

- [1 \(888\) 979-2398](#)
- [Your Courses](#)
- [Downloads](#)
- [Job Board](#)
- [Gifting](#)
- [Account](#)
- [Blog](#)
- [Log Out](#)
- [Cart](#)
- [Q](#)

[🔔 Tutorial Features](#) [📄 All Access Subscription](#)

Follow author to receive email updates about new content

Follow

Progress

Command Line 0%

Command

Line

0%

Chapter Progress

Chapter 1: Basics 0%

Chapter

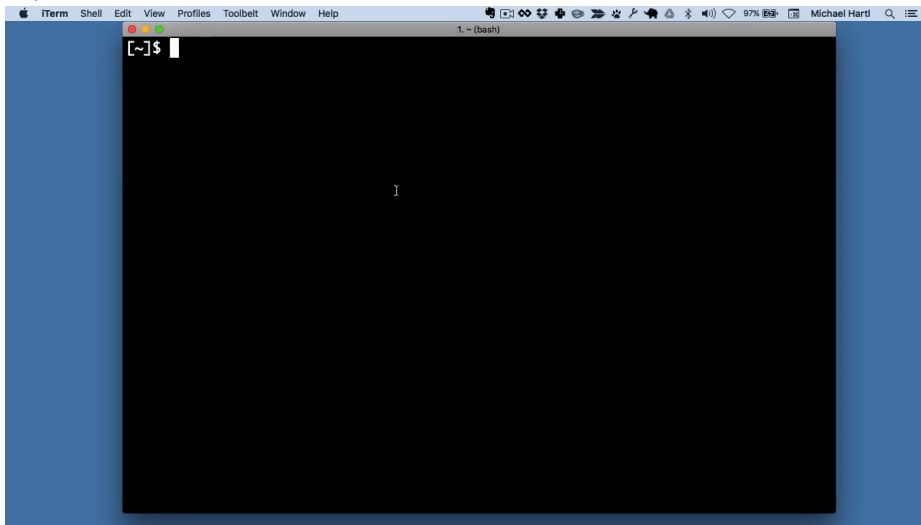
1:

Basics

0%

- [Frontmatter](#)
- [1. Basics](#)
  - [1.1 Introduction](#)
  - [1.2 Running a terminal](#)
    - [1.2.1 Exercises 0 / 2](#)
  - [1.3 Our first command](#)
    - [1.3.1 Exercises 0 / 2](#)
  - [1.4 Man pages](#)
    - [1.4.1 Exercises 0 / 2](#)
  - [1.5 Editing the line](#)
    - [1.5.1 Exercises 0 / 2](#)
  - [1.6 Cleaning up](#)
    - [1.6.1 Exercises 0 / 2](#)
  - [1.7 Summary](#)
    - [1.7.1 Exercises 0 / 3](#)
- [2. Manipulating files](#)
  - [2.1 Redirecting and appending](#)
    - [2.1.1 Exercises 0 / 3](#)
  - [2.2 Listing](#)
    - [2.2.1 Hidden files](#)
    - [2.2.2 Exercises 0 / 3](#)
  - [2.3 Renaming, copying, deleting](#)
    - [2.3.1 Unix terseness](#)
    - [2.3.2 Exercises 0 / 4](#)
  - [2.4 Summary](#)
    - [2.4.1 Exercises 0 / 4](#)

- [3. Inspecting files](#)
  - [3.1 Downloading a file](#)
    - [3.1.1 Exercises 0 / 4](#)
  - [3.2 Making heads and tails of it](#)
    - [3.2.1 Wordcount and pipes](#)
    - [3.2.2 Exercises 0 / 4](#)
  - [3.3 Less is more](#)
    - [3.3.1 Exercises 0 / 4](#)
  - [3.4 Grepping](#)
    - [3.4.1 Exercises 0 / 5](#)
  - [3.5 Summary](#)
    - [3.5.1 Exercises 0 / 5](#)
- [4. Directories](#)
  - [4.1 Directory structure](#)
    - [4.1.1 Exercises 0 / 3](#)
  - [4.2 Making directories](#)
    - [4.2.1 Exercises 0 / 3](#)
  - [4.3 Navigating directories](#)
    - [4.3.1 Exercises 0 / 4](#)
  - [4.4 Renaming, copying, and deleting directories](#)
    - [4.4.1 Grep redux](#)
    - [4.4.2 Exercises 0 / 4](#)
  - [4.5 Summary](#)
    - [4.5.1 Exercises 0 / 4](#)
  - [4.6 Conclusion](#)

[Pin Video](#)[Pop Out](#)

## [1.2 Running a terminal](#)

To run a command-line command, we first need to start a *terminal*, which is the program that gives us a command line. The exact details depend on the particular operating system you're using.

