

Raspberry Pi 5 Essentials

Program, build, and master over 60 projects with Python



Dogan Ibrahim

Raspberry Pi 5 Essentials

Program, build, and master over 60 projects with Python



Dogan Ibrahim

-
- This is an Elektor Publication. Elektor is the media brand of

Elektor International Media B.V.

PO Box 11, NL-6114-ZG Susteren, The Netherlands

Phone: +31 46 4389444

- All rights reserved. No part of this book may be reproduced in any material form, including photocopying, or storing in any medium by electronic means and whether or not transiently or incidentally to some other use of this publication, without the written permission of the copyright holder except in accordance with the provisions of the Copyright Designs and Patents Act 1988 or under the terms of a licence issued by the Copyright Licensing Agency Ltd., 90 Tottenham Court Road, London, England W1P 9HE. Applications for the copyright holder's permission to reproduce any part of the publication should be addressed to the publishers.

● Declaration

The author, editor, and publisher have used their best efforts in ensuring the correctness of the information contained in this book. They do not assume, and hereby disclaim, any liability to any party for any loss or damage caused by errors or omissions in this book, whether such errors or omissions result from negligence, accident or any other cause. All the programs given in the book are Copyright of the Author and Elektor International Media. These programs may only be used for educational purposes. Written permission from the Author or Elektor must be obtained before any of these programs can be used for commercial purposes.

- British Library Cataloguing in Publication Data

A catalogue record for this book is available from the British Library

- **ISBN 978-3-89576-586-5** Print
ISBN 978-3-89576-587-2 eBook

- © Copyright 2023: Elektor International Media B.V.
Editor: Clemens Valens
Prepress Production: D-Vision, Julian van den Berg
Print: Ipskamp Printing, Enschede (NL)

Elektor is the world's leading source of essential technical information and electronics products for pro engineers, electronics designers, and the companies seeking to engage them. Each day, our international team develops and delivers high-quality content - via a variety of media channels (including magazines, video, digital media, and social media) in several languages - relating to electronics design and DIY electronics. www.elektormagazine.com

Contents

Preface	11
Chapter 1 • The Raspberry Pi 5	13
1.1 Overview	13
1.2 The Raspberry Pi 5	13
Chapter 2 • Installing the Raspberry Pi 5 Operating System	17
2.1 Overview	17
2.2 Using a pre-installed SD card	17
2.3 Larger font in Console mode	18
2.4 Accessing your Raspberry Pi 5 Console from your PC – the Putty program	20
2.4.1 Configuring Putty	22
2.5 Accessing the Desktop GUI from your PC	23
2.6 Assigning a static IP address to your Raspberry Pi 5	24
2.7 Enabling Bluetooth.	26
2.8 Connecting the Raspberry Pi 5 to a wired network.	26
2.8.1 Unable to connect to a wired network	27
2.9 Installing the Raspberry Pi 5 Bookworm operating system on a blank microSD card	28
Chapter 3 • Using The Console Commands	31
3.1 Overview	31
3.2 The Command Prompt	31
3.3 Useful Console commands	31
3.3.1 System and user information	31
3.3.2 The Raspberry Pi 5 directory structure.	33
3.3.3 Resource monitoring on the Raspberry Pi 5	44
3.3.4 Shutting Down	46
3.3.5 Networking	47
3.3.6 System information and other useful commands	48
Chapter 4 • Desktop GUI – Desktop Applications	50
4.1 Overview	50
4.2 Desktop GUI Applications	50
4.2.1 Applications Menu.	51

4.2.2 Web browser	53
4.2.3 File manager	53
4.2.4 Terminal	54
4.2.5 Manage Bluetooth devices	54
4.2.6 Wi-Fi	54
4.2.7 Volume control	55
4.2.8 Date and time	55
Chapter 5 • Using a Text Editor in Console Mode.	56
5.1 nano text editor	56
5.2 vi text editor	61
Chapter 6 • Creating and Running a Python Program	65
6.1 Overview	65
6.2 Method 1 – Interactively from command prompt in console mode	65
6.3 Method 2 – Create a Python file in console mode.	65
6.4 Method 3 – Create a Python file in Desktop GUI mode	66
6.5 Which method?	68
Chapter 7 • Python Programming and Simple Programs.	69
7.1 Overview	69
7.2 Variable names	69
7.3 Reserved words.	70
7.4 Comments	70
7.5 Line continuation	70
7.6 Blank lines	70
7.7 More than one statement on a line	71
7.8 Indentation.	71
7.9 Python data types	71
7.10 Numbers.	72
7.11 Strings	75
7.11.1 String functions	76
7.11.2 Escape sequences.	77
7.12 Print statement	77
7.13 List variables.	78

