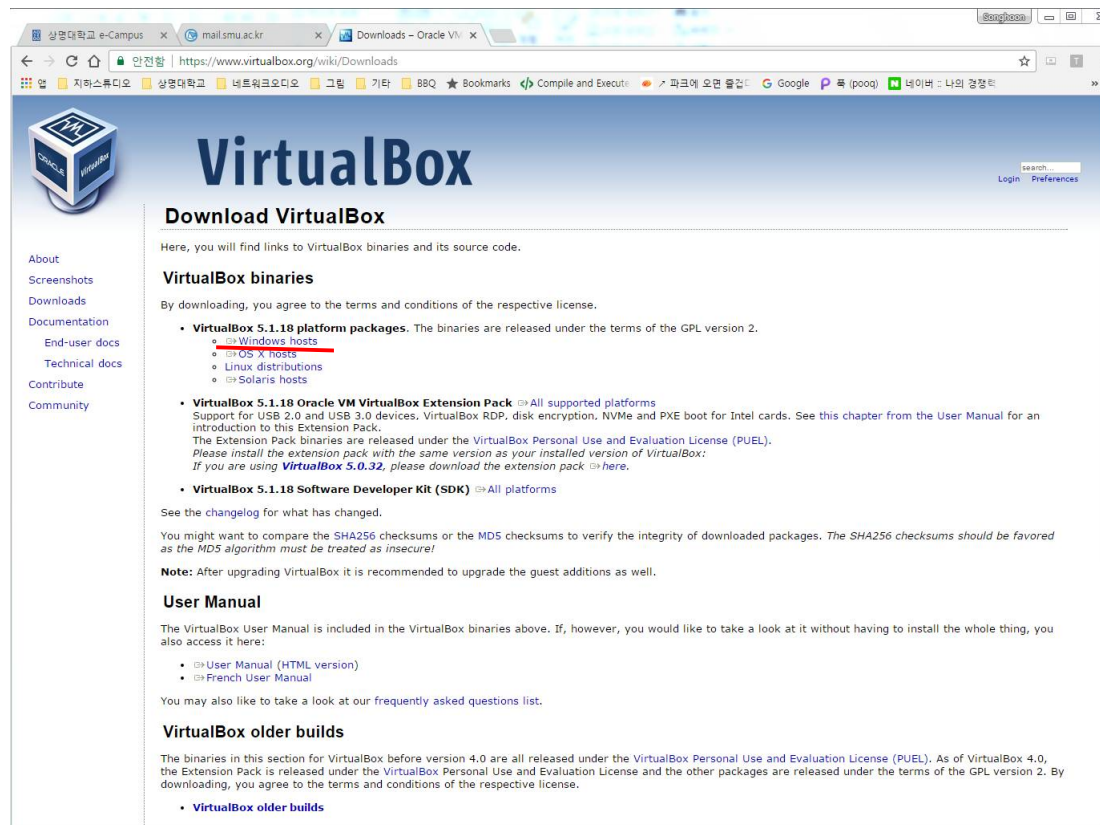
