CHAPTER 1

Introduction to Raspberry Pi

I hope you have gone through the Table of Contents and Introduction. If not, I highly recommend you go through them. This is the very first chapter of the book, and I welcome you all to the exciting journey of learning Linux with the Raspberry Pi Operating System.

In this chapter, we will learn the details about the most popular platform and single-board computer family of our times, the Raspberry Pi. Then we will learn a bit about Linux and the distribution of Linux that is popularly used with the Raspberry Pi family (hereafter, I will use the abbreviation RPi), the Raspberry Pi Operating System. We will learn how to install it on a RPi board. The following is the list of the topics that we will learn in this chapter:

- Single-board computers
- Raspberry Pi
- Linux and distributions
- Raspberry Pi OS setup
- Configuring the RPi board
- Connecting various RPi board models to the Internet

After completing this chapter, we will be comfortable with the installation and the basic usage of the RPi board and the RPi OS.

A. Pajankar, *Practical Linux with Raspberry Pi OS*,
https://doi.org/10.1007/978-1-4842-6510-9_1

Single-Board Computers

Single-board computers (also known as SBCs) have all the components of a fully functioning computer like the processor, GPU, RAM, and I/O on a single printed circuit board. This is in contrast with desktop or laptop computers that have a motherboard which has various slots for RAM, the processor, and the graphics card. Desktop or laptop computers can be upgraded by replacing processors and graphics cards. We can also add more RAM chips in the RAM slots. However, SBCs cannot be upgraded like that. This is one of the major differences between traditional desktops/laptops that are totally modular and SBCs. The key benefit of the lack of modularity of SBCs is that the size of an entire computer is very small. Most of the SBCs are a little bigger than a regular credit/debit card, and they are very compact.

SBCs are used as technology demonstrators (prototypes), educational computers, and embedded systems. There is a recent surge in the popularity of SBCs due to advances in the fabrication process and manufacturing technologies. We are living in an era where a new SBC or a new version of an existing one is announced almost on a monthly basis. The market is full of various SBCs and SBC families. A few prominent SBC families are Raspberry Pi, Banana Pro, BeagleBoards, and Orange Pi. Raspberry Pi is the most popular family of single-board computers available in the market, and it is one of the best-selling computers in the world. In the next section, we will have an overview of the Raspberry Pi family of computers.

Raspberry Pi

Raspberry Pi is a family of SBCs developed by the Raspberry Pi Foundation (www.raspberrypi.org/). It consists of many board models, and all the current models under production are listed on the foundation's products

page (www.raspberrypi.org/products/). Throughout the book, I will be using a Raspberry Pi 4 Model B (the latest board model in the family) with 4 GB RAM.

Table 1-1 lists the specifications of the Raspberry Pi 4 Model B.

Table 1-1. *Technical Specifications of Raspberry Pi 4 B*

Component	Specification
Processor	Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) 64-bit SoC at 1.5 GHz
RAM	LPDDR4-3200 SDRAM (2 GB or 4 GB or 8 GB)
Networking	2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE Gigabit Ethernet
USB	2 USB 3.0 ports, 2 USB 2.0 ports
General-Purpose Input/Output	Raspberry Pi standard 40-pin GPIO header
Display	2 micro-HDMI ports (up to 4kp60 supported) Two-lane MIPI Display Serial Interface port (www.mipi.org/specifications/dsi)
Camera connector	Two-lane MIPI Camera Serial Interface port
Audio	Four-pole stereo audio and composite video port
Secondary storage	MicroSD card slot for OS and data storage
Power	5 V DC via USB-C connector or 5 V DC via GPIO header (minimum 3A)

There are many models of the boards in this family that are currently under production, and if you visit the hobby electronics store near your home, you may find older out-of-production board models at a bargain price. To keep it brief, I will be discussing the technicalities and specifications of the other models only when needed. We can purchase Raspberry Pi boards at authorized retailers (the list can be found in the products page) or at popular ecommerce websites such as Amazon.

Figure 1-1 shows a RPi 4 B board.

Figure 1-1. *Photograph of a RPi 4 B board*

Figure 1-2 shows schematics of the components of RPi 4 B.

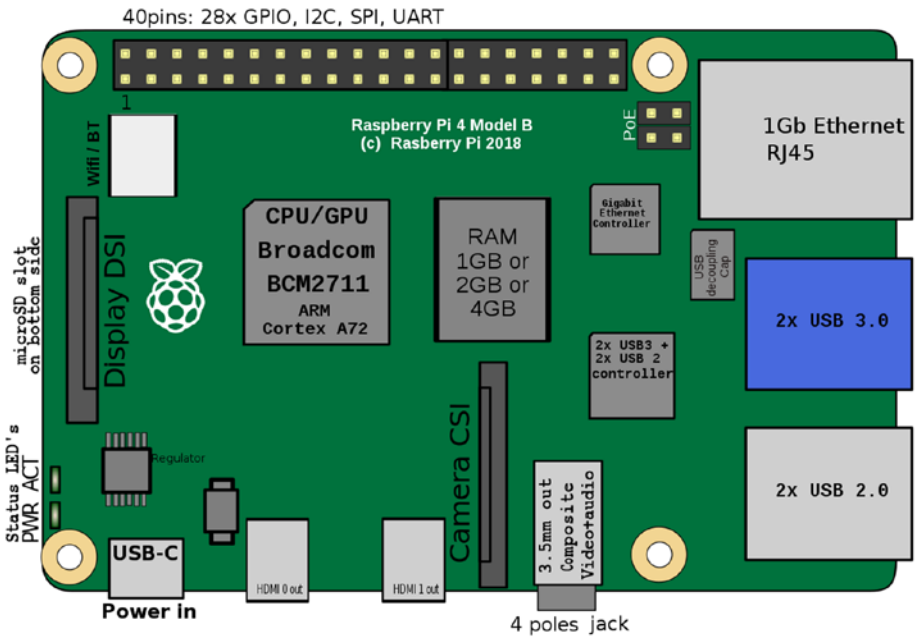

Figure 1-2. *Components on a RPi 4 B board*

Throughout this book, I will be explaining all the demonstrations using a 4 GB RAM model of this board.

