

Extracted from:

# Small, Sharp Software Tools

Harness the Combinatoric Power of  
Command-Line Tools and Utilities

This PDF file contains pages extracted from *Small, Sharp Software Tools*, published by the Pragmatic Bookshelf. For more information or to purchase a paperback or PDF copy, please visit <http://www.pragprog.com>.

Note: This extract contains some colored text (particularly in code listing). This is available only in online versions of the books. The printed versions are black and white. Pagination might vary between the online and printed versions; the content is otherwise identical.

Copyright © 2019 The Pragmatic Programmers, LLC.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

The Pragmatic Bookshelf

Raleigh, North Carolina

The  
Pragmatic  
Programmers

# Small, Sharp Software Tools

Harness the Combinatoric  
Power of Command-Line Tools  
and Utilities



Brian P. Hogan  
*edited by Tammy Coron*

# Small, Sharp Software Tools

Harness the Combinatoric Power of  
Command-Line Tools and Utilities

Brian P. Hogan

The Pragmatic Bookshelf

Raleigh, North Carolina



Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and The Pragmatic Programmers, LLC was aware of a trademark claim, the designations have been printed in initial capital letters or in all capitals. The Pragmatic Starter Kit, The Pragmatic Programmer, Pragmatic Programming, Pragmatic Bookshelf, PragProg and the linking *g* device are trademarks of The Pragmatic Programmers, LLC.

Every precaution was taken in the preparation of this book. However, the publisher assumes no responsibility for errors or omissions, or for damages that may result from the use of information (including program listings) contained herein.

Our Pragmatic books, screencasts, and audio books can help you and your team create better software and have more fun. Visit us at <https://pragprog.com>.

The team that produced this book includes:

Publisher: Andy Hunt  
VP of Operations: Janet Furlow  
Managing Editor: Susan Conant  
Development Editor: Tammy Coron  
Copy Editor: L. Sakhi MacMillan  
Indexing: Potomac Indexing, LLC  
Layout: Gilson Graphics

For sales, volume licensing, and support, please contact [support@pragprog.com](mailto:support@pragprog.com).

For international rights, please contact [rights@pragprog.com](mailto:rights@pragprog.com).

Copyright © 2019 The Pragmatic Programmers, LLC.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

ISBN-13: 978-1-68050-296-1  
Book version: P1.0—May 2019

# Preface

---

The *graphical user interface*, or GUI, is designed for the average user and provides most of the necessary features needed for user interaction. Your smartphone doesn't have a user-accessible command-line interface, or CLI, because it's aimed at the average consumer. You are only able to access basic features. But you're not an average consumer. You're a programmer, a software developer crafting the best code from the finest bits. With command line, you can do everything you can do in the GUI, and more—that is, if you know how.

In this book, you'll learn the basics and use those basic building blocks to tackle more advanced tasks. One of the best features of the CLI is that you can turn those basic commands into scripts that you can execute, meaning you can automate common tasks with ease. That's why popular programming languages and frameworks like Node.js, Ember, React, Elixir, Ruby, Python, and many others rely so heavily on the CLI. With this book, you'll learn how to put this interface to work for you.

## A Tale of Many CLIs

When it comes down to it, there are many command-line interfaces. If you've worked on a Linux or Unix server, you've likely had some interaction with its interface. If you're on Windows, chances are you've opened the Command Prompt or PowerShell to run some commands.

Many commands are different. For example, a POSIX system might use the command `ls` to list the files in a folder, whereas a Windows system uses the `dir` command instead. The command-line interfaces on macOS and POSIX systems use the `rm` command to remove files and directories, but the Windows CLI requires different commands for each.

This book covers the Bash command-line interface found on Linux, macOS, and Windows 10 through the Windows Subsystem for Linux.<sup>1</sup> You'll use the

---

1. <https://docs.microsoft.com/en-us/windows/wsl/about>

Bash shell. While there are other shells popular among developers, like Fish and ZSH, this book focuses on Bash because it's the default on many systems, so setup is minimal. Also, this book uses the GNU version of command-line utilities, instead of the BSD versions found on macOS. However, most of these differences are minor, and I'll point out when things are slightly different and show you how to install the GNU versions of these tools if necessary.

