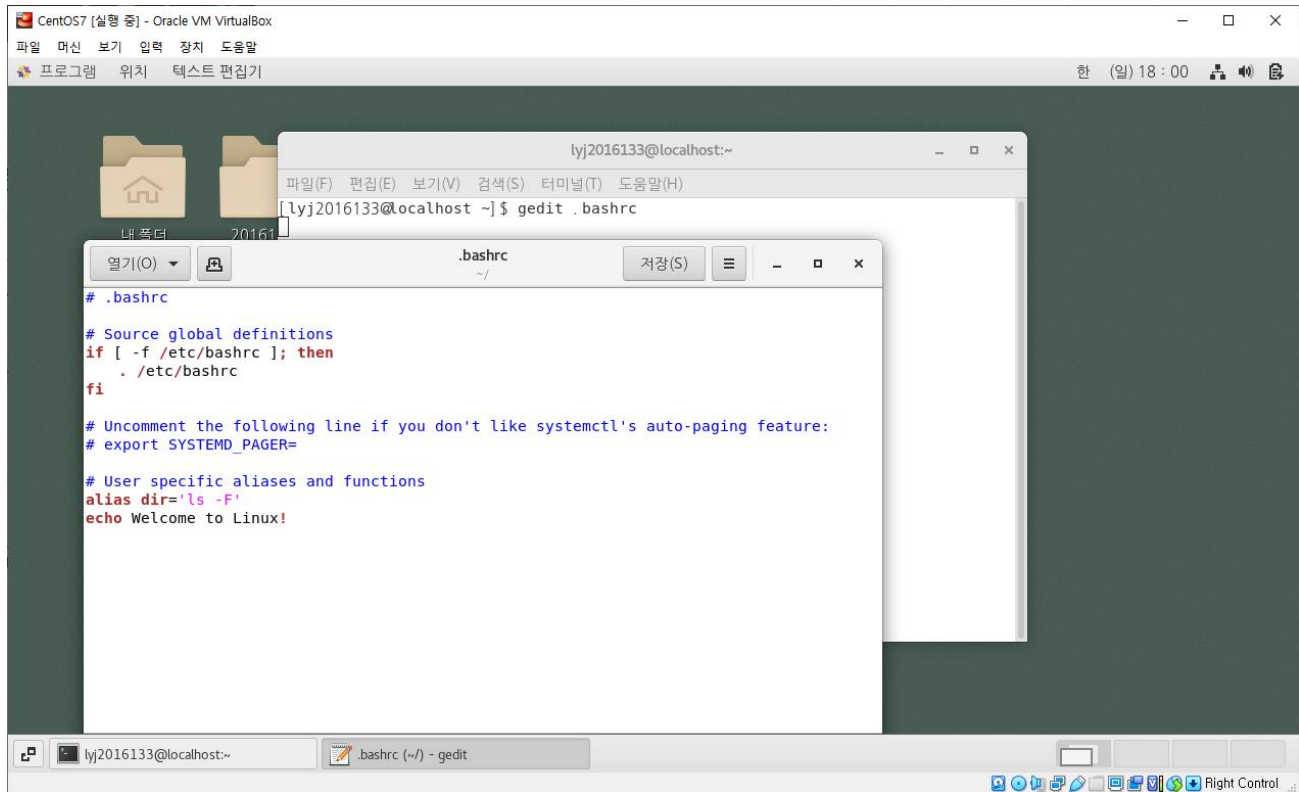


1. 시작 파일 실습

1) gedit으로 파일 .bashrc를 연 후

alias dir = 'ls -F', echo Welcome to Linux!를 입력하여 추가

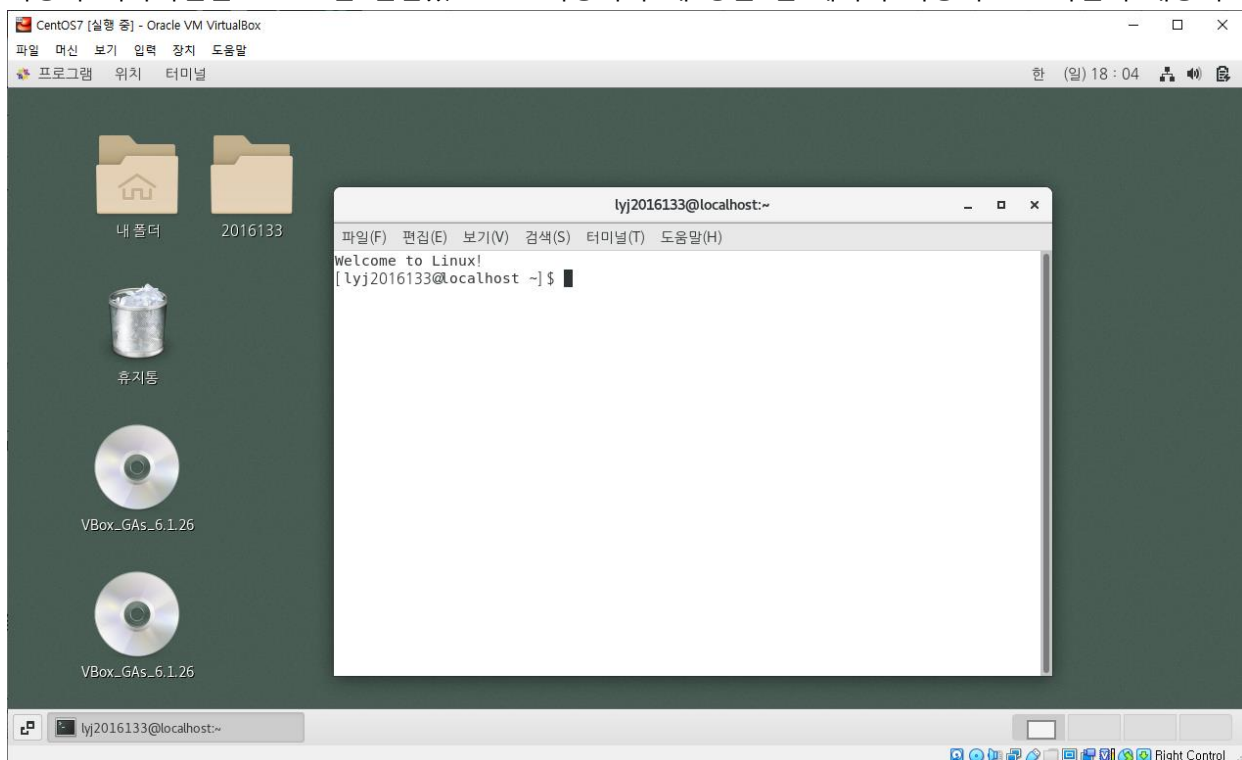


2) 새 창에서 출력된 내용을 확인하고 설명한다..

echo Welcome to Linux! : echo 뒤의 입력 Welcome to Linux!를 화면에 출력함.

(echo 뒤에 변수값을 사용할 경우 \$ 사용이 필수적이다.)

사용자 시작파일인 .bashrc를 편집했으므로 사용자가 새 창을 열 때마다 자동적으로 파일의 내용이 실행됨.