Linux and Distributions

Linux is a family of open source Unix-like free operating systems. It is based on the Linux kernel, a free and open source operating system kernel by Linus Torvalds. The Linux OS is packaged in a Linux distribution. A Linux distribution includes the Linux kernel developed and supporting system software, libraries, and APIs for programmers. Many of these components are part of the GNU Project (www.gnu.org/home.en.html), and that is why many people refer to Linux as GNU/Linux. As Linux is free and open source, anyone can create a custom distribution of Linux.

The following URLs have more information about GNU and Linux projects:

www.linux.org/
www.gnu.org/
www.fsf.org/

The top 25 distributions of Linux can be found here:

www.linux.org/pages/download/

Raspberry Pi OS

The Raspberry Pi Operating System is a derivative of a popular Linux distribution known as Debian. It is officially provided by the Raspberry Pi Foundation, and it is the most recommended operating system for the RPi family of SBCs. It is fully optimized for the RPi board models, and all the board models are supported by it. Formerly, it was known as the Raspbian OS, and it was created by Peter Green and Mike Thompson. In this chapter, we will learn in detail how to install the RPi OS on a microSD card and how to boot up a Pi board with that microSD card. The DistroWatch page about the OS can be found here: https://distrowatch.com/table.php?distribution=raspios.

Raspberry Pi OS Setup

As we have seen already, the Raspberry Pi OS is the most preferred OS for the Pi boards. In this section, we will learn how to set up the RPi OS on RPi boards. Though I will be using a RPi 4 B 4 GB model for all the demonstrations, for the convenience of the readers, we will discuss the setup process for all the board models ever produced by the foundation except the compute modules. Let us see all the components needed for the setup one by one:

1) We need a RPi board of any model.

2) We need an appropriate power supply. For Raspberry Pi 4 B, we need a USB-C power supply. Figure 1-3 is an image of a USB-C male pin.

Figure 1-3. *USB-C male header*

The Raspberry Pi Foundation has an official 15.3 W power supply for RPi 4 B. We can find more information at `www.raspberrypi.org/products/type-c-power-supply/`.

All other models of RPi boards need to be supplied by a micro-USB power supply. Figure 1-4 is an image of a micro-USB male pin.

Figure 1-4. *Micro-USB male pin*

The Raspberry Pi Foundation has an official universal power supply. We can find more information at www.raspberrypi.org/products/raspberry-pi-universal-power-supply/.

3) We also need a pair of a USB mouse and a USB keyboard. A USB keyboard and mouse combo that uses a sing USB port is preferred. It is available in the form of a keyboard with a built-in mousepad as shown in Figure 1-5.

Figure 1-5. *A USB keyboard with a built-in mousepad*

The board models RPi Zero and RPi Zero W have only a single micro-USB port, so this is mandatory for such models if we want to use them with a keyboard and a mouse. Also, for RPi Zero and RPi Zero W, we need a USB to micro-USB OTG converter as shown in Figure 1-6.

Figure 1-6. *A USB OTG converter*

4) The RPi board models use a microSD card to store OS and data. RPi 1 Model A and RPi 1 Model B use a SD card, and the rest of the models use a microSD card. We can get more information about the SD cards and compatibility at www.raspberrypi.org/documentation/installation/sd-cards.md and https://elinux.org/RPi_SD_cards. I recommend to purchase a class 10 card of 16 GB size. Also purchase a microSD to SD card converter if you are using RPi 1 Model A and RPi 1 Model B. Figure 1-7 is an image of a microSD card with a microSD to SD card converter.

Figure 1-7. *A microSD card and a microSD to SD card converter*

5) We need a HDMI or a VGA monitor for display.

6) All the models of RPi boards except the models RPi 4 B, RPi Zero, and RPi Zero W have HDMI output and can be directly connected to a HDMI monitor with a HDMI male-to-male cable. A HDMI male pin is shown in Figure 1-8.

Figure 1-8. *A HDMI male head*

RPi 4 B has micro-HDMI output. So we need a micro-HDMI to HDMI converter. RPi Zero and RPi Zero W have got mini-HDMI output. So, for them, we need a mini-HDMI to HDMI converter. Figure 1-9 shows us the HDMI, mini-HDMI, and micro-HDMI male pins, respectively.

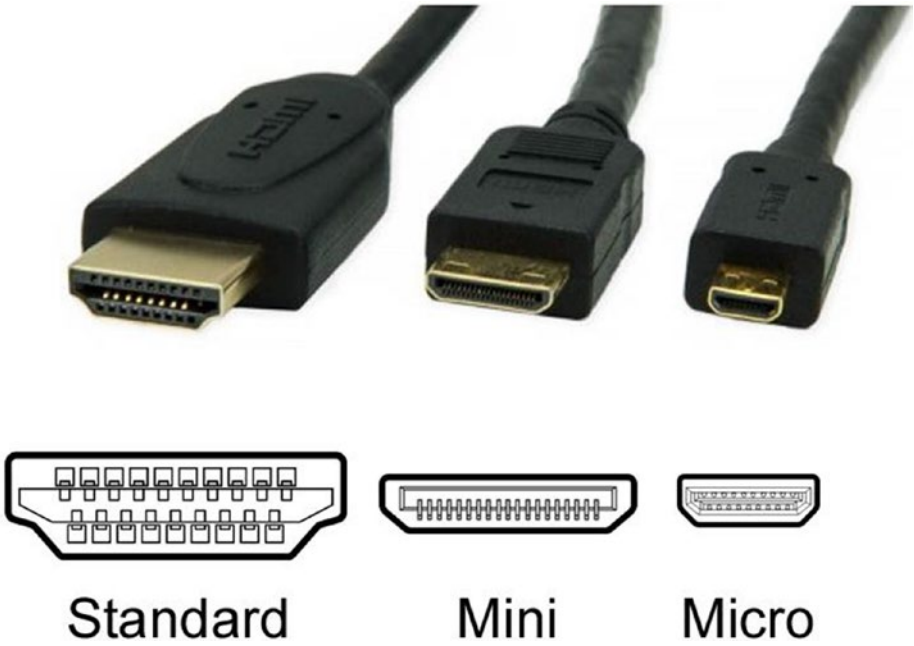

Figure 1-9. *Various types of HDMI male pins*

Figure 1-10 is an image of a HDMI to VGA converter, a mini-HDMI to HDMI converter, and a micro-HDMI to HDMI converter.

