

15-3. Foreground/Background Process

Chapter 15 Process

포그라운드 작업과 백그라운드 작업

- 사용자가 터미널에서 작업할 때 보통의 경우 한 번에 하나의 프로세스만 실행할 수 있다. 즉, 어떤 명령을 실행하면 명령이 끝날 때까지 기다려야 다시 터미널을 사용할 수 있는 것이다.
- 작업 제어(job control) 도구는 한 터미널에서 동시에 여러 프로세스를 실행하고 관리하기 위해 개발되었다.

포그라운드 작업(Foreground Process)

- 사용자가 입력한 명령이 실행되어 결과가 출력될 때까지 기다려야 하는 방식으로 처리되는 프로세스
- 이를 작업 제어에서는 포그라운드 작업이라고 한다.

```
[user1@itserver ~]$ sleep 100
```

백그라운드 작업(Background Process)

- 포그라운드 작업은 명령을 한번에 하나씩 실행하므로 동시에 여러 개의 프로세스를 실행할 수 없다.
- 그러나 작업 제어가 제공하는 백그라운드 기능을 사용하면 앞에서 포그라운드 프로세스가 실행되는 동안에 뒤에서 다른 프로세스가 실행될 수 있으므로 한 터미널에서 여러 개의 프로세스를 동시에 실행할 수 있다.
- 이와 같이 백그라운드 방식으로 처리되는 프로세스를 백그라운드 프로세스라고 하며, 작업 제어에서는 백그라운드 작업이라고 한다.
- 명령을 백그라운드로 실행하려면 명령의 마지막에 **& 기호**를 추가하면 된다.

```
[user1@itserver ~]$ sleep 100&
```

백그라운드 작업과 출력 방향 전환하기

- 백그라운드 작업의 실행이 끝나면 현재 터미널에서 무엇을 하고 있는지에 상관없이 결과가 그대로 화면에 출력된다.
- 기존 작업의 화면과 백그라운드 작업 결과가 뒤섞여 제대로 확인하기 어려워진다.
- 그래서 백그라운드로 처리할 때는 출력과 오류 방향을 전환하여 그 결과를 파일로 저장하는 방법을 많이 사용한다.

```
[user1@itserver ~]$ find / -name passwd > pw.dat 2>&1 &
```

작업 제어

- 작업 전환과 작업 일시 중지, 작업 종료를 뜻한다.
 - 작업 전환: 포그라운드 작업을 백그라운드 작업으로, 또는 백그라운드 작업을 포그라운드 작업으로 전환하는 것
 - 작업 일시 중지: 작업을 잠시 중단하는 것
 - 작업 종료: 프로세스를 종료하는 것처럼 작업을 종료하는 것

작업 목록 보기

- **jobs**

기능	백그라운드 작업을 모두 보여준다. 특정 작업번호를 지정하면 해당 작업의 정보만 보여준다.
형식	jobs %작업번호
%작업번호	%작업번호: 해당 번호의 작업 정보를 출력
사용 예	jobs %1 jobs

- jobs 명령의 출력 항목

항목	출력 예	의미
작업 번호	[1]	작업 번호로 백그라운드에서 실행할 때마다 순차적으로 증가
작업 순서	+	작업 순서를 나타냄 <ul style="list-style-type: none"> • +: 가장 최근에 접근한 작업 • -: + 작업보다 바로 전에 접근한 작업 • 공백: 그 외의 작업
상태	실행 중	작업 상태를 표시 <ul style="list-style-type: none"> • 실행중(Running): 현재 실행중 • 완료됨(Done): 작업이 정상적으로 종료 • 종료됨(Terminated): 작업이 비정상적으로 종료 • 정지됨(Stopped): 작업이 잠시 중단
명령	sleep 100 &	백그라운드에서 실행중인 명령

작업 전환하기

[Ctrl] + z 또는 stop [%작업번호]	포그라운드 작업을 중지
bg %작업번호	작업번호가 지시하는 작업을 백그라운드 작업으로 전환
fg %작업번호	작업번호가 지시하는 작업을 포그라운드 작업으로 전환

작업 종료하기: [Ctrl] + c

- 포그라운드 작업을 [Ctrl]+c를 입력하면 대부분 종료된다.
- 또 다른 방법은 해당 프로세스의 PID를 찾아 강제로 종료하는 것이다.
 - 백그라운드의 작업은 kill 명령으로 강제 종료해야 한다.
 - kill 명령의 인자로 PID 대신 '%작업번호'를 지정해도 된다.

```
lima@itserver: ~  
lima@itserver:~$ sleep 100%  
sleep: 잘못된 시간 간격 '100%'  
Try 'sleep --help' for more information.  
lima@itserver:~$ sleep 100&  
[1] 2076  
lima@itserver:~$ kill %1  
lima@itserver:~$ jobs  
[1]+  종료됨                  sleep 100  
lima@itserver:~$
```

로그아웃 후에도 백그라운드 작업 계속 실행하기: nohup

- 백그라운드 작업을 실행한 터미널이 종료되거나 사용자가 로그아웃하면 실행 중이던 백그라운드 작업도 같이 종료된다.
- 로그아웃한 다음에도 작업이 완료될 때까지 백그라운드 작업을 실행해야 하는 경우가 있다.
- **nohup**

기능	로그아웃 한 후에도 백그라운드 작업을 계속 실행한다.
형식	nohup 명령&

- nohup 명령은 반드시 백그라운드로 해야 한다.
- 출력 방향 전환을 하지 않으면 명령의 실행 결과와 오류 메시지가 현재 디렉터리에 **nohup.out** 파일에 자동 저장된다.

```
[user1@itserver ~]$ nohup find / -name passwd &
```

```
[user1@itserver ~]$ more nohup.out
```