

# make

make — Unix program for “making” things...

...by running commands based on what's changed

what commands? based on *rules* in *makefile*

## make rules

```
main.o: main.c main.h extra.h  
▶      clang -c main.c
```

before colon: target(s) (file(s) generated/updated)

after colon: prerequisite(s)

following lines prefixed by a tab character:  
command(s) to run

make will run the commands if any prerequisite is newer than the target (or the target does not exist)

# make rules

```
main.o: main.c main.h extra.h  
    clang -c main.c
```

before colon: **target(s)** (file(s) generated/updated)

after colon: prerequisite(s)

following lines prefixed by a tab character:  
command(s) to run

make will run the commands if any prerequisite is newer than the target (or the target does not exist)

# make rules

```
main.o: main.c main.h extra.h  
▶      clang -c main.c
```

before colon: target(s) (file(s) generated/updated)

after colon: prerequisite(s)

following lines prefixed by a tab character:  
command(s) to run

make will run the commands if any prerequisite is newer than the target (or the target does not exist)

## make rules

```
main.o: main.c main.h extra.h  
▶      clang -c main.c
```

before colon: target(s) (file(s) generated/updated)

after colon: prerequisite(s)

following lines prefixed by a **tab** character:  
command(s) to run

make will run the commands if any prerequisite is newer than the target (or the target does not exist)

## make rules

```
main.o: main.c main.h extra.h
```

```
▶      clang -c main.c
```

before colon: target(s) (file(s) generated/updated)

after colon: prerequisite(s)

following lines prefixed by a tab character:

command(s) to run

make will run the commands if any prerequisite is newer than the target (or the target does not exist)

# running make

“make *target*”

- look in `Makefile` in current directory for rules

- check if *target* is up-to-date

- if not, rebuild it (and dependencies, if needed) so it is

“make”

- look in `Makefile` in current directory for rules

- build target for first rule in `Makefile`

- convention: first target called “all”

- (and not file called all actually generated)