

## 비동기 처리

Process

Thread

일반적인 프로그램 구현 (싱글 쓰레드 형태)

멀티 쓰레딩 형태 구현

# 비동기 처리

## Process

현재 실행중인 하나의 프로그램

내가 작성한 소스코드도 실행하는 동안 하나의 프로세스가 된다.

## Thread

하나의 프로세스 안에서 가동되는 작업 단위.

기본적으로 모든 프로세스는 싱글 쓰레드 단위로 작업을 수행한다.

두 개 이상의 쓰레드를 가동하게 되면 동시에 쓰레드 수 만큼의 작업을 수행할 수 있게 된다.

## 일반적인 프로그램 구현 (싱글 쓰레드 형태)

```
import time
import threading
```

```
# 다소 시간이 소요되는 작업을 가정한 함수 (ex: OpenAPI 연동 등)
def sum(myname: str, low: int, high: int) → None:
    total: int = 0
```

비동기 처리

Process

Thread

일반적인 프로그램 구현 (싱글 스레드 형태)

멀티 스레딩 형태 구현

```
for i in range(low, high):  
    total += i  
    time.sleep(1)  
    print("[%s] total=%d" % (myname, total))
```

```
sum("A", 1, 5)  
sum("B", 5, 10)  
sum("C", 10, 15)  
print("Done !!")
```

```
[A] total=1  
[A] total=3  
[A] total=6  
[A] total=10  
[B] total=5  
[B] total=11  
[B] total=18  
[B] total=26  
[B] total=35  
[C] total=10  
[C] total=21  
[C] total=33  
[C] total=46  
[C] total=60  
Done !!
```

## 멀티 스레딩 형태 구현

## 비동기 처리

Process

Thread

일반적인 프로그램 구현 (싱글 쓰레드 형태)

멀티 쓰레딩 형태 구현

```
# threading.Thread 클래스를 상속받는 클래스 정의
class MyThread(threading.Thread):
    # 함수 구현시 전달하는 파라미터는 생성자로 전달하여 멤버변수에 저장
    def __init__(self, myname: str, low: int, high: int):
        # 상위 클래스에 대한 초기화 --> 멀티 쓰레딩 초기화
        threading.Thread.__init__(self)

        # 전달된 파라미터를 멤버변수에 복사
        self.myname = myname
        self.low = low
        self.high = high

    # 처리 해야 하는 작업을 run() 이름의 메서드로 구현
    # 이 메서드의 이름은 사전에 약속되어 있으며 파라미터를 전달할 수 없다.
    def run(self):
        total: int = 0

        for i in range(self.low, self.high):
            total += i
            time.sleep(1)
            print("[%s] total=%d" % (self.myname, total))
```

```
# Thread 객체 생성
a = MyThread("A", 1, 5)
b = MyThread("B", 5, 10)
c = MyThread("C", 10, 15)

# 각 Thread 가동 --> 해당 객체의 run() 메서드가 호출된다.
a.start()
```

## 비동기 처리

Process

Thread

일반적인 프로그램 구현 (싱글 쓰레드 형태)

멀티 쓰레딩 형태 구현

```
b.start()  
c.start()
```

# 각 Thread의 작업이 종료될까지 대기

```
a.join()  
b.join()  
c.join()
```

```
print("Done!!!")
```

```
[C] total=10  
[B] total=5  
[A] total=1  
[C] total=21[A] total=3  
[B] total=11
```

```
[B] total=18  
[C] total=33  
[A] total=6  
[B] total=26[A] total=10  
[C] total=46
```

```
[B] total=35[C] total=60
```

```
Done!!!
```