Figure 1-10. *Various types of HDMI converters*

If we are using a VGA monitor, then we need to use the HDMI to VGA converter shown in the preceding image.

7) We need a SD card reader. Many laptops have a built-in SD card reader. If your laptop does not have one, you need a separate card reader. Figure 1-11 is a representational image of a SD card reader.

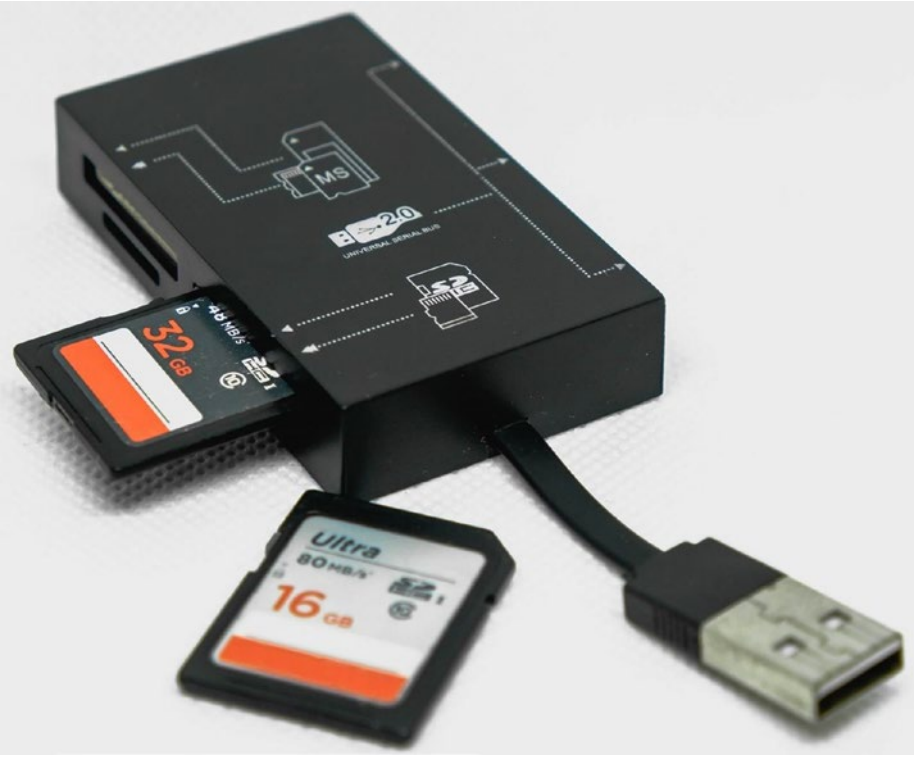

Figure 1-11. *SD card reader*

8) Finally, we need a computer with the Windows, Linux, or macOS operating system.

Preparing the SD Card Manually

Preparing the SD card manually is recommended as it gives us the full control, and we can change the settings before the very first booting of the board. For that, the Raspberry Pi Foundation provides us a utility known as the **Raspberry Pi Imager**. It can be downloaded from www.raspberrypi.org/downloads/. It is available for Windows, Linux, and macOS. Download it and install it on your OS. Once you open it, it shows the window in Figure 1-12.

Figure 1-12. *Raspberry Pi Imager window*

As we can see in Figure 1-12, we have an option for choosing a SD card. Insert the microSD card you have into the card reader device or the card reader slot of your laptop. The Imager will detect it. Then click the **CHOOSE OS** button. It will open the window in Figure 1-13.

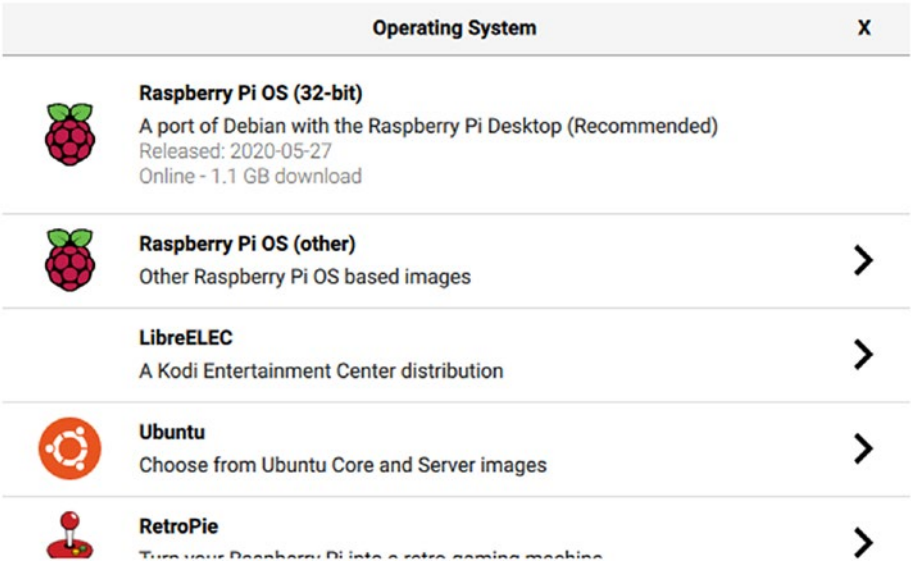

Figure 1-13. *Option for choosing the OS*

We can see the options for various operating systems. We need to choose the second option, **Raspberry Pi OS (other)**. Then it shows the options displayed in Figure 1-14.

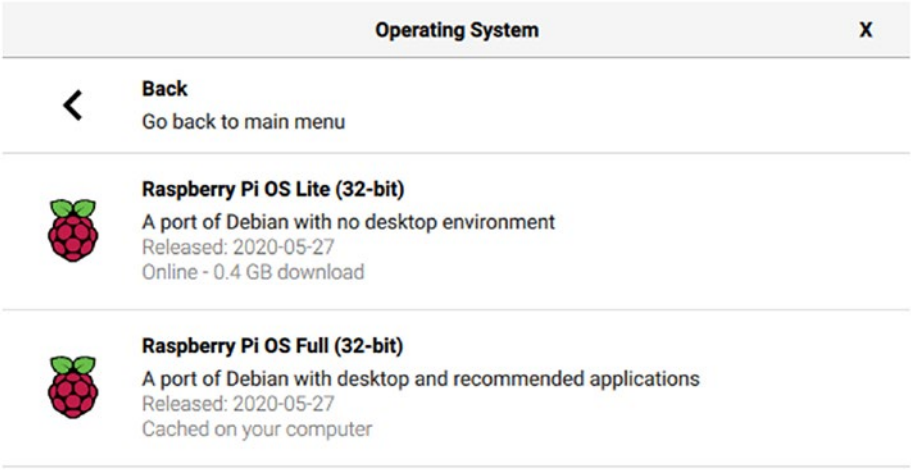

Figure 1-14. *More choices under Raspberry Pi OS (other)*

Choose **Raspberry Pi OS Full (32-bit)**. Then click the button labeled as **WRITE**. It will start writing the OS to the microSD card as shown in Figure 1-15.