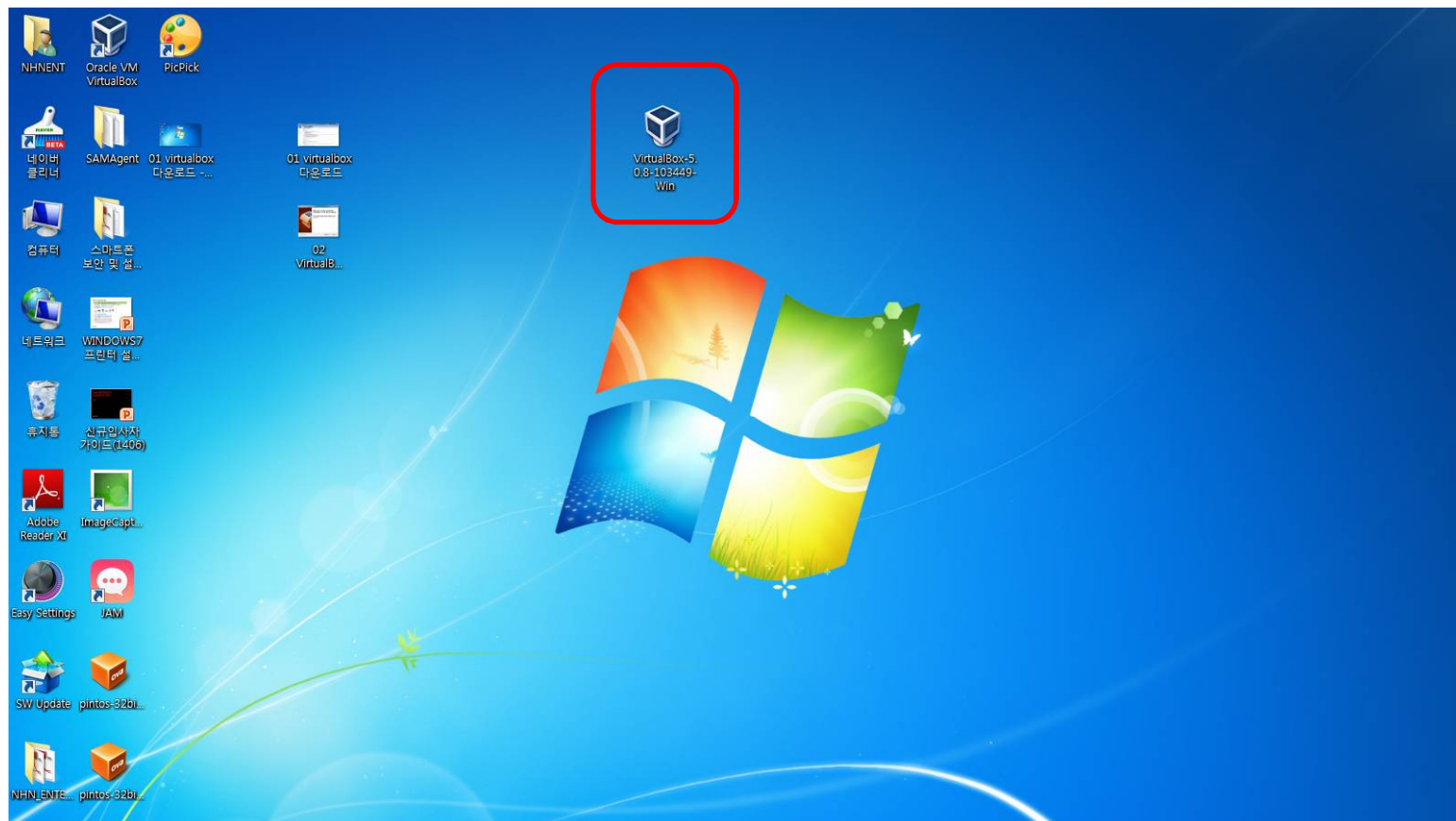


