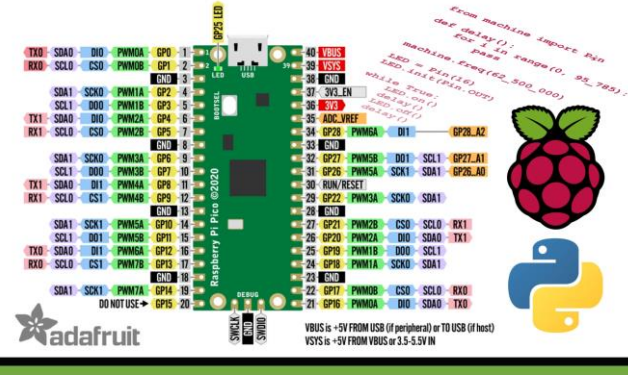


Getting Started with Pico and Micropython

Opening saved projects from the Raspberry Pi Pico

Francis Kanene
Choma District Resource Center
0975273153

Raspberry Pi Pico
Interfacing and Programming with
MicroPython



The image shows a Raspberry Pi Pico board with a detailed pin header diagram overlaid. The diagram lists various pins and their functions, such as TX0, RX0, SCL1, SDA1, TX1, RX1, SCL2, SDA2, TX2, RX2, SCL3, SDA3, TX3, RX3, SCL4, SDA4, TX4, RX4, SCL5, SDA5, TX5, RX5, SCL6, SDA6, TX6, RX6, SCL7, SDA7, TX7, RX7, SCL8, SDA8, TX8, RX8, SCL9, SDA9, TX9, RX9, SCL10, SDA10, TX10, RX10, SCL11, SDA11, TX11, RX11, SCL12, SDA12, TX12, RX12, SCL13, SDA13, TX13, RX13, SCL14, SDA14, TX14, RX14, SCL15, SDA15, TX15, RX15, SCL16, SDA16, TX16, RX16, SCL17, SDA17, TX17, RX17, SCL18, SDA18, TX18, RX18, SCL19, SDA19, TX19, RX19, SCL20, SDA20, TX20, RX20. The code snippet shows a simple program that prints the number of pins on the board.

```
from machine import Pin
def delay():
    for i in range(0, 95, 700):
        machine.freq(62_500_000)
        while True:
            led.on()
            delay()
            led.off()
            delay()
```