7.13.1 List functions	79
7.14 Tuple variables	80
7.15 Dictionary variables	80
7.15.1 Dictionary functions	81
7.16 Keyboard input	81
7.17 Comparison operators	82
7.18 Logical operators	82
7.19 Assignment operators	82
7.20 Control of flow.	82
7.20.1 if, if...else, and elif	82
7.20.2 for statement	84
7.20.3 while statement	85
7.20.4 continue statement	85
7.20.5 break statement	86
7.20.6 pass statement.	86
7.21 Example 1 – 4-Band resistor colour code identifier.	87
7.22 Example 2 – Series or parallel resistors	89
7.23 Example 3 – Resistive potential divider	91
7.24 Trigonometric functions	93
7.25 User-defined functions	93
7.26 Examples	96
7.27 Recursive functions	106
7.28 Exceptions	106
7.29 try/final exceptions	109
7.31 Creating your own modules.	111
Chapter 8 • Raspberry Pi 5 LED Projects	114
8.1 Overview	114
8.2 Raspberry Pi 5 GPIO pin definitions	114
8.3 Project 1 – Flashing an LED	115
8.4 Project 2 – Alternately flashing LEDs	118
8.5 Project 3 – Binary counting with 8 LEDs.	119
8.6 Project 4 – Christmas lights (random flashing 8 LEDs)	124

8.7 Project 5 – Chasing LEDs	126
8.8 Project 6 – Rotating LEDs with push-button switch	128
8.9 Project 7 – Morse Code exerciser with LED or buzzer	132
8.10 Project 8 – Electronic dice.	136
Chapter 9 • Using an I²C LCD.	141
9.1 Overview	141
9.2 The I ² C Bus	141
9.3 I ² C pins of Raspberry Pi 5.	142
9.4 Project 1 – Using an I ² C LCD – Seconds counter	142
9.5 Project 2 – Using an I ² C LCD – Display time	147
9.6 Project 3 – Using an I ² C LCD – Display IP address of Raspberry Pi 5	148
9.7 Project 4 – Voltmeter – Output to the screen	149
9.8 Project 5 – Voltmeter – Output to LCD	154
9.9 Project 6 – Analog temperature sensor thermometer – output to the screen.	156
9.10 Project 7 – Analog temperature sensor thermometer – output on LCD	159
9.11 Project 8 – Reaction timer – output to screen	161
9.12 Project 9 – Reaction timer – output to LCD	163
9.13 Project 10 – Automatic dusk lights.	165
9.14 Project 11 – Ultrasonic distance measurement	167
9.15 Project 12 – Car parking sensors	170
9.16 Project 13 – Fading LED	172
9.17 Project 14 – Melody maker	173
Chapter 10 • Plotting Graphs with Python and Raspberry Pi 5	176
10.1 Overview	176
10.2 The Matplotlib graph plotting library.	176
10.3 Project 1 – RC transient circuit analysis - Charging	189
10.4 Project 2 – RC transient circuit analysis - Discharging	191
10.5 Transient RL circuits.	194
10.6 Project 3 – RCL transient circuit analysis	194
10.7 Project 4 – Temperature, pressure and humidity measurement – Display on the screen	198

10.8 Project 5 – Temperature, pressure and humidity measurement – Plotting the data	202
Chapter 11 • Waveform Generation – Using the Digital-to-Analog Converter (DAC)	204
11.1 Overview	204
11.2 The MCP4921 DAC	204
11.3 Project 1 – Generating a square wave signal with any peak voltage up to +3.3 V .	205
11.4 Project 2 – Generating a sawtooth wave signal	209
11.5 Project 3 – Generating a triangle wave signal	211
11.6 Project 4 – Generating an arbitrary wave signal	213
11.7 Project 5 – Generating a sine wave signal.	215
Chapter 12 • Using the Sense HAT	219
12.1 Overview	219
12.2 The Sense HAT interface.	219
12.3 Programming the Sense HAT.	221
12.4 Project 1 – Displaying text on Sense HAT	221
12.5 Project 2 – Test your math skills - multiplication	224
12.6 Project 3 – Learning the times tables	225
12.7 Project 4 – Display the temperature, humidity, and pressure	226
12.8 Project 5 – ON-OFF temperature controller	228
12.9 Project 6 – Generate two dice numbers	233
12.10 Project 7 – Display the current time	235
12.11 Project 8 – Displaying two-digit integer numbers	236
12.12 Project 9 – Up counter	240
12.13 The inertial measurement sensor.	241
12.13.1 Project 10 – Reading the acceleration	241
12.13.2 Project 11 – Accelerometer-based dice	241
12.13.3 Project 12 – Accelerometer-based LED shapes	243
Chapter 13 • Using a 4×4 Keypad	245
13.1 Overview	245
13.2 Project 1 – Using a 4×4 keypad.	245
Chapter 14 • Communication over Wi-Fi	256