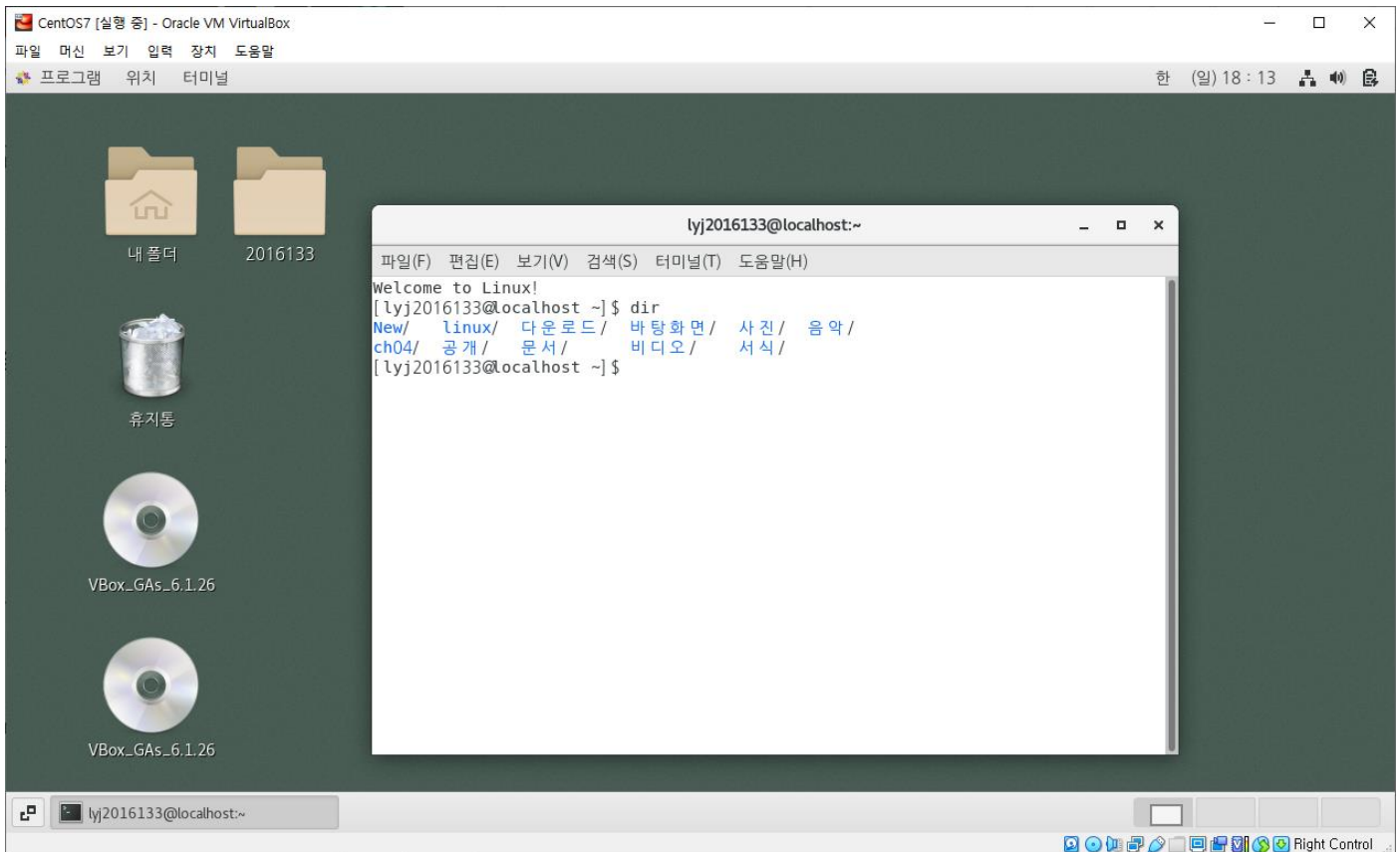


3) 새 창에서 dir 명령어를 실행하고 그 결과를 설명한다.

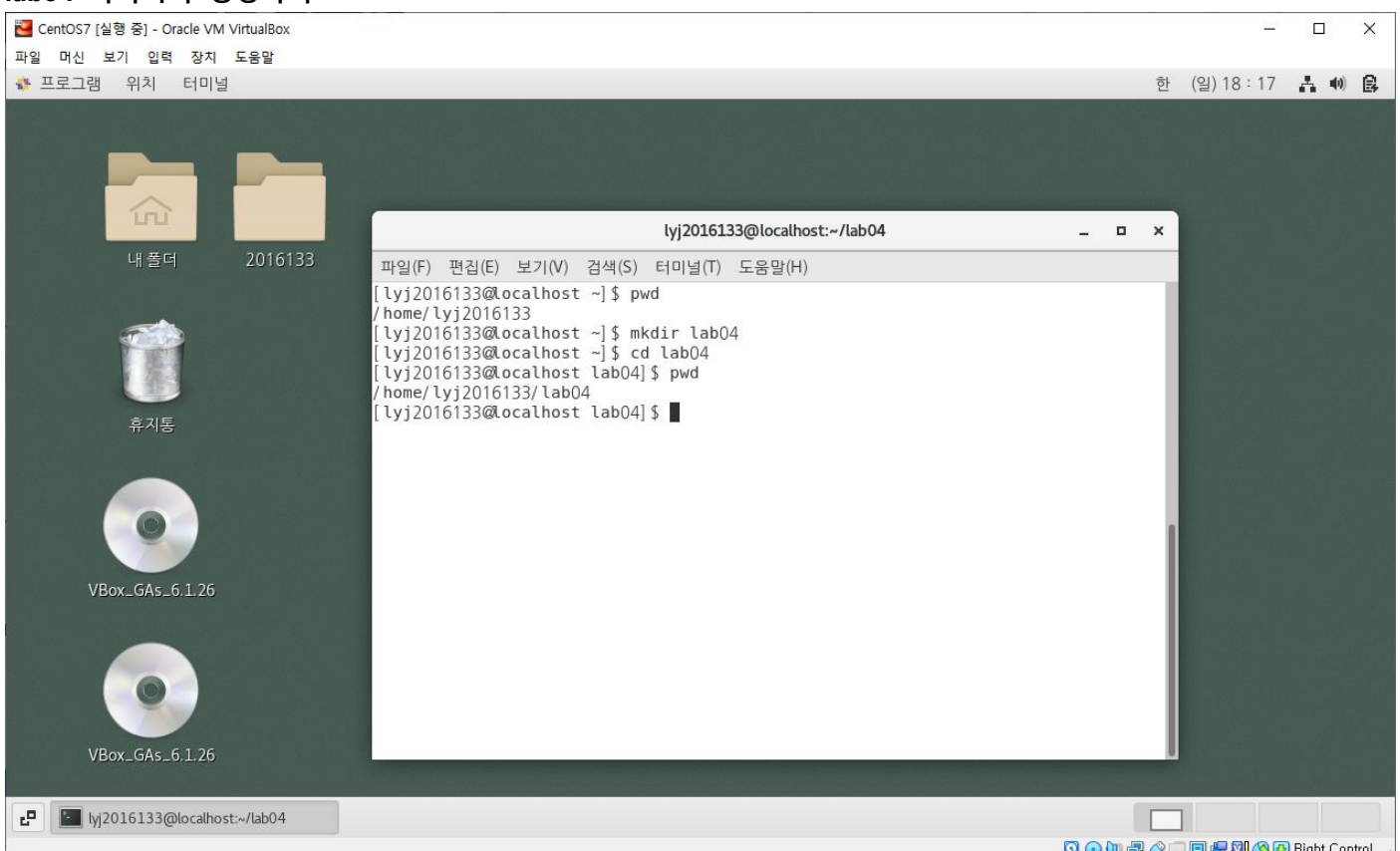
alias 별명 = '명령어' : 명령어를 간소화하여 다른 이름으로 사용할 수 있도록 하는 쉘 내부 명령어이다.

따라서 dir 입력시 ls -F 명령어 실행

사용자 시작파일인 .bashrc를 편집했으므로 사용자가 새 창을 열 때마다 자동적으로 파일의 내용이 실행됨.