Figure 1-15. *Writing the OS to the microSD card*

Once the OS is written to the microSD card, safely remove the card from the SD card reader and reinsert it. In the Windows OS, it will show as a drive with the label **boot**. In this drive, there is a file labeled as **config.txt**. This file stores all the options related to booting, and it acts in the same way as the BIOS (Basic Input/Output System) to initialize the booting. In case we are using a HDMI monitor, we do not have to modify the settings. But if we are using a VGA monitor, we need to make changes to a few lines as follows:

- Change `#disable_overscan=1` to `disable_overscan=1`.
- Change `#hdmi_force_hotplug=1` to `hdmi_force_hotplug=1`.
- Change `#hdmi_group=1` to `hdmi_group=2`.

- Change `#hdmi_mode=1` to `hdmi_mode=16`.
- Change `#hdmi_drive=2` to `hdmi_drive=2`.
- Change `#config_hdmi_boost=4` to `config_hdmi_boost=4`.
- Save the file.

By default, the commented options (which have the symbol # at the beginning) are disabled. We must enable these options by uncommenting their respective lines by removing the symbol # at the beginning of these commented lines.

Booting Up the Pi Board for the First Time

Let us boot up our RPi board for the first time with the microSD card we prepared. The following are the steps:

- Insert the microSD card into the microSD card slot of the RPi board. RPi 1 Model A and RPi 1 Model B have slots for a SD card. So, for these board models, we must use a microSD to SD card converter. Insert the microSD card into the microSD to SD card converter and then insert the converter into the RPi 1 Model A and RPi 1 Model B SD card slot.
- Connect the Pi to the HDMI monitor. As discussed earlier, in case you a VGA have monitor, connect it using the HDMI to VGA converter.
- Connect the USB mouse and USB keyboard. It is recommended to have a single keyboard with a mousepad. For RPi Zero or RPi Zero W, you need to first connect it to a USB OTG cable and then connect the USB OTG cable to the RPi Zero or RPi Zero W board.

- Connect the RPi board to an appropriate power supply (we have discussed this earlier). Connect the monitor to a power source too. Make sure that the power to the RPi board and the monitor is switched off at this point.
- Check all the connections once and then switch on the power supply of the RPi and the monitor.

At this stage, our RPi board will start booting up. We will see a green light blinking on the Pi board. It means that it is booting up. Well, congratulations on our very first success!

Note If the HDMI monitor is showing the message **No Signal** and not showing any visual output, then power down the RPi board and change the line `#hdmi_force_hotplug=1` to `hdmi_force_hotplug=1` in the file `/boot/config.txt` on the microSD card. Boot up the RPi again with this changed setting, and the HDMI monitor will definitely show the output.

Configuring the RPi Board

Once the RPi boots for the first time, it will show a configuration wizard window as shown in Figure 1-16.

Figure 1-16. *Welcome window*

Click the button Next, and the window in Figure 1-17 will appear.

Figure 1-17. *Country and Language*

In the preceding image, set the **Country** and the **Language**; it will automatically select the **Timezone** according to the **Country** selected. We can change that too if we wish. Click the Next button, and the window in Figure 1-18 will appear.

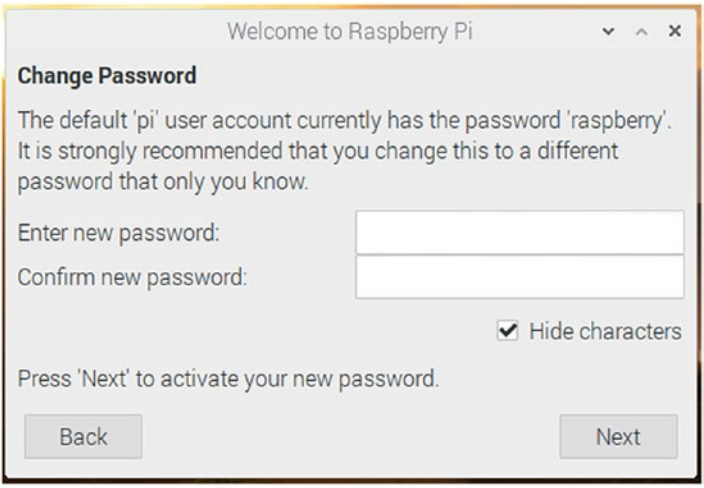

Figure 1-18. *Change the password*

Here, we can change the default password of the user **pi** if we want. If we leave it blank, then it will retain the default password. Click the **Next** button, and it will show the window in Figure 1-19.

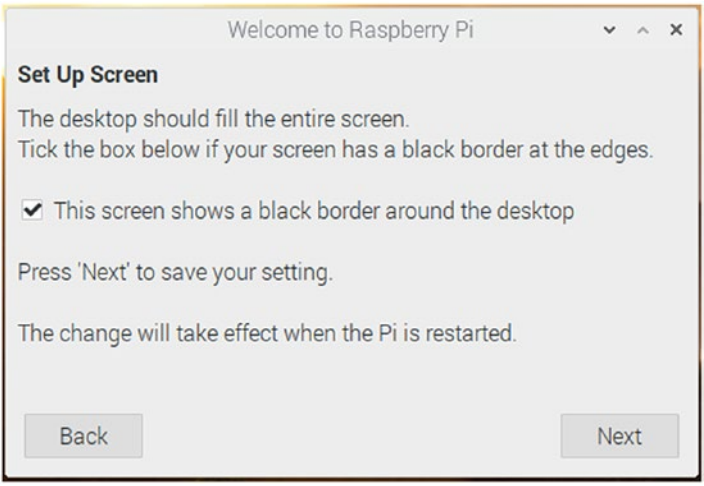

Figure 1-19. *Set Up Screen window*

Check the checkbox if black borders are visible at the edges of the desktop view. The Raspberry Pi OS will correct it on the next boot. The window in Figure 1-20 will appear after we click the **Next** button only if the RPi board model has WiFi.