adafruit

Shujen Chen
Muhammad Ali Mazidi
Nasim Yazdani

Figure 1: RP Pico board with pin headers

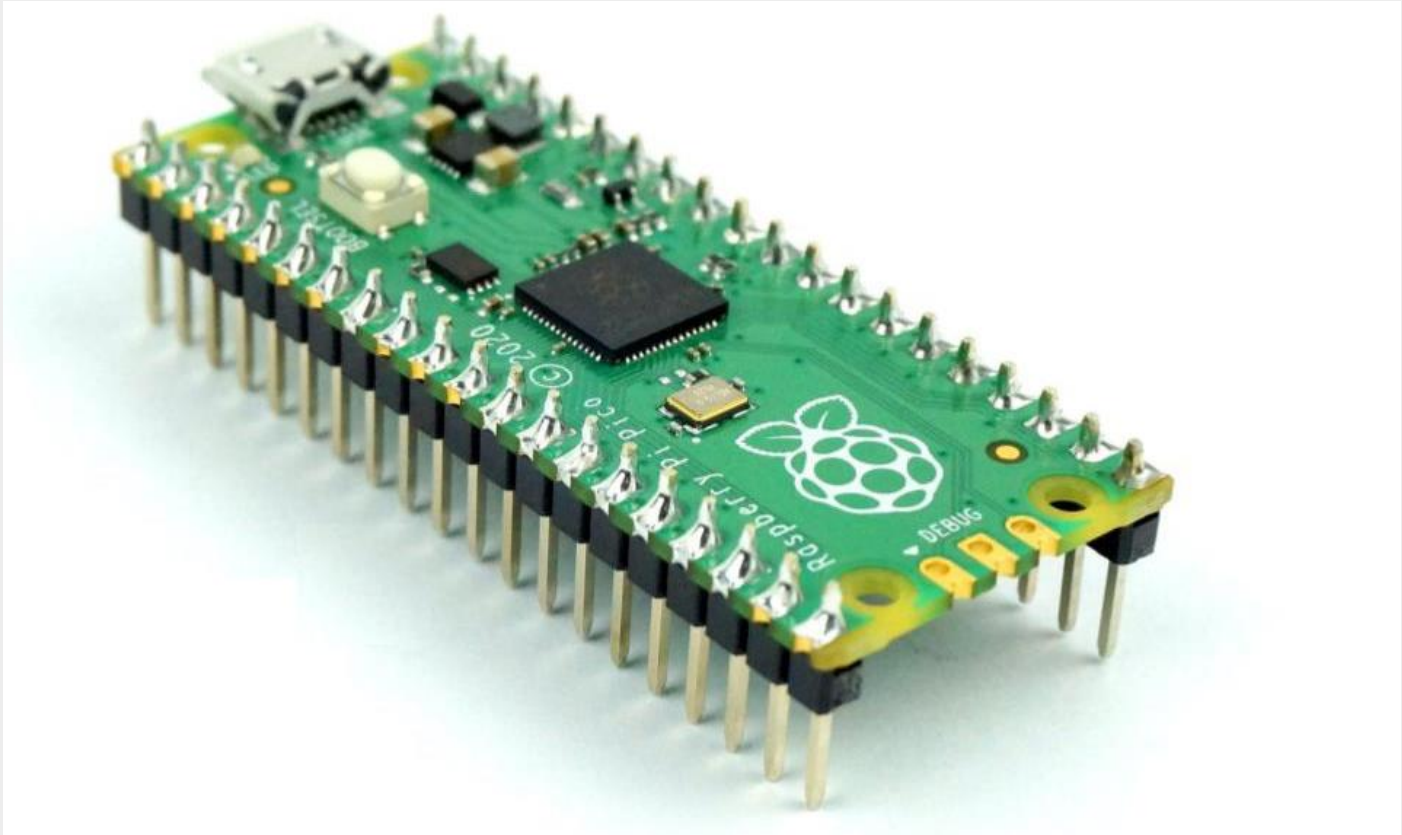


Figure 3: Establish connection between the computer and Pico by opening Thonny