### [macOS](#)

On macOS, you can open a terminal window using the macOS application [Spotlight](#), which you can launch either by typing `⌘_` (Command-space) or by clicking on the magnifying glass in the upper right part of your screen. Once you've launched Spotlight,

you can start a terminal program by typing “terminal” in the Spotlight Search bar. (If you are interested in using a more advanced and customizable terminal program, I recommend installing [iTerm](#), but this step is optional.)

At this point, you might see the alert shown in [Listing 1.1](#).

Listing 1.1: A macOS terminal alert.

The default interactive shell is now zsh.  
To update your account to use zsh, please run `chsh -s /bin/zsh`.  
For more details, please visit <https://support.apple.com/kb/HT208050>.

```
[~]$
```

This alert is the result of a change made in [macOS Catalina](#). You don’t need to do anything about it right now; we’ll address this issue the first time it makes any difference in this tutorial ([Section 2.3](#)). For more information, see the Learn Enough blog post “[Using Z Shell on Macs with the Learn Enough Tutorials](#)”.

## [Linux](#)

On Linux, you can click the terminal icon as shown in [Figure 1.3](#). The result should be something like [Figure 1.4](#), although the exact details on your system will likely differ.

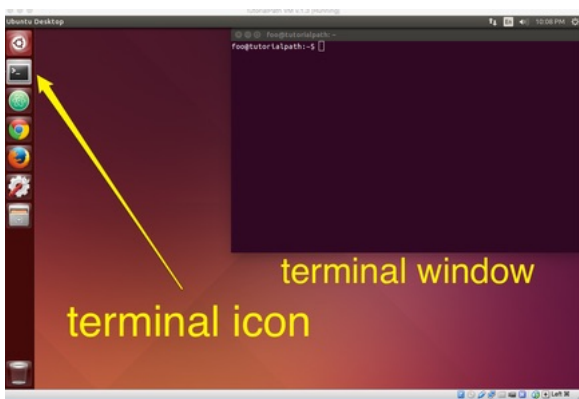


Figure 1.3: The Linux terminal icon.

## [Windows](#)

On Windows, the recommended option is to install Linux (which, incredibly, Microsoft has decided to support natively) as described in the [Windows section](#) of the free tutorial [Learn Enough Dev Environment to Be Dangerous](#). Once Linux is installed, you should look for a terminal icon as described in [Section 1.2.2](#). Apply your technical sophistication ([Box 1.4](#)) if you get stuck.

## [Terminal window](#)

Regardless of which operating system you use, your terminal window should look something like [Figure 1.4](#), though details may differ.

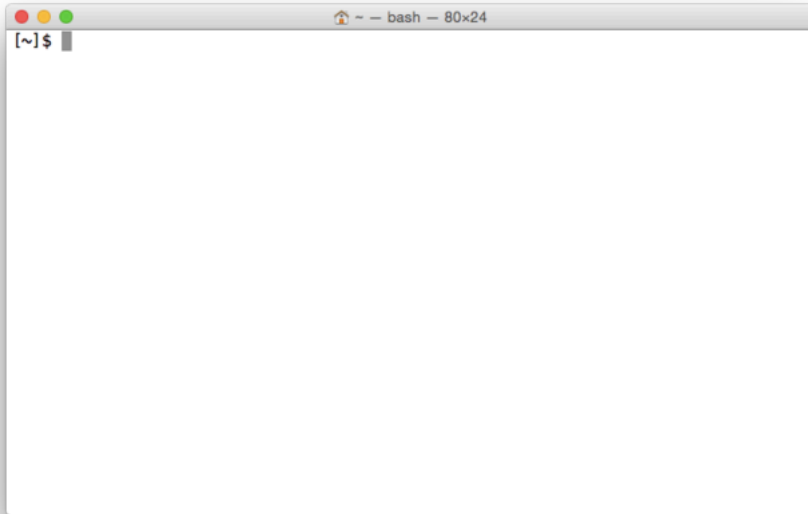


Figure 1.4: A terminal window.

The example we saw in [Figure 1.2](#) includes all of the typical elements of a command, as illustrated in [Figure 1.5](#): the *prompt* (to “prompt” the user to do something) followed by a *command* (as in “give the computer a command”), an *option* (as in “choose a different option”),<sup>3</sup> and an *argument* (as in the “[argument of a function](#)” in mathematics). It’s essential to understand that the prompt is supplied automatically by the terminal, and you do not need to type it. (Indeed, if you do type it, it will likely result in an error.) Moreover, the exact details of the prompt will differ, and are not important for the purposes of this tutorial ([Box 1.2](#)).