lab04 디렉터리 생성하기



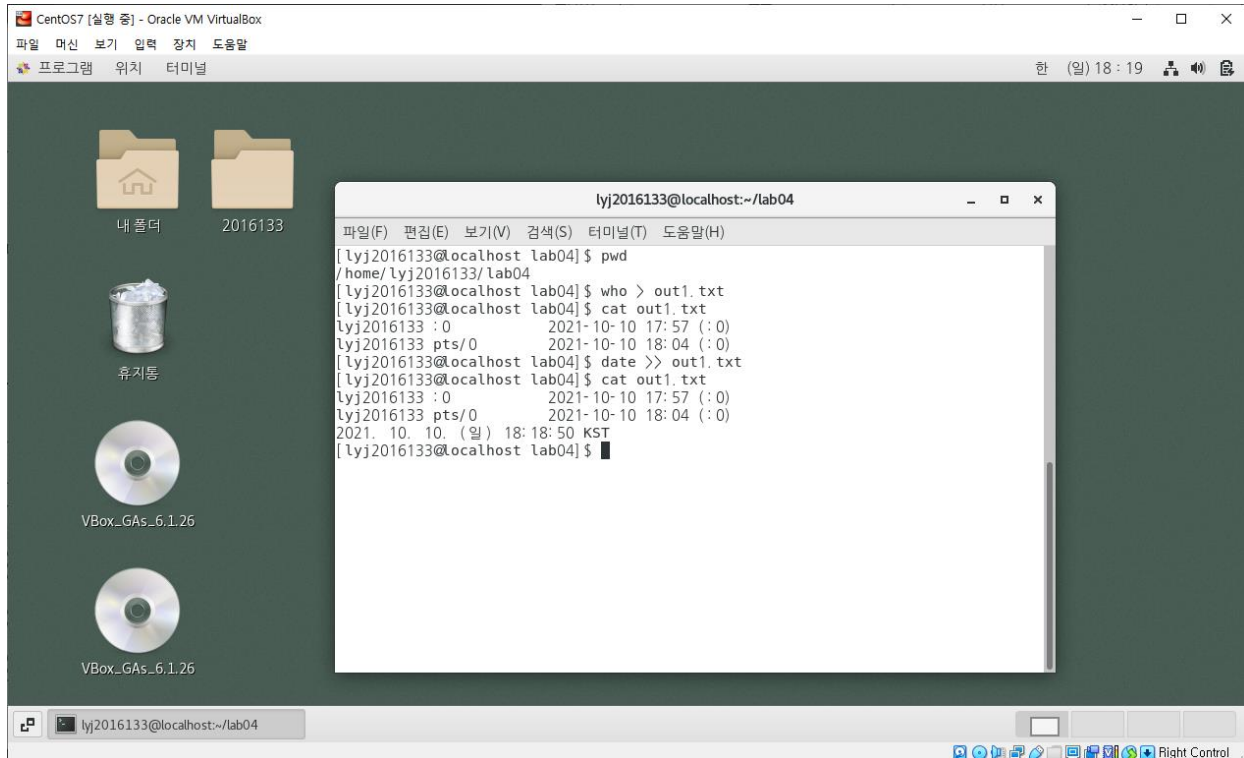
2. 출력 재지정 연습

who > out1.txt : 명령어 who를 통해 list된 사용자들을 out1.txt에 저장함

cat out1.txt : out1.txt를 화면에 출력

date >> out1.txt : date 정보를 out1.txt에 출력 추가(>>)함. overwrite하는 것이 아닌 뒤에 추가하는 것/

cat out1.txt : out1.txt를 화면에 출력 -> date의 출력이 추가된 것을 볼 수 있음



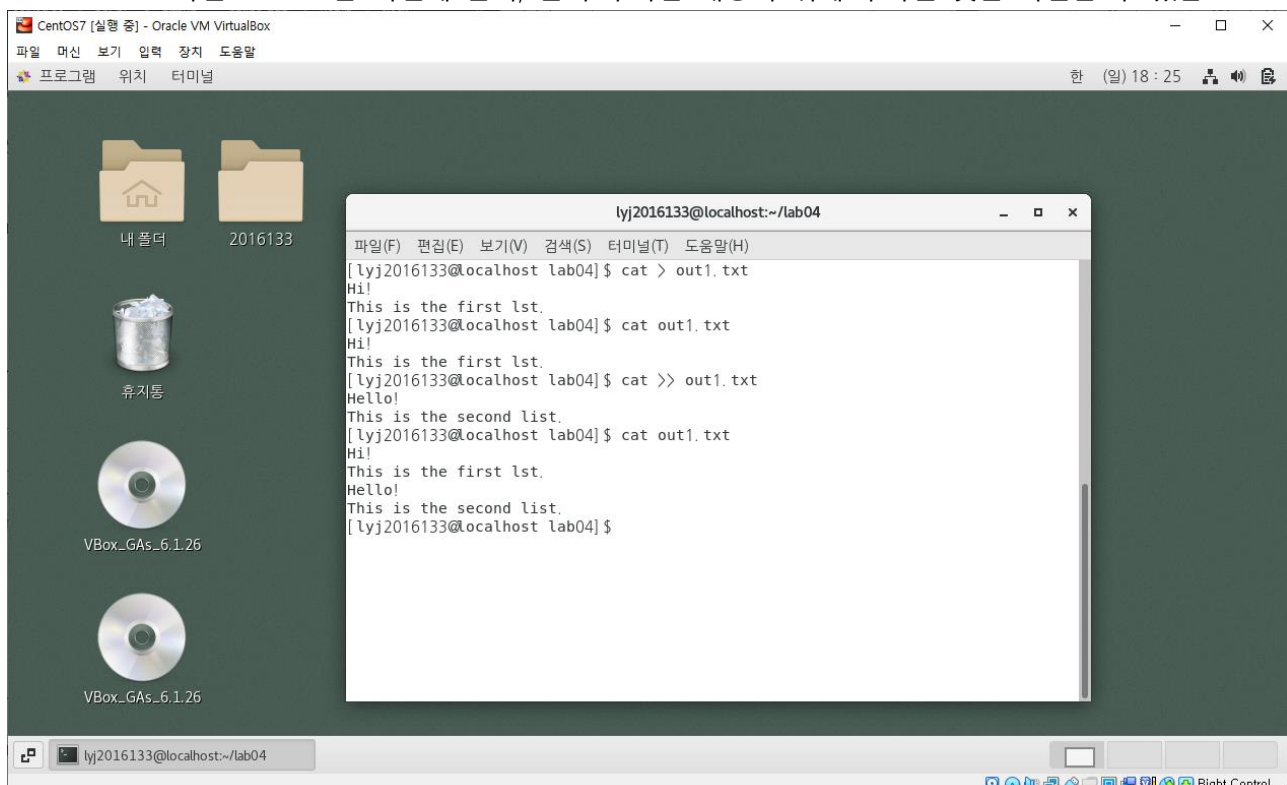
2. 출력 재지정 실습

cat > out1.txt : 입력 받은 내용을 파일 out1.txt에 저장

cat out1.txt : 파일 out1.txt를 화면에 출력

cat >> out1.txt : 파일 out1.txt에 출력 추가(>>), overwrite하는 것이 아닌 뒤에 추가하는 것

cat out1.txt : 파일 out1.txt를 화면에 출력, 출력 추가한 내용이 뒤에 추가된 것을 확인할 수 있음.

