

Linux Development Tools - make

Prerequisite


- ✚ [Linux CLI Basics](#)
- ✚ [gcc Basics](#)

Agenda

- ✚ Why do we need a makefile?
- ✚ Writing a makefile to create an executable using multiple source files
- ✚ Makefile processing using make

Learning Outcome

- ✚ Understand the project directory structure
- ✚ Understand the contents of a makefile specific to target (all, clean), dependencies and macros
- ✚ To be able to create a makefile to compile, link and generate an executable consisting of one or more source (.c, .h) files
- ✚ To be able to interpret and resolve the errors in makefile execution

Topic	Reference Video	Reference Links
Linux Development Tool – make	 Automating Build using make - makefile execution using dependency tree, make, clean target, make working, missing tab separator, rm -f (44)	Why make ?
		Build Automation using makefile and make
		Common mistakes in makefile and debugging
		Detailed help - A-Z of Gnu Make
		Sample Make
		FAQ on make

Summary

- ✚ For issues in makefile, use -n option with make to view the substituted gcc command statements to be executed
- ✚ Use rm -f in clean target to override issue if any and to forcefully delete *.o and exe
- ✚ Use make with -f option to run makefile named differently

Assignments

Additional References