



Lab Environments

SCE212: Computer Organization and Architecture
Spring 2023

Jeongseob Ahn (안정섭)
csl.ajou.ac.kr

Linux is our playground

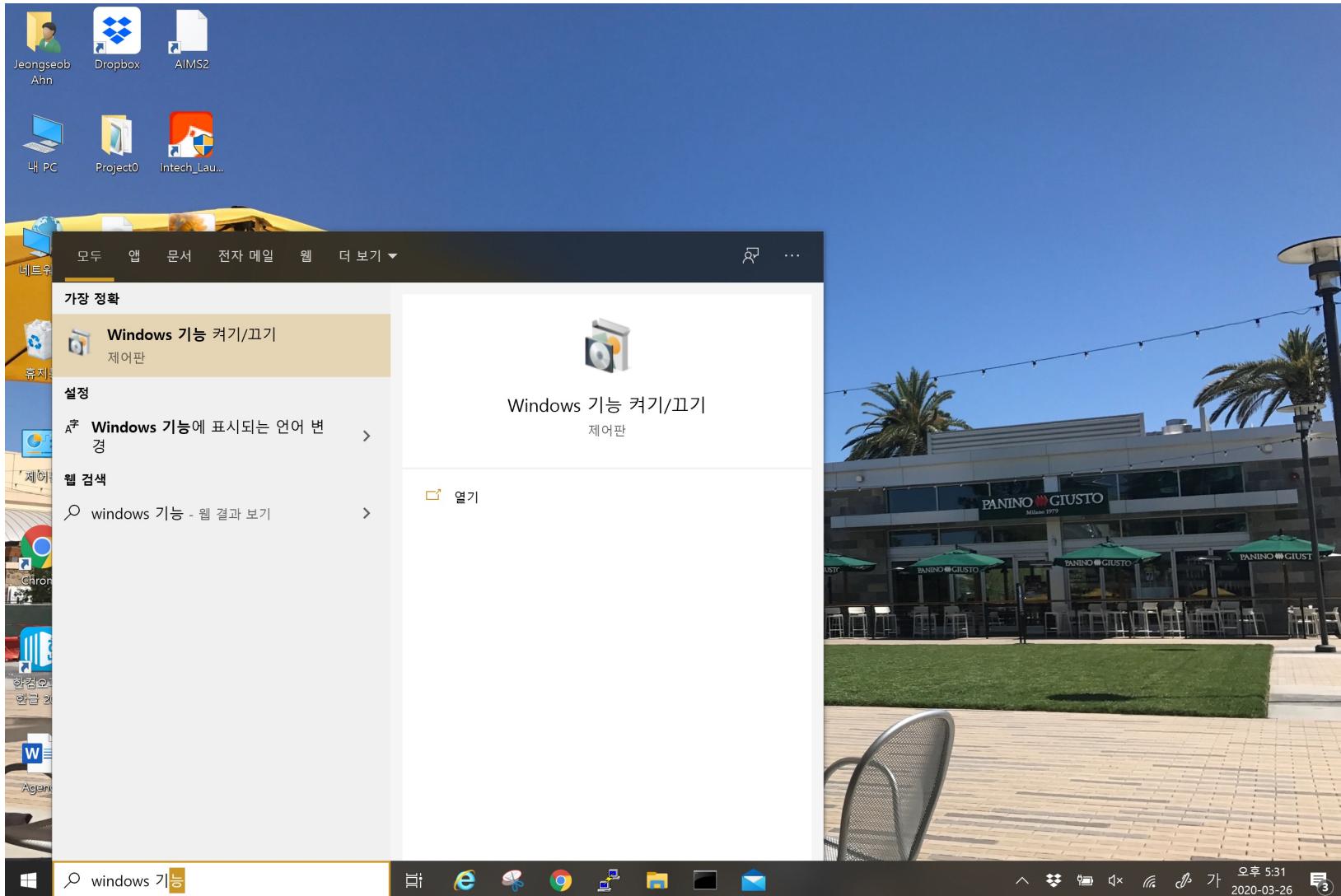
- Why are we supposed to use the Linux operating system?
 - Many real-world applications have been developed under the Linux environments
 - It is very popular from mobile/embedded to server
 - If you would like to be a system engineer, it is destiny
 - Please do not worry about using Linux



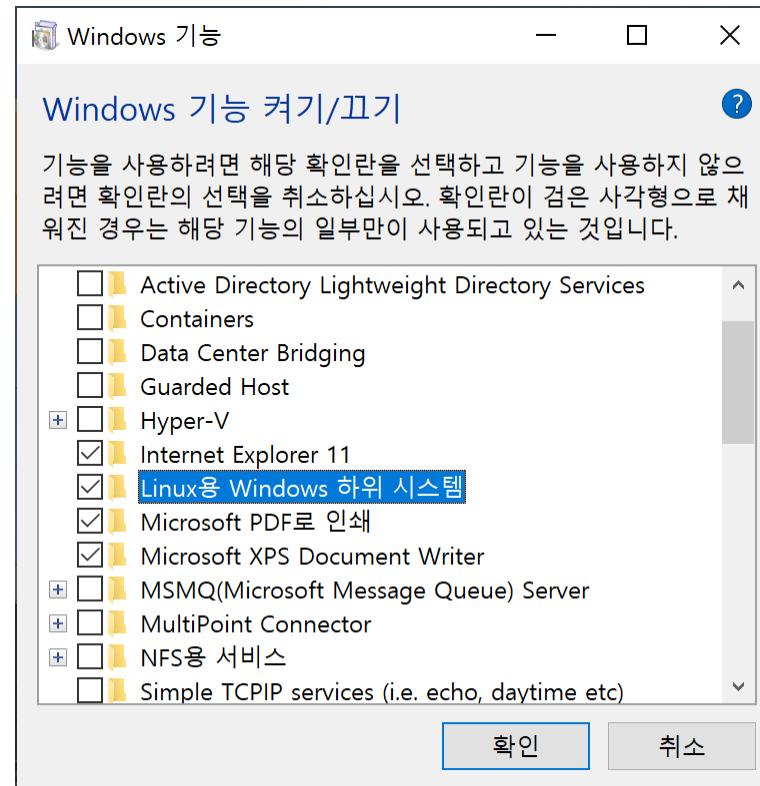
Prepare your environment

- Option 1. Use your own PC or laptop to run Linux
 - Turn on Windows Subsystem for Linux
- Option 2. Use the Linux server we provide
 - You don't need to have a virtual machine
 - Download Putty (Windows) or iTerm (Mac)
 - Skip the slides from 4 to 14

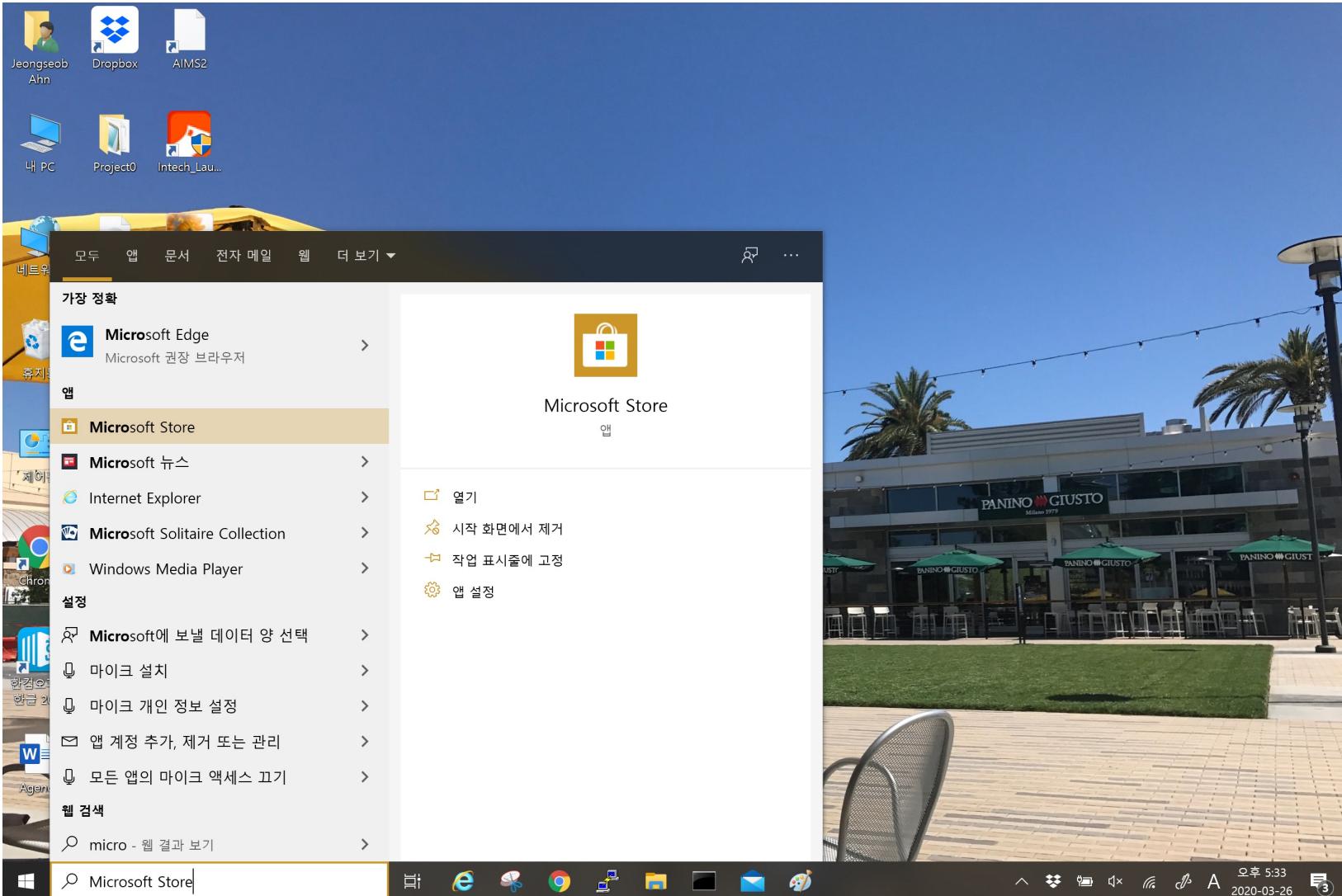
Windows Subsystem for Linux 활성화



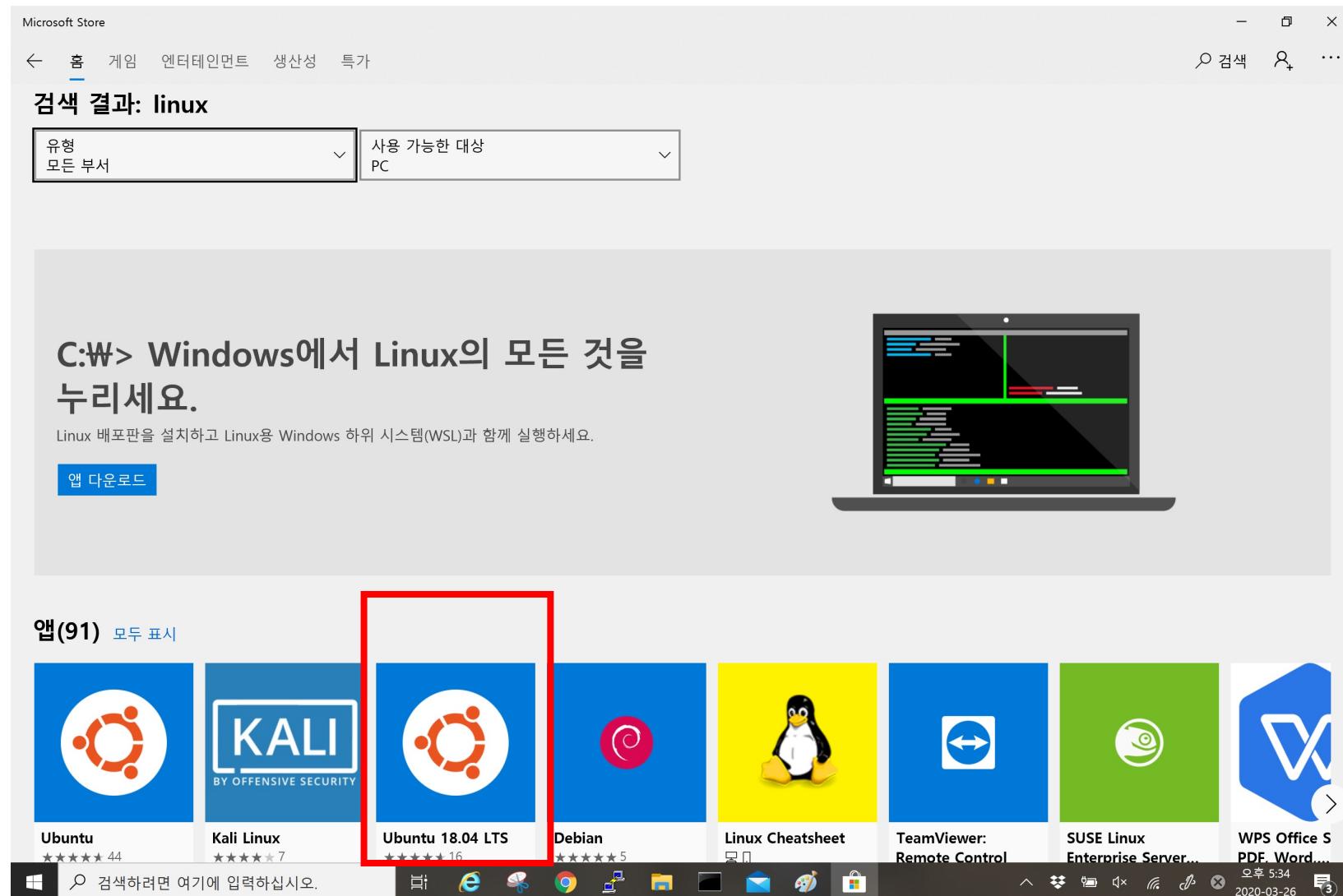
Windows Subsystem for Linux 활성화



Microsoft Store 이동



Linux를 검색하여 Ubuntu (LTS) 설치



설치 중

Microsoft Store

← 홈 게임 엔터테인먼트 생산성 특가 ⌂ 검색 ⌂ 7 ⌂ ...