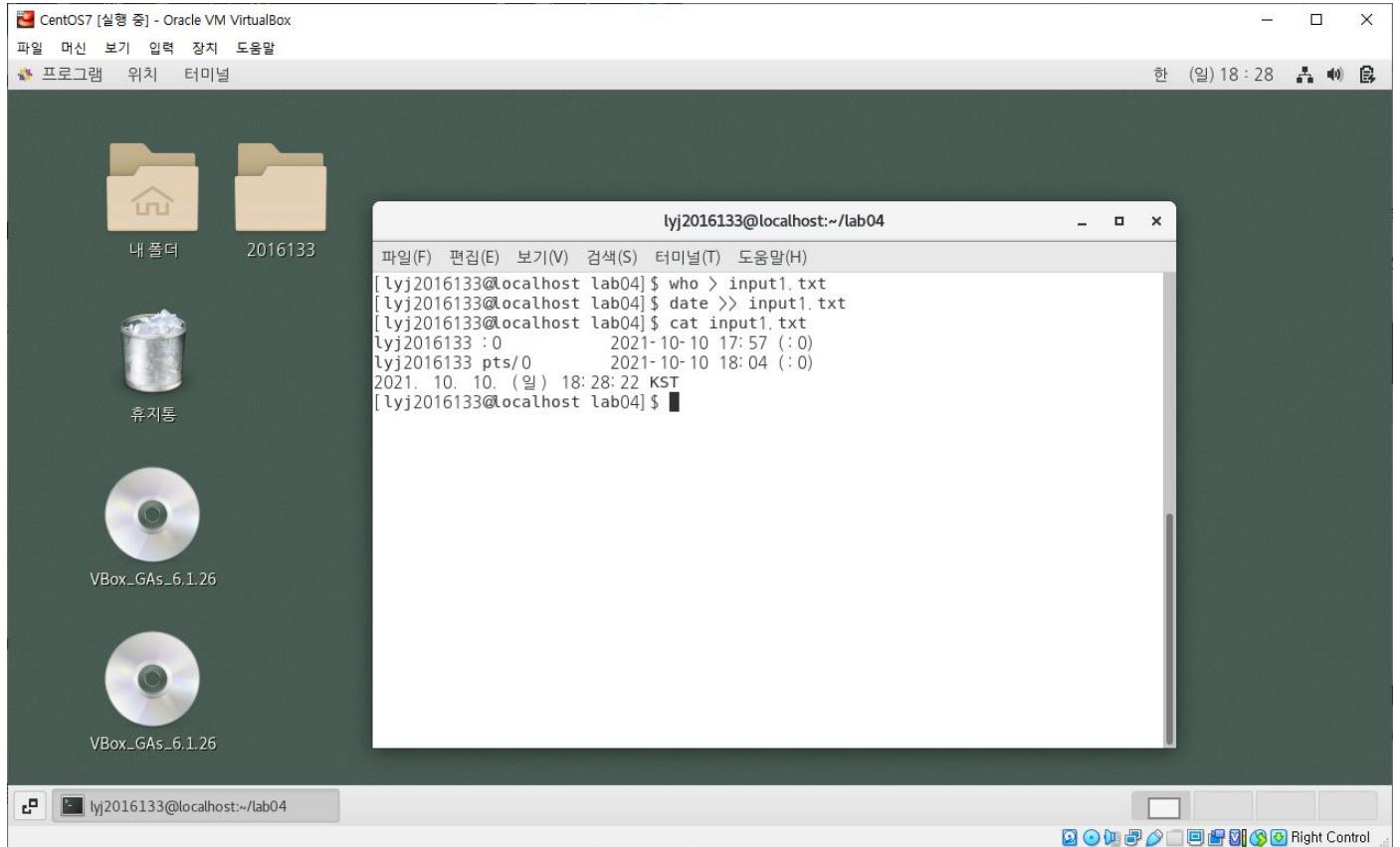


3. 입력 재지정 실습

who > input1.txt : 명령어 who를 통해 list된 사용자들을 input1.txt에 저장함

date >> input1.txt : 명령어 date의 표준출력을 모니터 대신 파일 input1.txt에 추가함

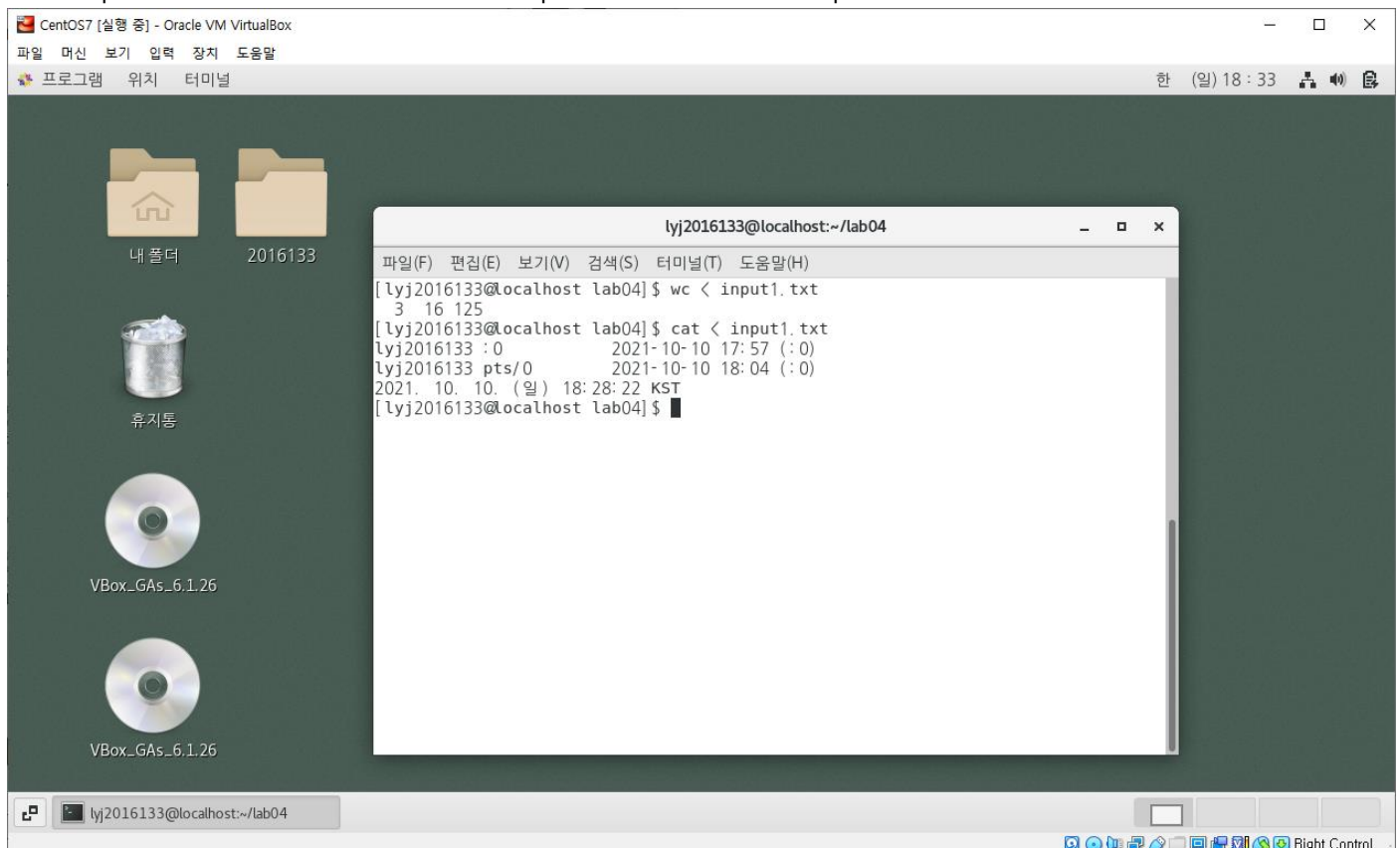
cat input1.txt : 화면에 파일 input1.txt 내용 출력, date의 표준출력이 추가된 것을 확인할 수 있음.



```
lyj2016133@localhost:~/lab04
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[lyj2016133@localhost lab04]$ who > input1.txt
[lyj2016133@localhost lab04]$ date >> input1.txt
[lyj2016133@localhost lab04]$ cat input1.txt
lyj2016133 :0                2021-10-10 17:57 (:0)
lyj2016133 pts/0            2021-10-10 18:04 (:0)
2021. 10. 10. (일) 18:28:22 KST
[lyj2016133@localhost lab04]$
```

wc < input1.txt : wc의 표준 입력을 키보드 대신 파일에서 받음. 즉 input1.txt에 word count 명령어를 실행함.

cat < input1.txt : cat의 표준 입력을 파일 input1.txt에서 받음. 즉 input1.txt를 화면에 출력함



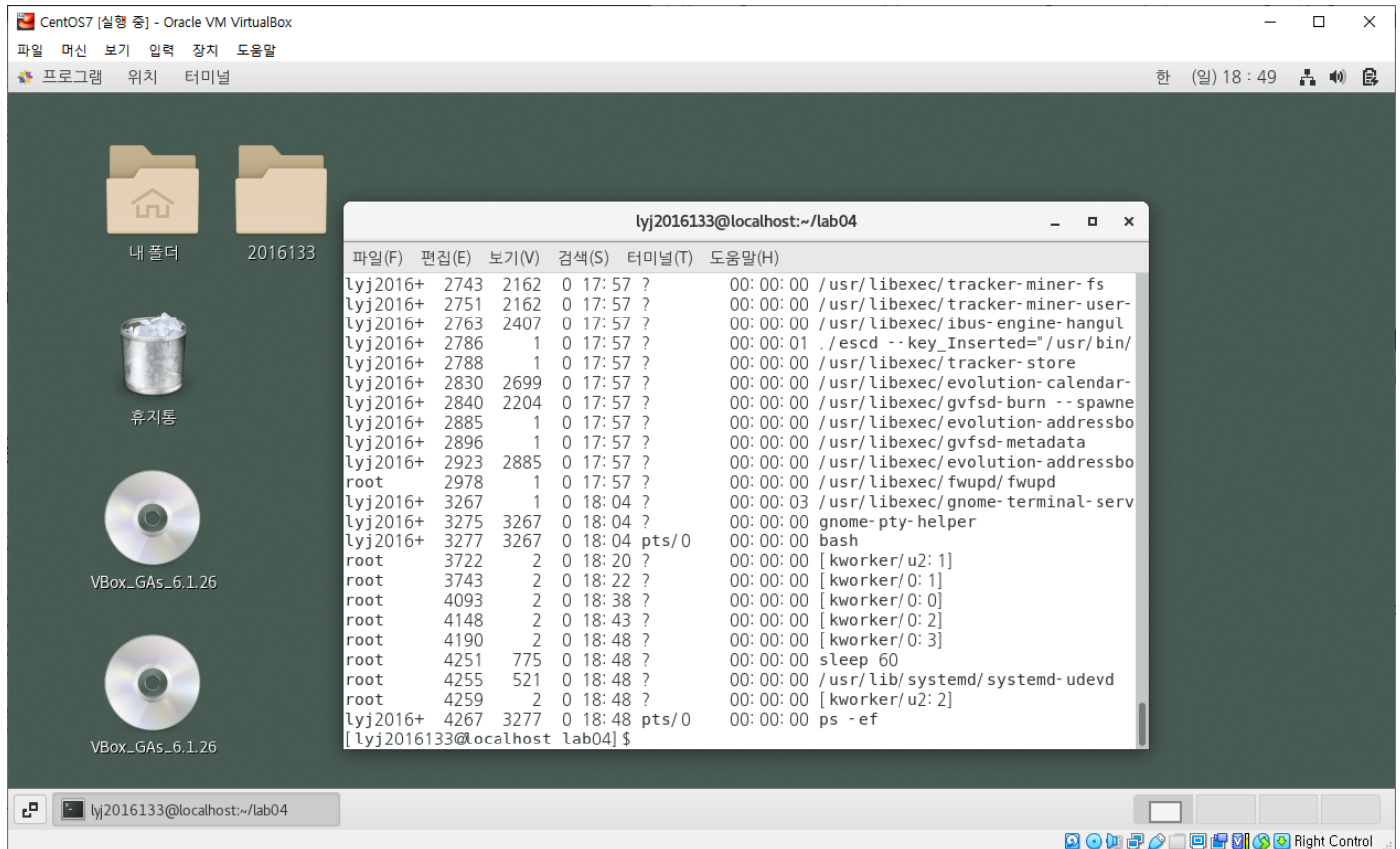
```
lyj2016133@localhost:~/lab04
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[lyj2016133@localhost lab04]$ wc < input1.txt
 3 16 125
[lyj2016133@localhost lab04]$ cat < input1.txt
lyj2016133 :0                2021-10-10 17:57 (:0)
lyj2016133 pts/0            2021-10-10 18:04 (:0)
2021. 10. 10. (일) 18:28:22 KST
[lyj2016133@localhost lab04]$
```