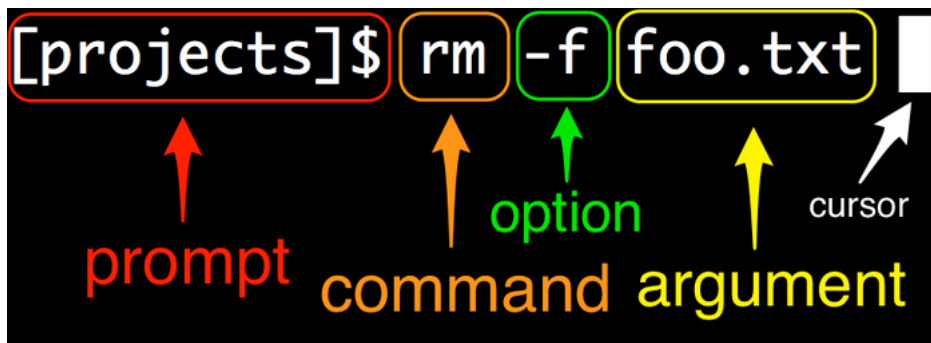


Figure 1.5: Anatomy of a command line. (Your prompt may differ.)

#### Box 1.2. What is the prompt?

Every command line starts with some symbol or symbols designed to “prompt” you to action. The prompt usually ends with a dollar sign \$ or a percent sign %, and is preceded by information that depends on the details of your system. For example, on some systems the prompt might look like this:

```
Michael's MacBook Air:~ mhartl$
```

In [Figure 1.4](#), the prompt looks like this instead:

```
[~]$
```

and in [Figure 1.5](#) it looks like this:

```
[projects]$
```

Finally, the prompt I’m looking at right now looks like this:

```
[learn_enough_command_line (first-draft)]$
```

For the purposes of this tutorial, the details of the prompt are not important, but we will discuss useful ways to customize the prompt starting in the next tutorial after this one ([Learn Enough Text Editor to Be Dangerous](#)).

### 1.2.1 Exercises

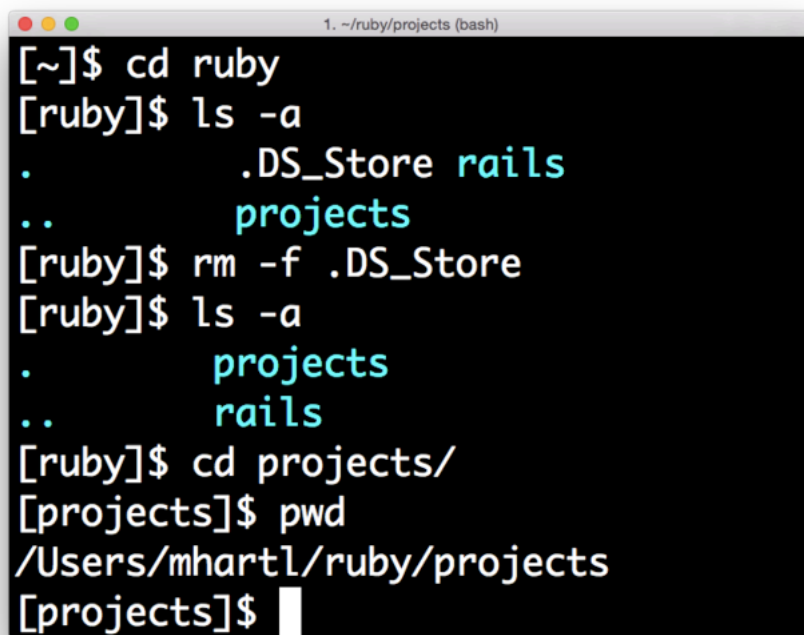
*Learn Enough Command Line to Be Dangerous* includes a large number of exercises. I strongly recommend getting in the habit of attempting them before moving on to the next section, as they reinforce the material we've just covered and will give you essential practice in using the many commands discussed. It's not generally the case that they are *required* to proceed, though, so if you get stuck it's sometimes a good idea to continue forward and then revisit the exercise at a later time. Indeed, this is good advice for the main text as well—you'll be surprised how often a seemingly impossible idea or intractable problem will look easy the second time around.

1. By referring to [Figure 1.5](#), identify the prompt, command, options, arguments, and cursor in each line of [Figure 1.6](#).

[👤 View all \( 347 \)](#) [+ Add Answer](#)

2. Most modern terminal programs have the ability to create multiple tabs ([Figure 1.7](#)), which are useful for organizing a set of related terminal windows.<sup>4</sup> By examining the menu items for your terminal program ([Figure 1.8](#)), figure out how to create a new tab. Extra credit: Learn the keyboard shortcut for creating a new tab. (Learning keyboard shortcuts for your system is an excellent habit to cultivate.)