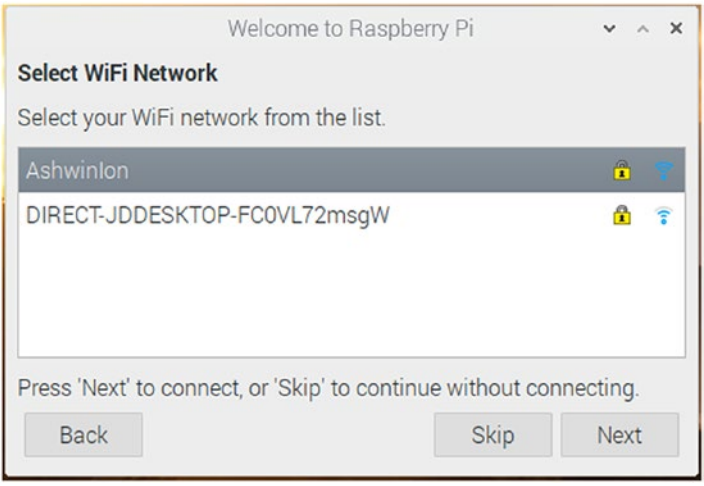

Figure 1-20. *WiFi network selection window*

Choose the WiFi network for which you know the credentials and click the **Next** button, and the window in Figure 1-21 will appear.

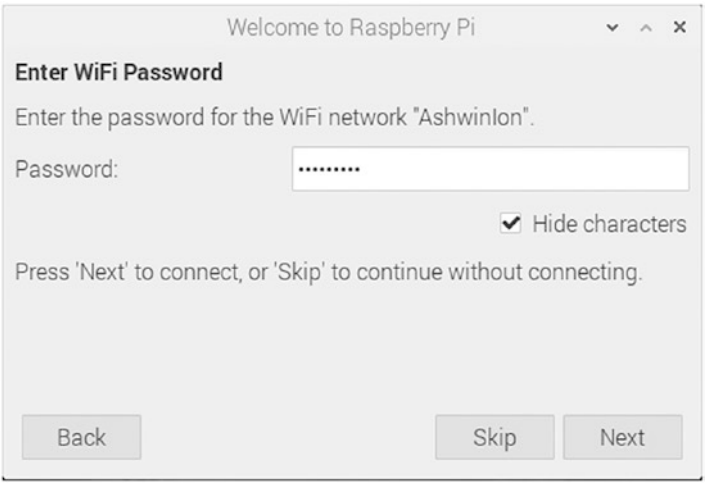

Figure 1-21. *WiFi network password*

Enter the password of the selected WiFi network and click the **Next** button. It will show Update Software window as shown in Figure 1-22.

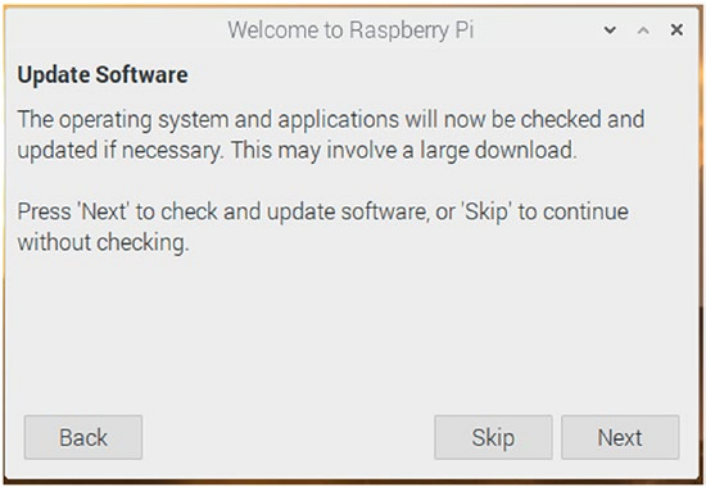

Figure 1-22. *Update Software window*

If we click the **Next** button, then it will update the RPi OS. We will learn how to do it manually in the next chapter. For now, we will skip this by clicking the **Skip** button. It will then show the window in Figure 1-23.

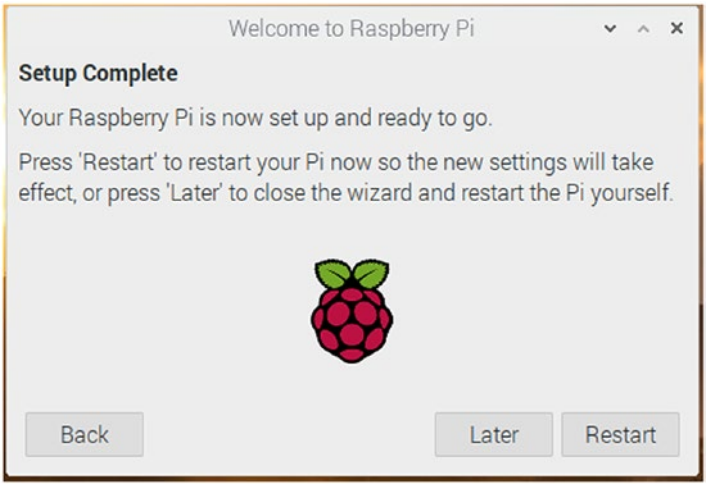

Figure 1-23. *Setup Complete*

This means that the configuration is successful. We need to configure a few more settings manually before a reboot, so click the **Later** button.

In the top-left corner of the desktop, we see a Raspberry icon. It is the menu for the Raspberry Pi OS, and it functions in the same way as the Windows logo in the Microsoft Windows OS. Click that Raspberry logo and navigate to **Preferences ➤ Raspberry Pi Configuration** as shown in Figure 1-24.