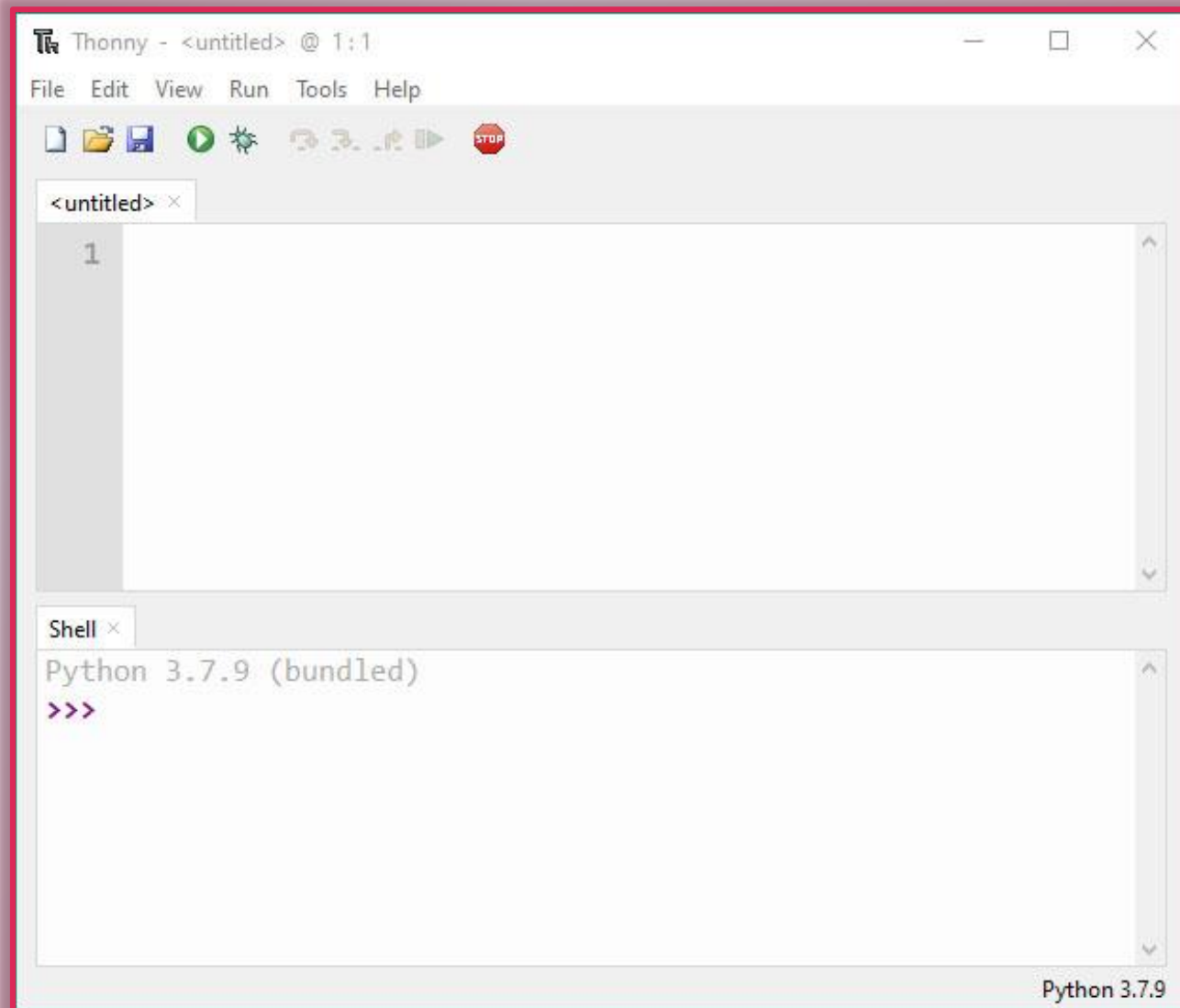


Figure 4: When Thonny detects an external Python connection, it displays the possible connections to choose from and select Miropython(Raspberry Pi Pico)

