

# C++ For C Coders 1

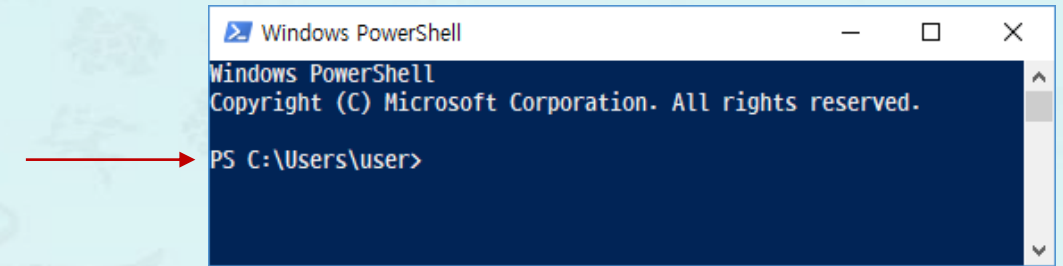
**Data Structures**  
**C++ for C Coders**

한동대학교 김영섭교수  
idebtor@gmail.com

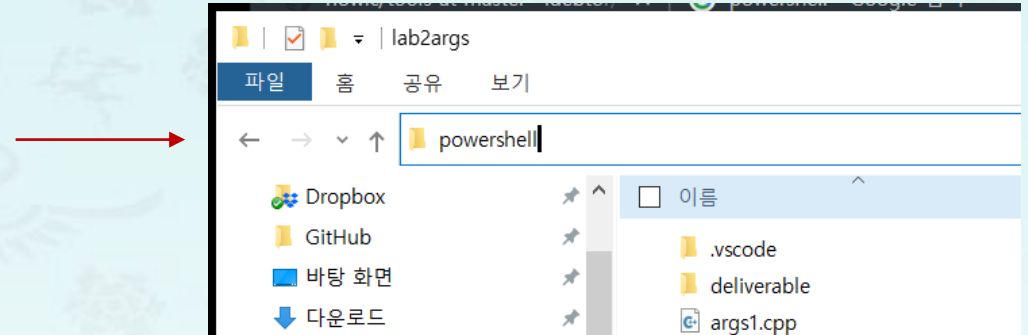
Start a console  
PS, cmd, bash  
Revisiting installation of g++  
C vs C++  
In-house programming principles  
VS Code  
Hello World

# Start a console at your folder

1. Start PowerShell through "Start" menu

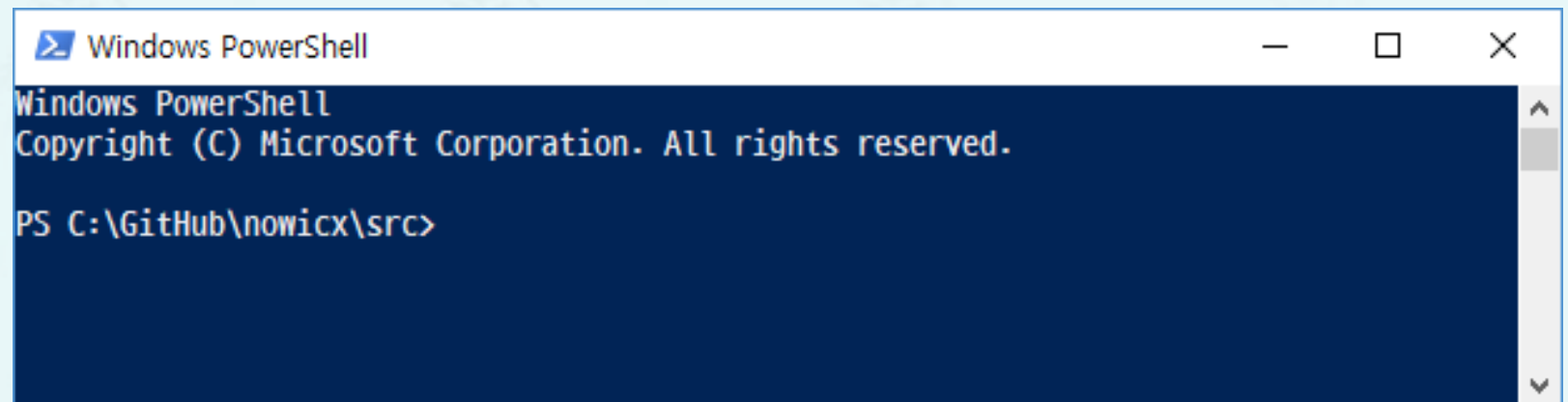
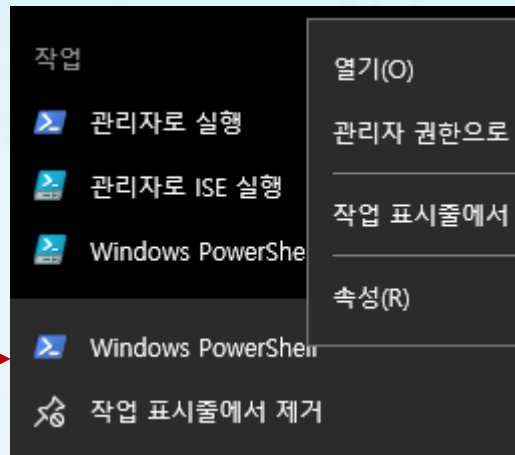