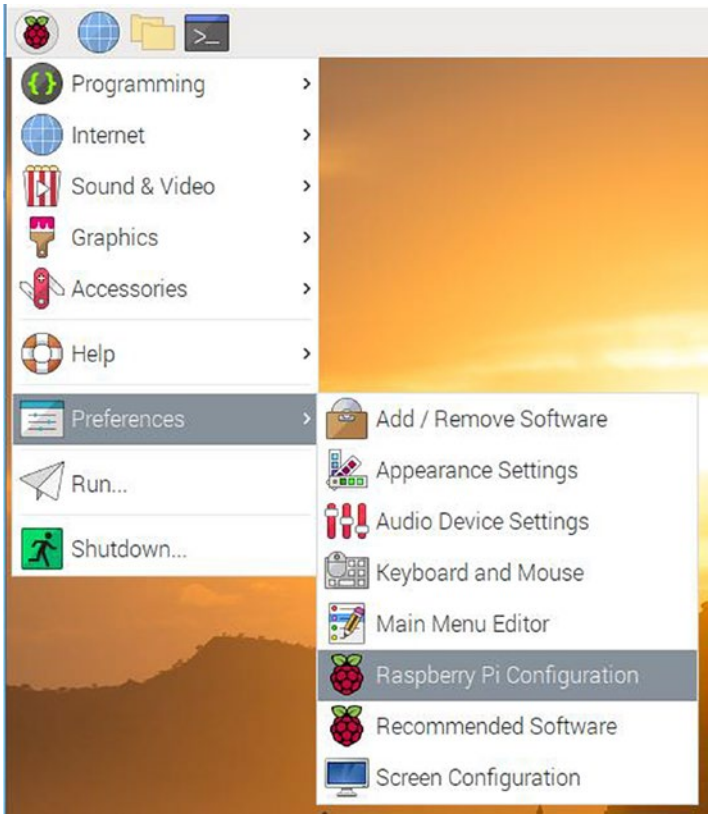

Figure 1-24. *Raspberry Pi Configuration in the RPi OS menu*

Clicking the option shown in the preceding image will open the configuration window shown in Figure 1-25.

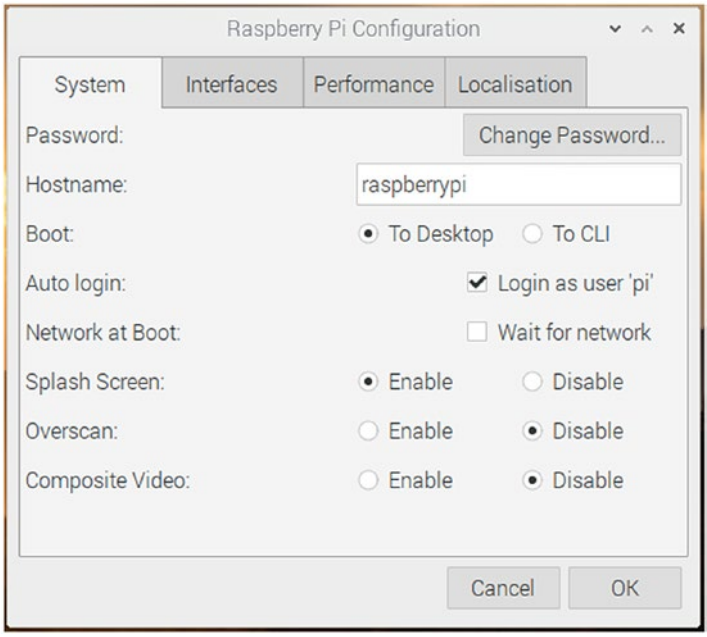

Figure 1-25. *Raspberry Pi Configuration window*

Click the tab Interfaces and it will show options for interfaces as shown in Figure 1-26.

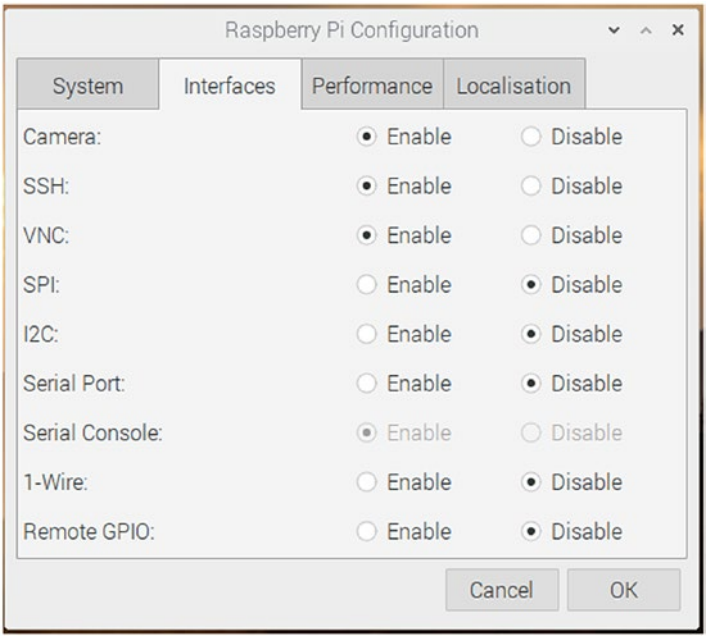

Figure 1-26. *Raspberry Pi Interfaces window*

In this window, click the **Enable** radio buttons for the options **Camera**, **SSH**, and **VNC**. Then click the **Performance** tab and it will show performance options as shown in Figure 1-27.

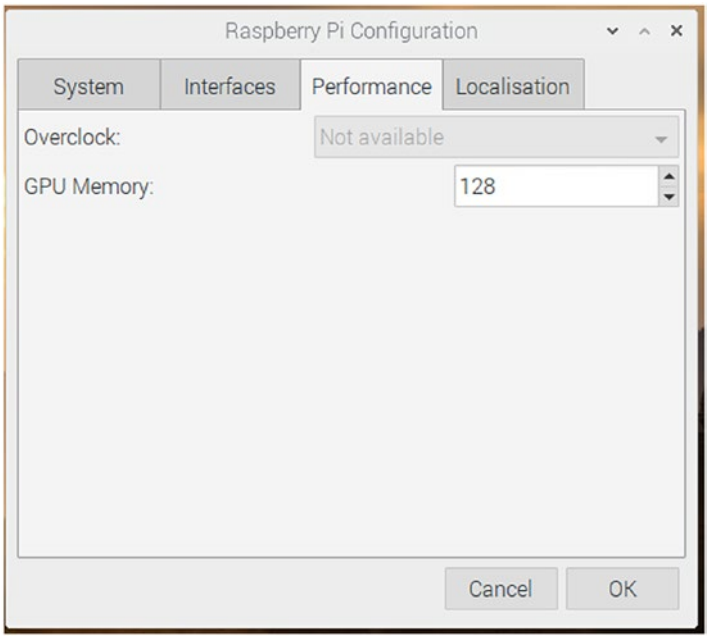

Figure 1-27. *Raspberry Pi Performance*

The option to overclock through this utility is disabled for many board models. Here, we can set the GPU memory. I recommend setting it to 128 MB. This much amount of RAM is used by GPU as video memory (RPi does not have a dedicated GPU memory). Finally, click the **Localisation** tab and it will show Localization options as shown in Figure 1-28.