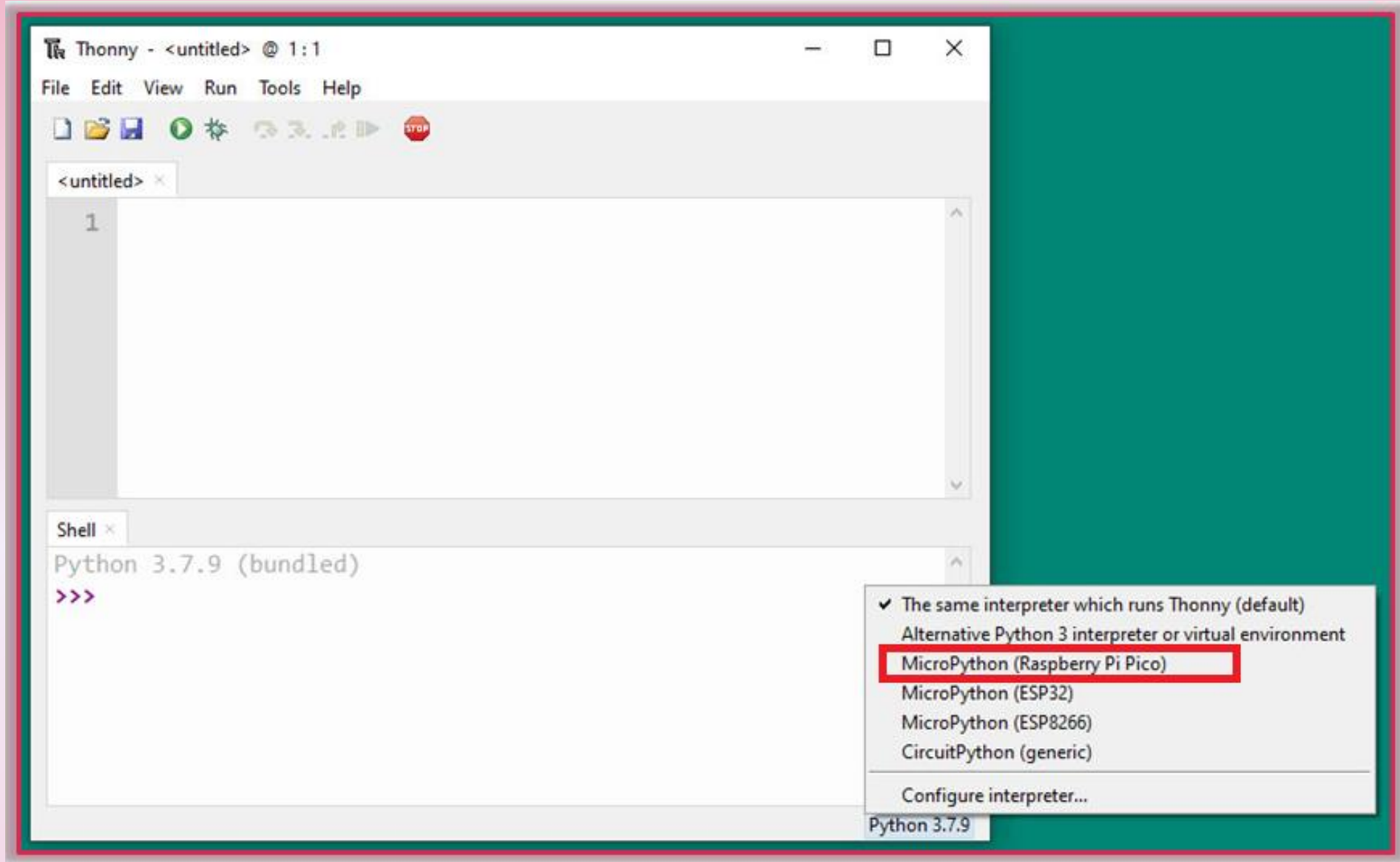


Figure 5: Accessing saved codes on the Pico by clicking the folder

