

Getting ready

We will be using a tool from the suite of programs called ImageMagick, which was used in the previous recipe as well. Install it using your package manager.

How to do it...

Let's get started with using the `import` command from ImageMagick to take screenshots:

1. Take the screenshot of the whole screen:

```
$ import -window root screenshot.png
```
2. Manually select a region to take the screenshot:

```
$ import screenshot.png
```

3. Take a screenshot of a specific window:

```
$ import -window window_id screenshot.png
```

To find out `window_id`, run the command `xwininfo` and click on the window you want. Then pass this `window_id` to the `-window` option of `import .x`.

Managing multiple terminals from one

If you have been using the shell extensively, you would have noticed that at times you will need to have access to more than one terminal at once. If you are using a graphical terminal emulator like Konsole, you might use multiple tabs to achieve this.

However, what to do when you want a solution without using a graphical terminal emulator? Or you are logged into a remote machine and want multiple shells? In the latter case, opening multiple `ssh` connections will basically waste network bandwidth and even slow down things. We will see how to achieve multiple shells while avoiding these problems.

Getting ready

To achieve this, we will be using a utility called **GNU screen**. If the screen is not installed on your distribution by default, install it using the package manager.

How to do it...

- ▶ **Creating screen windows:** To create a new screen, run the command `screen` from your shell. You will see a welcome message with some information about the screen. Press space or return to begin. You will be given a new shell to enter the commands. To create a new window (which basically means a new shell), press `Ctrl + A` and then `C` (these are case sensitive).
- ▶ **Viewing a list of open windows:** While running the screen, if you want to see a list of all open windows, use the keystrokes `Ctrl + A` and `"`.
- ▶ **Switching between windows:** A usual requirement while using the screen is to cycle through the open windows in a next/previous fashion. For this, use the keystrokes `Ctrl + A` and `Ctrl + N` for the next window and `Ctrl + A` and `Ctrl + P` for the previous window.
- ▶ **Attaching to and detaching screens:** A useful feature of the screen is that you can save and load screen sessions, called detaching and attaching in screen terminology. To detach from the current screen session, press `Ctrl + A` and `Ctrl + D`. When starting the screen, to attach to an existing screen use:

```
screen -r -d
```

- ▶ Which tells the screen to attach the last screen session. If you have more than one detached sessions, the screen will output a list of them and then you can use:

```
screen -r -d PID
```

- ▶ Where, `PID` is the pid of the screen session you want to attach to.