Ubuntu 18.04 LTS 다운로드 중... 30.13MB/258.9MB
보다 빠른 다운로드에 대한 자세한 내용 알아보기 1.7 Mbps

Ubuntu 18.04 LTS

Canonical Group Limited · 개발자 도구 > 서버

★★★★★ 16 공유

Ubuntu 18.04 on Windows allows one to use Ubuntu Terminal and run Ubuntu command line utilities including bash, ssh, git, apt and many more.

자세히

IARC 3+

개요 시스템 요구 사항 리뷰 관련 계정

지원 플랫폼

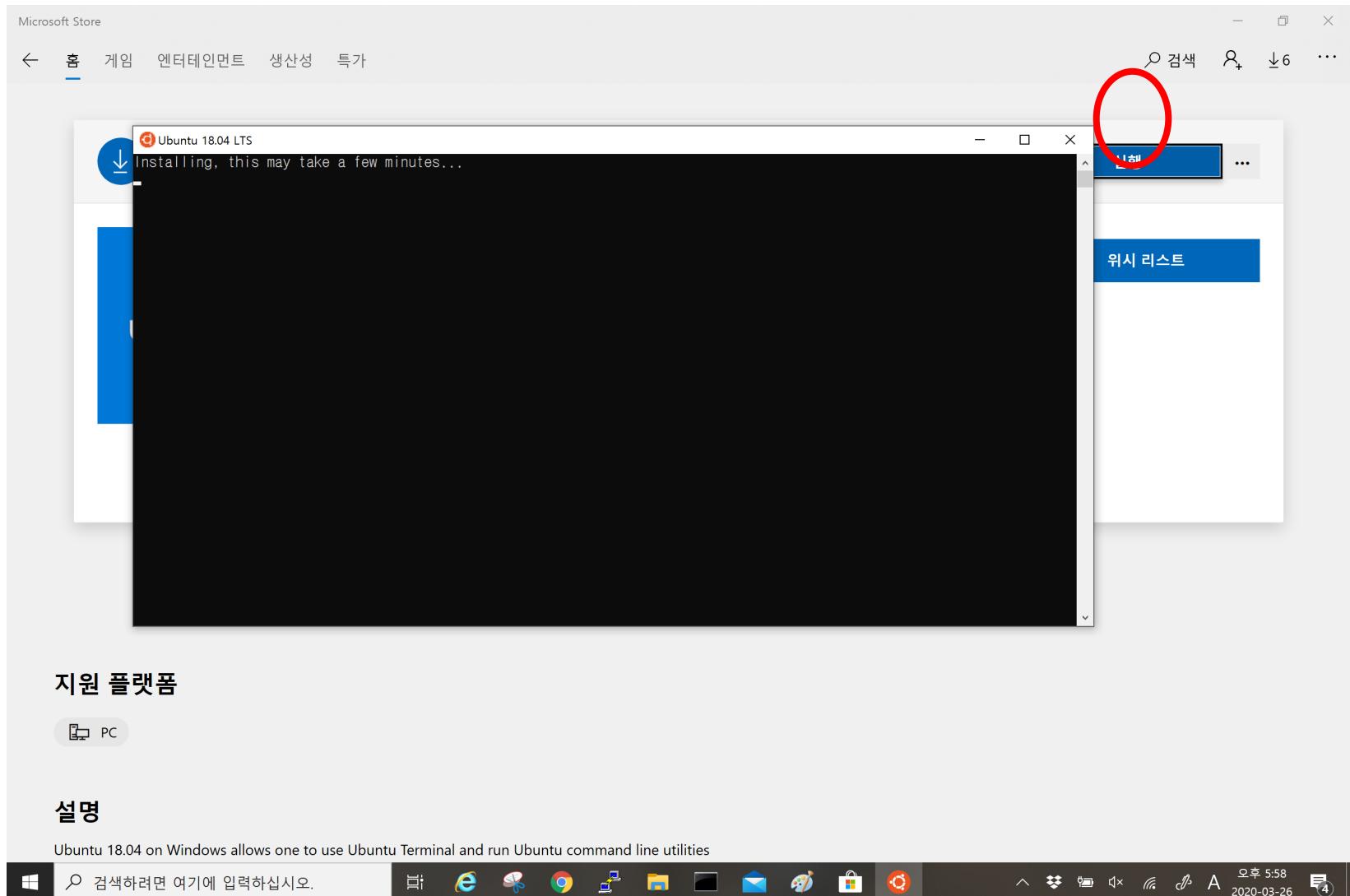
PC

설명

검색하려면 여기에 입력하십시오. ⌂

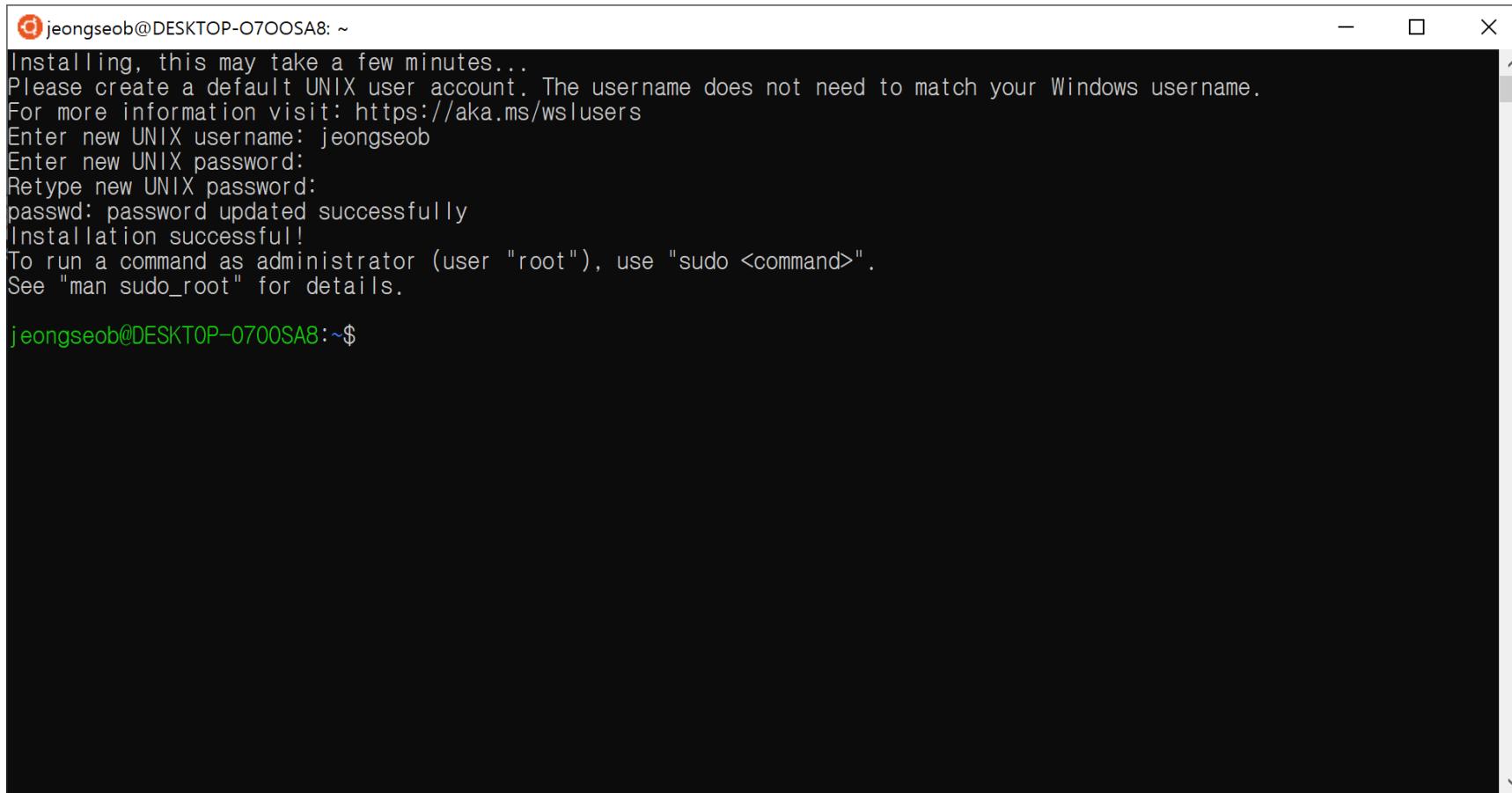
Windows Taskbar icons: File Explorer, Edge, File Manager, Mail, Paint 3D, Task View, Taskbar settings, Volume, Battery, Network, Notifications, Date/Time (2020-03-26), Taskbar count (4)

설치 완료 후 실행



Setting your account ID and PASSWD

사용할 ID와 Password 생성



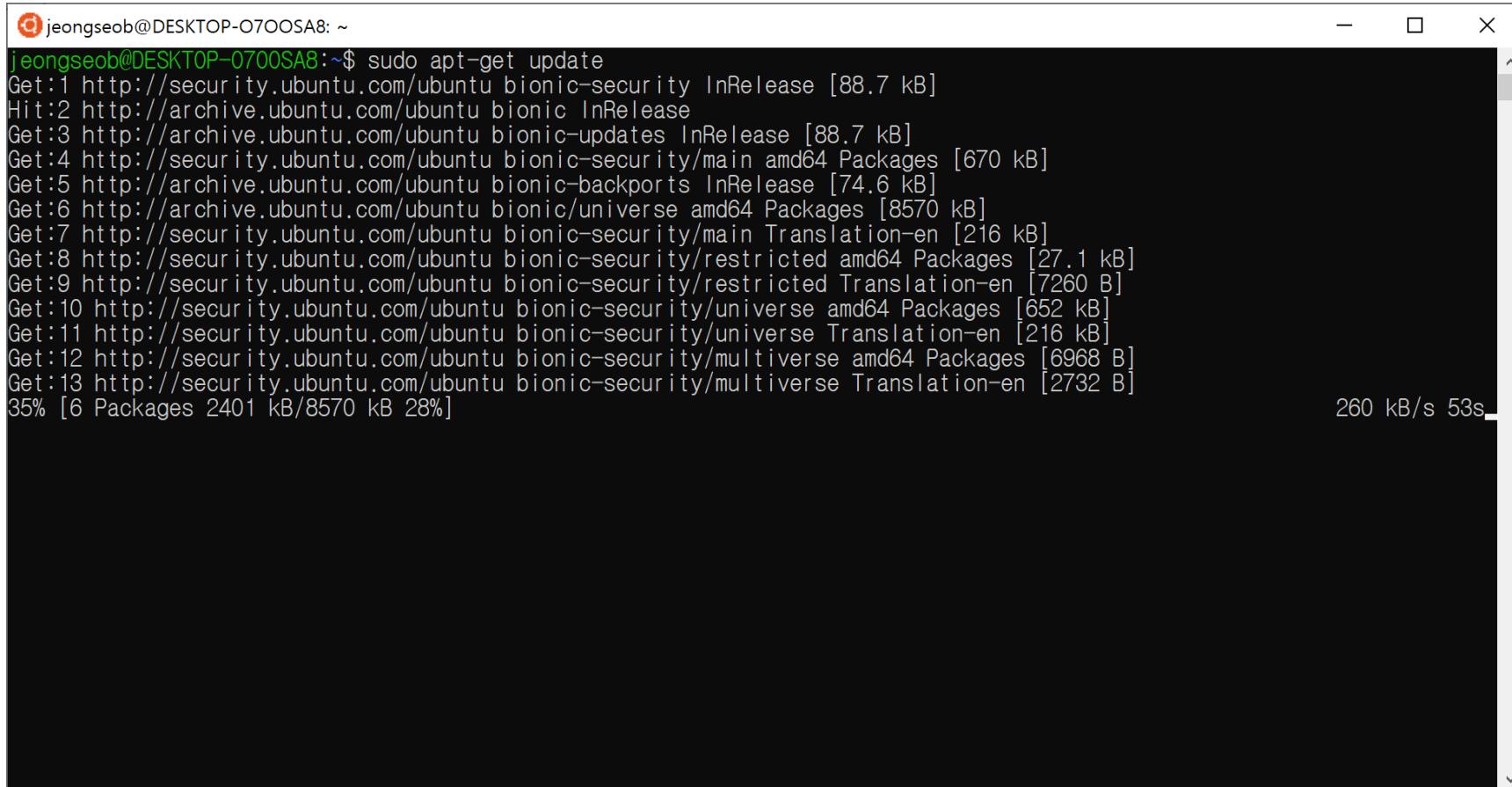
A screenshot of a Windows Terminal window titled "jeongseob@DESKTOP-0700SA8: ~". The window displays the process of creating a new UNIX user account. The text output is as follows:

```
Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: https://aka.ms/wslusers
Enter new UNIX username: jeongseob
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Installation successful!
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

jeongseob@DESKTOP-0700SA8:~$
```