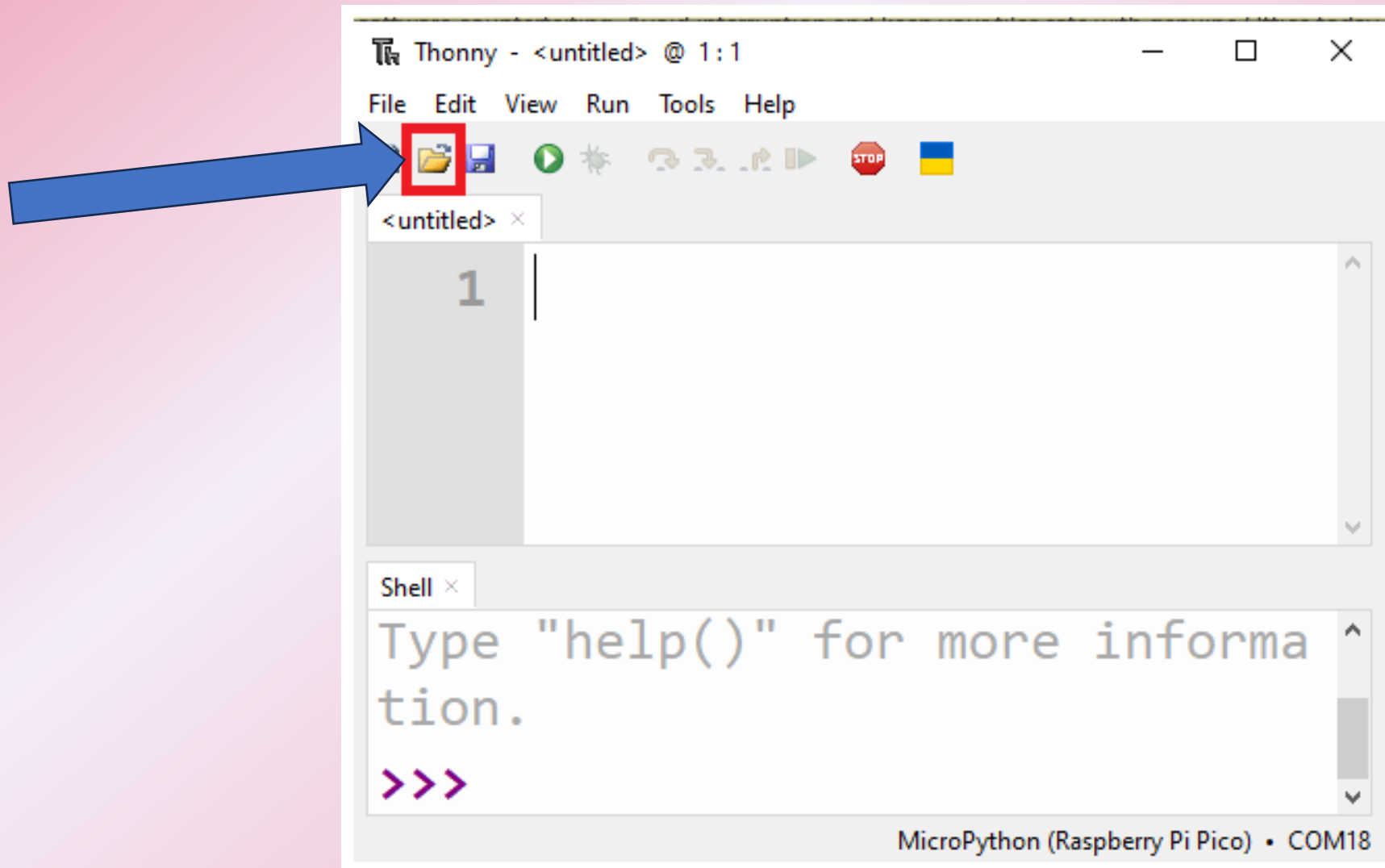


Figure 6: Select the Pico to open the saved codes

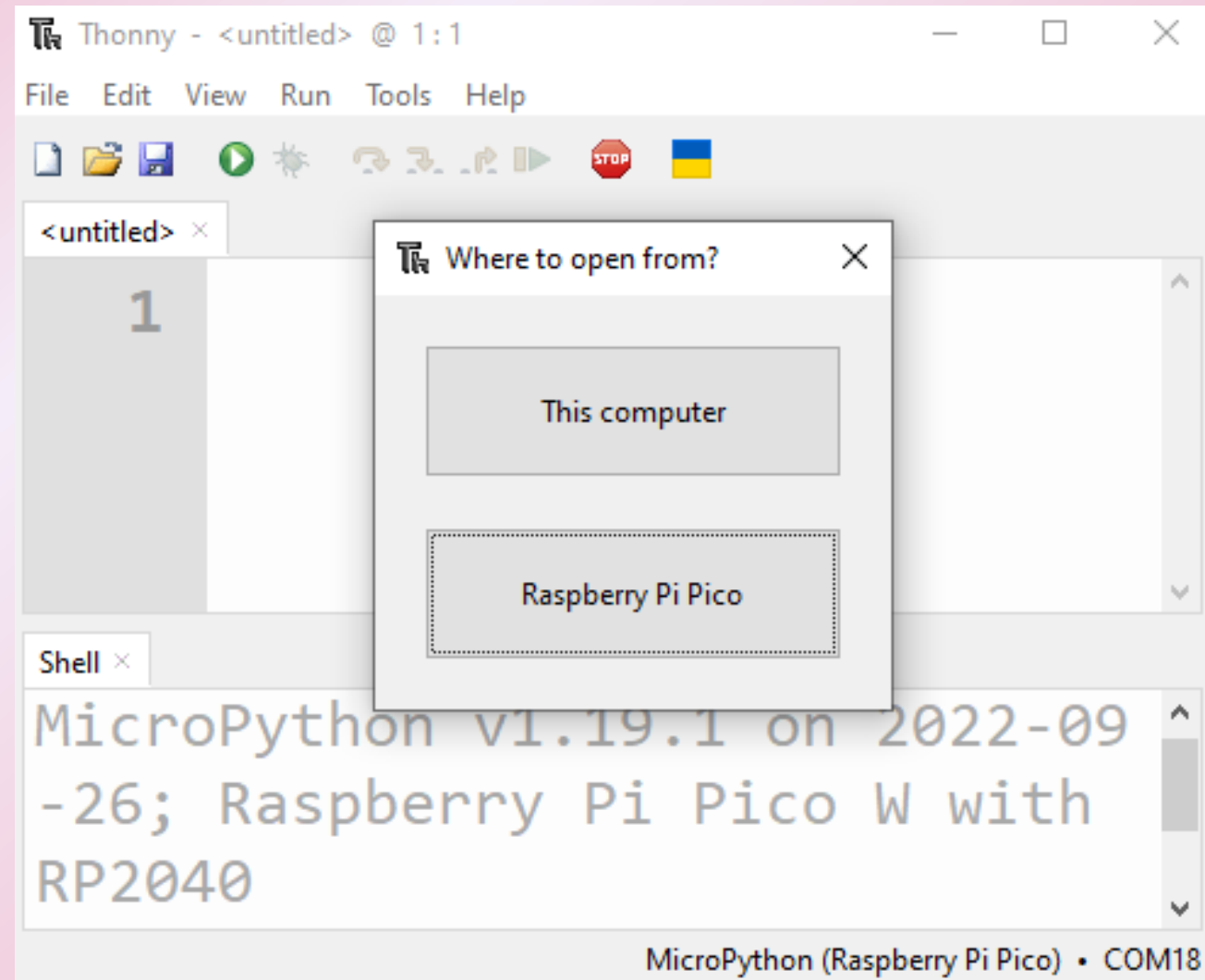