2. Move to the folder you want to start.
  - Enter 'powershell' at the File Explorer



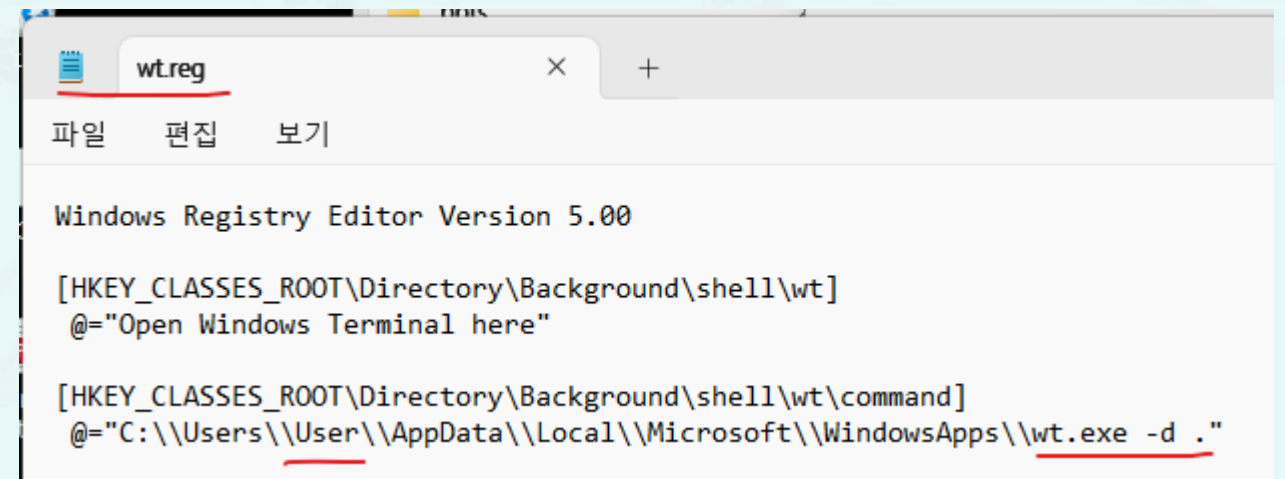
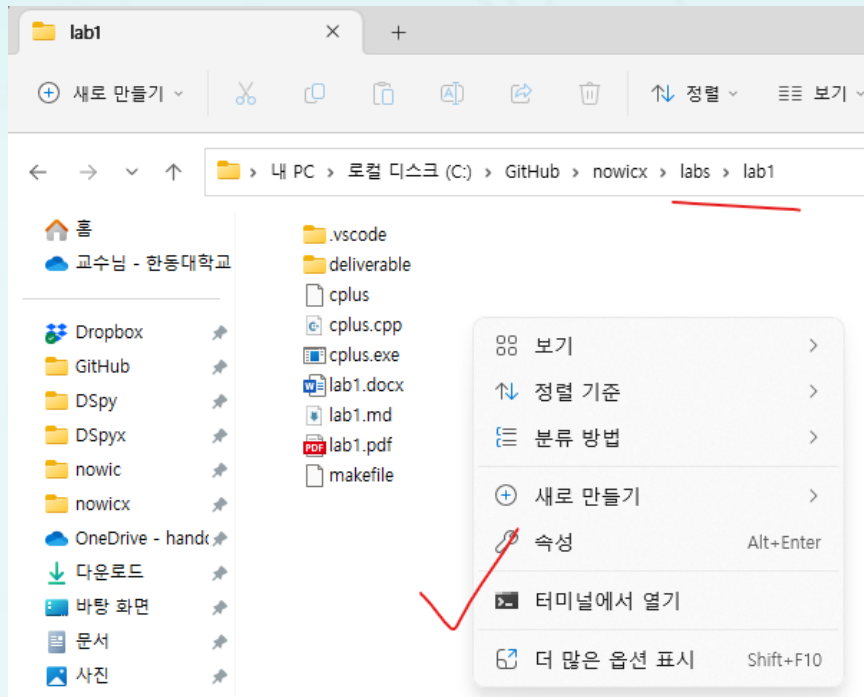
# Taskbar Icon: Start a console at your favorite folder