14.1 Overview	256
14.2 UDP and TCP	256
14.2.1 UDP communication	257
14.2.2 TCP communication	257
14.3 Project 1 – Sending a text message to a smartphone using TCP/IP	258
14.4 Project 2 – Two-way communication with the smartphone using TCP/IP	262
14.5 Project 3 – Communicating with a PC using TCP/IP	263
14.6 Project 4 – Controlling an LED connected to Raspberry Pi 5 from a smartphone using TCP/IP	266
14.7 Project 5 – Sending a text message to a smartphone using UDP	268
14.8 Project 6 – Controlling an LED connected to Raspberry Pi 5 from a smartphone using UDP	271
14.9 Communicating with the Raspberry Pi Pico W over Wi-Fi	273
14.9.1 Project 7 – Raspberry Pi 5 and Raspberry Pi Pico W communication – controlling a relay over Wi-Fi	276
14.10 Project 8 – Storing ambient temperature and atmospheric pressure data in the Cloud	280
Chapter 15 • Communication over Bluetooth	288
15.1 Overview	288
15.2 Project 1 – Exchanging text with a smartphone	288
15.3 Project 2 – Bluetooth control of LED from a smartphone	295
15.4 Arduino UNO – Raspberry Pi 5 Bluetooth communication	297
15.4.1 Project 3 – Communicating with an Arduino UNO over Bluetooth	298
15.4.2 Project 4 – Play audio (e.g. music) on Bluetooth speaker via Raspberry Pi 5	303
Chapter 16 • Raspberry Pi 5 Camera Projects	305
16.1 Overview	305
16.2 Installing the Camera	305
16.3 Project 1 – Still camera commands	306
16.3.1 libcamera	306
16.4 Project 2 – Building a time-lapse camera – Who is in my parking place?	309
16.5 Project 3 – Video camera commands	315
16.6 Project 4 – Who is ringing my doorbell?	315
Index	320

Preface

The Raspberry Pi 5 is a credit-card-sized computer from Raspberry Pi that can be used in many applications, such as in audio and video media centers, as a desktop computer, in industrial controllers, robotics, and in many domestic and commercial applications. In addition to the many features found in other Raspberry Pi computers, the Raspberry Pi 5 offers Wi-Fi and Bluetooth 5.0 (with BLE support), which makes it highly desirable in remote and Internet-based control and monitoring applications.

The Raspberry Pi 5 is based on a 64-bit quad-core ARM Cortex-A76 processor running at 2.4 GHz. This implies a performance boost of two to three times compared to the Raspberry Pi 4. Raspberry Pi 5 comes with an enhanced graphic performance, using an 800 MHz VideoCore VII graphics chip. Additionally, the Raspberry Pi 5 features the RP1 southbridge chip made by Raspberry Pi. With the help of this RP1 southbridge, Raspberry Pi 5 delivers higher performance and more functionality for peripheral devices. It should now be possible to carry out many real-time operations such as audio digital signal processing, real-time digital control and monitoring, and many other real-time operations using this tiny powerhouse.

This book is about the Raspberry Pi 5 computer and its use in various control and monitoring applications. The book explains in simple terms and with many tested and working example projects how to configure the Raspberry Pi 5 computer, how to use the latest operating system (Bookworm), and how to write application programs using the popular Python programming language.

The book starts with an introduction to the Raspberry Pi 5 computer and covers the important topics of accessing the computer locally and remotely. Use of the console command language as well as accessing and using the desktop GUI have been described with working examples.

