

Python - Tópico Principal

Todo programa Python possui pelo menos um thread de execução chamado **thread principal**. O **thread principal** por padrão é um thread não-daemon.

Às vezes, podemos precisar criar threads adicionais em nosso programa para executar o código simultaneamente.

Aqui está a **sintaxe** para criar um novo thread -

```
object = threading.Thread(target, daemon)
```

O construtor Thread() cria um novo objeto. Ao chamar o método start(), o novo thread começa a ser executado e chama automaticamente uma função dada como argumento para o parâmetro de destino cujo padrão é **run**. O segundo parâmetro é "daemon" que por padrão é Nenhum.

Exemplo

```
from time import sleep
from threading import current_thread
from threading import Thread

# function to be executed by a new thread
def run():
    # get the current thread
    thread = current_thread()
    # is it a daemon thread?
    print(f'Daemon thread: {thread.daemon}')

# create a new thread
thread = Thread(target=run)

# start the new thread
thread.start()

# block for a 0.5 sec
sleep(0.5)
```

Ele produzirá a seguinte **saída** -



Daemon thread: False

Então, criando um thread com a seguinte instrução -

```
t1=threading.Thread(target=run)
```

Esta instrução cria um thread não-daemon. Quando iniciado, ele chama o método run().