## What's In (and not in) This Book

In [Chapter 1, Getting Your Feet Wet, on page ?](#), you'll get a quick tour through some of the most useful parts of the CLI. You'll navigate around the filesystem, run commands as a superuser, learn to install a few programs you'll use in the book using the package managers, create files, learn about streams of text, and learn how to use the help system.

Next, in [Chapter 2, Creating an Ubuntu Virtual Machine with VirtualBox, on page ?](#), you'll create an Ubuntu virtual machine you can use to practice the commands in the rest of this book. While you don't have to follow this chapter, using virtualization is a great way to practice. You'll create the virtual machine, install the Ubuntu operating system, and take a snapshot, so you can restore your environment if something goes wrong.

[Chapter 3, Navigating the Filesystem, on page ?](#), lets you get comfortable moving around the filesystem on the command line. You'll learn where to look for things and how to get around quickly using absolute paths, relative paths, and a few handy shortcuts. You'll also learn where to find things on a Unix-like filesystem, and how to find how much space is available on your filesystem.

Next, in [Chapter 4, Working with Files and Directories, on page ?](#), you'll work with files and directories. You'll expand on what you learned in the first chapter, and do more advanced operations in more detail. You'll concatenate files, read larger files, and manage permissions. You'll move, rename, and copy files and directories, learn about links, and learn about the filesystem itself. You'll learn how to find files quickly, and how to identify file types.

In [Chapter 5, Streams of Text, on page ?](#), you'll dive deeper into redirecting program output and use `grep` to filter the output. You'll learn how to redirect output to another program, and learn the difference between `STDIN`, `STDOUT`, and `STDERR`. You'll learn about the `sed` stream editor, `xargs`, and `awk`.

In [Chapter 6, The Shell and Environment, on page ?](#), you'll accelerate your workflow by making customizations to your command-line environment. You'll create a custom prompt that tells you information about your session. You'll create aliases for complicated commands so you can type fewer keystrokes.

You'll create configuration files that let you start your session with the customizations you make, and you'll learn how to set environment variables you can refer to later in scripts or applications.

In [Chapter 7, Running and Managing Programs, on page ?](#), you'll explore additional ways to run programs, leverage your command history to reuse commands effectively, and explore subshells. You'll also manage the processes running on your system and run programs in the background.

In [Chapter 8, Networking Tools, on page ?](#), you'll use command-line tools to look up IP addresses and domain names, fetch files and interact with web APIs, transfer files with `scp` and `rsync`, inspect open ports, and do low-level network communications with `netcat`.

In [Chapter 9, Automation, on page ?](#), you'll reduce repetitive tasks and create workflows. You'll use `make` to build a very basic static site generator that wraps content with headers and footers. Then, you'll use `Bash` to create an interactive script for creating a web project.

In [Chapter 10, Additional Programs, on page ?](#), you'll explore a handful of tools that can improve how you work. You'll use `ranger` to navigate and manage files and `direnv` to set environment variables for specific projects. You'll use some Python CLI tools to keep a journal of your work, run a basic web server, and view source code with syntax highlighting. You'll use `jq` to process JSON, manipulate documentation with `pandoc`, run commands when files change with `entr`, and work with APIs with `Siege` and `HTTpie`.

Finally, you'll find two appendices in the book as well. [Appendix 1, Command Quick Reference, on page ?](#), provides a list of the commands used in this book. [Appendix 2, Installing GNU Utilities on macOS, on page ?](#), explains how to install the GNU versions of many of the tools in this book on a Mac.

But you also won't find a few things in this book. First, there are a ton of commands available, and this book won't cover them all. I've selected the tools in this book based on research, interactions with students in classes and workshops, and personal experience. In addition, you'll use only a handful of options for each tool, with the assumption that you'll explore other options on your own. You'll use the Linux versions of these tools, rather than the BSD versions found on OpenBSD or FreeBSD.