Prepare the build environment (1/2)

```
$] sudo apt-get update
```



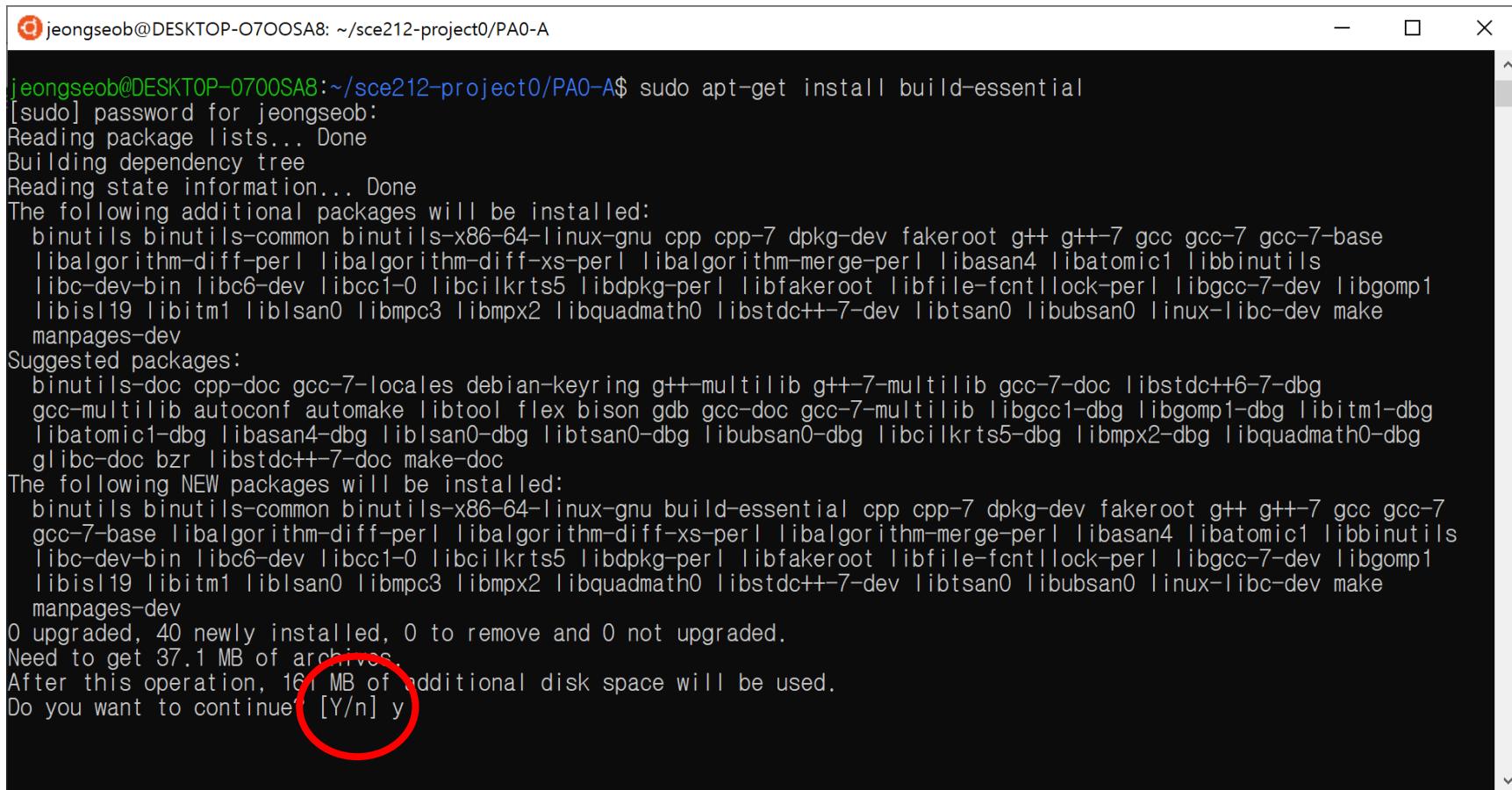
A screenshot of a terminal window titled 'jeongseob@DESKTOP-O700SA8: ~'. The window displays the output of the command 'sudo apt-get update'. The output shows the progress of fetching packages from various Ubuntu repositories. The progress bar at the bottom right indicates 35% completion at 2401 kB / 8570 kB, with a download speed of 260 kB/s and a total estimated time of 53s.

```
jeongseob@DESKTOP-O700SA8:~$ sudo apt-get update
Get:1 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Hit:2 http://archive.ubuntu.com/ubuntu bionic InRelease
Get:3 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:4 http://security.ubuntu.com/ubuntu bionic-security/main amd64 Packages [670 kB]
Get:5 http://archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:6 http://archive.ubuntu.com/ubuntu bionic/universe amd64 Packages [8570 kB]
Get:7 http://security.ubuntu.com/ubuntu bionic-security/main Translation-en [216 kB]
Get:8 http://security.ubuntu.com/ubuntu bionic-security/restricted amd64 Packages [27.1 kB]
Get:9 http://security.ubuntu.com/ubuntu bionic-security/restricted Translation-en [7260 B]
Get:10 http://security.ubuntu.com/ubuntu bionic-security/universe amd64 Packages [652 kB]
Get:11 http://security.ubuntu.com/ubuntu bionic-security/universe Translation-en [216 kB]
Get:12 http://security.ubuntu.com/ubuntu bionic-security/multiverse amd64 Packages [6968 B]
Get:13 http://security.ubuntu.com/ubuntu bionic-security/multiverse Translation-en [2732 B]
35% [6 Packages 2401 kB/8570 kB 28%]
```

컴파일러 설치를 위해 필요한 과정

Prepare the build environment (2/2)

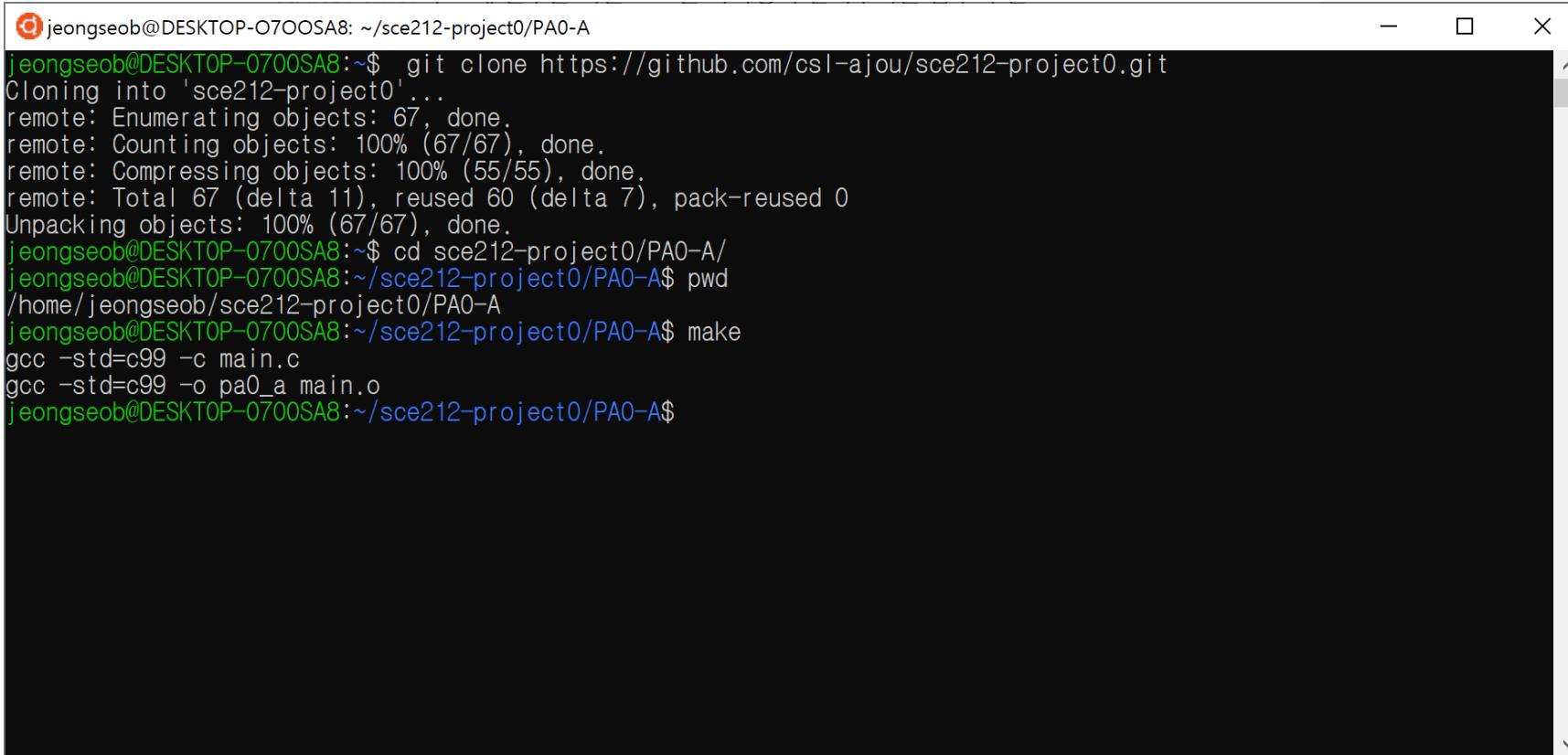
```
$] sudo apt-get build-essential
```



```
jeongseob@DESKTOP-O7OOSA8: ~/sce212-project0/PA0-A$ sudo apt-get install build-essential
[sudo] password for jeongseob:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  binutils binutils-common binutils-x86_64-linux-gnu cpp cpp-7 dpkg-dev fakeroot g++ g++-7 gcc gcc-7-base
  libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan4 libatomic1 libbinutils
  libc-dev-bin libc6-dev libcc1-0 libcilkrt5 libdpkg-perl libfakeroot libfile-fcntllock-perl libgcc-7-dev libgomp1
  libisl19 libitm1 liblsan0 libmpc3 libmpx2 libquadmath0 libstdc++-7-dev libtsan0 libubsan0 linux-libc-dev make
  manpages-dev
Suggested packages:
  binutils-doc cpp-doc gcc-7-locales debian-keyring g++-multilib g++-7-multilib gcc-7-doc libstdc++6-7-dbg
  gcc-multilib autoconf automake libtool flex bison gdb gcc-doc gcc-7-multilib libgcc1-dbg libgomp1-dbg libitm1-dbg
  libatomic1-dbg libasan4-dbg liblsan0-dbg libtsan0-dbg libubsan0-dbg libcilkrt5-dbg libmpx2-dbg libquadmath0-dbg
  glibc-doc bzr libstdc++-7-doc make-doc
The following NEW packages will be installed:
  binutils binutils-common binutils-x86_64-linux-gnu build-essential cpp cpp-7 dpkg-dev fakeroot g++ g++-7 gcc gcc-7
  gcc-7-base libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan4 libatomic1 libbinutils
  libc-dev-bin libc6-dev libcc1-0 libcilkrt5 libdpkg-perl libfakeroot libfile-fcntllock-perl libgcc-7-dev libgomp1
  libisl19 libitm1 liblsan0 libmpc3 libmpx2 libquadmath0 libstdc++-7-dev libtsan0 libubsan0 linux-libc-dev make
  manpages-dev
0 upgraded, 40 newly installed, 0 to remove and 0 not upgraded.
Need to get 37.1 MB of archives.
After this operation, 161 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Y 입력 후 Enter 눌러서 설치!

Start the programming assignment



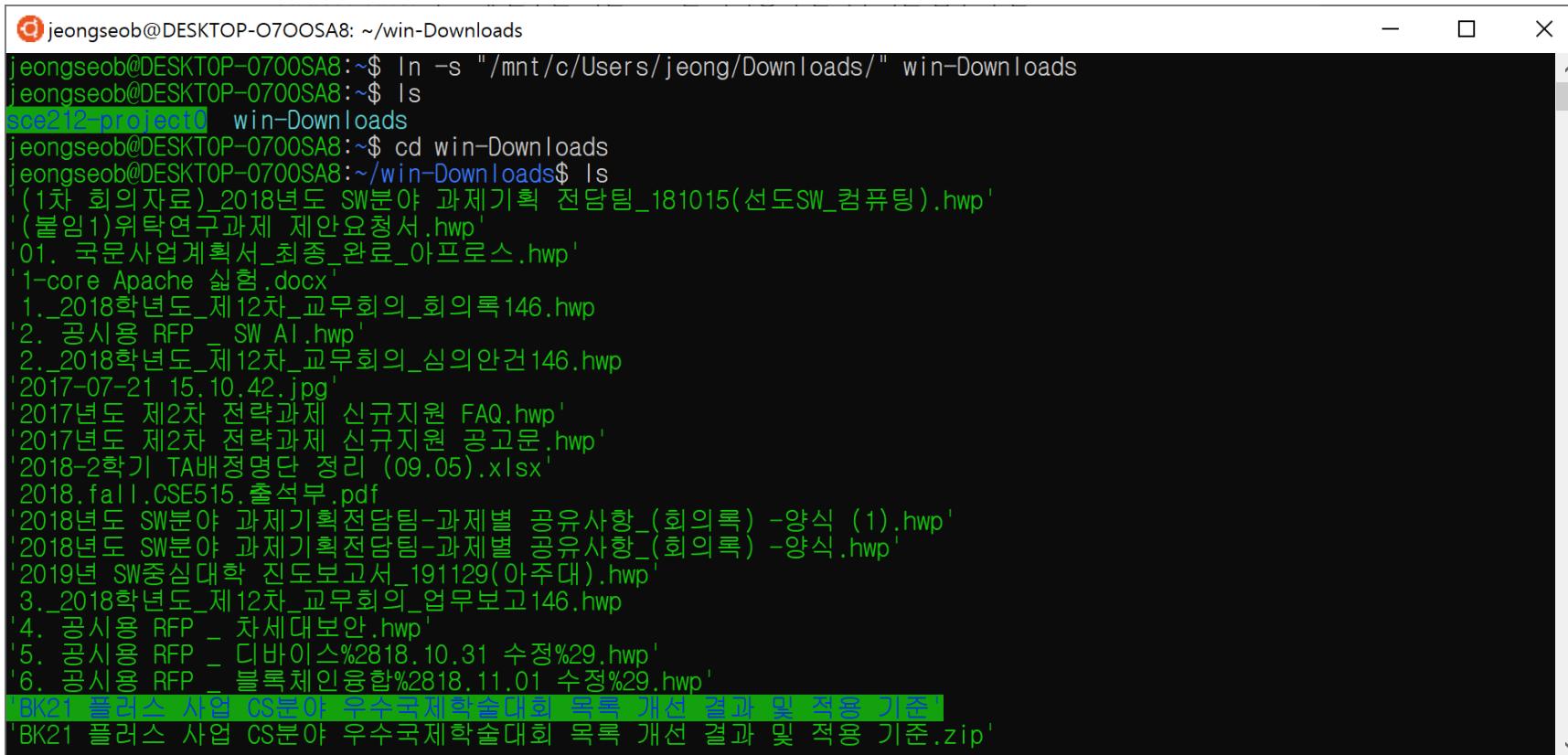
```
jeongseob@DESKTOP-0700SA8: ~/sce212-project0/PA0-A
jeongseob@DESKTOP-0700SA8:~$ git clone https://github.com/csl-ajou/sce212-project0.git
Cloning into 'sce212-project0'...
remote: Enumerating objects: 67, done.
remote: Counting objects: 100% (67/67), done.
remote: Compressing objects: 100% (55/55), done.
remote: Total 67 (delta 11), reused 60 (delta 7), pack-reused 0
Unpacking objects: 100% (67/67), done.
jeongseob@DESKTOP-0700SA8:~/sce212-project0/PA0-A/
jeongseob@DESKTOP-0700SA8:~/sce212-project0/PA0-A$ pwd
/home/jeongseob/sce212-project0/PA0-A
jeongseob@DESKTOP-0700SA8:~/sce212-project0/PA0-A$ make
gcc -std=c99 -c main.c
gcc -std=c99 -o pa0_a main.o
jeongseob@DESKTOP-0700SA8:~/sce212-project0/PA0-A$
```

Handout에 나와있는 내용과 동일

File Sharing with your Windows

