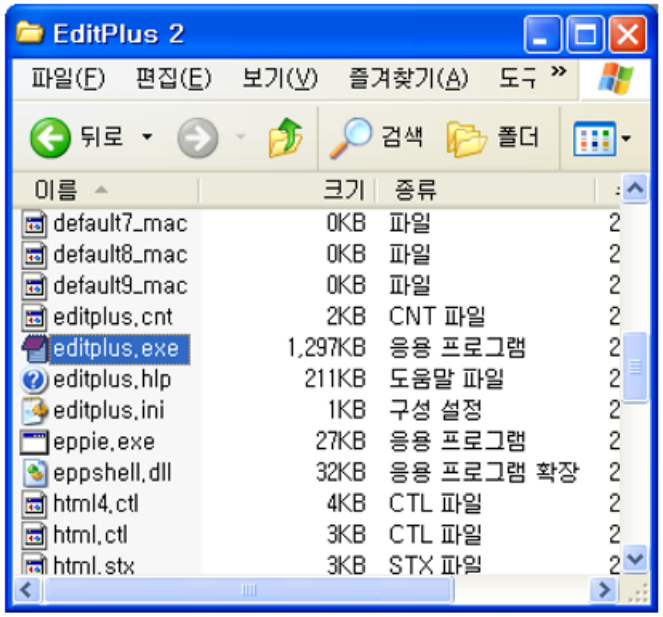
프로세스(process) 와 쓰레드(thread)

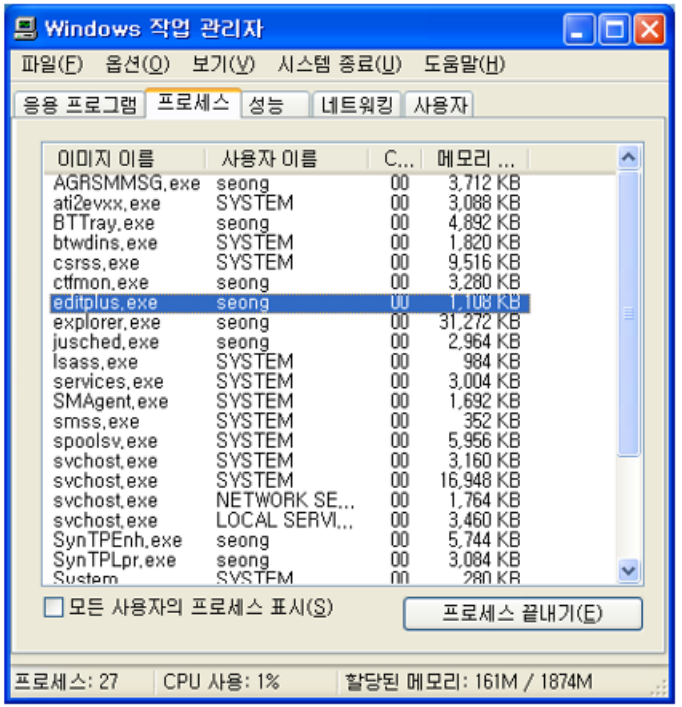
실행

프로그램 → 프로세스

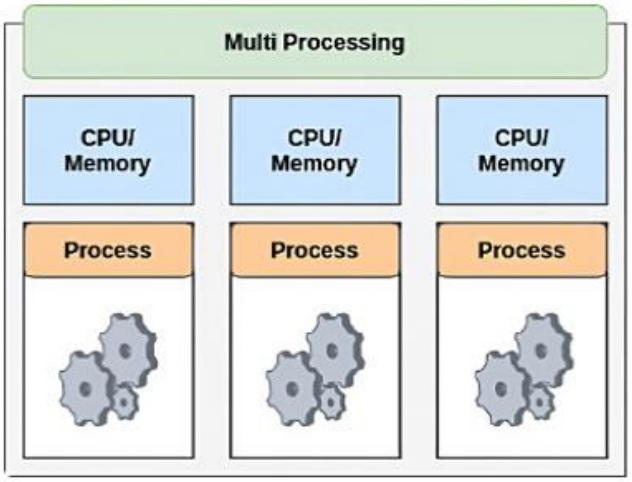
▶ 프로그램 : 실행 가능한 파일(HDD)