1. Start PowerShell if you don't have a PowerShell icon in Taskbar.
2. In the Taskbar right-click and pin to keep a link there.
3. **Again right click the icon in taskbar** and then right-click Windows PowerShell and choose Properties
4. Enter your preferred directory in the Start in: input field and press OK.
5. Start from the taskbar icon



# Start a "Windows Terminal" at the current folder in File Explorer

1. Download "**Windows Terminal**" from MS Store.
2. Have "**wt.reg**" file ready to add it to the registry.  
Replace "**User**" in "wt.reg" shown below with **your user's name of the Windows**.
3. Merge[병합] it to the registry. (Click right on the file "wt.reg" and choose "병합")
4. You may restart the Windows if it is not effective immediately.



# PowerShell, Cmd, vs Bash (Borne-again shell), sh, ksh, csh

PowerShell (Cmdlet)	PowerShell (Alias)	Windows Cmd	Unix shell	Description
Get-ChildItem	gci, dir, <b>ls</b>	dir	<b>ls</b>	Lists all files and folders in the current or given folder
Get-Content	gc, type, <b>cat</b>	type	<b>cat</b>	Gets the content of a file
Get-Command	gcm	help	<b>type, which</b>	Lists available commands
Get-Help	help, man	help	<b>man</b>	Prints a command's documentation on the console
Clear-Host	cls, <b>clear</b>	cls	<b>clear</b>	Clears the screen
Copy-Item	cp, copy, <b>cp</b>	copy, xcopy	<b>cp</b>	Copies files and folders to another location
Move-Item	mi, move, <b>mv</b>	move	<b>mv</b>	Moves files and folders to a new location
Remove-Item	ri, del, <b>rmdir</b> , rd, <b>rm</b>	del, rmdir, rd	<b>rm, rmdir</b>	Deletes files or folders
Rename-Item	rni, ren, <b>mv</b>	ren, rename	<b>mv</b>	Renames a single file, folder, hard link or symbolic link
Get-Location	gl, cd, <b>pwd</b>	cd	<b>pwd</b>	Displays the working path (current folder)
Pop-Location	popd	popd	<b>popd</b>	Changes the working path to the location most recently pushed onto the stack
Push-Location	pushd	pushd	<b>pushd</b>	Stores the working path onto the stack
Set-Location	sl, <b>cd</b> , chdir	cd, chdir	<b>cd</b>	Changes the working path, cd .. (move to parent folder), cd ~ (move to HOME folder)
Write-Output	<b>echo</b> , write	echo	<b>echo</b>	Prints strings or other objects to the standard output
Get-Process	gps, <b>ps</b>	tlist, tasklist	<b>ps</b>	Lists all running processes
Stop-Process	spps, <b>kill</b>	kill, taskkill	<b>kill</b>	Stops a running process
Select-String	sls	findstr	<b>find, grep</b>	Prints lines matching a pattern
Set-Variable	sv, set	set	<b>env, export, set, setenv</b>	Creates or alters the contents of an environment variable

# PowerShell, Cmd, vs Bash (Borne-again shell), sh, ksh, csh

---

This Linux Command Handbook covers **60 core Bash commands** you will need as a developer. Each entry includes example code and tips for when to use that command.