```
$] ln -s "/mnt/c/Users/[계정아이디]/Downloads/" win-Downloads  
$] cd win-Downloads  
$] ls
```

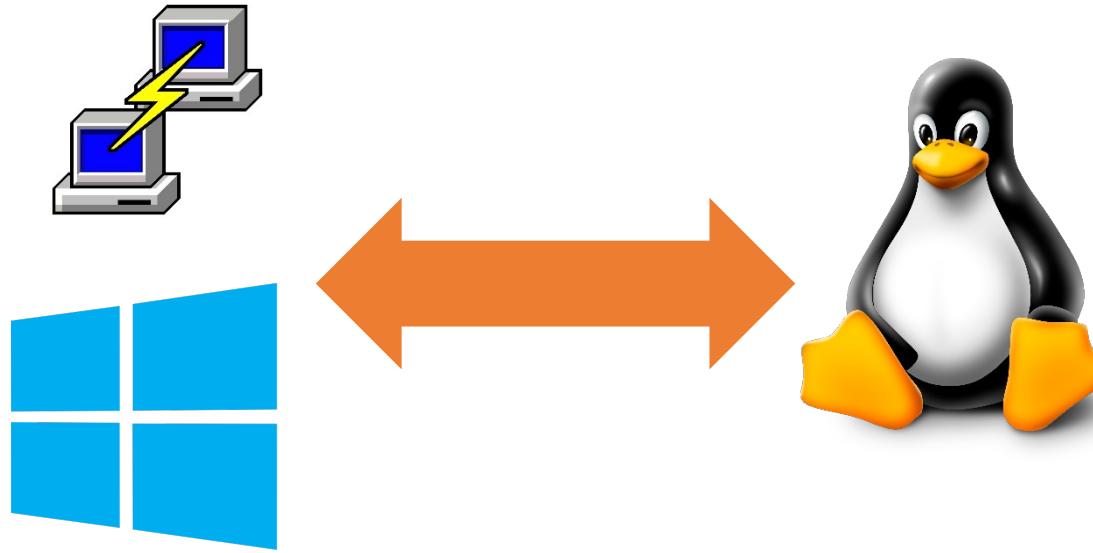
각 학생이 사용하는 Windows의 계정 이름



A screenshot of a terminal window titled 'jeongseob@DESKTOP-0700SA8: ~/win-Downloads'. The window shows a list of files and directories. The first few lines of the terminal output are:

```
jeongseob@DESKTOP-0700SA8:~$ ln -s "/mnt/c/Users/jeong/Downloads/" win-Downloads  
jeongseob@DESKTOP-0700SA8:~$ ls  
sce212-project0 win-Downloads  
jeongseob@DESKTOP-0700SA8:~$ cd win-Downloads  
jeongseob@DESKTOP-0700SA8:~/win-Downloads$ ls
```

The list of files includes various documents and images, such as HWP files for class projects and RFPs, and a PDF for a TA assignment. A red oval highlights the path '/mnt/c/Users/[계정아이디]/Downloads/' in the terminal command, which corresponds to the circled '계정아이디' in the text above. The bottom of the terminal window shows two green status bars: 'BK21 플러스 사업 CS분야 우수국제학술대회 목록 개선 결과 및 적용 기준' and 'BK21 플러스 사업 CS분야 우수국제학술대회 목록 개선 결과 및 적용 기준.zip'.



Connecting the server through a
terminal client (**Putty**) on Windows

Download the Putty binary

Download PutTY: latest release [chiark.greenend.org.uk/~sgtatham/putty/latest.html](https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html)

You probably want one of these. They include versions of all the PuTTY utilities.

(Not sure whether you want the 32-bit or the 64-bit version? Read the [FAQ entry](#).)

MSI ('Windows Installer')

32-bit:	putty-0.73-installer.msi	(or by FTP)	(signature)
64-bit:	putty-64bit-0.73-installer.msi	(or by FTP)	(signature)

Unix source archive

.tar.gz:	putty-0.73.tar.gz	(or by FTP)	(signature)
----------	-----------------------------------	------------------------------	-------------------------------

Alternative binary files

The installer packages above will provide versions of all of these (except PuTTYtel), but y

(Not sure whether you want the 32-bit or the 64-bit version? Read the [FAQ entry](#).)

putty.exe (the SSH and Telnet client itself)

32-bit:	putty.exe	(or by FTP)	(signature)
64-bit:	putty.exe	(or by FTP)	(signature)

pscp.exe (an SCP client, i.e. command-line secure file copy)

32-bit:	pscp.exe	(or by FTP)	(signature)
64-bit:	pscp.exe	(or by FTP)	(signature)

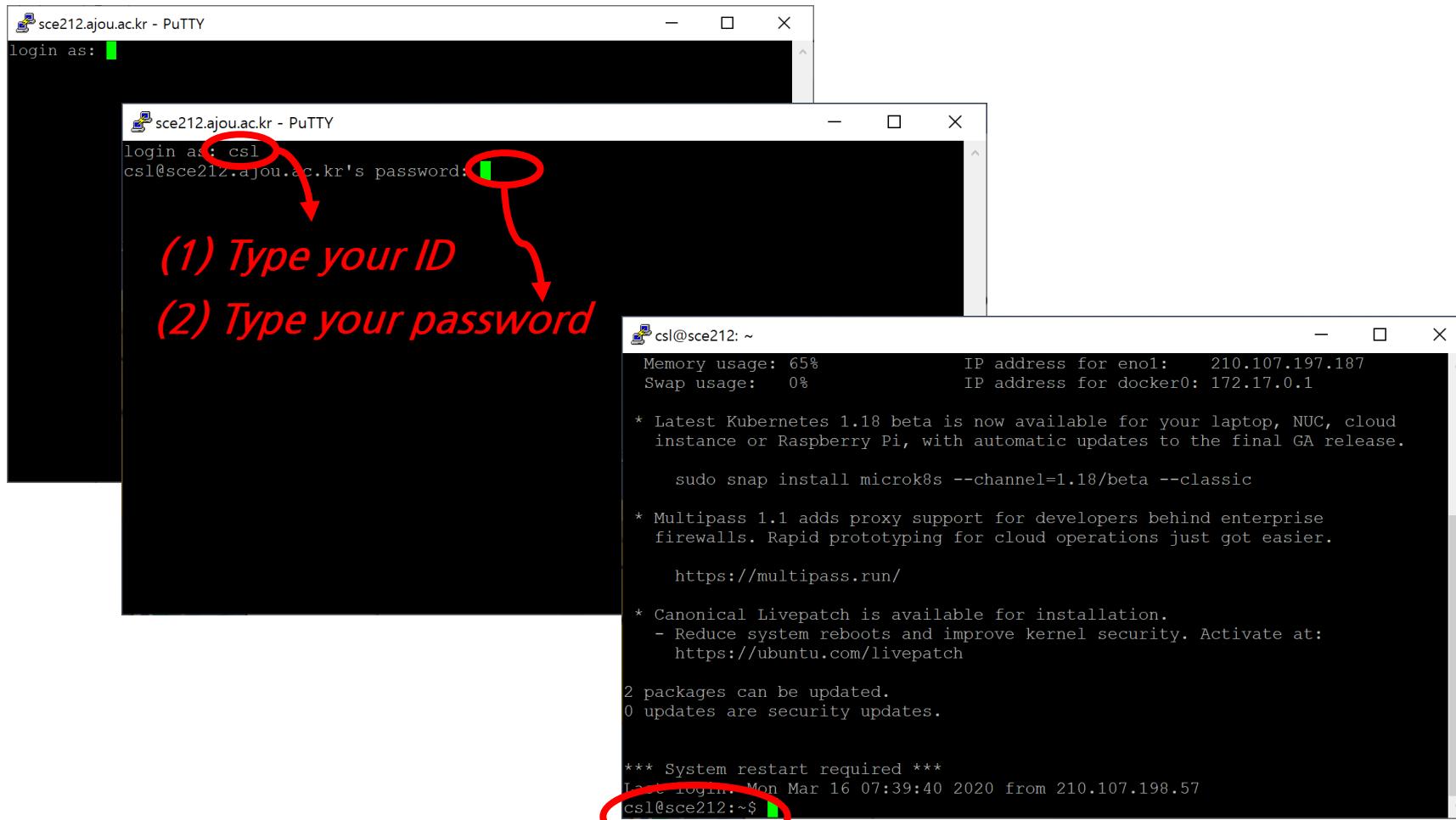
psftp.exe (an SFTP client, i.e. general file transfer sessions much like FTP)

32-bit:	psftp.exe	(or by FTP)	(signature)
64-bit:	psftp.exe	(or by FTP)	(signature)

PutTY Configuration window showing session setup. The host name is set to "sce212.ajou.ac.kr" and the port is 22. The connection type is set to SSH. The "Open" button is highlighted with a red circle. The "About" button at the bottom left is also circled in red.

본인 이메일로 전달받은 서버 주소 이용

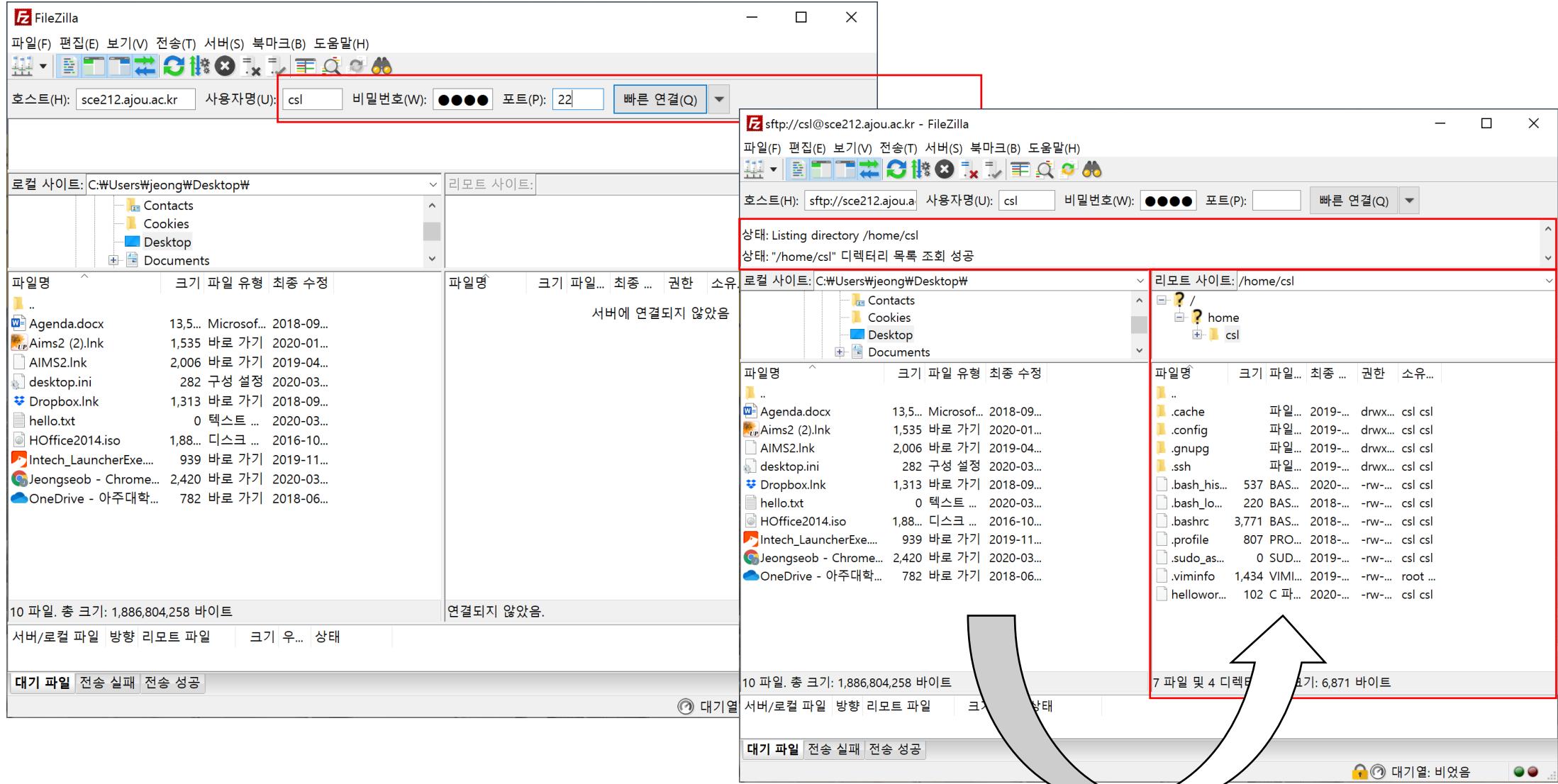
Connecting the server through Putty

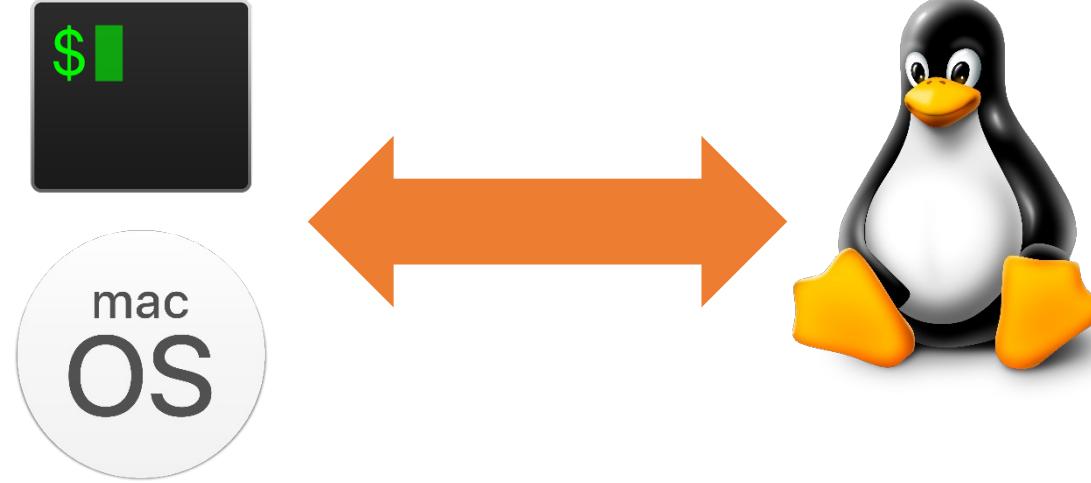


(제공해주는 서버를 이용하는 경우) 첫번째 login 이후 원하는 비밀번호로 변경을 하시기 바랍니다

Copy a file to the server using FileZilla

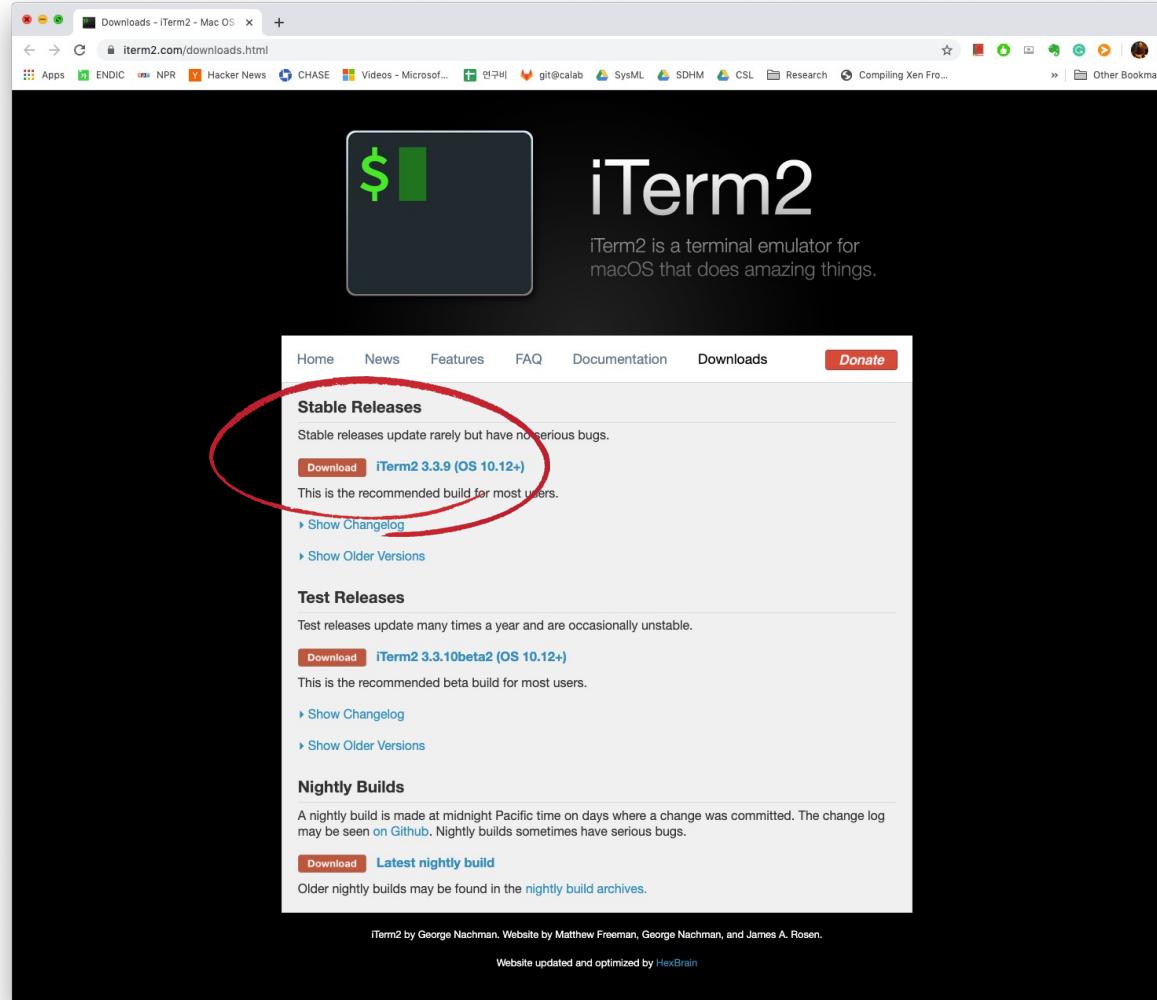
The screenshot shows a web browser displaying the official FileZilla download page at filezilla-project.org/download.php?type=client. The page features the FileZilla logo and the tagline "The free FTP solution". A promotion for "FileZilla® Pro" is visible, showing icons for various platforms and a "GET IN NOW >" button. The main section is titled "Download FileZilla Client for Windows (64bit)". It states that the latest stable version is 3.47.2.1 and asks the user to select their platform. A large green button with the text "Download FileZilla Client" and a red arrow points to it. Below the button, a note says "This installer may include bundled offers. Check below for more options." and mentions support for Windows 8.1 and 10. Other download options for different platforms are listed, along with links for "More download options" and "Show additional download options". On the left, a sidebar contains links for Home, FileZilla (Features, Screenshots, Download, Documentation, FileZilla Pro), FileZilla Server (Download), Community (Forum, Project page, Wiki), General (FAQ, Support, Contact, License, Privacy Policy, Trademark Policy), Development (Source code, Nightly builds, Translations, Version history, Changelog, Issue tracker), and Other projects (libfilezilla, Octochess). Sponsors are also listed at the bottom.

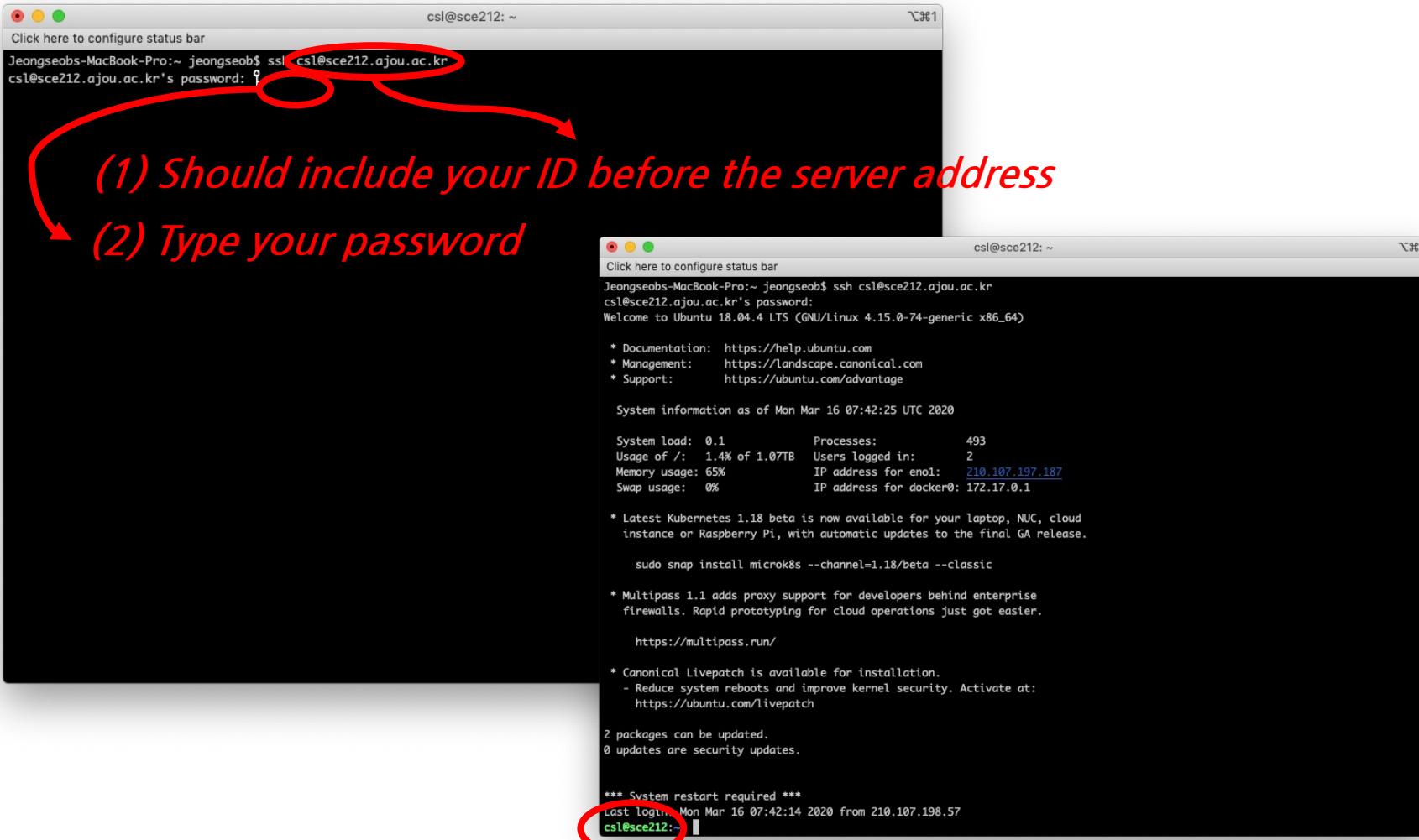




Connecting the server through a
terminal client (`iTerm`) on MacOS

Download the iTerm binary





The image shows two screenshots of a macOS terminal window. The top screenshot shows a command being entered: `Jeongseob-MacBook-Pro:~ jeongseob$ ssh csl@sce212.ajou.ac.kr`. A red circle highlights the password field where the user has typed 'csl'. A red arrow points from the text '(1) Should include your ID before the server address' to the 'jeongseob\$' part of the command. Another red arrow points from the text '(2) Type your password' to the password field. The bottom screenshot shows the terminal after logging in, displaying a welcome message for Ubuntu 18.04.4 LTS, system information, and various system status messages. A red circle highlights the prompt 'csl@sce212:~' at the bottom of the screen.

```
Jeongseob-MacBook-Pro:~ jeongseob$ ssh csl@sce212.ajou.ac.kr
csl@sce212.ajou.ac.kr's password: csl

(1) Should include your ID before the server address
(2) Type your password

csl@sce212:~
```

```
Jeongseob-MacBook-Pro:~ jeongseob$ ssh csl@sce212.ajou.ac.kr
csl@sce212.ajou.ac.kr's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-74-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

System information as of Mon Mar 16 07:42:25 UTC 2020

System load: 0.1      Processes:          493
Usage of /: 1.4% of 1.07TB  Users logged in:  2
Memory usage: 65%      IP address for eno1:  210.107.197.187
Swap usage: 0%          IP address for docker0: 172.17.0.1

* Latest Kubernetes 1.18 beta is now available for your laptop, NUC, cloud
instance or Raspberry Pi, with automatic updates to the final GA release.

sudo snap install microk8s --channel=1.18/beta --classic

* Multipass 1.1 adds proxy support for developers behind enterprise
firewalls. Rapid prototyping for cloud operations just got easier.

https://multipass.run/

* Canonical Livepatch is available for installation.
- Reduce system reboots and improve kernel security. Activate at:
https://ubuntu.com/livepatch

2 packages can be updated.
0 updates are security updates.

*** System restart required ***
Last login: Mon Mar 16 07:42:14 2020 from 210.107.198.57
csl@sce212:~
```

(3) You can see the prompt

Copy a file to the server using SCP

The image shows two terminal windows side-by-side. The left terminal window is on a MacBook Pro and displays the following command and its output:

```
Jeongseobs-MacBook-Pro:~ jeongseob$ cat helloworld.c
#include <stdio.h>

int main (int argc, char* argv[])
{
    printf("Hello Linux!\n");
    return 0;
}
Jeongseobs-MacBook-Pro:~ jeongseob$ Jeongseobs-MacBook-Pro:~ jeongseob$ scp helloworld.c csl@sce212.ajou.ac.kr's password: helloworld.c
Jeongseobs-MacBook-Pro:~ jeongseob$
```

The right terminal window is on a server (Ubuntu 18.04.4 LTS) and shows the following session:

```
Click here to configure status bar
Jeongseobs-MacBook-Pro:~ jeongseob$ ssh csl@sce212.ajou.ac.kr
csl@sce212.ajou.ac.kr's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-74-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

System information as of Mon Mar 16 08:03:46 UTC 2020

System load: 0.02      Processes:      502
Usage of /: 1.4% of 1.07TB Users logged in: 2
Memory usage: 66%      IP address for eno1: 210.107.197.187
Swap usage: 0%          IP address for docker0: 172.17.0.1

* Latest Kubernetes 1.18 beta is now available for your laptop, NUC, cloud
instance or Raspberry Pi, with automatic updates to the final GA release.

sudo snap install microk8s --channel=1.18/beta --classic

* Multipass 1.1 adds proxy support for developers behind enterprise
firewalls. Rapid prototyping for cloud operations just got easier.

https://multipass.run/

* Canonical Livepatch is available for installation.
- Reduce system reboots and improve kernel security. Activate at:
https://ubuntu.com/livepatch

2 packages can be updated.
0 updates are security updates.