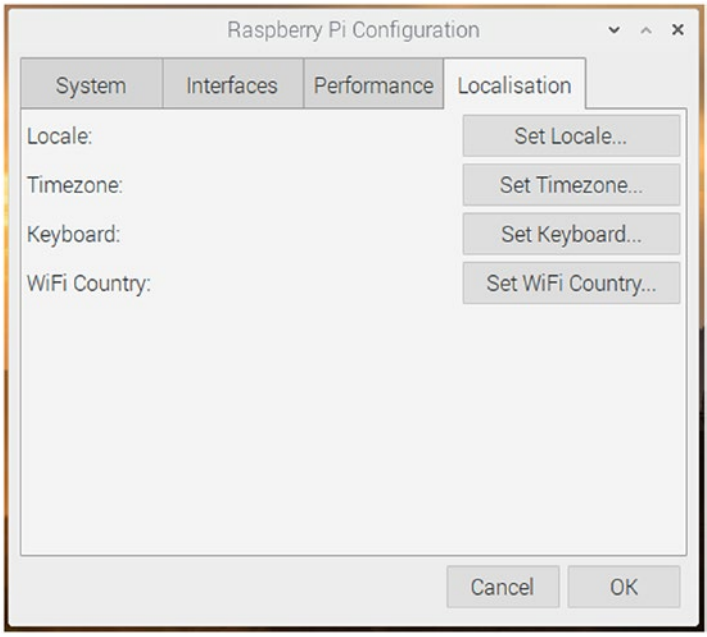

Figure 1-28. *Raspberry Pi Localisation*

Here, we can set the options as per our localization requirements.

Once all these settings are changed as per our choice, we can reboot the RPi board by clicking the last button labeled as Shutdown in the RPi OS. It opens the window in Figure 1-29.

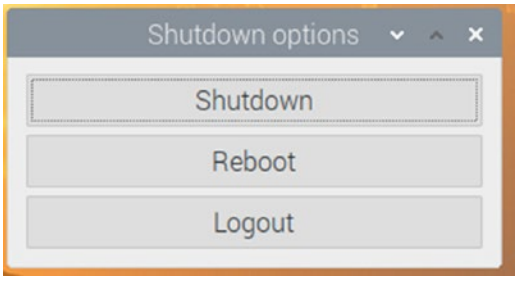

Figure 1-29. *Raspberry Pi Shutdown options*

Just click the **Reboot** button, and the RPi will reboot. All our changes will take effect after the reboot is completed. If we have not changed the default password for the user **pi**, then at startup, the message in Figure 1-30 will be shown.

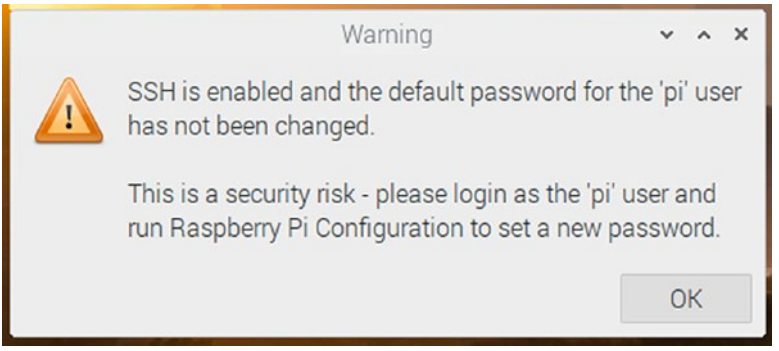

Figure 1-30. *Raspberry Pi warning message after booting to the desktop*

Connecting Various RPi Board Models to the Internet

We can directly plug in the Ethernet cable to the RJ45 Ethernet port on the RPi boards that have it. It will automatically detect the connection and connect to the Internet. Just make sure that DHCP (Dynamic Host Configuration Protocol) is enabled at the WiFi router or the managed switch or the Internet gateway. The models RPi 1 A, RPi 1 A+, RPi Zero, RPi Zero W, and RPi 3 A+ do not have an Ethernet port. However, RPi Zero W and RPi 3 A+ have built-in WiFi for connecting to WANs. We can use a simple USB WiFi dongle for the rest of the models. Figure 1-31 is an image of a USB WiFi dongle.

Figure 1-31. *A USB WiFi dongle*

Plug in this USB WiFi adapter to one of the USB ports. If the USB ports are not enough, then use a powered USB hub. For Raspberry Pi Zero, we need to plug in this dongle to a USB OTG cable and then plug that into the micro-USB port of RPi Zero.

After plugging in the USB WiFi adapter, we need to open the **lxterminal** utility. It is the terminal command-line utility of the RPi OS. We can find it as a small black icon in the RPi OS's taskbar, and we can also find it in **Accessories** in the RPi OS menu. Another way to invoke it is to press Ctrl+F2. The **Run** window in Figure 1-32 will appear. Here, you can type in **lxterminal** and then press the Enter key on the keyboard or click the **OK** button.

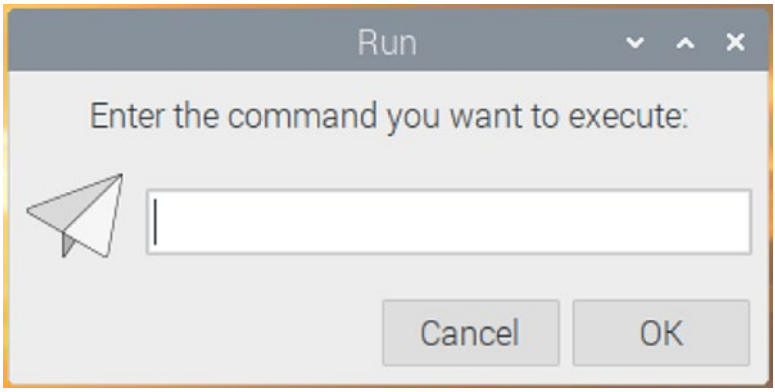

Figure 1-32. *A Run window*

The lxterminal is the terminal emulator for the Raspberry Pi OS, and Figure 1-33 is a screenshot of an instance of it.