Figure 7: Select the saved code from the Pi Pico and click **OK**

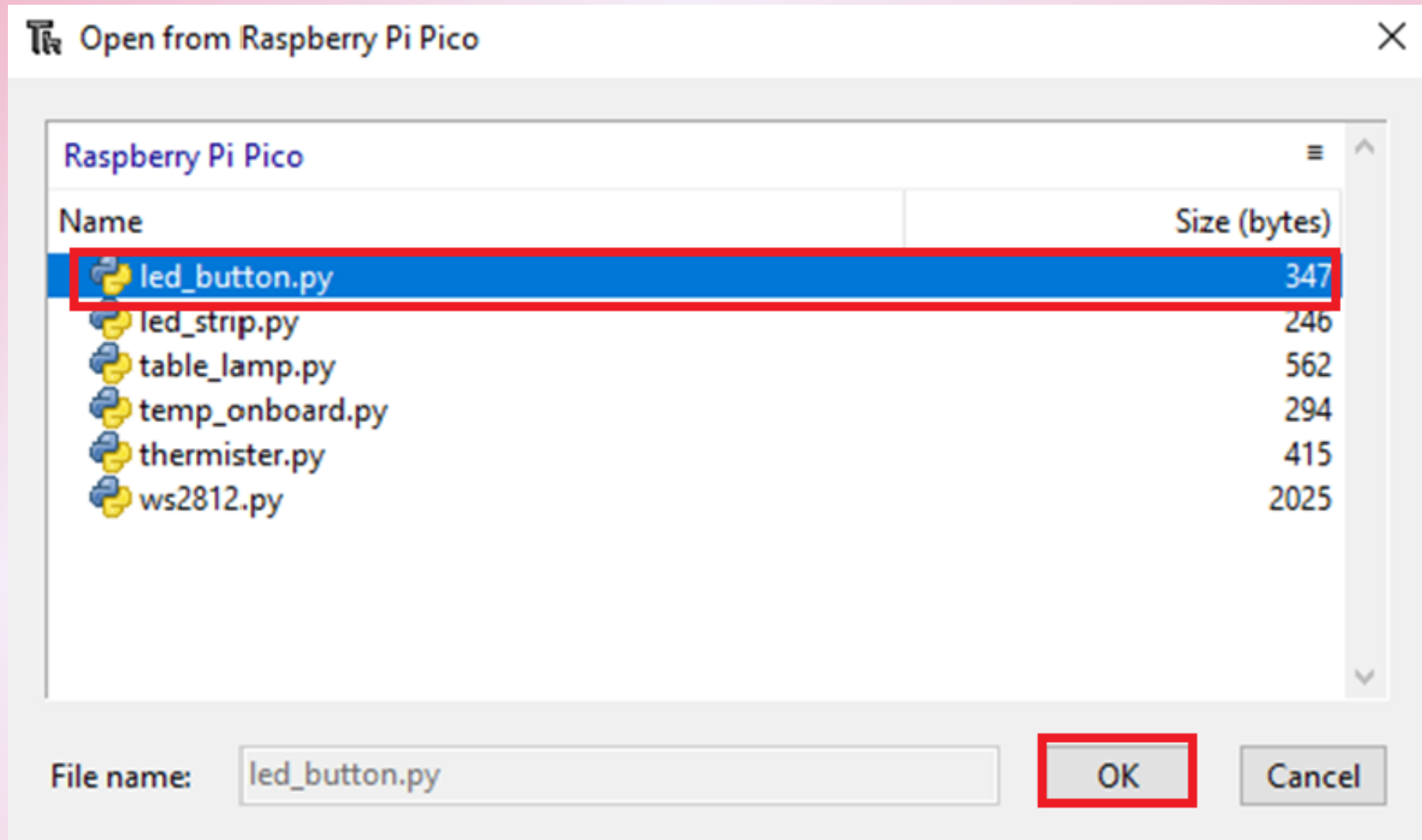