*** System restart required ***
Last login: Mon Mar 16 07:42:26 2020 from 210.107.198.57
csl@sce212:~$ ls
helloworld.c
csl@sce212:~$
```

Annotations in red text are overlaid on the right terminal window:

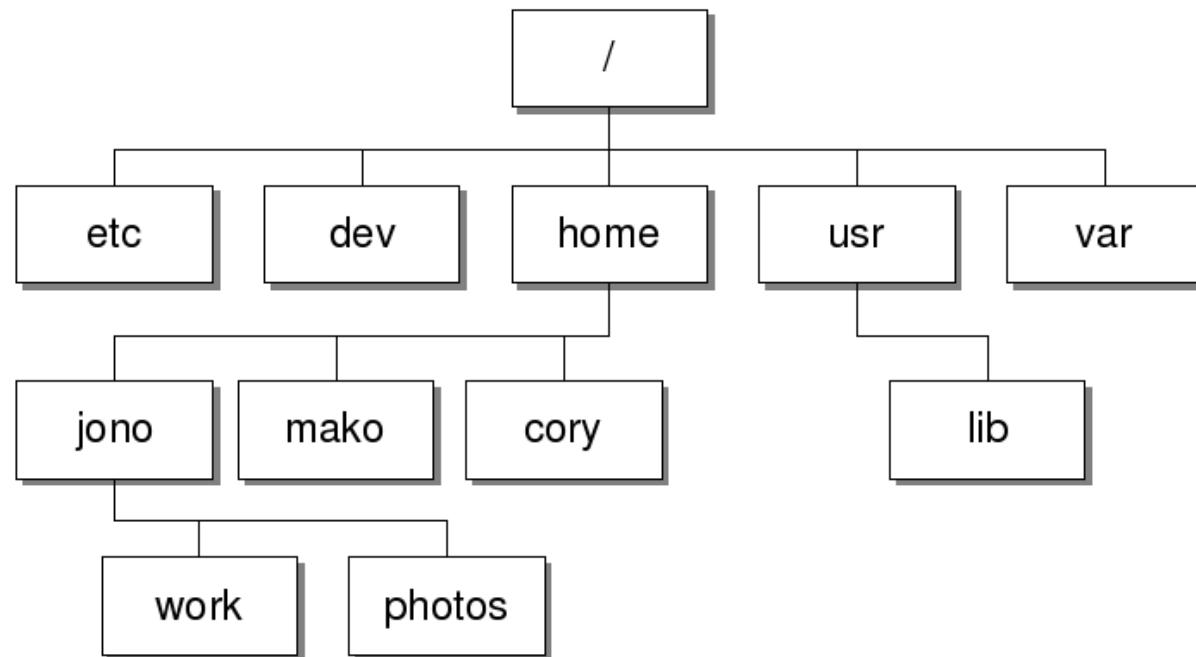
- "Login the server" is circled around the password prompt.
- "You can see the copy file" is circled around the "helloworld.c" file listed in the directory.

Linux Environment



File Structure of Linux

- There are some truncated directories, but they usually have the same structure as the picture.



Path

- Absolute path: Full path of file or directory
 - E.g.,) /home/user/application, /usr/lib, ...
- Relative path: At time that current path view other files and directories,
 - . : current working directory
 - .. : current working directory's upper directory
 - ~ : user's directory path.
- Example) If current directory is /home/cs1/app/hello,
 - . : /home/cs1/app/hello
 - .. : /home/cs1/app
 - ~ : /home/cs1 (if current username is 'cs1')

Shell

- A command interpreter used in GNU/Linux



A screenshot of a terminal window on a Linux system. The window has a dark background and light-colored text. At the top, it shows the command `ls` and its output, which includes the file `helloworld.c`. At the bottom, there is a red circle highlighting the text `cs1@sce212:~$`, which is the standard Linux shell prompt. Below the terminal window, the text *Shell prompt* is written in red.

```
cs1@sce212:~$ ls
helloworld.c
cs1@sce212:~$
```

Shell prompt

- Linux also supports the GUI interface like Windows or MacOS

Basic Commands in Linux: pwd and ls

- pwd: print working directory
 - Print name of current/working directory
- ls: list directory
 - Print current directory contents where you are
 - ls <directory> : Print the directory's contents what you wants

```
vagrant@dev [10:44:00] ~
$ pwd
/home/vagrant

vagrant@dev [10:44:02] ~
$ ls
project1

vagrant@dev [10:44:03] ~
$ ls project1/
hello.txt

vagrant@dev [10:44:10] ~
$ ls /home
ubuntu vagrant
```

Basic Commands in Linux: cd

- cd: change directory
 - Change your directory where the user wants
- How to use
 - \$ cd .. : change path to upper directory
 - \$ cd or cd ~ : change path to user's home directory
 - \$ cd <directory name>

```
vagrant@dev [11:02:20] ~
$ pwd
/home/vagrant

vagrant@dev [11:02:25] ~
$ cd project1/

vagrant@dev [11:02:30] ~/project1
$ cd .

vagrant@dev [11:02:38] ~/project1
$ cd ..

vagrant@dev [11:02:39] ~
$ cd ..

vagrant@dev [11:02:43] /home
$ cd ~

vagrant@dev [11:02:48] ~
$ █
```

Basic Commands in Linux: mkdir and rm

- `mkdir`: make directory
 - Make a new directories
 - `$ mkdir <directory name>`
- `rm`: remove
 - Remove files or directories
 - `-r`: remove directories and their contents recursively
 - `-f`: force. ignore nonexistent files and arguments, never prompt
 - `rm [-rf] <file or directory name>`

```
vagrant@dev [11:17:02] ~
$ mkdir foo

vagrant@dev [11:17:08] ~
$ ls
foo  project1

vagrant@dev [11:17:10] ~
$ rm foo
rm: cannot remove 'foo': Is a directory

vagrant@dev [11:17:14] ~
$ rm -r foo

vagrant@dev [11:17:18] ~
$ ls
project1
```

Basic Commands in Linux: cp and mv

- cp: copy
 - Copy files and directories
 - -r : Copy directories recursively
 - \$ cp [-r] <source> <destination>
- mv: move
 - Move (rename) files
 - \$ mv <source> <destination>
 - Actually, mv equals cp & rm
 - \$ mv foo bar = \$ cp foo bar; rm foo

```
vagrant@dev [11:37:01] ~
$ ls
bye.txt  project1

vagrant@dev [11:37:01] ~
$ cp bye.txt project1/

vagrant@dev [11:37:27] ~
$ cd project1/
vagrant@dev [11:37:30] ~/project1
$ ls
bye.txt  hello.txt
```

C/C++ Compiler in Linux: GCC

- gcc(GNU Compiler Collection)
 - includes front ends for C, C++, Objective-C, Fortran, Ada, Go, and D, as well as libraries for these languages
- Originally, it was a C compiler. It was extended to compile C++ and other languages



GCC commands

- `gcc -E`
 - Stop after the preprocessing stage
- `gcc -S`
 - Stop after the stage of compilation proper
- `gcc -c`
 - Compile or assemble the source files, but do not link.
- `gcc -o <file>`
 - Place output in `<file>`
- `gcc -g`
 - Produce debugging information in the operating system's native format
 - [Debugging options](#)

Build your program with Makefile

- make utility is a simple tool to organize code compilation
- Advantages
 - Utility to aid in the design and maintenance of large programs
 - Relieves the programmer of the burden of remembering which files need recompilation
- Disadvantages
 - It is difficult to manage source code which consists of many files
 - Every change requires long compilation
- Use cases
 - Linux Kernel
 - Redis
 - Any other bunch of popular applications

What should we use this?

- Assume that there are a hundred of source files and they are correlated to each other such as
 - a001.c, a002.c, a003.c, ..., a100.c

```
gcc -c a001.c
```

```
gcc -c a002.c
```

```
gcc -c a003.c
```

...

- If you have to modify a single line inside of codes, you should compile all the files

How to use (1)

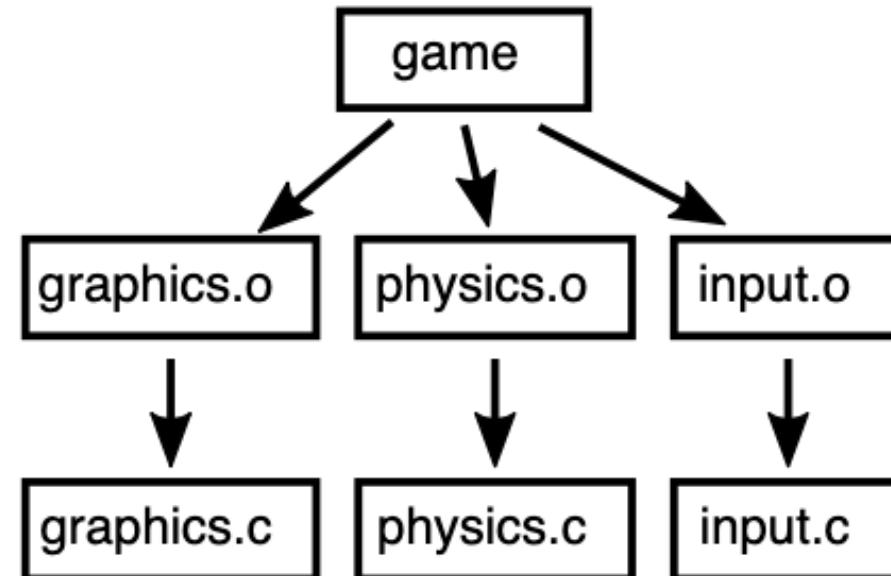
Rule

Target: dependencies

command

...

command

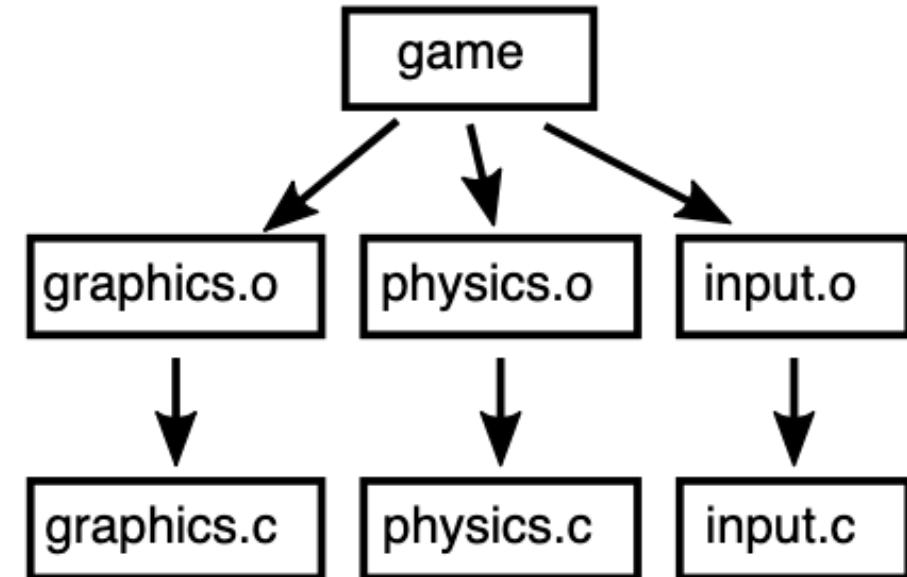


How to use (2)

```
game: graphics.o physics.o input.o  
      gcc -o game graphics.o physics.o \  
           input.o
```