<https://www.freecodecamp.org/news/the-linux-commands-handbook/>



# Install GNU C/C++ Compiler (**MSYS2** first, **MinGW-w64** next)

- MSYS = "**M**inimal **SYS**tem", is a Bourne Shell command line interpreter system  
MinGW = "**M**inimalist **G**NU for **W**indows"

- Install MSYS2 first from <http://www.msys2.org/>  
Then use a command at the msys2 window.

```
$ pacman -Syu
```

```
$ pacman -S mingw-w64-x86_64-gcc
```

- Add the following path to the PATH environment variable

```
C:/msys64/mingw64/bin
```

```
# for g++
```

```
C:/msys64/usr/bin
```


```
# for commands such as ls, cat, rm
```

 be cautious, not 'user'

- Add .bash\_profile

```
C:/msys64/home/user/.bash_profile
```


```
# a sample is provided
```


 check your home folder

# .bash\_profile (an example) in C:\msys64\home\user

```
alias ls='ls -Gp --color=auto'
alias ll='ls -alkF'
alias rm='rm -i'
alias h='history'
LS_COLORS=$LS_COLORS':no=00:di=36;01'
LS_COLORS=$LS_COLORS':*.h=1;33:*.exe=31:*.o=1;32:*.md=1;33'
export LS_COLORS
export PATH=$PATH:"/c/msys64/mingw64/bin"
export PATH=$PATH:"/c/msys64/usr/bin"
export PATH=$PATH:"/c/Program Files/Git/bin"
export PATH=$PATH:"/c/Users/$USER/AppData/Local/Programs/Microsoft VS Code/bin"
# code () { VSCODE_CWD="$PWD" open -n -b "com.microsoft.VSCode" --args $* ;} For macOS
# Setting my dev folder as a startup folder of msys.
HOME="/c/GitHub/nowic"
cd $HOME
echo $PWD

# @$(hostname) may be added, if necessary, after $(whoami)
GREEN="$(tput setaf 2)"
RESET="$(tput sgr0)"
PS1='${GREEN}$(pwd -W)> ${RESET}'
```

 This is the folder Code is installed.

 check your github folder



# .bash\_profile (an example) in C:\msys64\home\user

For macOS

```
alias ls='ls -Gp --color=auto'
alias ll='ls -alkF'
alias rm='rm -i'
alias h='history'
LS_COLORS=$LS_COLORS':no=00:di=36;01'
LS_COLORS=$LS_COLORS':*.h=1;33:*.exe=31:*.o=1;32:*.md=1;33'
export LS_COLORS
export PATH=$PATH:"/c/msys64/mingw64/bin"
export PATH=$PATH:"/c/msys64/usr/bin"
export PATH=$PATH:"/c/Program Files/Git/bin"
export PATH=$PATH:"/c/Users/$USER/AppData/Local/Programs/Microsoft VS Code/bin"
# code () { VSCODE_CWD="$PWD" open -n -b "com.microsoft.VSCode" --args $* ;}
# Setting my dev folder as a startup folder of msys.
HOME="/c/GitHub/nowic"
cd $HOME
echo $PWD

# @$(hostname) may be added, if necessary, after $(whoami)
GREEN="$(tput setaf 2)"
RESET="$(tput sgr0)"
PS1='${GREEN}$(pwd -W)> ${RESET}'
```

Open a terminal & run the following commands

참고: <https://qastack.kr/superuser/886132/where-is-the-zshrc-file-on-mac>

> **touch ~/.zshrc** # 폴더와 파일 생성

> **open ~/.zshrc** # 파일 열고, .bash\_profile 내용 입력

Close the file and reopen a terminal.

↙ This is the folder Code is installed.