4. 파이프 실습

1) 다음 명령들을 사용하여 특정 사용자의 프로세스들을 리스트하고 그 결과를 설명한다

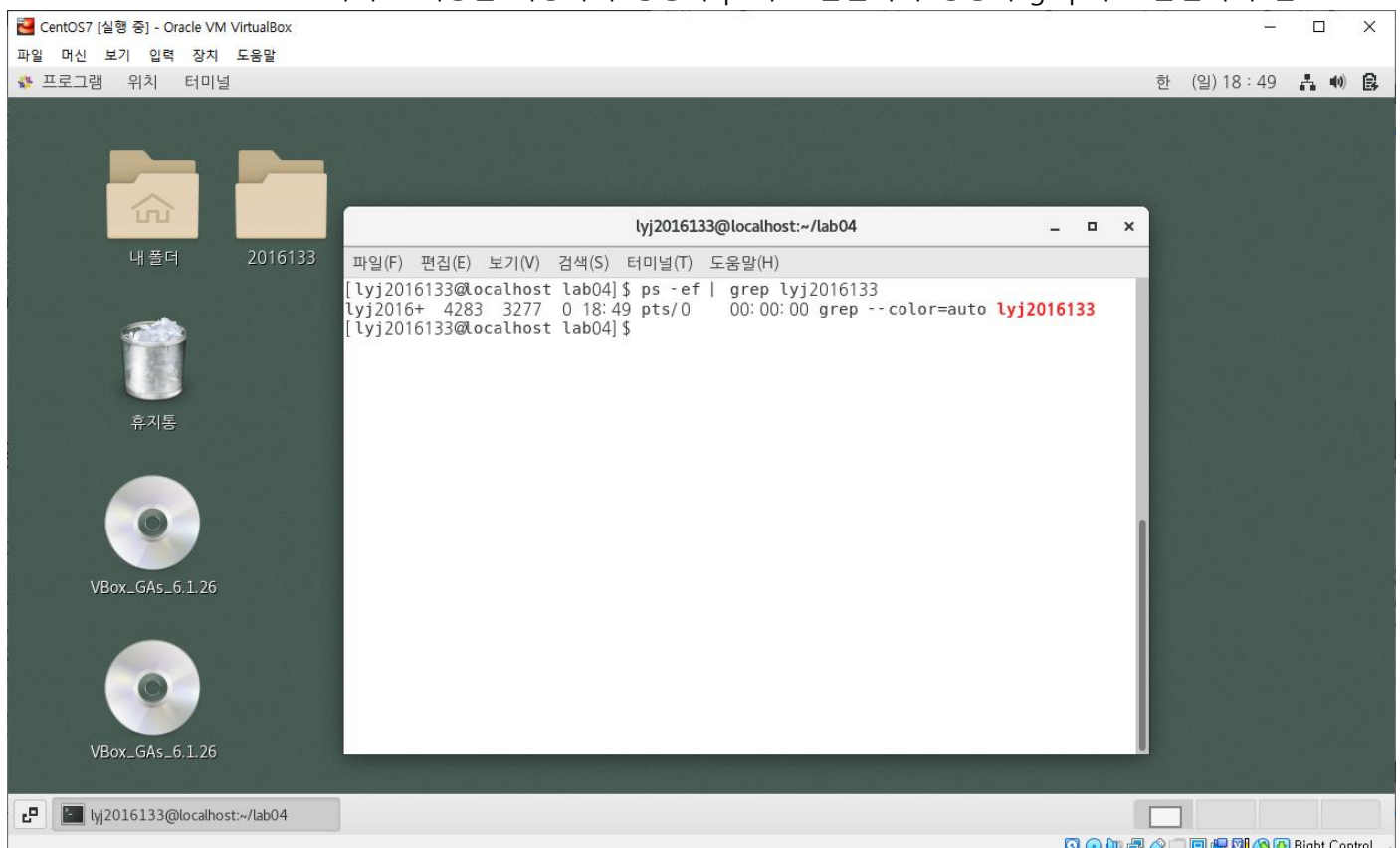
ps -ef : 실행중인 프로세스 리스트를 출력하는 명령어 ps

커널 프로세스를 제외한 모든 프로세스를(-e) 풀 포맷으로 보여준다(-f)



ps -ef | grep lyj2016133 : 특정 문자열을 찾는 명령어 grep, 뒤에 사용자 이름을 입력함으로써 사용자이름이 들어간 파일만을 찾음

파이프 기능을 이용하여 명령어 ps의 표준출력이 명령어 grep의 표준입력이 됨



2) 다음 명령을 사용하여 특정 디렉토리 내의 .c 파일의 개수를 출력하고 그 결과를 설명한다.

touch out1.c, touch out2.c, touch out3.c : 각각 out1.c, out2.c, out3.c 라는 파일 생성

ls ./* .c : 현재 경로에서 .c로 끝나는 모든 파일을 보임

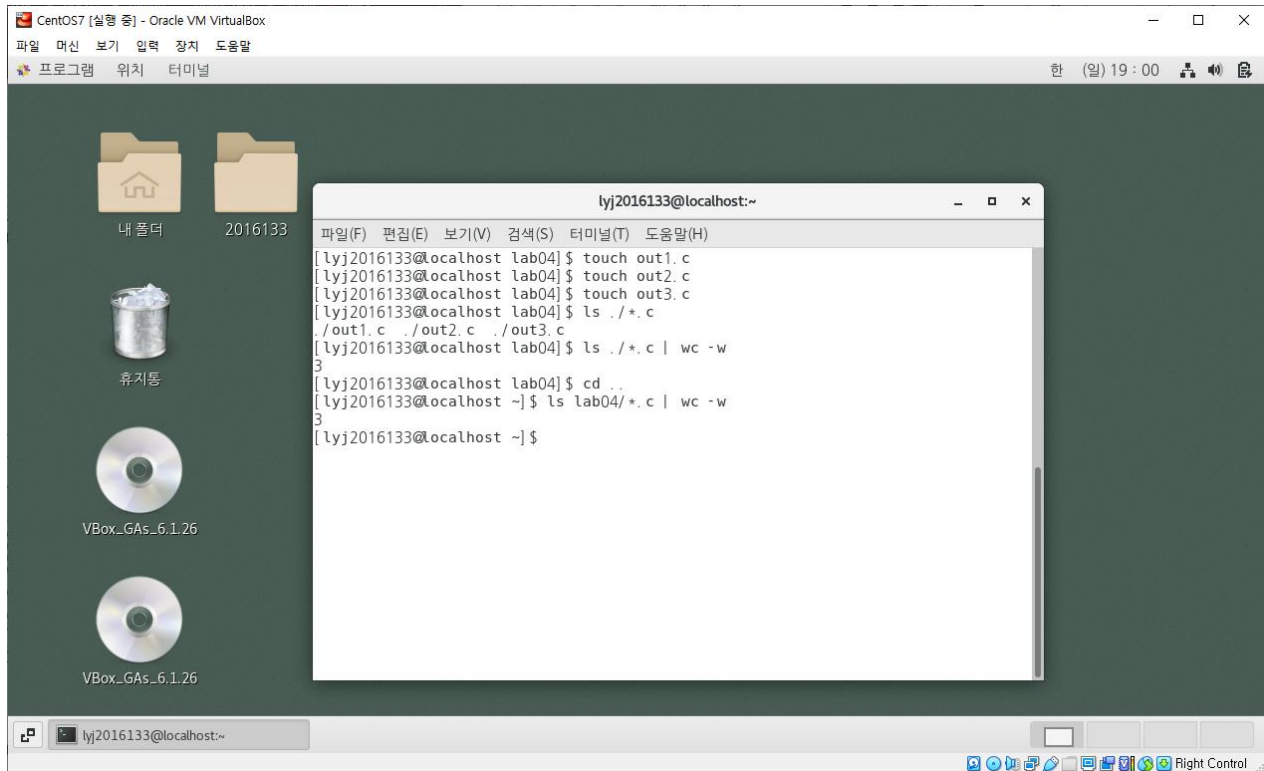
ls ./* .c | wc -w : 파이프기능을 이용하여 ls의 표준출력을 wc의 표준입력으로 이용

