Now that we have a function written that handles the task at hand in a more general manner (i.e. it can handle any package and any date), it is worth taking a closer look at the function and asking whether it is written in the most useful possible manner. In particular, it could be argued that this function does too many things:

1. Construct the path to the remote and local log file
2. Download the log file (if it doesn't already exist locally)
3. Read the log file into R
4. Find the package and return the number of downloads

It might make sense to abstract the first two things on this list into a separate function. For example, we could create a function called check_for_logfile() to see if we need to download the log file and then num_download() could call this function.

```
check_for_logfile <- function(date) {
        year <- substr(date, 1, 4)
        src <- sprintf("http://cran-logs.rstudio.com/%s/%s.csv.gz",
                       year, date)
        dest <- file.path("data", basename(src))
        if(!file.exists(dest)) {
                val <- download.file(src, dest, quiet = TRUE)
                if(!val)
                        stop("unable to download file ", src)
        }
        dest
}
```

This file takes the original download code from num_download() and adds a bit of error checking to see if download.file()was successful (if not, an error is thrown with stop()).

Now the num_download() function is somewhat simpler.

```
num_download <- function(pkgname, date = "2016-07-20") {
        dest <- check_for_logfile(date)
        cran <- read_csv(dest, col_types = "ccicccccci", progress = FALSE)
        cran %>% filter(package == pkgname) %>% nrow
}
```

In addition to being simpler to read, another key difference is that the num_download() function does not need to know anything about downloading or URLs or files. All it knows is that there is a function check_for_logfile() that just deals with getting the data to your computer. From there, we can just read the data with read_csv() and get the information we need. This is the value of abstraction and writing functions.

Mark as completed

👍 👎 ⚐