**Multitasking**

in programming refers to the ability of a system to handle multiple tasks (or processes) at the same time. In Python, multitasking can be achieved through various techniques like threading, multiprocessing, and asynchronous programming.

We need to Import the Threading and if we want to know the current thread name then we need to use

Threading.current\_thread().name

import threading

print("The current Thread is",threading.current\_thread())

print("The current Thread is",threading.current\_thread().name)

**Here are the main types of multitasking in Python:**

1. **Using threading.Thread with a function**.

2. **Subclassing threading.Thread**.

3. **Using threading.Thread with arguments**.

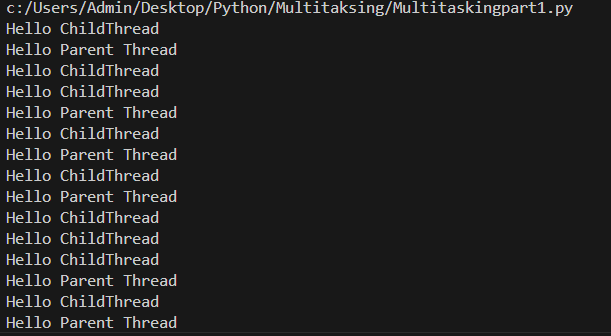
4. **Creating multiple threads with a loop**.

5. **Creating daemon threads**.

**1. Creating Thread using threading.Thread with a function**

* In this method, you create a thread by passing a function (or callable) to the Thread class.
* from threading import Thread
* def display():
* for i in range(1,10):
* print("Hello ChildThread")
* # threading.sleep(1)
* t = Thread(target=display)
* t.start()
* for i in range(1,11):
* print("Hello Parent Thread"

**output: Always unpreditable**



**Ex2:** import threading

def print\_numbers():

    for i in range(1, 6):

        print(f"Number: {i}")

# Create a thread by passing the target function

thread1 = threading.Thread(target=print\_numbers)

# Start the thread

thread1.start()

# Join the thread to wait for it to finish

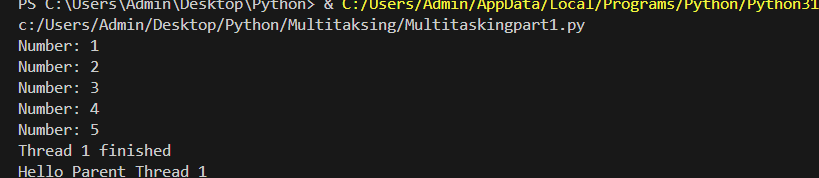
thread1.join()

print("Thread 1 finished")

for i in range(1,6):

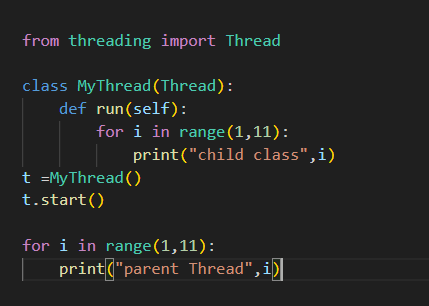
    print("Hello Parent Thread",i)

**Here join() method will make the function to execute first and then the next for loop**

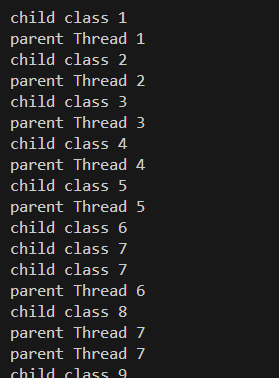
**Output:** 

**2. Creating Thread by Subclassing threading.Thread**

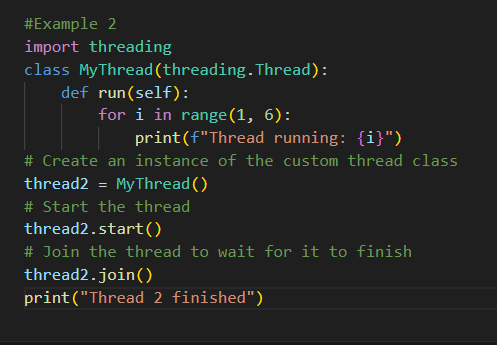
* In this method, you create a subclass of threading.Thread and override its run() method.
* Ex1



Output:



Example 2:



Output:

Thread running: 1

Thread running: 2

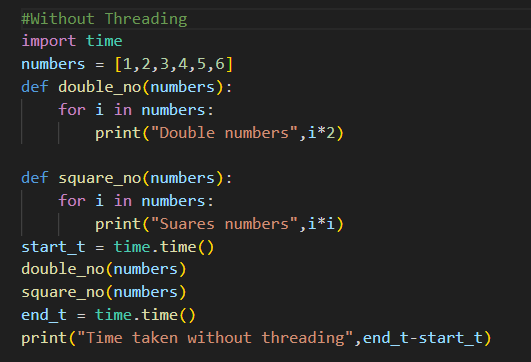
Thread running: 3

Thread running: 4

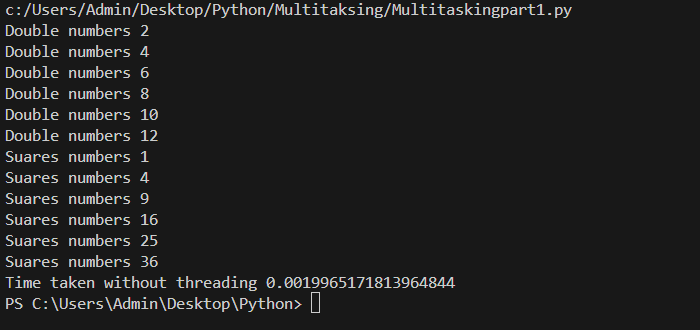
Thread running: 5

Thread 2 finished

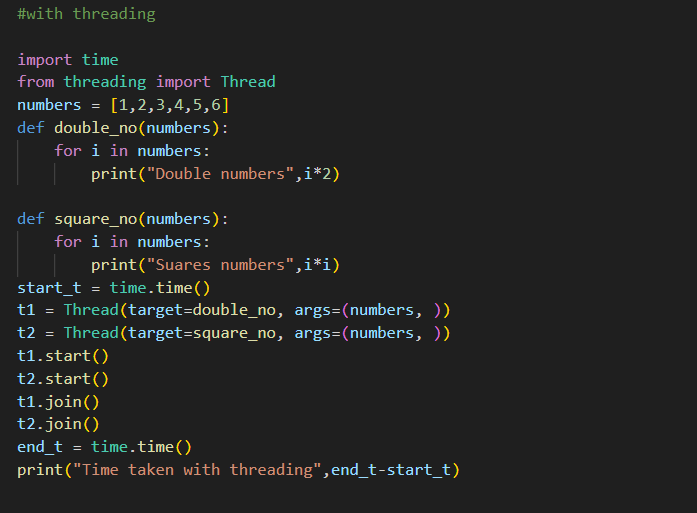
**With and Without Multi-Threading difference:**



**Output:**



**With Threading**



**Output:**