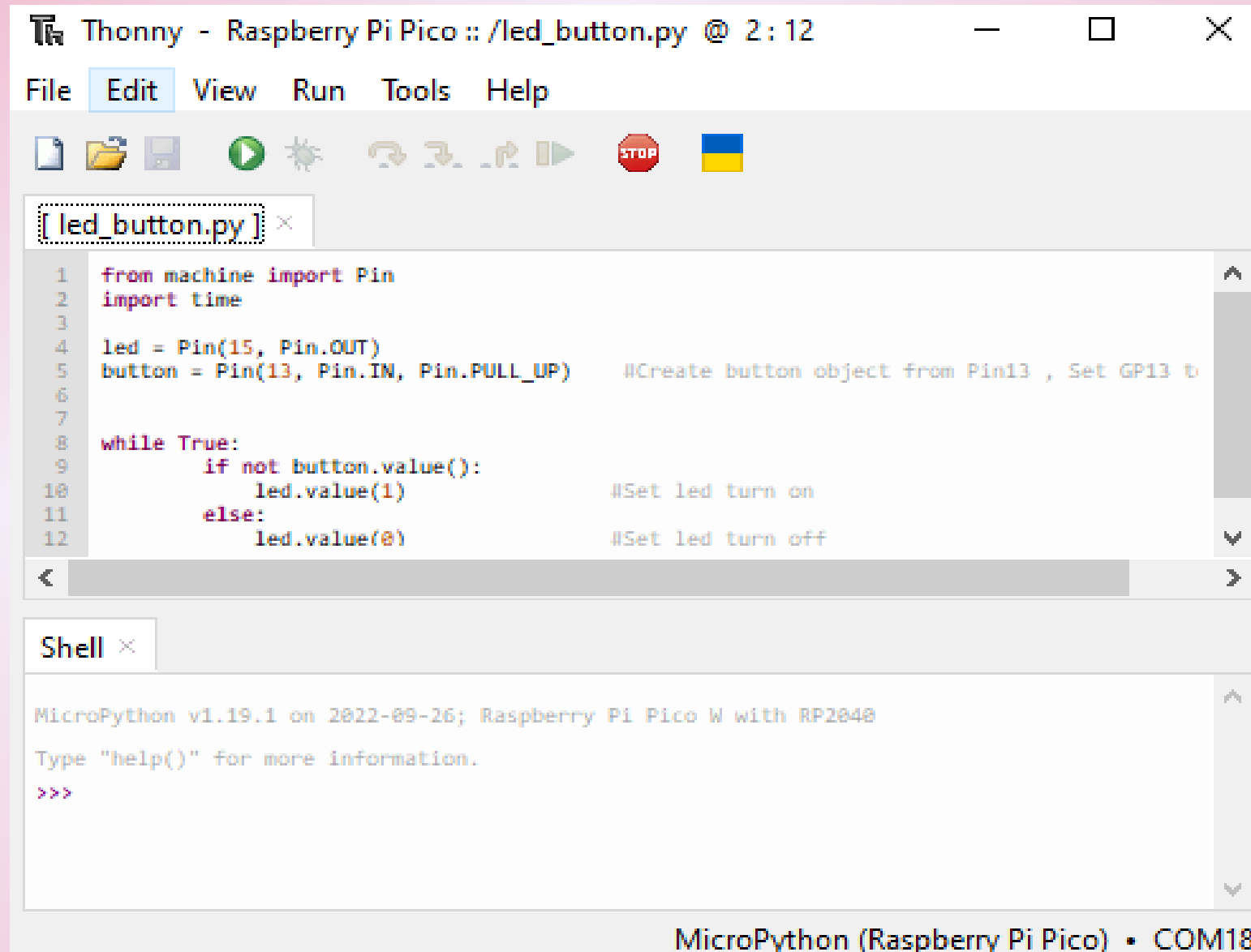


Figure 8: Selected code opens in Thonny





THANK YOU