시스템SW 실습 환경 구축

1. VirtualBox 다운로드

- www.virtualbox.org 방문, virtualbox 5.0 다운로드 (VirtualBox 5.1.18 for Windows hosts용 선택)



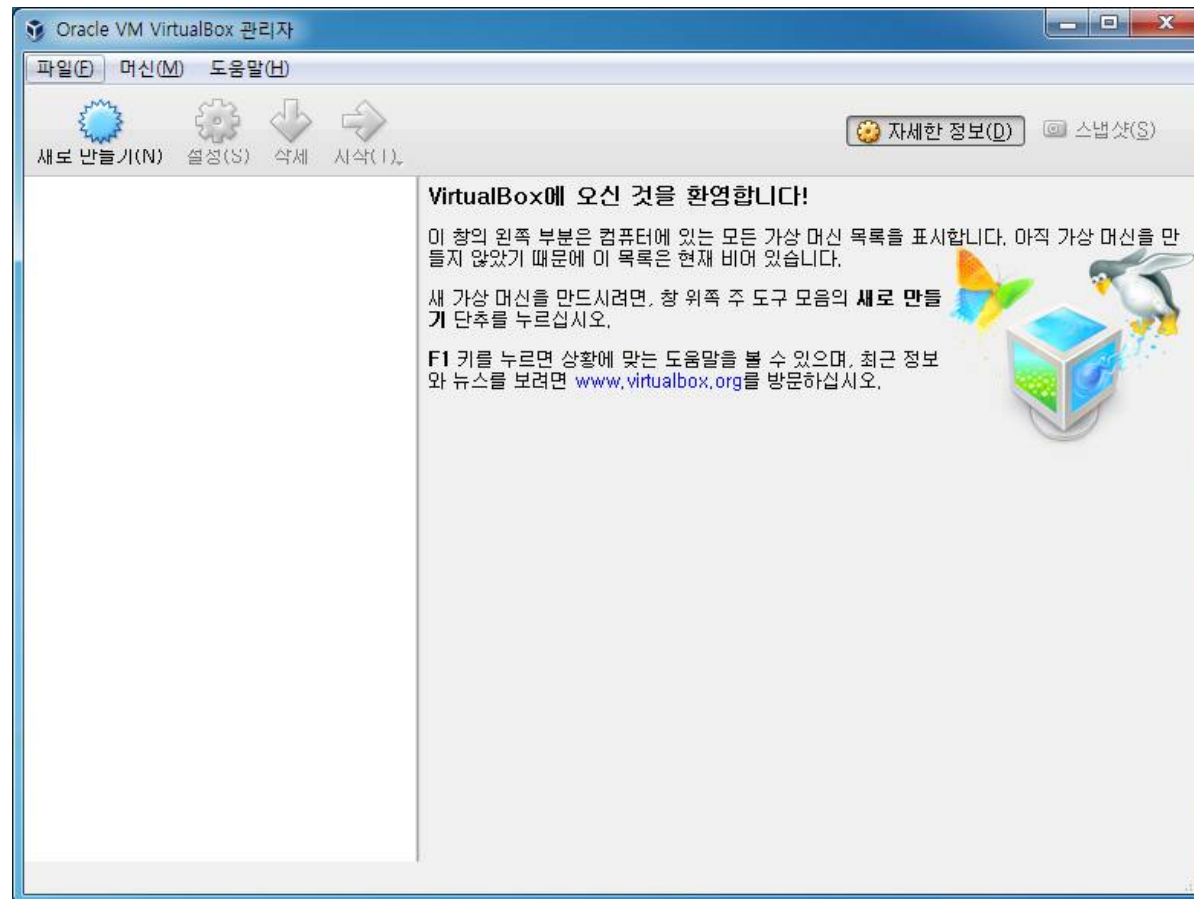


2. VirtualBox 설치 1/2