The remaining parts of the book cover many Raspberry-Pi-5-based hardware projects using components and devices such as LEDs, buzzers, LCDs, ultrasonic sensors, temperature sensors, Sense HAT, camera modules, etc. Example projects are given using Wi-Fi and Bluetooth modules to send and receive data from smartphones, from the PC, and sending real-time temperature and atmospheric pressure data to the cloud.

All the projects presented in the book have been tested and are working. Complete circuit diagrams and full program listings are given for each project, with detailed descriptions of the operation of each project. The following subheadings are used in every project whenever necessary:

- Project title
- Project description
- Block diagram
- Circuit diagram
- Program listing
- Suggestions for future work

I hope the readers find the book helpful and enjoy reading it, and use a Raspberry Pi 5 in their next new projects.

Prof Dr. Dogan Ibrahim
London

Chapter 1 • The Raspberry Pi 5

1.1 Overview

The Raspberry Pi 5 is the latest credit card size computer from Raspberry Pi. In this chapter, we will look at the specifications of this new computer and compare it with the Raspberry Pi 4.

1.2 The Raspberry Pi 5

Raspberry Pi 4 was released in June 2019. There has been a long wait for a newer model and finally the Raspberry Pi 5 was launched in October 2023.

The Raspberry Pi 5 is claimed to have two or three times the processing power of The Raspberry Pi 4, which is already a very popular single board computer. The Raspberry Pi 5 is currently available in 4 GB and 8 GB memory capacities, but smaller memory devices may appear later. Although the Raspberry Pi 5 is the same size and shape as the Model 4B, it has a number of interesting new features such as PCIe connector, power button, built-in real-time clock and some others that we will investigate in this chapter.

The Raspberry Pi 5 is based on a 2.4 GHz Cortex-A76 ARM processor with a new south-bridge for handling the peripheral interface. A new VideoCore VII GPU is provided with 800 MHz speed. The dual camera interface is another nice feature of the Raspberry Pi 5. The microSD card interface now supports cards that work at much higher speeds.

Table 1.1 shows a comparison of the Raspberry Pi 4 and 5. Notice that both devices have dual 2 × 4kp60 HDMI display interfaces, although Pi 5 supports HDR output. The 2 × 20 pin GPIO interface is the same in both devices. The Raspberry Pi 5 additionally has two camera interfaces, a PCIe bus connector, a UART interface, an RTC clock power connector, and a fan power connector. Wi-Fi and Bluetooth are supported by both devices. The on-board power switch on Pi 5 is a useful addition and was requested by many users. Pi 5 is powered from 5 V/4 A USB-C type power supply, where Pi 4 is powered from a 3 A power supply. Pi 5 is slightly more expensive than Pi 4.

	Raspberry Pi 4	Raspberry Pi 5
SoC	BCM2711 SoC Cortex-A72 CPU at 1.8 GHz	BCM2712 SoC Cortex-A76 CPU at 2.4 GHz
CPU	4 core	4 core
Instruction set	ARMv8-A	ARMv8-2
Display	500 MHz VideoCore Vi GPU	800 MHz VideoCore VII GPU
L2 Cache	1 MB (shared)	2 MB
L3 Cache	None	2 MB (shared)
RAM	1, 2, 4, 8 GB LPDDR4	4, 8 GB LPDDR4X
SD Card	microSD	microSD (high speed SDR104 compatible)
GPIO	2 × 20 pin	2 × 20 pin

USB ports	2× USB2 2× USB3	2× USB2 2× USB3
Networking	Gigabit Ethernet port	Gigabit Ethernet port
Connectors	2-lane MIPI display port 2-lane MIPI CSI camera port 4-pole stereo audio and composite video port	2× MIPI camera 2× 4-lane MIPI camera/display PCIe 2.0 interface UART port RTC clock power port Fan power port
Wi-Fi/Bluetooth	802.11ac, Bluetooth 5/BLE	802.11ac, Bluetooth 5/BLE
Power button	None	Yes
Power	5 V, 3 A USB-C	5 V, 4 A USB-C
Size	85 × 56 mm	85 × 56 mm

Table 1.1 Comparison of Raspberry Pi 5 and Raspberry Pi 4

There are two micro-HDMI based monitor ports on both devices, with both having the same specifications.

The Ethernet port and USB ports are swapped. As a result of this, the Raspberry Pi 4 case is incompatible with the Pi 5 and a new case is required.