현재 경로에서 .c로 끝나는 모든 파일의 개수를 화면에 출력

cd .. : 상위 디렉토리로 이동

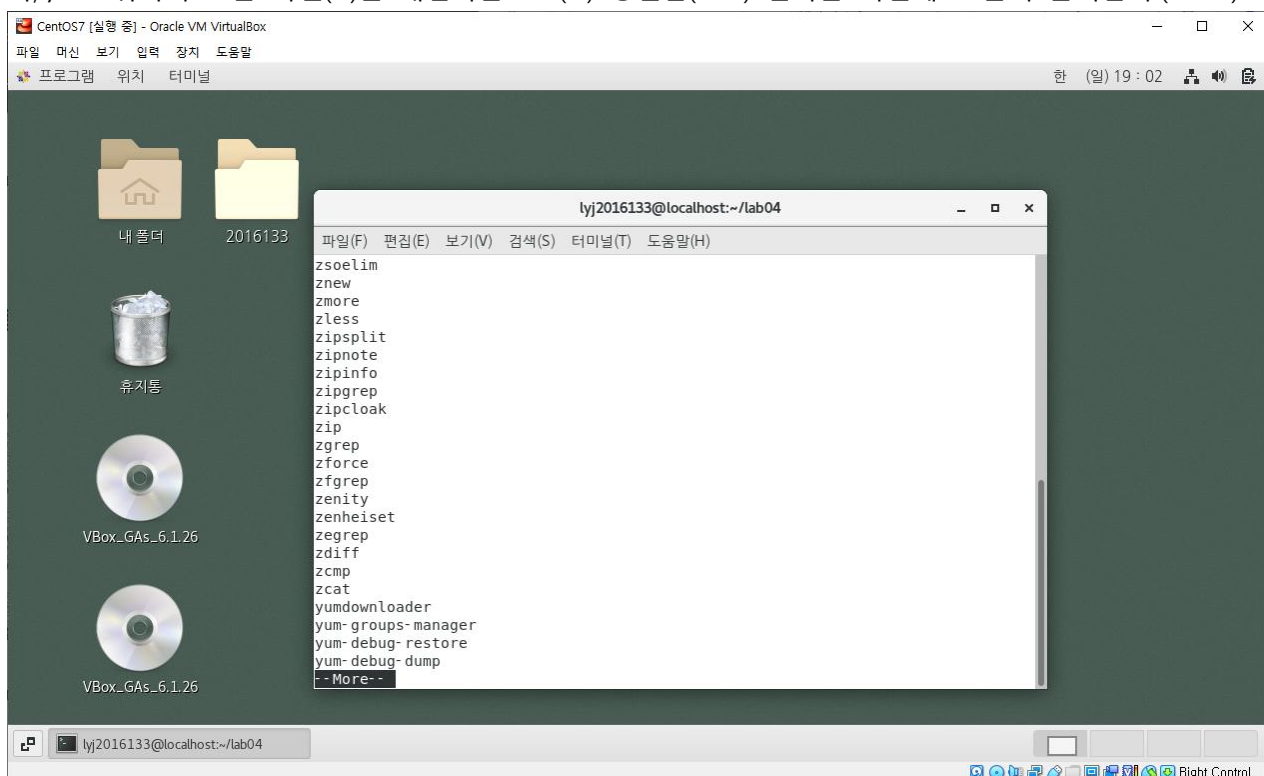
ls lab04/* .c | wc -w : 파이프기능을 이용하여 ls의 표준출력을 wc의 표준입력으로 이용

lab04 경로에서 .c로 끝나는 모든 파일의 개수를 화면에 출력



3) ls /bin | sort -r | more 명령어를 사용하고 그 결과를 설명한다.

파이프 기능을 사용하여 ls의 표준출력을 sort의 표준입력으로, sort의 표준출력을 more의 표준입력으로 이용
즉, /bin 위치의 모든 파일(ls)을 내림차순으로(-r) 정렬한(sort) 결과를 화면에 10줄씩 출력한다.(more)



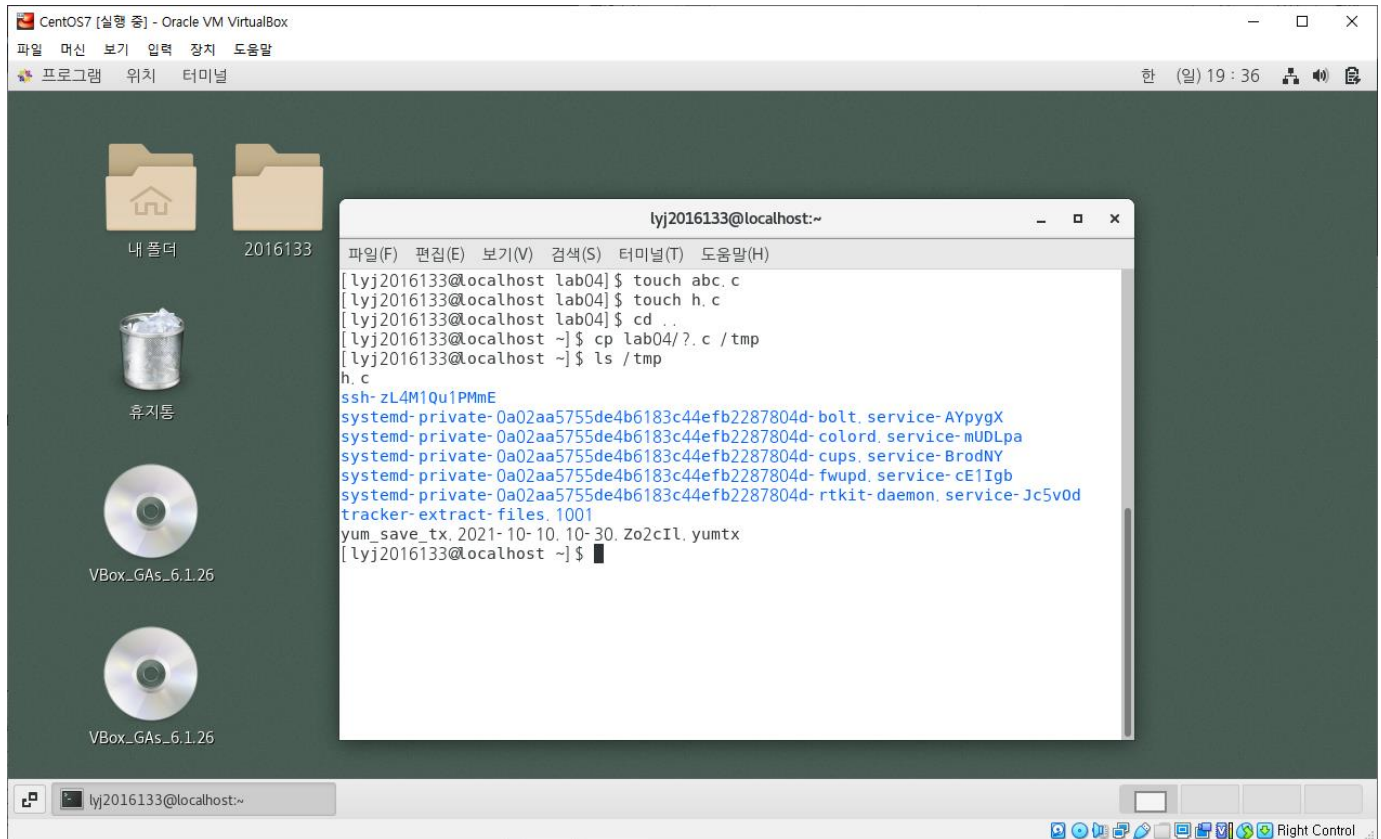
5. 대표문자 실습

예시와 같이 실행하였음.

