2.7 Rules for Cleaning the Directory

Compiling a program is not the only thing you might want to write rules for. Makefiles commonly tell how to do a few other things besides compiling a program: for example, how to delete all the object files and executables so that the directory is ‘clean’.

编译程序并不是您可能想要为其编写规则的唯一事情。 Makefiles 通常告诉除了编译程序之外如何做一些其他事情：例如，如何删除所有目标文件和可执行文件以使目录 “干净” 。

Here is how we could write a make rule for cleaning our example editor:

```
clean:
        rm edit $(objects)
```

In practice, we might want to write the rule in a somewhat more complicated manner to handle unanticipated situations. We would do this:

在实践中，我们可能希望以更复杂的方式编写规则来处理意料之外的情况。 我们会这样做：

```
.PHONY : clean
clean :
        -rm edit $(objects)
```

This prevents make from getting confused by an actual file called clean and causes it to continue in spite of errors from rm. (See Phony Targets, and Errors in Recipes.)

这可以防止 make 被一个名为 clean 的实际文件混淆，并导致它继续运行，尽管来自 rm 的错误。 （请参阅虚假目标和配方中的错误。）

A rule such as this should not be placed at the beginning of the makefile, because we do not want it to run by default! Thus, in the example makefile, we want the rule for edit, which recompiles the editor, to remain the default goal.

不应将这样的规则放在 `makefile` 的开头，因为我们不希望它默认运行！ 因此，在示例 makefile 中，我们希望重新编译 editor 的 edit 规则保持默认目标。

Since clean is not a prerequisite of edit, this rule will not run at all if we give the command ‘make’ with no arguments. In order to make the rule run, we have to type ‘make clean’. See How to Run make.

因为 clean 不是 edit 的先决条件，所以如果我们给出不带参数的命令 “make” ， 这条规则根本不会运行。 为了使规则运行，我们必须输入 “make clean” 。 请参阅如何运行 make。