2. VirtualBox 설치 2/2

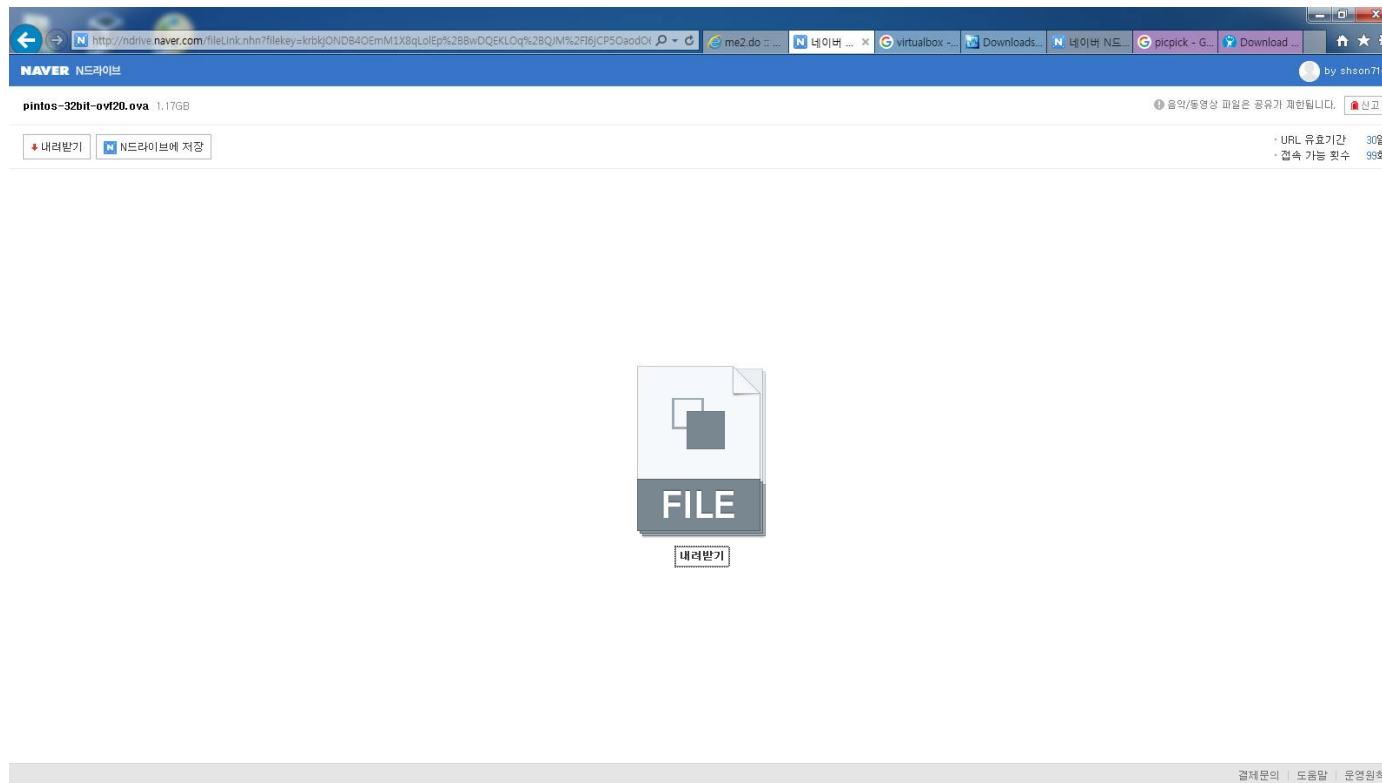
- 실행된 VirtualBox

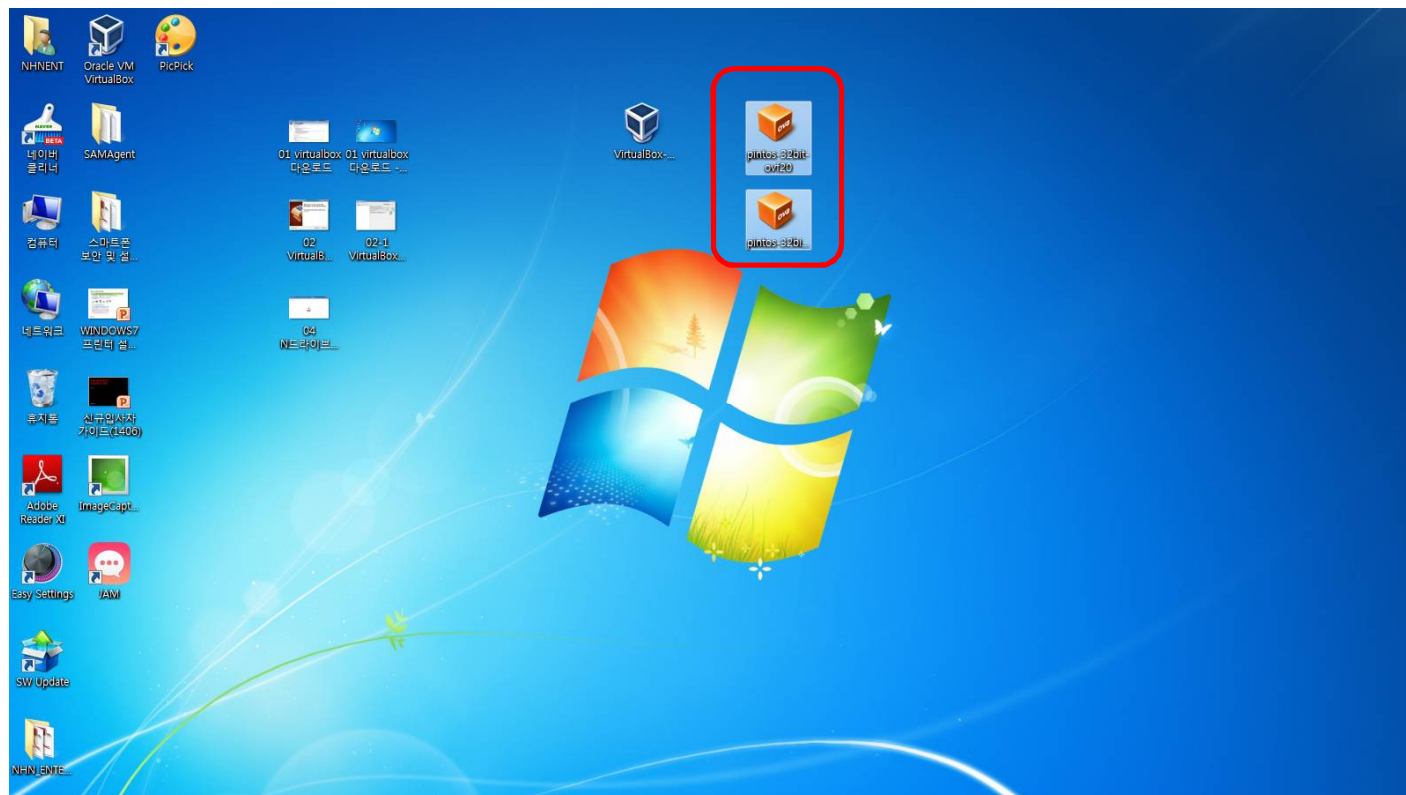


3. 네이버 계정 준비

- 가상머신 파일을 네이버 클라우드를 통해 배포
- 네이버에 계정을 새로 만들기 (이미 계정을 가지고 있으면 이 과정 생략 가능)
- 다운로드 URL
 - <http://naver.me/52r34vFX> 에 접속, virtual machine file 다운로드 받기 (파일명 sys-ubuntu-32bit-ovf-2.ova)
 - 다운로드 유효기간은 30일
 - 파일 크기는 각각 약 1.1GB