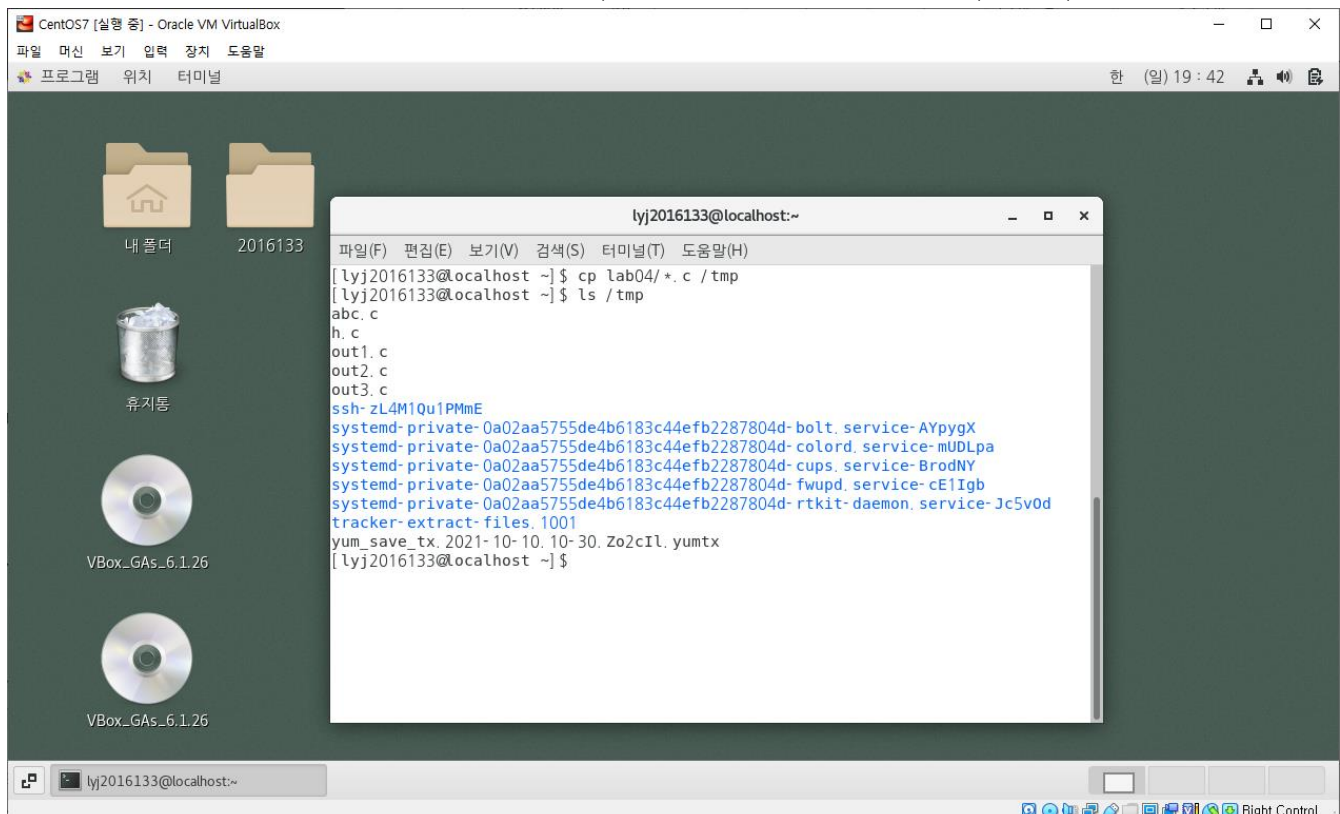
touch abc.c , touch h.c : 각각 abc.c, h.c라는 파일을 생성

cd .. : 상위 디렉터리로 이동

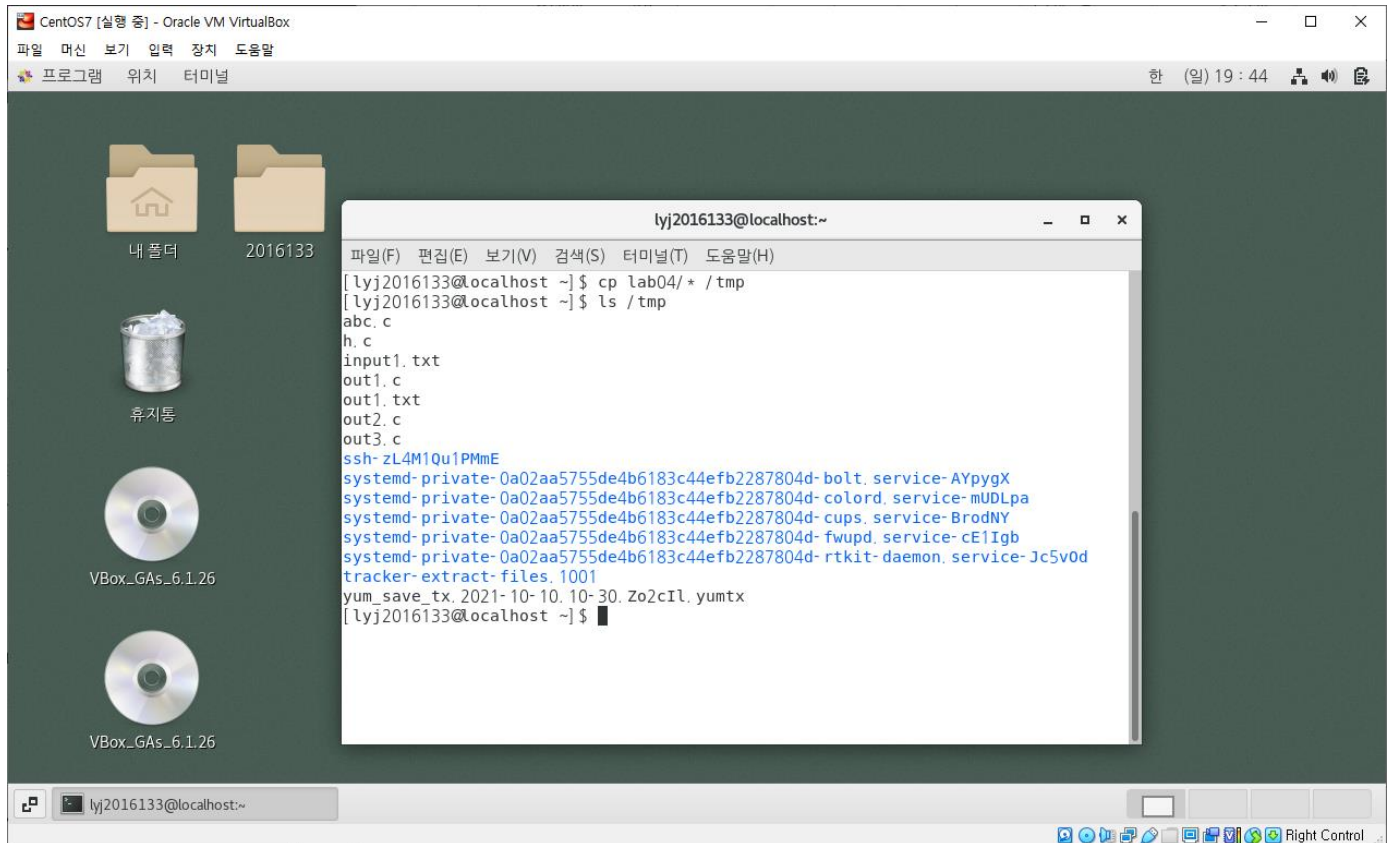
cp lab04/?c tmp : ?는 임의의 한 문자를 대치함. 즉 tmp라는 위치에, lab04에 있는 파일 중 이름이 □.c와 같은 형태로 한 글자인 모든 파일을 복사함 -> h.c만 복사됨



cp lab04/*.c /tmp : *는 임의의 스트링을 대치함. 즉 tmp라는 위치에, lab04에 있는 파일 중 .c로 끝나는 모든 파일을 복사함. -> abc.c, h.c와 기존에 만들었던 out1.c, out2.c, out3.c 모두 복사됨.



cp lab04/* /tmp : *는 임의의 스트링을 대치함. 즉 tmp라는 위치에, lab04에 있는 모든 파일을 복사함
-> 기존에 만들었던 input1.txt도 복사된 것을 확인할 수 있음