```
graphics.o: graphics.c  
      gcc -c graphics.c
```

...



How do we edit the source files?

- Option 1: Use vim on Linux
 - You don't need to use additional editor, but you may need some time to be familiar with that
 - Reference
 - Vim official tutorial for Korean (터미널에 명령어 입력)
\$ vimtutor -g ko
 - [완전 초보를 위한 Vim](#)
 - [Vim에 대해 점진적으로 학습하기](#)
 - [Interactive Vim tutorial](#)
 - [VIM Adventures: 게임으로 배우는 Vim](#)
- Option 2: Choose your preferred editor
 - If you dislike to use vim on Linux, you may need to edit your file on your PC and copy the file to Linux
 - It is annoying because you need to copy file whenever you change your source code

<https://missing-semester-kr.github.io/>

여러분의 CS 교육에서 누락된 학기

CS 교과과정은 운영체제에서 기계학습에 이르기까지 CS 내의 고급 주제에 대해 모두 가르쳐 주지만, 거의 다루지 않는 중요한 한 가지 과목이 있으며, 대신 학생들이 스스로 알아내야 하는 툴의 숙련도가 있습니다. 이 수업에서 우리는 여러분에게 command line 을 마스터하는 방법, 강력한 text editor 사용, version control system의 화려한 기능 사용 방법 등을 가르쳐 줄 것입니다!

학생들은 교육 과정 동안 이러한 툴을 사용하여 수백 시간을 보내므로, 가능한 한 유연하고 마찰이 없는 경험을 하는 것이 이치에 맞습니다. 이러한 툴들을 마스터하게 되면 자유자재로 툴을 알맞게 사용하는 방법을 알아가는 시간을 덜 쓸 수 있을 뿐만 아니라 이전에 불가능할 정도로 복잡해 보였던 문제들을 해결할 수 있습니다.

이 [수업의 동기](#)에 대해 읽어보세요.

시간표

- [1/13: 수업 개요 + 셸\(SHELL\)](#)
- [1/14: 셸 툴과 스크립팅](#)
- [1/15: Editors \(Vim\)](#)
- [1/16: Data Wrangling](#)
- [1/21: Command-line Environment](#)
- [1/22: Version Control \(Git\)](#)
- [1/23: Debugging and Profiling](#)
- [1/27: Metaprogramming](#)
- [1/28: Security and Cryptography](#)
- [1/29: 기타 흥미로운 주제들](#)
- [1/30: Q&A](#)

녹화된 수업 영상은 [YouTube](#)에서 볼 수 있습니다.

수업 내용

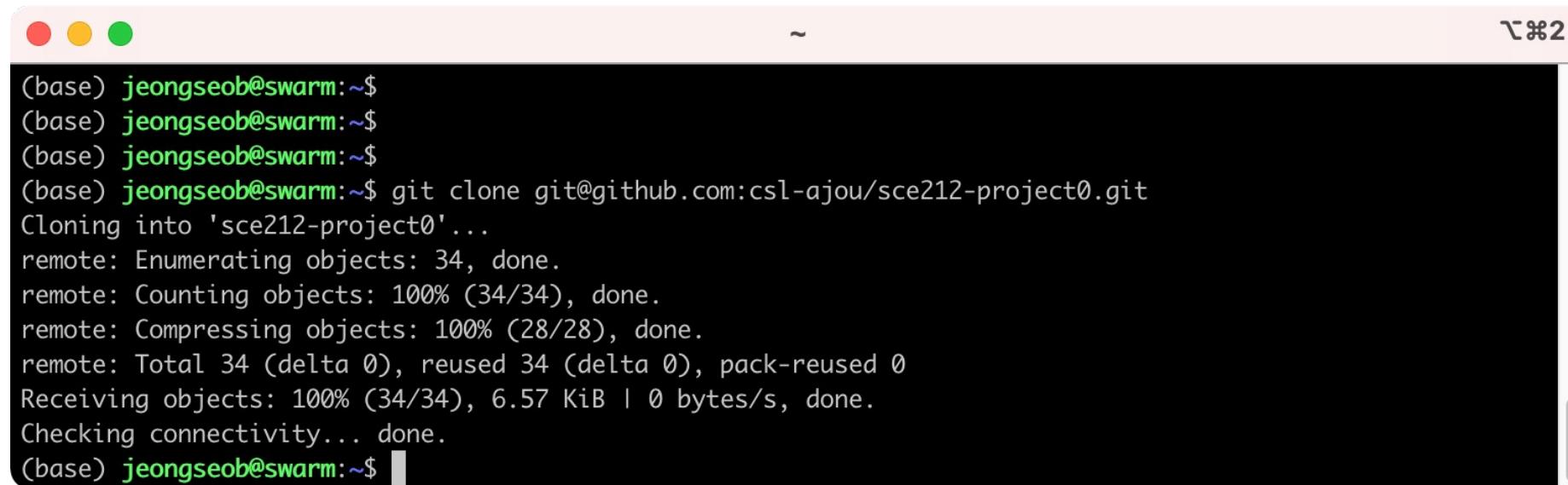
Staff: 이 수업은 [Anish](#), [Jon](#), 그리고 [Jose](#)가 함께 가르칩니다. **Questions:** missing-semester@mit.edu로 Email을 보내주시면 됩니다.

Let's prepare the programming assignment #0

0. Carefully read the handout file
1. Download the skeleton codes from Github we provided
2. Try to compile the skeleton code
 - You can see some errors or wrong output because the skeleton code is not completed
3. Coding
 - Write your own code on top of the provided skeleton code
 - Check whether my output is correct or not
4. Submit your code

1. Download the skeleton code

```
$] git clone git@github.com:csl-ajou/sce212-project0.git
```



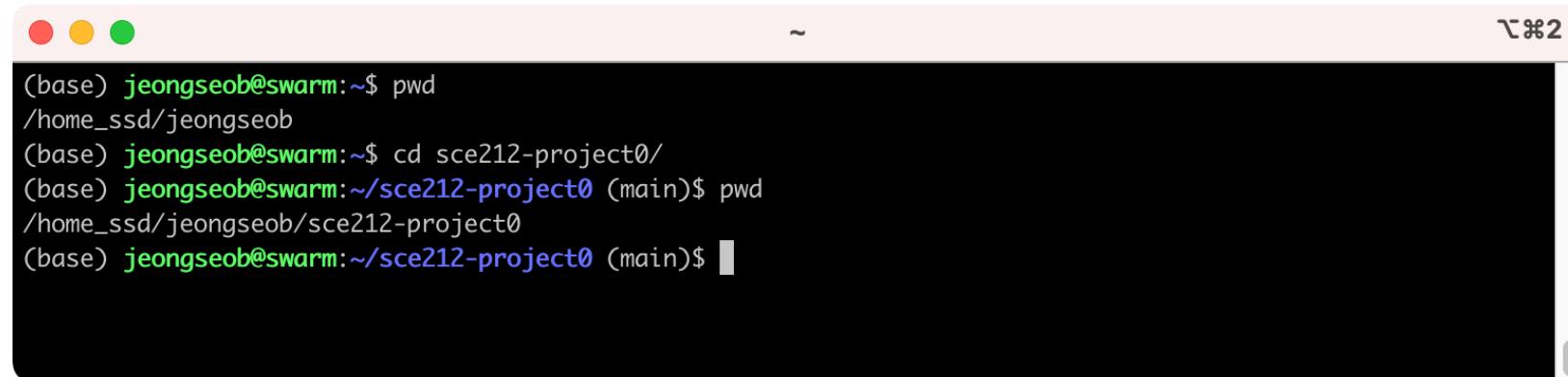
A screenshot of a macOS terminal window. The window has a light pink header bar with three circular icons (red, yellow, green) on the left and a close button on the right. The title bar on the right says "Terminal". The main area is a black terminal window with white text. It shows a user's session starting at the '(base)' prompt, followed by several identical prompts, then the command 'git clone git@github.com:csl-ajou/sce212-project0.git'. The terminal then displays the progress of cloning a repository, including object enumeration, counting, compressing, and receiving objects, along with a final check for connectivity. The session ends with another '(base)' prompt.

```
(base) jeongseob@swarm:~$  
(base) jeongseob@swarm:~$  
(base) jeongseob@swarm:~$  
(base) jeongseob@swarm:~$ git clone git@github.com:csl-ajou/sce212-project0.git  
Cloning into 'sce212-project0'...  
remote: Enumerating objects: 34, done.  
remote: Counting objects: 100% (34/34), done.  
remote: Compressing objects: 100% (28/28), done.  
remote: Total 34 (delta 0), reused 34 (delta 0), pack-reused 0  
Receiving objects: 100% (34/34), 6.57 KiB | 0 bytes/s, done.  
Checking connectivity... done.  
(base) jeongseob@swarm:~$
```

2. Try to compile the skeleton code

```
$ ] pwd
```

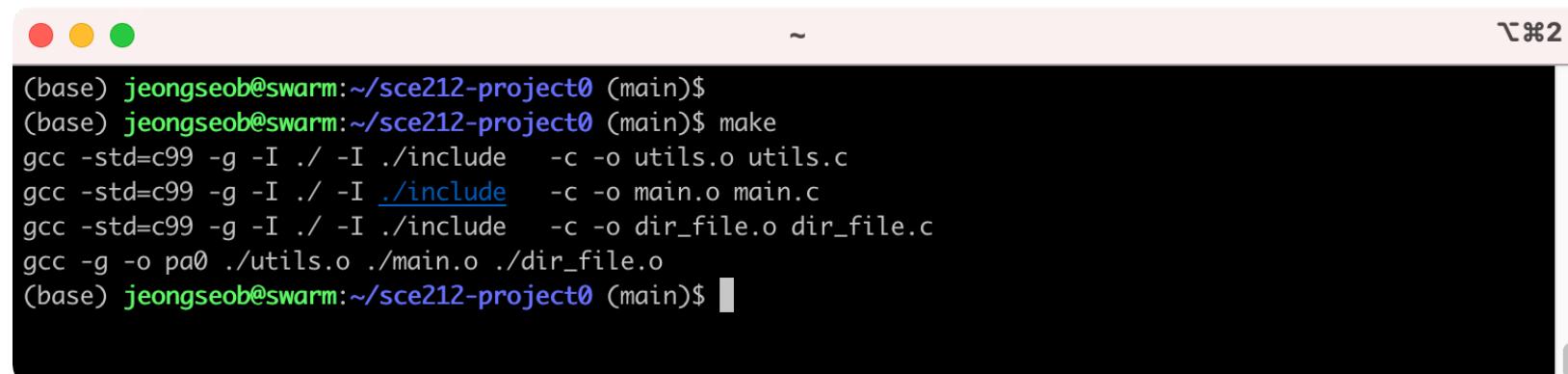
```
$ ] cd sce212-project0/
```



A screenshot of a macOS terminal window titled '终端' (Terminal). The window has three colored title bar buttons (red, yellow, green) on the top left and a close button on the top right. The terminal prompt '(base)' is followed by the user's name 'jeongseob' and the host name 'swarm'. The command entered is 'pwd', which prints the current working directory '/home_ss.../jeongseob'. The terminal window is set against a light blue background.

```
(base) jeongseob@swarm:~$ pwd  
/home_ss.../jeongseob  
(base) jeongseob@swarm:~$ cd sce212-project0/  
(base) jeongseob@swarm:~/sce212-project0 (main)$ pwd  
/home_ss.../jeongseob/sce212-project0  
(base) jeongseob@swarm:~/sce212-project0 (main)$ 
```

```
$ ] make
```

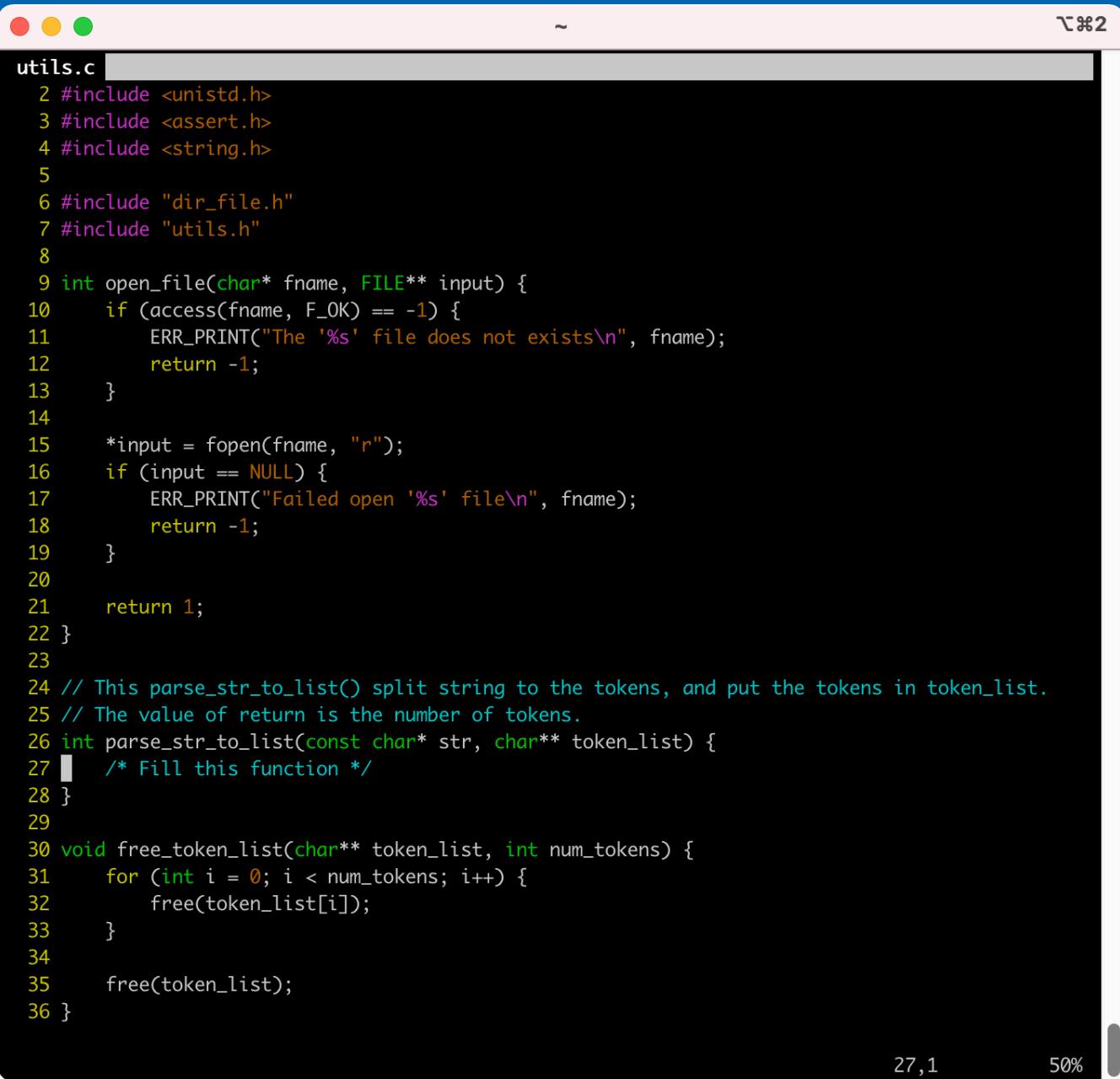


A screenshot of a macOS terminal window titled '终端' (Terminal). The window has three colored title bar buttons (red, yellow, green) on the top left and a close button on the top right. The terminal prompt '(base)' is followed by the user's name 'jeongseob' and the host name 'swarm'. The command entered is 'make', which triggers a series of GCC compilation commands for C source files. The terminal window is set against a light blue background.