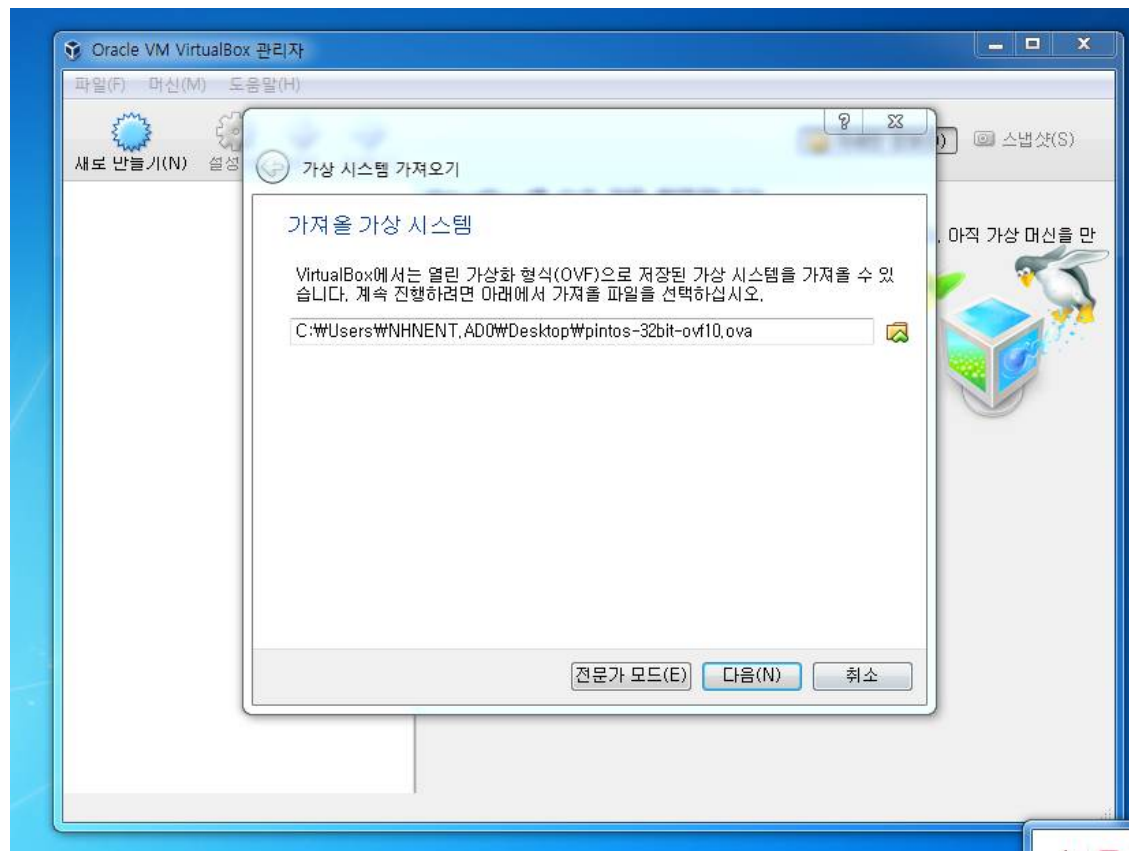
4. 가상 머신 파일 다운로드





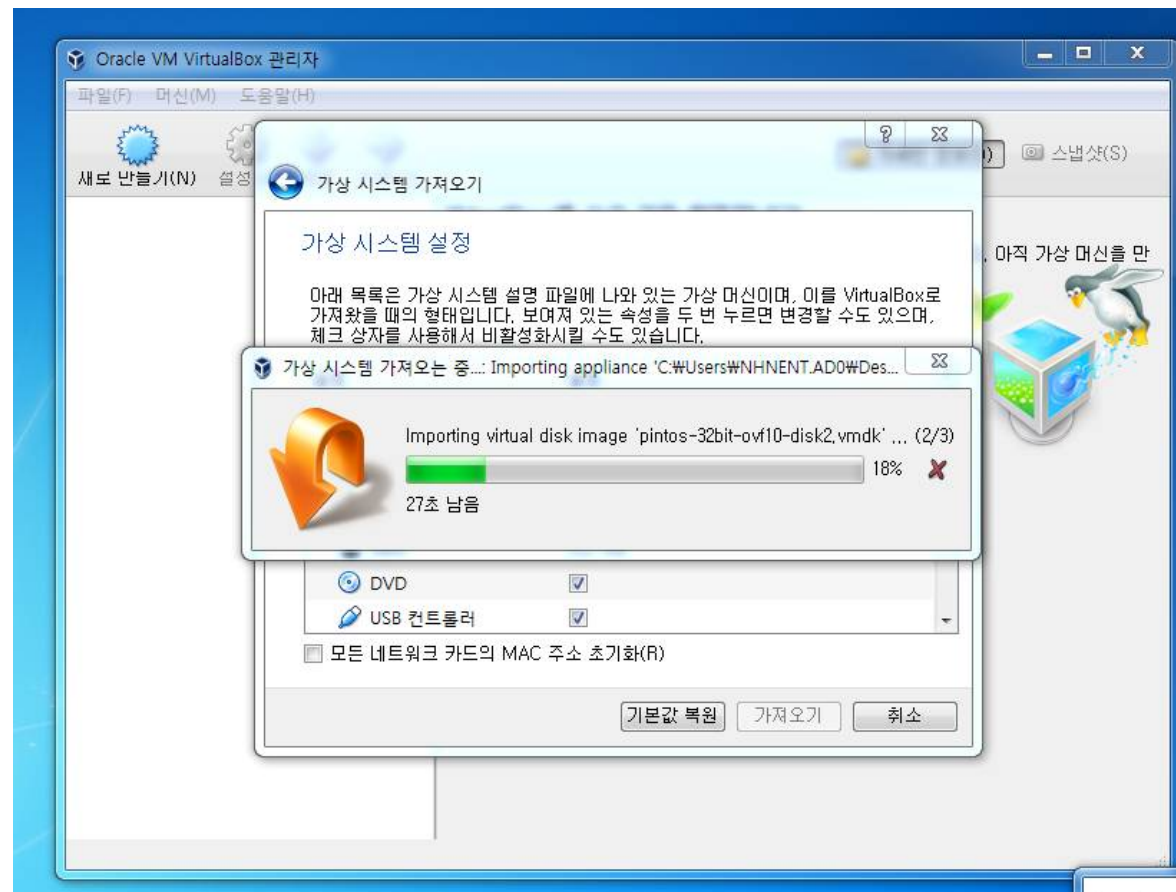
5. 가상 머신 가져오기 1/3

- VirtualBox 실행 후, [파일]-[가상 시스템 가져오기] 메뉴를 실행, 앞서 다운로드 받은 파일을 열기



가상 머신 가져오기 2/3

- 가상 머신 가져오는 과정



가상 머신 가져오기 3/3

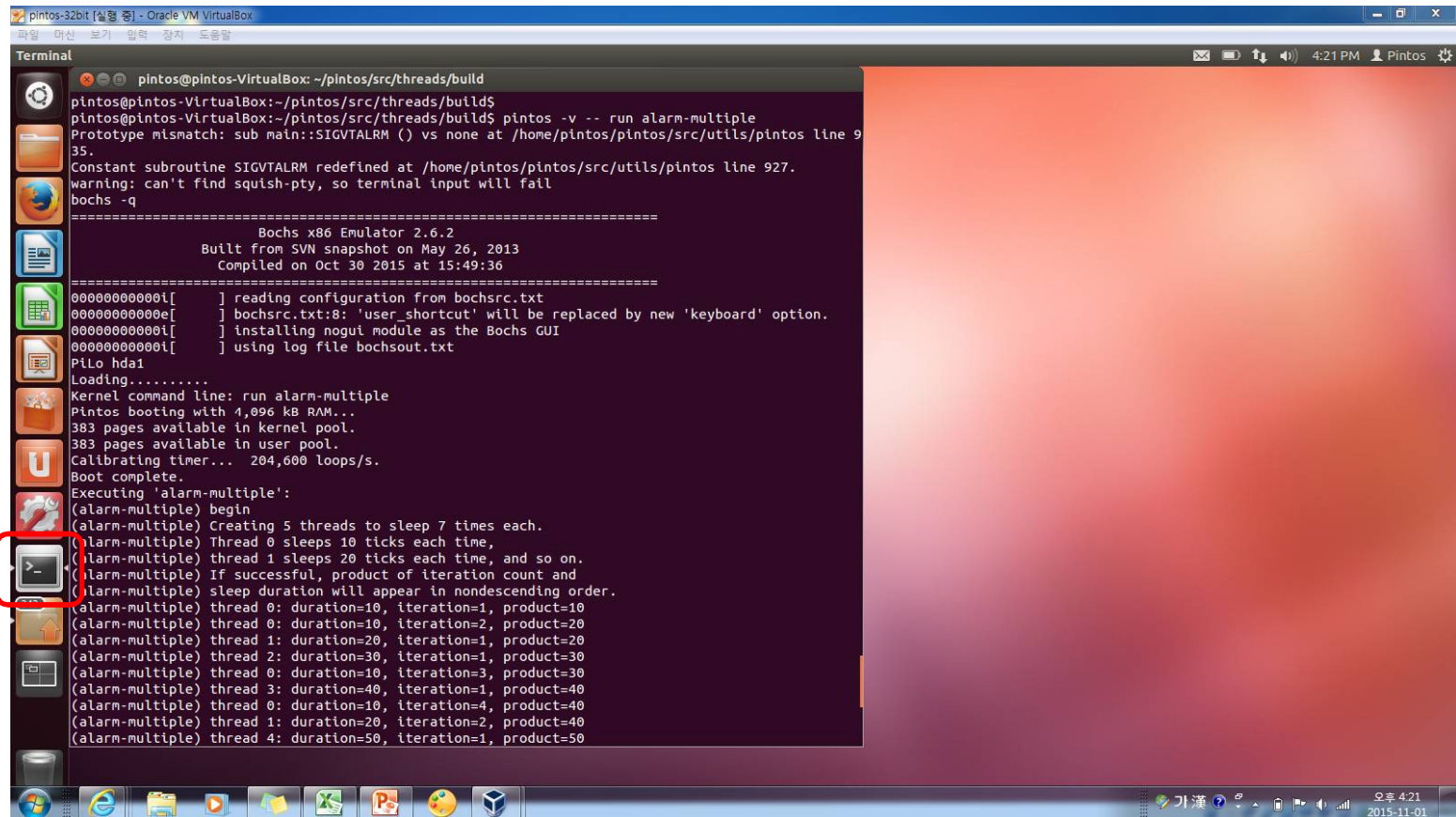
- 가져오기 완료된 가상 머신 실행



6. 가상 머신 실행

- 가상 머신의 실행 (가상머신 암호는 syssw)

terminal
프로그램
아이콘



```
pintos@pintos-VirtualBox: ~/pintos/src/threads/build
pintos@pintos-VirtualBox:~/pintos/src/threads/build$
pintos@pintos-VirtualBox:~/pintos/src/threads/build$ pintos -v -- run alarm-multiple
Prototype mismatch: sub main::SIGVTALRM () vs none at /home/pintos/pintos/src/utlis/pintos line 9
35.
Constant subroutine SIGVTALRM redefined at /home/pintos/pintos/src/utlis/pintos line 927.