[👤 View all \( 309 \)](#) [+ Add Answer](#)



```
1. ~/ruby/projects (bash)
[~]$ cd ruby
[ruby]$ ls -a
.      .DS_Store rails
..     projects
[ruby]$ rm -f .DS_Store
[ruby]$ ls -a
.      projects
..     rails
[ruby]$ cd projects/
[projects]$ pwd
/Users/mhartl/ruby/projects
[projects]$
```

Figure 1.6: A series of typical commands.

```

2. ~/rails_projects/sample_app_3rd_edition (bash)
~/rails_projects/sa... 361  ~ (bash) 362  ~ (bash) 363
[sample_app_3rd_edition (master)]$ ls -l
total 72
-rw-r--r--  1 mhartl  staff  1081 Nov 23 13:13 Gemfile
-rw-r--r--  1 mhartl  staff  8295 Nov 25 20:22 Gemfile.lock
-rw-r--r--  1 mhartl  staff  1803 Nov 25 20:11 Guardfile
-rw-r--r--  1 mhartl  staff   39 Sep 28 20:49 Procfile
-rw-r--r--  1 mhartl  staff  1414 Nov 25 20:11 README.md
-rw-r--r--  1 mhartl  staff  249 Nov 17 2014 Rakefile
drwxr-xr-x  9 mhartl  staff  306 Sep 28 20:49 app
drwxr-xr-x  7 mhartl  staff  238 Nov 17 2014 bin
drwxr-xr-x 12 mhartl  staff  408 Nov 25 20:10 config
-rw-r--r--  1 mhartl  staff  154 Nov 17 2014 config.ru
drwxr-xr-x  7 mhartl  staff  238 Nov 25 20:10 db
drwxr-xr-x  4 mhartl  staff  136 Nov 17 2014 lib
drwxr-xr-x  5 mhartl  staff  170 Nov 17 2014 log
drwxr-xr-x  8 mhartl  staff  272 Feb 19 2015 public
drwxr-xr-x  9 mhartl  staff  306 Sep 28 20:49 test
drwxr-xr-x  6 mhartl  staff  204 Feb 19 2015 tmp
drwxr-xr-x  3 mhartl  staff  102 Nov 17 2014 vendor
[sample_app_3rd_edition (master)]$

```

Figure 1.7: A terminal window with three tabs.

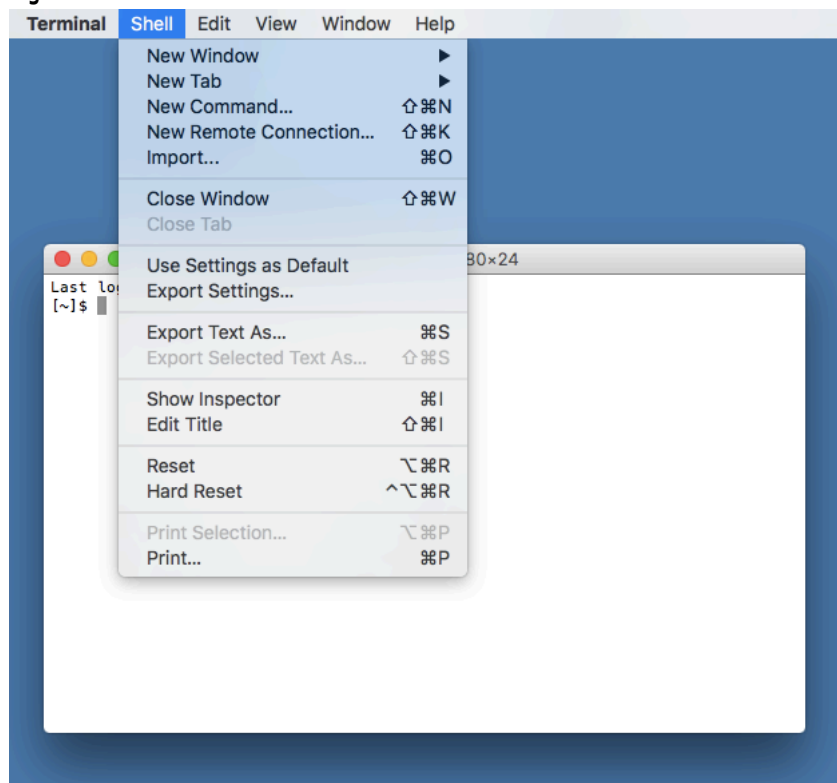


Figure 1.8: Some menu items for the default macOS terminal.

[NEXT: 1.3 Our first command](#)

## Join the Mailing List

Get occasional notifications about things like product discounts, blog posts, and new or updated tutorials. Unsubscribe at any time.