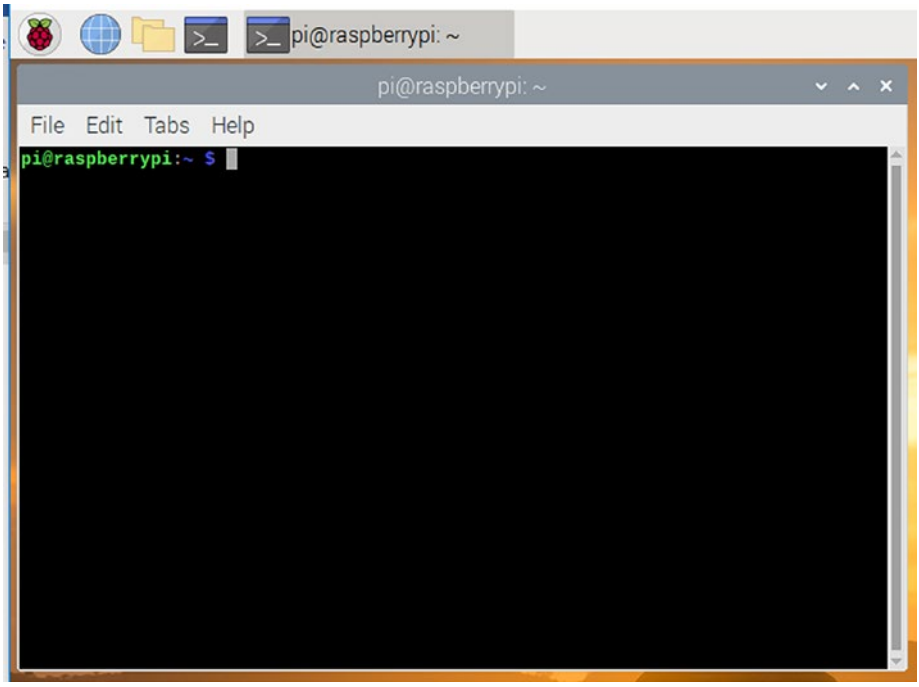

Figure 1-33. *A screenshot of the lxterminal*

Here, we can type in the Linux commands; and after typing in, we must press the Enter key to execute the current command. Let us manually configure the networking using this. This will also give you a decent practice to work with the lxterminal which we will be primarily using throughout the book.

All the network-related information is stored in a file at `/etc/network/interfaces`. Do not bother yourself too much at this stage. We will learn all these things in detail from the next chapter onward. To connect to WiFi after plugging in the USB WiFi dongle, we need to add a few entries to this file I mentioned. First, take the backup of the original file by executing the following command in the lxterminal:

```
mv /etc/network/interfaces /etc/network/interfaces.bkp
```

Then we can create the network interfaces file from scratch by running the following command:

```
sudo nano /etc/network/interfaces
```

The preceding command will open the network interfaces file with a plaintext editor known as the nano editor. It is a simple **WYSIWYG** (**What You See Is What You Get**) plaintext editor. Enter the following lines there:

```
source-directory /etc/network/interfaces.d
auto lo
iface lo inet loopback
auto wlan0
allow-hotplug wlan0
iface wlan0 inet dhcp
wpa-ssid "AshwinIon"
wpa-psk "internet1"
```

After entering the lines, press **Ctrl+X** and then enter **Y**. In the preceding settings, substitute **AshwinIon** with your own SSID and **internet1** with the password for the same WiFi network. Then run the following command on the command prompt:

```
sudo service networking restart
```

It will restart the networking service and will connect to the WiFi. In any case (Ethernet or WiFi), the RPi is assigned with a unique IP address. We can find it out by running the Linux networking command `ifconfig` at the **lxterminal**. The output of the command will have the IPV4 and the MAC addresses of the RPi board.

The other way of knowing the IP address of the RPi is by checking the active client list in the WiFi router or the managed switch to which the RPi board is connected. Figure 1-34 is a screenshot of my WiFi router's active client list where we can see an entry for the RPi connected to it.

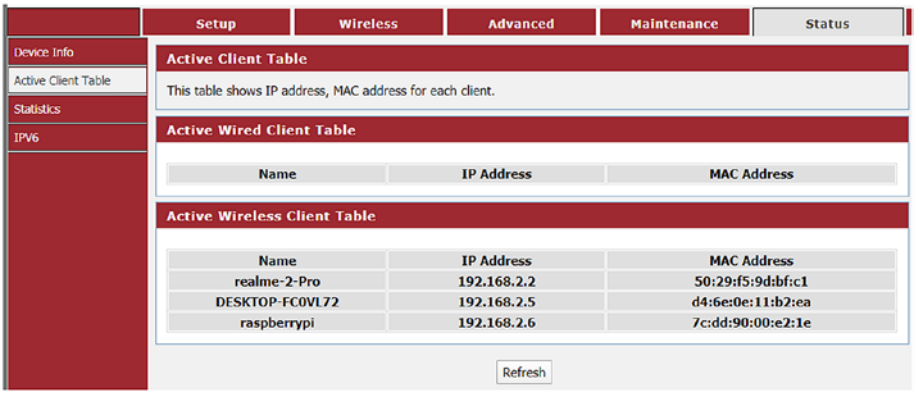

Figure 1-34. *A screenshot of the active client list*

The last entry corresponds to the RPi board connected to it.

Summary

In this chapter, we got started with the basics of Linux and the Raspberry Pi OS. Then we installed the Raspberry Pi OS on a microSD card and learned how to boot up various models of Raspberry Pi. We have also had a bit of hands-on with the terminal emulator **lxterminal,** and we will explore this in detail in the next chapter.

In the next chapter, we will learn the basics of the Linux filesystem and GUI (Graphical User Interface). We will learn what an OS shell is and how to communicate with it using the terminal emulator. We will also learn how to update the RPi OS with commands and how to remotely connect to it.