warning: can't find squish-pty, so terminal input will fail
bochs -q

=====
Bochs x86 Emulator 2.6.2
Built from SVN snapshot on May 26, 2013
Compiled on Oct 30 2015 at 15:49:36
=====
00000000000i[ ] reading configuration from bochsrc.txt
00000000000e[ ] bochsrc.txt: 'user_shortcut' will be replaced by new 'keyboard' option.
00000000000i[ ] installing nogui module as the Bochs GUI
00000000000i[ ] using log file bochsout.txt
Pilo hda1
Loading.....
Kernel command line: run alarm-multiple
Pintos booting with 4,096 kB RAM...
383 pages available in kernel pool.
383 pages available in user pool.
Calibrating timer... 204,600 loops/s.
Boot complete.
Executing 'alarm-multiple':
(alarm-multiple) begin
(alarm-multiple) Creating 5 threads to sleep 7 times each.
(alarm-multiple) Thread 0 sleeps 10 ticks each time,
(alarm-multiple) thread 1 sleeps 20 ticks each time, and so on.
(alarm-multiple) If successful, product of iteration count and
(alarm-multiple) sleep duration will appear in nondescending order.
(alarm-multiple) thread 0: duration=10, iteration=1, product=10
(alarm-multiple) thread 0: duration=10, iteration=2, product=20
(alarm-multiple) thread 1: duration=20, iteration=1, product=20
(alarm-multiple) thread 2: duration=30, iteration=1, product=30
(alarm-multiple) thread 0: duration=10, iteration=3, product=30
(alarm-multiple) thread 3: duration=40, iteration=1, product=40
(alarm-multiple) thread 0: duration=10, iteration=4, product=40
(alarm-multiple) thread 1: duration=20, iteration=2, product=40
(alarm-multiple) thread 4: duration=50, iteration=1, product=50
```