

# Your First Filter

---

Now, let's look at writing our own filters. We'll continue working with the **Replit** project we used when learning about redirection. You can open it from your Replit account, or click the "running man", and re-fork it.



The program `filter.cpp` is the simplest possible version of the built-in filter `cat`. Remember that the `cat` command reads a character from **standard input** and sends it on to **standard output**, stopping only when there is no more input to be processed.

There are three ways to process input:

- Raw, or **unformatted input** (a byte or character at a time)
- **Line-oriented** input (one line at a time)
- **Formatted** or token-based input (a "word" at a time)

The program `filter.cpp` uses **raw, unformatted input**. Build and run the program in **Replit** like this. First, open the **Shell** tab and then type:

1. `make filter`. If you have only a single file, you can build it by giving make the name of the output file. The `make` program finds `filter.cpp` file and then compiles and links it.
2. `./filter < input.txt` to run the program. This should produce **exactly the same output** as using `cat`.

Note, in Unix, to run a program located in **the current working directory**, first type the directory `./` and then the name of the program. You can repeat any of the previous exercises replacing `cat` with `./filter`, and the results should be the same.



This course content is offered under a [CC Attribution Non-Commercial](#) license. Content in this course can be considered under this license unless otherwise noted.