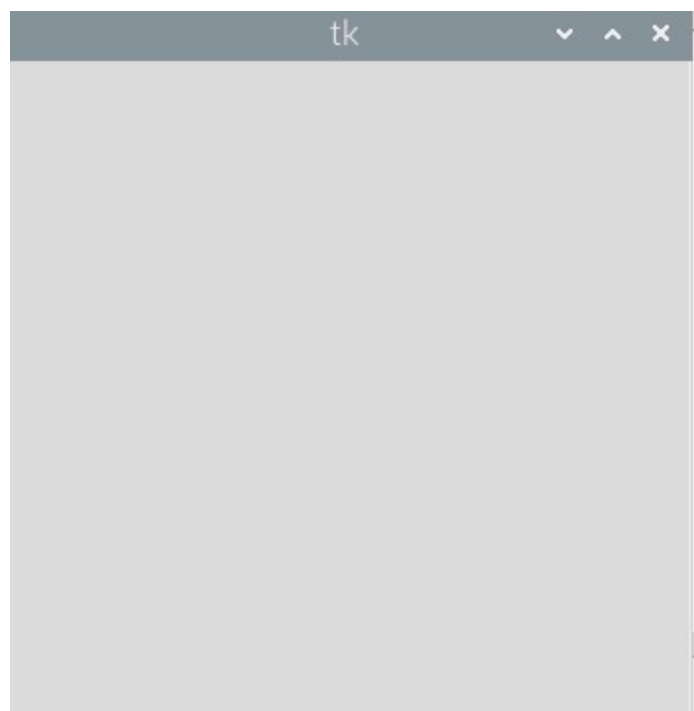
Como toda biblioteca do python, usamos a instrução ***import*** para carregar a biblioteca em nosso programa. Vejamos a seguir um trecho de código para iniciar uma janela gráfica.

```
from tkinter import * # carrega todos os módulos da biblioteca

janela = Tk()         # Inicializa uma janela vazia

janela.mainloop()    # Cria um sistema de loop e fica a espera de algum evento.
```

Ao executar o código acima, teremos como resultado a janela abaixo, vazia e sem nenhuma funcionalidade.



## Populando a janela e colocando alguns atributos

```
from tkinter import *

def sair():
    janela.destroy()

janela = Tk()
janela.configure(background='green')
janela.geometry('500x400')

janela.title("Controle de Relés via interface gráfica")

btn = Button(janela,text="Sair", command = sair)
btn.pack(padx=100,pady=100)
janela.mainloop()
```

No código acima configuramos a cor de fundo como sendo verde ('green'), também redimensionamos o tamanho da janela para '500x400', definimos um título para a mesma e inserimos um componente botão (Button), o qual quando pressionado exerce uma ação, que nada mais é que chamar a função sair, que por sua vez encerra a aplicação.