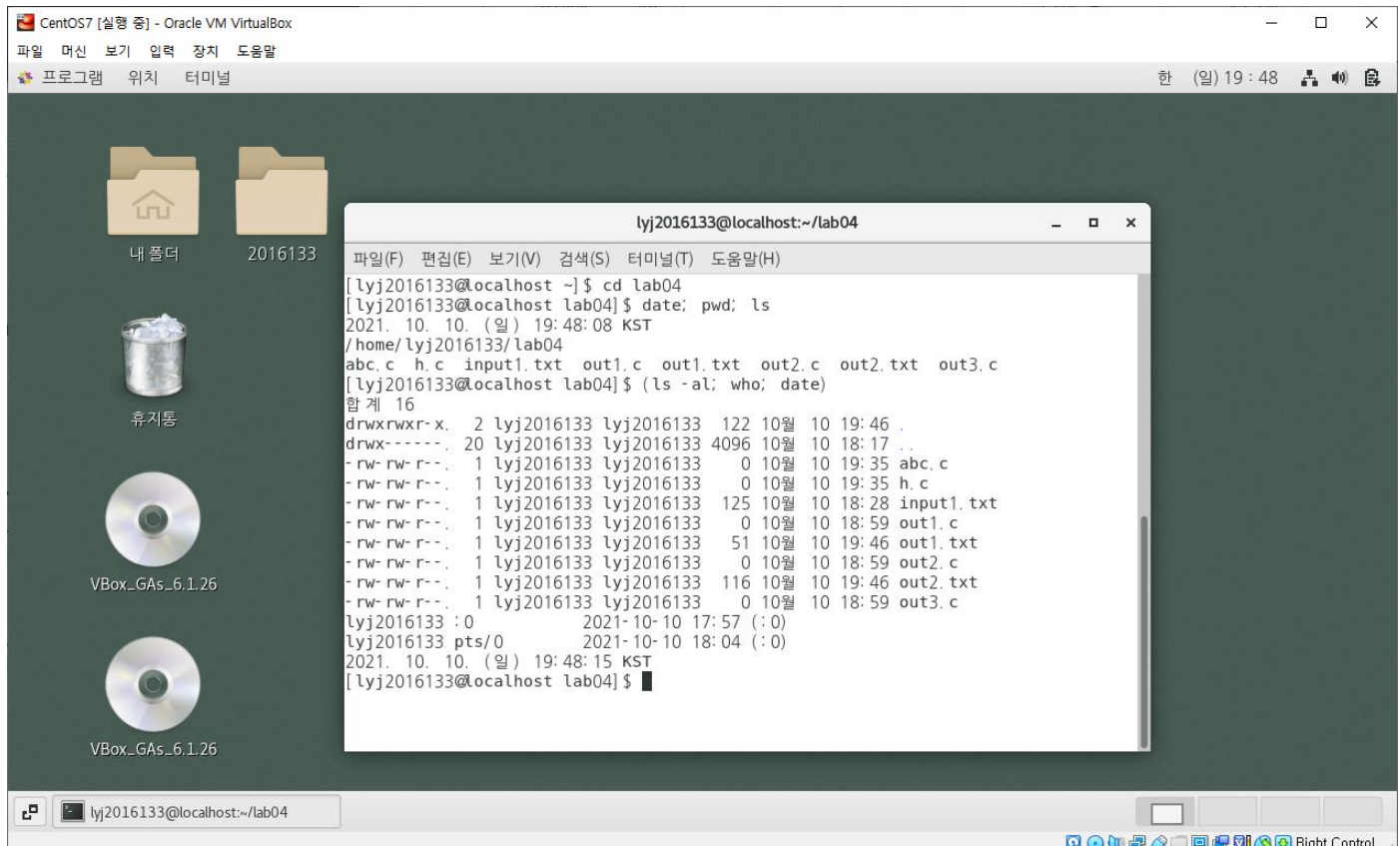


6. 명령어 열/그룹 실습

예시와 같이 실행하였음.

date; pwd; ls : 명령어 열로, 나열된 명령어들을 순차적으로 실행한다. 각각 다른 줄로 화면에 결과가 출력됨.

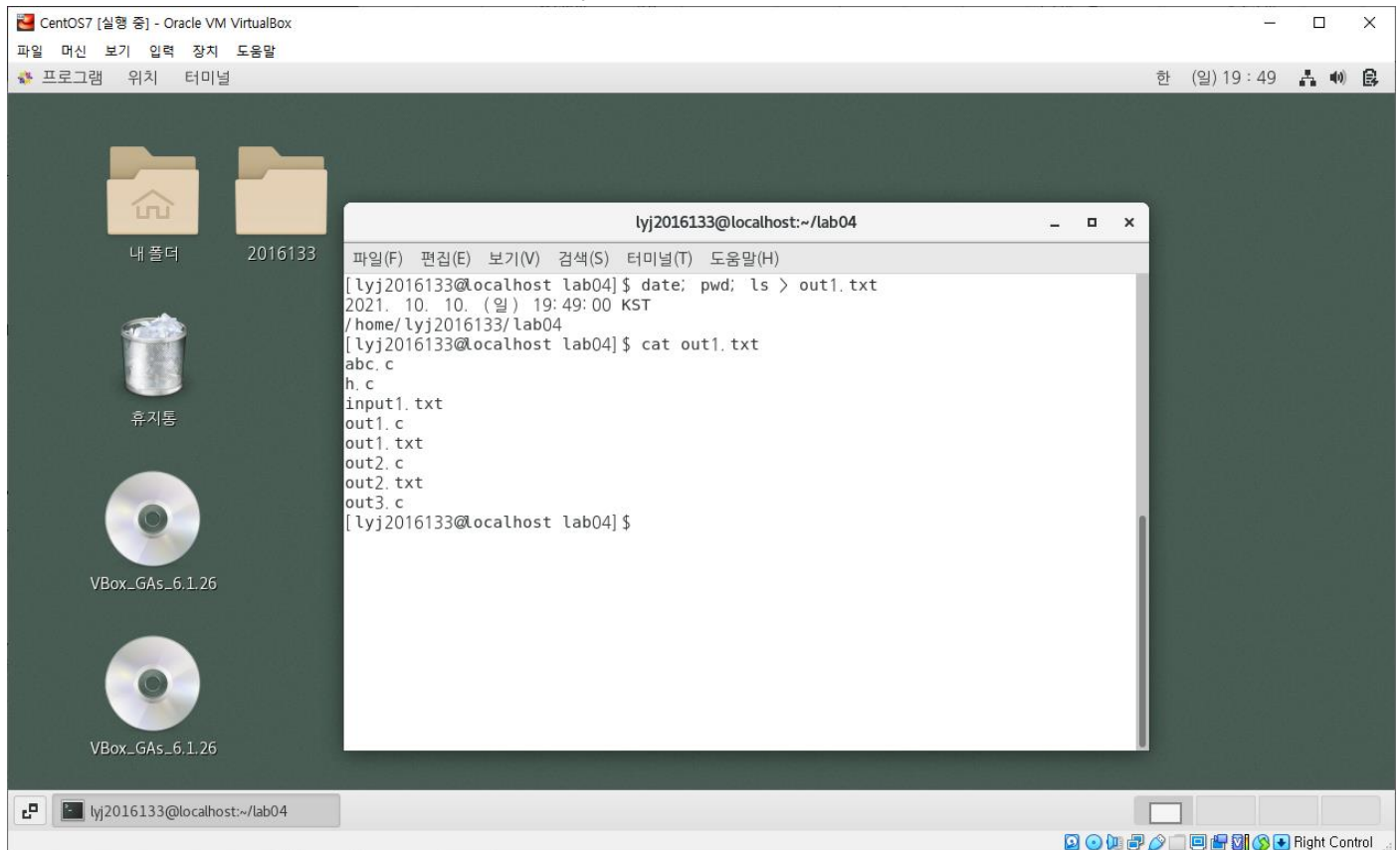
(ls -al; who; date) : 명령어 그룹으로, 나열된 명령어들을 하나의 그룹으로 묶어 순차적으로 실행한다.



date; pwd; ls > out1.txt : 명령어 열을 사용하였으므로 차례로 명령어가 실행되고,

마지막 ls만 출력 재지정으로 결과가 out1.txt에 저장됨.

cat out1.txt : out1.txt의 내용을 출력함, date와 pwd에 대한 내용은 저장되지 않은 것을 확인할 수 있음.



(date; pwd; ls) > out2.txt : 명령어 그룹을 사용하였으므로 차례로 명령어가 실행되고 이 전체 결과가

출력 재지정으로 out2.txt에 저장됨

cat out2.txt : out2.txt의 내용을 출력함, date와 pwd의 내용도 저장된 것을 확인할 수 있음.