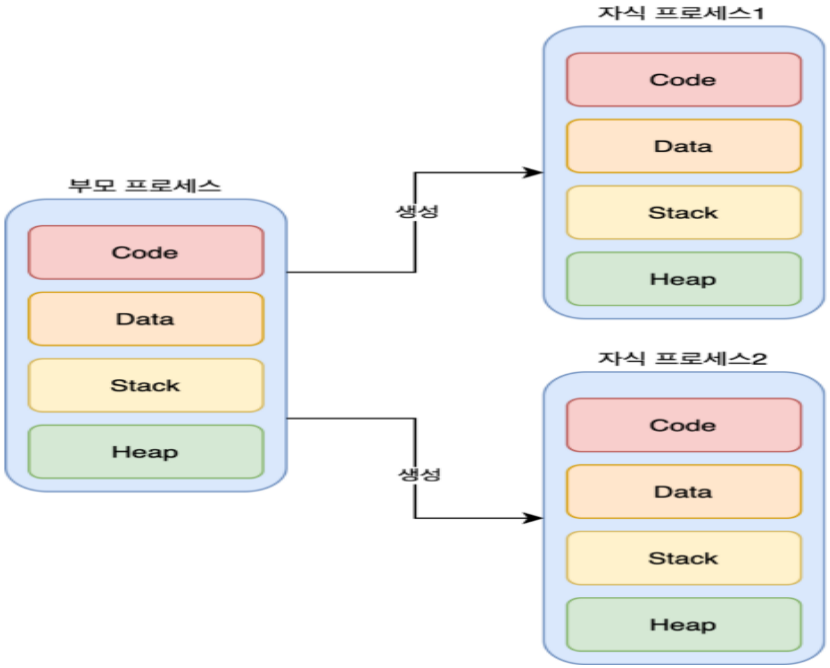


▶ 프로세스 : 실행 중인 프로그램(메모리)

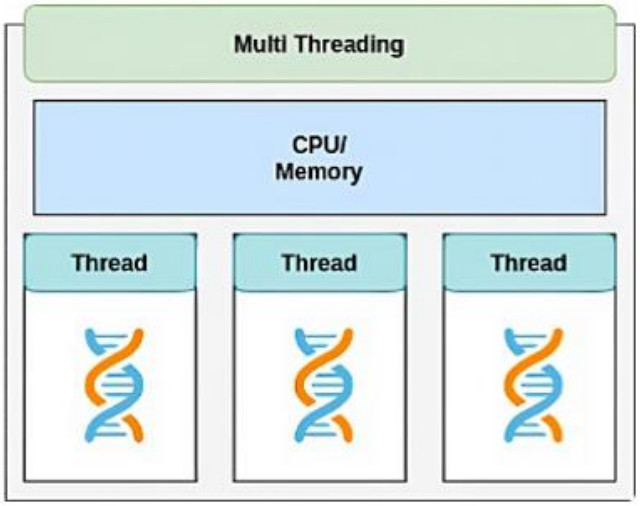


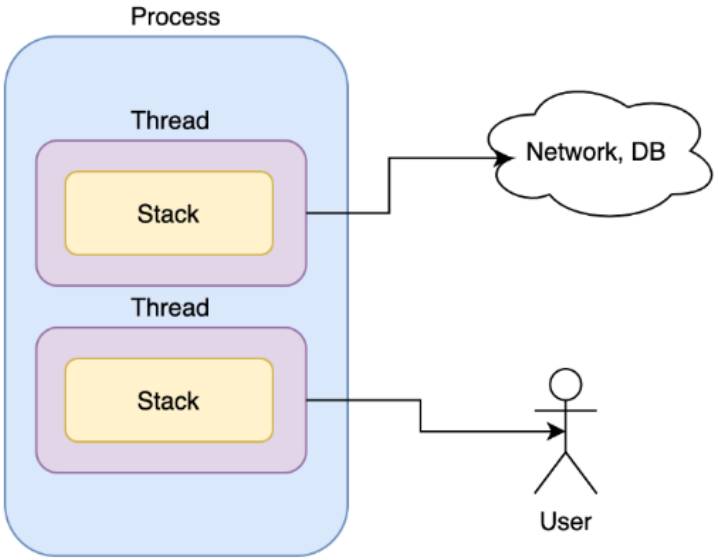
각 Process에 독립적으로 자원을 할당해서 사용





Thread는 Process에 할당된 자원을 공유해서 사용





▶ 프로세스 : 실행 중인 프로그램, 자원(resources)과 쓰레드로 구성

▶ 쓰레드 : 프로세스 내에서 실제 작업을 수행.

모든 프로세스는 하나 이상의 쓰레드를 가지고 있다.