I've also left out a few tools that I think are fantastic, but aren't a good fit. This book won't go into using alternative shells, it won't go over Windows or PowerShell command-line tools, and it doesn't spend time using tools like `Vim`, `Emacs`, or `tmux`. You also won't work with `Git` in this book, except in

one case when you'll use it to download and install another tool. When you need to use a command-line based editor, you'll use nano. When talking about running something in a detached terminal, you'll use screen. If you know how to use something different, you should feel comfortable doing so.

You'll work with real-world examples of useful tools that will accelerate your work, and you'll find exercises to practice them. When you're done, you'll know where to go next to learn more about each tool.

## How to Use This Book

You can use Windows, Linux, or macOS to follow along with the activities and exercises in this book—as long as you run a Bash shell. Thankfully, that's the default on macOS and most Linux variants, and you can set up Bash on Windows 10. However, you may want to use VirtualBox<sup>2</sup> to install the Ubuntu operating system and use that instead.

Using VirtualBox, you can run the commands in this book without worrying about doing anything to harm your computer, and you can take snapshots of your virtualized environment which you can revert if something goes wrong while you're working on the exercises in this book. You'll also avoid inconsistencies between the various operating systems while you're learning. [Chapter 2, Creating an Ubuntu Virtual Machine with VirtualBox, on page ?](#), explains this in more detail and walks you through setting this up.

Throughout the book, you'll see commands and output. Commands you'll type will look like this:

```
$ echo hello > hello.txt
```

The dollar sign (\$) represents the prompt from the Bash shell session. You won't type it when you type the command. It just denotes that this is a command you should type.

Occasionally, you'll see a series of commands. The dollar sign prompts let you see what you should type, and the rest is output from the program:

```
$ ls -l
total 36
drwxr-xr-x 2 brian brian 4096 Mar  2 11:59 Desktop
drwxr-xr-x 2 brian brian 4096 Mar  2 11:59 Documents
drwxr-xr-x 2 brian brian 4096 Mar  2 11:59 Downloads
drwxr-xr-x 2 brian brian 4096 Mar  2 11:59 Music
drwxr-xr-x 2 brian brian 4096 Mar  2 11:59 Pictures
drwxr-xr-x 2 brian brian 4096 Mar  2 11:59 Public
```

2. <https://virtualbox.org>



```
drwxr-xr-x 2 brian brian 4096 Mar  2 15:48 sharptools
drwxr-xr-x 2 brian brian 4096 Mar  2 11:59 Templates
drwxr-xr-x 2 brian brian 4096 Mar  2 11:59 Videos
$ mkdir Code
$ cd Code
```

Additionally, sometimes you'll see commands that span multiple lines, like this:

```
$ cat << 'EOF' > hello.txt
> Hello
> World
> EOF
```

The > symbol indicates that the command continues on that line. You'll see the > character in your shell whenever it's waiting for you to continue typing. You won't type that character either.

Finally, each chapter has some exercises for you to try, so you can practice what you learned. In addition, each chapter assumes you have experience with the content in the previous chapter. Feel free to jump around and explore, but don't go so fast that you don't get some practice time in. The commands in this book will help you be more productive, but only once you've practiced them enough so they're second nature.

## Online Resources

The book's website<sup>3</sup> has links to download some companion files for this book, including data files you may want for exercises, as well as scripts and configuration files you'll write in this book. You'll download those files from the CLI as you work through the book.

If you're reading the electronic version of this book, you can click the box above the code excerpts to download that source code directly.

In addition, the companion site for the book<sup>4</sup> has supplemental material, including videos and additional tutorials you'll find useful.

With all that out of the way, it's time to dive in. You'll start by doing some very basic command-line stuff on your own machine so you can see how the command-line interface works and what it can do for you.

3. <https://pragprog.com/titles/bhcldev>

4. <https://smallsharpsoftwaretools.com>