```
(base) jeongseob@swarm:~/sce212-project0 (main)$  
(base) jeongseob@swarm:~/sce212-project0 (main)$ make  
gcc -std=c99 -g -I ./ -I ./include -c -o utils.o utils.c  
gcc -std=c99 -g -I ./ -I ./include -c -o main.o main.c  
gcc -std=c99 -g -I ./ -I ./include -c -o dir_file.o dir_file.c  
gcc -g -o pa0 ./utils.o ./main.o ./dir_file.o  
(base) jeongseob@swarm:~/sce212-project0 (main)$ 
```

3. Coding

```
(base) jeongseob@swarm:~/sce212-project0 (main)$  
(base) jeongseob@swarm:~/sce212-project0 (main)$ vim utils.c
```



The image shows a Mac OS X desktop environment with a terminal window open. The title bar of the terminal window says "utils.c". The terminal prompt "(base)" is visible at the top. The main area of the terminal shows the following command:

```
jeongseob@swarm:~/sce212-project0 (main)$ vim utils.c
```

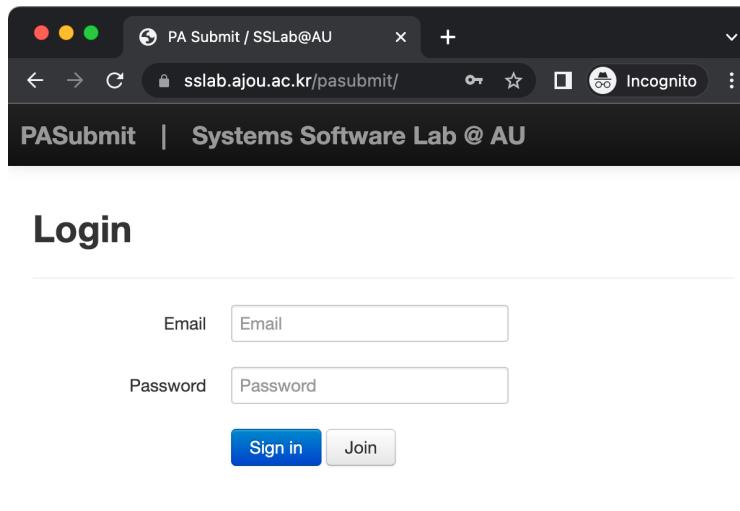
Below the terminal window, there is a large code editor window displaying a C program. The code is color-coded for syntax highlighting. The code includes functions for file operations and token lists. The code editor has a dark theme with syntax highlighting in green, blue, and yellow. The status bar at the bottom right of the code editor shows "27,1" and "50%".

```
utils.c  
1 #include <unistd.h>  
2 #include <assert.h>  
3 #include <string.h>  
4  
5 #include "dir_file.h"  
6 #include "utils.h"  
7  
8 int open_file(char* fname, FILE** input) {  
9     if (access(fname, F_OK) == -1) {  
10         ERR_PRINT("The '%s' file does not exists\n", fname);  
11         return -1;  
12     }  
13     *input = fopen(fname, "r");  
14     if (*input == NULL) {  
15         ERR_PRINT("Failed open '%s' file\n", fname);  
16         return -1;  
17     }  
18     return 1;  
19 }  
20  
21 // This parse_str_to_list() split string to the tokens, and put the tokens in token_list.  
22 // The value of return is the number of tokens.  
23 int parse_str_to_list(const char* str, char** token_list) {  
24     /* Fill this function */  
25 }  
26  
27 void free_token_list(char** token_list, int num_tokens) {  
28     for (int i = 0; i < num_tokens; i++) {  
29         free(token_list[i]);  
30     }  
31     free(token_list);  
32 }
```

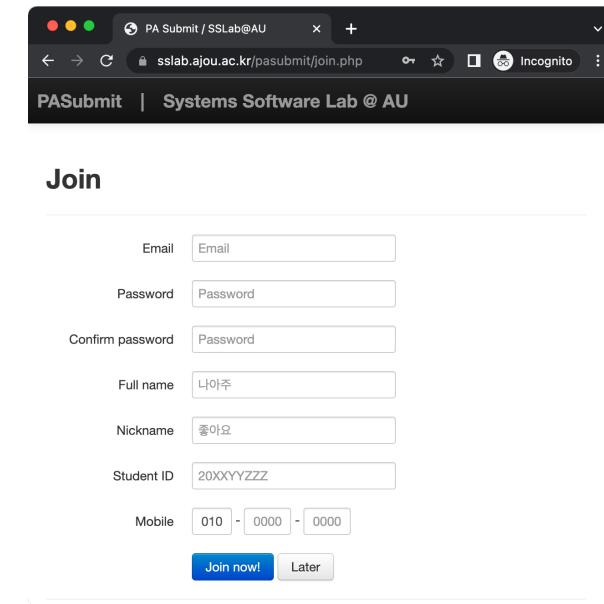
4. Submitting your PA #0

- <https://sslab.ajou.ac.kr/pasubmit/>

- Make your account if you do not have the account already
- If you took Prof. Sang-Hoon Kim's courses before, you may have your account
- If you are the first time, you need to join the site



The screenshot shows the login page of the PA Submit system. The title bar reads "PA Submit / SSLab@AU". The main content area has a dark header with the text "PASubmit | Systems Software Lab @ AU". Below this is a "Login" section with two input fields: "Email" and "Password". At the bottom are two buttons: "Sign in" (in blue) and "Join" (in grey).



The screenshot shows the "Join" page of the PA Submit system. The title bar reads "PA Submit / SSLab@AU". The main content area has a dark header with the text "PASubmit | Systems Software Lab @ AU". Below this is a "Join" section with several input fields:

- Email: Email
- Password: Password
- Confirm password: Password
- Full name: 나아주
- Nickname: 좋아요
- Student ID: 20XXYYZZZ
- Mobile: 010 - 0000 - 0000

At the bottom are two buttons: "Join now!" (in blue) and "Later".

Dashboard of pasubmit

The screenshot shows a web browser window for the PA Submit system at sslab.ajou.ac.kr/pasubmit/main.php. The title bar reads "PA Submit / SSLab@AU". The main content area displays the following information:

PASubmit | Systems Software Lab @ AU

ACTIVE CLASSES
[SCE212] Computer Organization and Architecture

CLASS
[Assignments](#) (highlighted)
[Class Homepage](#)

CLASS MANAGEMENT
[Students](#)
[Submissions](#)

SYSTEM MANAGEMENT
[Testing Queue](#)
[Assignments](#)
[My Info](#)
[Logout](#)

PA0: Warming up C programming

Let's review your C programming coding skills

September 8 (Thu), 6:00pm ~ September 22 (Thu), 11:59pm

[Handout »](#) [Submit »](#)

Submission

PA Submit / SSLab@AU [Handout »](#)

sslab.ajou.ac.kr/pasubmit/main.php?classIdx=23&menu=Assignments&paldx=90

PASubmit | Systems Software Lab @ AU

ACTIVE CLASSES
[SCE212]
Computer
Organization
and Architecture

CLASS
Assignments
Class
Homepage

CLASS MANAGEMENT
Students
Submissions

SYSTEM MANAGEMENT
Testing Queue
Assignments
My Info
Logout

PA0: Warming up C programming

Let's review your C programming coding skills

Due on September 22 (Thu), 11:59pm

Code

Time
2022/09/08 11:51:56

File
pa0-submission.tar.gz

Report

Time
N/A

Git

Time
URL
N/A

Submit **Submit** **Register**

#	Date/Time	File	Compile	Execution	Final	Test	-
rev.1	2022/09/08 11:51:56	pa0-submission.tar.gz	Passed	Success!!	0	End	Final

PA Submit / SSLab@AU

sslab.ajou.ac.kr/pasubmit/main.php?classIdx=23&menu=Assignments&paldx=90

PASubmit | Systems Software Lab @ AU

ACTIVE CLASSES

[SCE212] Computer Organization and Architecture

CLASS

Assignments

Class Homepage

CLASS MANAGEMENT

Students Submissions

SYSTEM MANAGEMENT

Testing Queue Assignments My Info Logout

PA0 - Warning up C programming

Downloads

Name Date Modified Size Kind

Name	Date Modified	Size	Kind
pa0-handout.pdf	Today 11:58 AM	298 KB	PDF Document
pa0-submission.tar.gz	Today 11:51 AM	2 KB	gzip compressed archive
2022f-sce...eton.tar.gz	Today 11:07 AM	3 KB	gzip compressed archive
2022f-sce...nce.tar.gz	Today 11:07 AM	4 KB	gzip compressed archive
> pa0-b-skeleton	Today 10:39 AM	--	Folder
pa0-b-skeleton.tar.gz	Today 10:39 AM	3 KB	gzip compressed archive
sce-212-수강생학번.xlsx	Yesterday 6:05 PM	10 KB	Microsoft Excel (.xlsx)
2022 기업...련 서식.hwp	Yesterday 4:11 PM	121 KB	Microsoft Word Document
> ????.?????...사업 계획서	Yesterday 3:10 PM	--	Folder
sce212-pr...master.zip	Yesterday 2:50 PM	429 KB	ZIP archive
고성능 컴퓨...집) (2).hwp	Sep 2, 2022 1:32 PM	90 KB	Microsoft Word Document
[불임.1] 20...안내서.hwp	Sep 2, 2022 1:28 PM	121 KB	Microsoft Word Document
[불임.2] 초...신청서.hwp	Sep 2, 2022 1:28 PM	101 KB	Microsoft Word Document
> Project7	Sep 1, 2022 4:02 PM	--	Folder

Cancel Open

Final Test

© Systems Software Lab at Ajou University, 2019-2022.

PA Submit / SSLab@AU

sslab.ajou.ac.kr/pasubmit/main.php?classIdx=12&menu=Assignments&paldx=33

SSLab@AU

ACTIVE CLASSES
[SCE212] Computer Organization and Architecture

CLASS
Assignments
Class Homepage

SYSTEM MANAGEMENT
Testing Queue
My Info
Logout

sslab.ajou.ac.kr says
Update Complete

OK

PA0-A

Handout

Comprehension about the pointer used in C programming

Due on March 31 (Tue) 11:59pm

Code

Time
2020/03/20 11:51:42

File
main.c

▶ Submit

Report

Time
File
N/A

▶ Submit

Git

Time
URL
N/A

▶ Register

#	Date/Time	File	Compile	Execution	Final	Test	-
rev.1	2020/03/20 11:51:42	main.c	Passed	Success!!	0	End	Final

We recommend to use the Google Chrome Browser. © Ajou University Systems Software Laboratory, 2019.

This screenshot shows a web browser window for the SSLab@AU assignment submission system. The URL is sslab.ajou.ac.kr/pasubmit/main.php?classIdx=12&menu=Assignments&paldx=33. The page displays information for assignment PA0-A, which is a comprehension about pointers in C programming, due on March 31 at 11:59pm. The sidebar shows the user is in the 'Assignments' section of the [SCE212] Computer Organization and Architecture class. A modal dialog box from 'sslab.ajou.ac.kr' informs the user that the update is complete. Below the modal, there are three main sections: 'Code' (Time: 2020/03/20 11:51:42, File: main.c), 'Report' (Time: N/A, File: N/A), and 'Git' (Time: N/A, URL: N/A). At the bottom, a table lists a single submission entry: rev.1, Date/Time 2020/03/20 11:51:42, File main.c, Compile Passed, Execution Success!!, Final 0, Test End, and a final column with a 'Final' button. The footer suggests using Google Chrome and credits Ajou University Systems Software Laboratory, 2019.

PA Submit / SSLab@AU

sslab.ajou.ac.kr/pasubmit/main.php?classIdx=23&menu=Assignments&paldx=90

PASubmit | Systems Software Lab @ AU

PA0: Warming up C programming

Let's review your C programming coding skills

Due on September 22 (Thu), 11:59pm

Code

Time
2022/09/08 12:02:51

File
pa0-submission.tar.gz

Report

Time

File
N/A

Git

Time

URL
N/A

#	Date/Time	File	Compile	Execution	Final	Test	-
rev.2	2022/09/08 12:02:51	pa0-submission.tar.gz	-	-	-	Run	Final
rev.1	2022/09/08 11:51:56	pa0-submission.tar.gz	Passed	Success!!	0	End	Final

© Systems Software Lab at Ajou University, 2019-2022.

PA Submit / SSLab@AU

sslab.ajou.ac.kr/pasubmit/main.php?classIdx=23&menu=Assignments&paldx=90

PASubmit | Systems Software Lab @ AU

PA0: Warming up C programming

Let's review your C programming coding skills

Due on September 22 (Thu), 11:59pm

Code

Time
2022/09/08 12:02:51

File
pa0-submission.tar.gz

Report

Time

File
N/A

Git

Time

URL
N/A

#	Date/Time	File	Compile	Execution	Final	Test	-
rev.2	2022/09/08 12:02:51	pa0-submission.tar.gz	Passed	Success!!	0	<button>End</button>	<button>Final</button>
rev.1	2022/09/08 11:51:56	pa0-submission.tar.gz	Passed	Success!!	0	<button>End</button>	<button>Final</button>

PA Submit / SSLab@AU

sslab.ajou.ac.kr/pasubmit/main.php?classIdx=23&menu=Assignments&paldx=90

PASubmit | Systems Software Lab @ AU

PA0: Warming up C programming

Execution Result

Let's review your submission.

Due on Sep 08, 2022

Code

Time: 2022/09/08 12:00:00

File: pa0-submission

Testcase: test_0 (1/5)

- (0)main.c
- (0)README.md
- (0).gitignore
- (0)figure1.png
- (0)figure2.png

Testcase: test_1 (2/5)

- (0)lecture1.pdf
- (0)lecture2.pdf
- (0)lecture3.pdf
- (0)lecture4.pdf
- (0)lecture5.pdf
- (0)lecture6.pdf
- (0)computer_organization_and_design.pdf

Testcase: test_2 (3/5)

Close

Git

Time

URL: N/A

Register

Action	Final	Test	-
Pass!!	0	End	Final
Pass!!	0	End	Final

FAQ

- 몇번 제출 (시도) 가능한가요?
 - 특별히 언급하지 않는 이상 제한 없음
- 늦게 제출 허용 되나요?
 - 토큰을 사용 하면 되고 토큰이 부족한 경우 딜레이로 인한 패널티 (50%) 있음
 - 늦게 제출하는 것이 제출하지 않은것 보다는 점수를 많이 받을 수 있음
- 토큰은 어떻게 사용하나요?
 - 기본적으로 늦게 제출시 토큰을 먼저 사용하는 것으로 시스템에서 간주 함
 - 늦게 제출했지만 다음에 토큰을 쓰고 싶다? → 담당교수에게 따로 문의