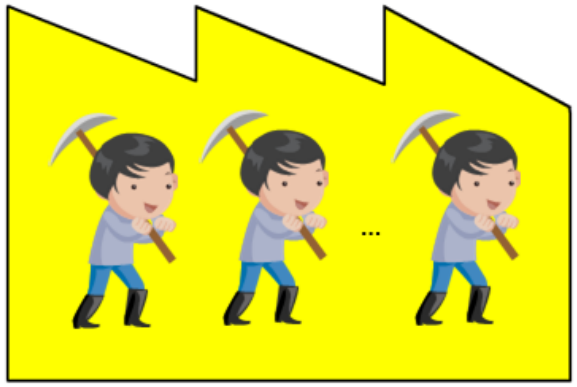
프로세스 : 쓰레드 = 공장 : 일꾼

▶ 싱글 쓰레드 프로세스

= 자원+ 쓰레드

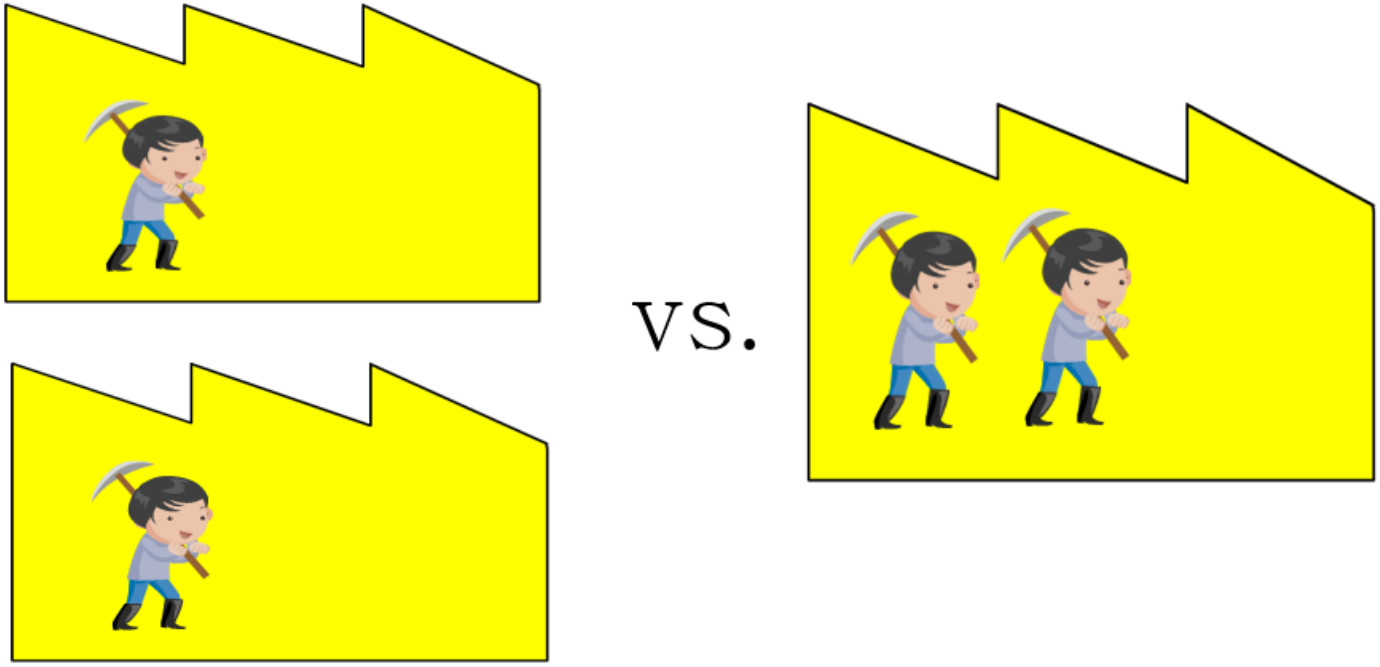
▶ 멀티 쓰레드 프로세스

= 자원+ 쓰레드+ 쓰레드+ …+ 쓰레드



"하나의 새로운 프로세스를 생성하는 것보다
하나의 새로운 쓰레드를 생성하는 것이 더 적은 비용이 든다."

- 2 프로세스 1 쓰레드 vs. 1 프로세스 2 쓰레드



"많은 프로그램들이 멀티쓰레드로 작성되어 있다.
그러나, 멀티쓰레드 프로그래밍이 장점만 있는 것은 아니다."

장점	- 자원을 보다 **효율적**으로 사용할 수 있다. - 사용자에 대한 **응답성(responseness)이 향상**된다. - 작업이 분리되어 **코드가 간결**해 진다. **"여러 모로 좋다."**
단점	- 동기화(synchronization)에 주의해야 한다. - 교착상태(dead-lock)가 발생하지 않도록 주의해야 한다. - 각 쓰레드가 효율적으로 고르게 실행될 수 있게 해야 한다. **"프로그래밍할 때 고려해야 할 사항들이 많다."**