

Python - Sincronizando Threads

O módulo de threading fornecido com Python inclui um mecanismo de bloqueio simples de implementar que permite sincronizar threads. Um novo bloqueio é criado chamando o método `Lock()` , que retorna o novo bloqueio.

O método `adquirir(blocking)` do novo objeto de bloqueio é usado para forçar os threads a serem executados de forma síncrona. O parâmetro de bloqueio opcional permite controlar se o encadeamento aguarda para adquirir o bloqueio.

Se o bloqueio for definido como 0, o thread retornará imediatamente com valor 0 se o bloqueio não puder ser adquirido e com 1 se o bloqueio foi adquirido. Se o bloqueio for definido como 1, o thread será bloqueado e aguardará a liberação do bloqueio.

O método `release()` do novo objeto lock é usado para liberar o bloqueio quando ele não for mais necessário.

Exemplo

```
import threading
import time

class myThread (threading.Thread):
    def __init__(self, threadID, name, counter):
        threading.Thread.__init__(self)
        self.threadID = threadID
        self.name = name
        self.counter = counter
    def run(self):
        print ("Starting " + self.name)
        # Get lock to synchronize threads
        threadLock.acquire()
        print_time(self.name, self.counter, 3)
        # Free lock to release next thread
        threadLock.release()

def print_time(threadName, delay, counter):
    while counter:
        time.sleep(delay)
        print ("%s: %s" % (threadName, time.ctime(time.time())))
        counter -= 1

threadLock = threading.Lock()
```



```
threads = []

# Create new threads
thread1 = myThread(1, "Thread-1", 1)
thread2 = myThread(2, "Thread-2", 2)

# Start new Threads
thread1.start()
thread2.start()

# Add threads to thread list
threads.append(thread1)
threads.append(thread2)

# Wait for all threads to complete
for t in threads:
    t.join()
print ("Exiting Main Thread")
```

Saída

Quando o código acima é executado, ele produz a seguinte saída -

```
Starting Thread-1
Starting Thread-2
Thread-1: Thu Jul 13 21:10:11 2023
Thread-1: Thu Jul 13 21:10:12 2023
Thread-1: Thu Jul 13 21:10:13 2023
Thread-2: Thu Jul 13 21:10:15 2023
Thread-2: Thu Jul 13 21:10:17 2023
Thread-2: Thu Jul 13 21:10:19 2023
Exiting Main Thread
```