Uncomment for macOS

↙ check your github folder

# How to test .bash\_profile in C:\msys64\home\user

Open a terminal & run the following commands

참고: <https://qastack.kr/superuser/886132/where-is-the-zshrc-file-on-mac>

> **touch ~/.zshrc** # 폴더와 파일 생성

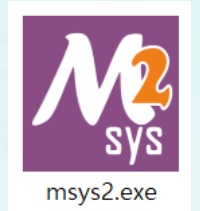
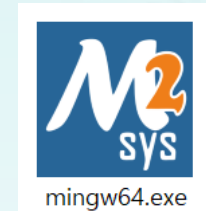
> **open ~/.zshrc** # 파일 열고, .bash\_profile 내용 입력

Close the file and reopen a terminal.

- For windows users, after adding .bash\_profile as shown below:

C:/msys64/home/user/.bash\_profile # a sample is provided

- Create a short-cut from either mingw64.exe or msys2.exe.
- Double click the short-cut and test g++ installation as shown below:



```
MSYS2 MSYS Shell
c:/msys64/home/User/.bash_profile
/home/User
C:/GitHub/nowic> g++ --version
g++.exe (Rev5, Built by MSYS2 project) 10.3.0
Copyright (C) 2020 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

C:/GitHub/nowic> |
```

# C++ for C Coders

---

- C
- Dennis Ritchie
- 1972
- 29 Keywords
  - Procedural programming
- C++
- Stroustrup
- 1985
- 63 Keywords by 1996
- C++ as a better C
  - Object-oriented programming
  - **Generic programming**  
**STL(Standard Template Library)**  
“Swiss Army Knife”

# C++ specific keywords

---

asm	dynamic_cast	namespace	reinterpret_cast	try
bool	explicit	new	static_cast	typeid
catch	false	operator	template	typename
class	friend	private	this	using
const_cast	inline	public	throw	virtual
delete	mutable	protected	true	wchar_t

# From C to C++

---

- Even if C++ is slanted toward object-oriented programming (OOP), you can nevertheless use any C++ compiler to compile C code and benefits from some C++ goodies.
- C++ as a better C

# In-house Programming Principles

---

- DRY or DIE
  - Don't repeat yourself, or Duplication is Evil.
  - To make you professional coders and save time.
- KISS
  - Keep it simple, stupid!, Keep it simple and short.
  - To make you professional coders – code maintainable
- NMN
  - No magic number
  - To make you professional coders - code robust.
- NSE
  - No side effects
  - To make you professional coders - code clean.



# Source Code Editor: Visual Studio Code

---

- Code is an open source and a cross-platform editor software developed by Microsoft. It becomes one of the best code editors in the market, lately.
- Features:
  - IntelliSense - autocomplete
  - Refactoring
  - Run and Debug
  - Built-in Git
  - Extensions
- Refer to the following documents in nowic/tools
  - [VSCode Step1. GCC 및 MSYS2 설치.pdf](#)
  - [VSCode Step2. C C++ 코딩 준비.pdf](#)
  - [VSCode Step3. Makefile 사용 방법.pdf](#)
  - [VSCode Step4. JSON 파일 설정 및 사용 방법.pdf](#)
  - [VSCode Step5. 코딩 및 디버깅 테스트.pdf](#)

# Write "Hello World" program in C++

- Open Code editor

```
$ code hello.cpp
```

- Add the source code.
- Save the file.

- Compile and Execute

```
$ g++ hello.cpp
```

```
$ ./a.exe
```

```
$ ./a
```

```
$ a
```

← **PowerShell**

← **macOS**

← **cmd**

```
// file: hello.cpp
#include <iostream>

int main() {
    std::cout << "Hello World!" << std::endl;
}
```

← Scope Resolution Operator(::)

```
// file: hello.cpp
#include <iostream>
using namespace std;

int main() {
    cout << "Hello World!" << endl;
}
```

Without using Scope Resolution Operator(::)

# C++ For C Coders 1

**Data Structures**  
**C++ for C Coders**

한동대학교 김영섭교수  
idebtor@gmail.com

Start a console  
PS, cmd, bash  
Revisiting installation of g++  
C vs C++  
In-house programming principles  
Hello World