Compilation

g++

- Compiler you need to use for this class

- Transforms each source file (.cpp) into an object file (.o)

- Then it links all of the object files into a single executable

g++

g++ -c

g++ -o

Makefiles

- Automate the build process
- Comprised of a series
    - Actions (targets)
    - Prerequisites (dependency list)
    - Commands

Example

Target

build: programming_review.cpp

Dependency List

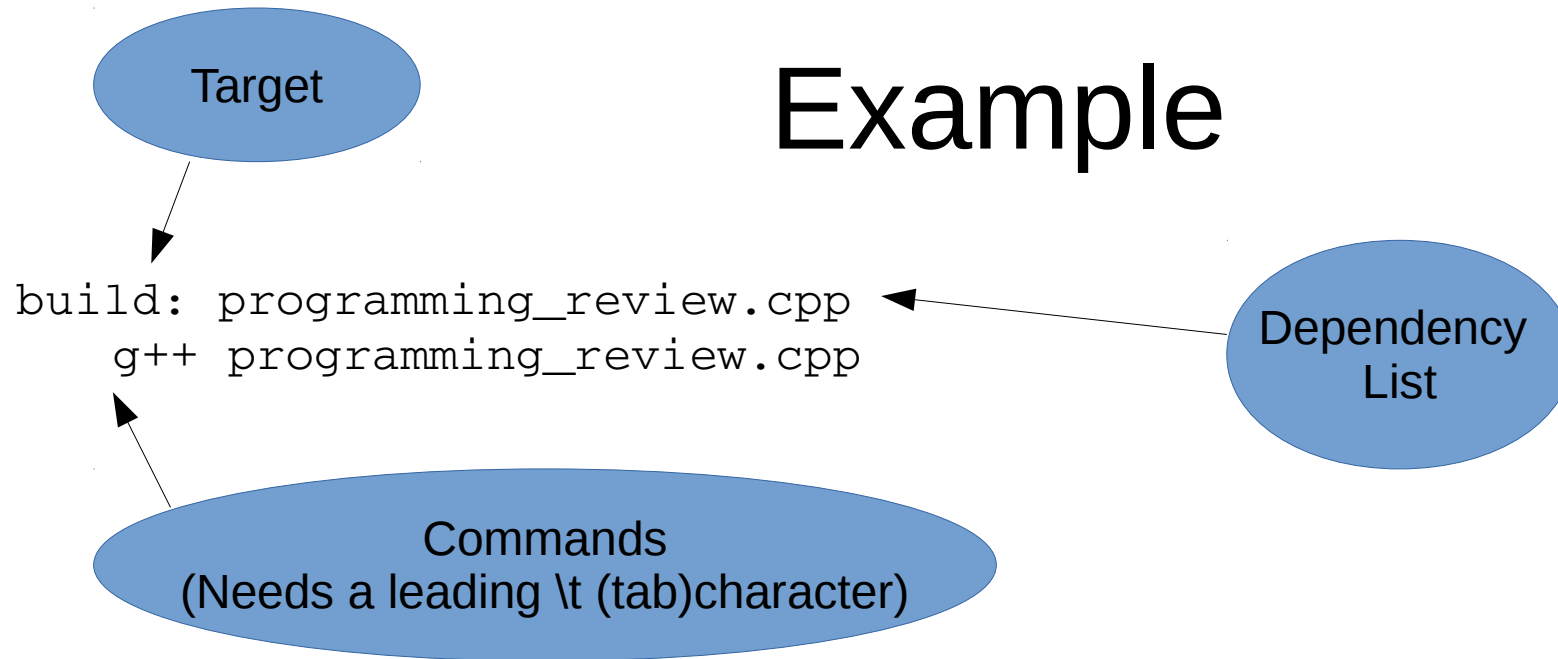g++ programming_review.cpp

Commands
(Needs a leading \t (tab)character)

```
dennis@dennis-Gazelle-Professional:~/Documents/TA/2016 fall cop3330/code$ make
g++ programming_review.cpp
dennis@dennis-Gazelle-Professional:~/Documents/TA/2016 fall cop3330/code$ ls
a.out  Makefile  programming_review.cpp
dennis@dennis-Gazelle-Professional:~/Documents/TA/2016 fall cop3330/code$
```

Additional Targets

```
build: programming_review.o
    g++ -o programming_review.x programming_review.o

compile: programming_review.cpp
    g++ -c -o programming_review.o programming_review.cpp

clean:
    Rm -f programming_review.x
```

Other Ideas for Targets

- Archival / backup
- Clearing the screen
- Running the program
- Testing the program
- Searching across multiple files
- Number of lines of code

Features

- Make has many more features, e.g.
  - Pattern matching
  - Variables
  - Recursive make
  - Suppressing output
- I'll provide more detailed examples as they become relevant