The camera and display connectors on the Raspberry Pi 5 are 15-pin and smaller, instead of the original 22-pin connector used on Pi 4. A ribbon cable with 22-pin on one side and 15-pin on the other side is required to connect an existing Raspberry Pi 4 camera to the Raspberry Pi 5. The Raspberry Pi 5 has two connectors, allowing two cameras or DSI displays (or a mix of either) to be connected. The PCIe connector is for fast external PCIe compatible peripherals, such as SSDs.

The new power button on the Raspberry Pi 5 could be very useful. When the device is On, pressing the button brings the shutdown (logout) menu. A safe shutdown will occur with another press of the power button.

Figure 1.1 shows the front view of the Raspberry Pi 5 with the components labelled for reference.

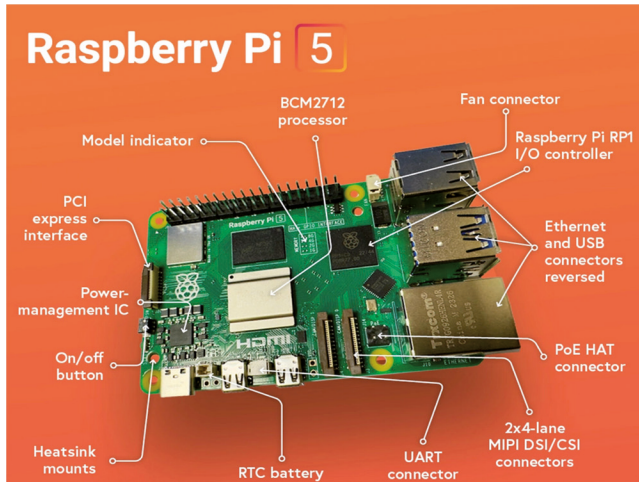


Figure 1.1 Raspberry Pi 5

The Raspberry Pi 5 gets rather hot, and it is recommended to use a cooler to lower the CPU temperature. Although the idle CPU temperature is around 50°C, it can go higher than 85°C under a stress test. An active cooler is available for the Raspberry Pi 5. Holes and power points are provided on the board to install and power the active cooler. Figure 1.2 shows the Raspberry Pi 5 with the active cooler installed. The active cooler cools down the SoC, RAM, and the southbridge chip. When the CPU is idle, the active cooler keeps the CPU temperature at around 40°C. The fan of the cooler operates automatically when the CPU temperature goes just above 50°C.

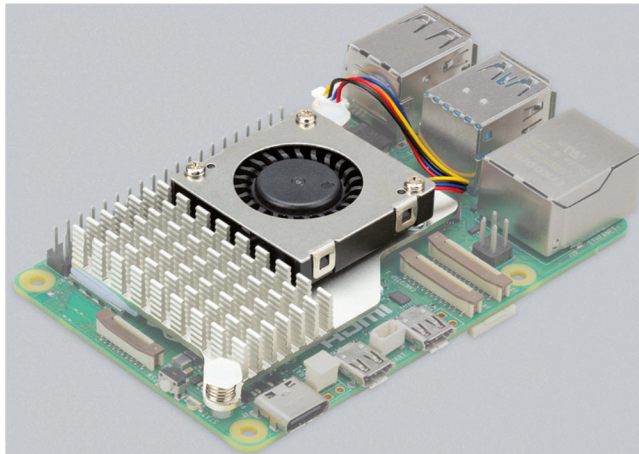


Figure 1.2 The Raspberry Pi 5 with active cooler

The Raspberry Pi 5 operating system (OS) is based upon Debian 12 with the code name **Bookworm**. This OS, released in July 2023, comes with a new Python interpreter (Python 3.11). This means that a Python package cannot be installed using the **pip** commands.

Another major software change is that the RPi.GPIO library (created by Ben Croston) was not available at the time of writing this book. As a result of this, all the GPIO-based Python programs in the book have been developed using the **gpiozero** library. Most third party HATs are based on RPi.GPIO and these will not work until their software is changed by their manufacturers. It is hoped that the manufacturers will change their software by the time Raspberry Pi 5 becomes officially widely available.

Chapter 2 • Installing the Raspberry Pi 5 Operating System

2.1 Overview

The Raspberry Pi 5 operating system **Bookworm** is available either on a pre-installed microSD card, or you can download the operating system image on a blank microSD card. In this chapter, you will learn to install the operating system using both methods.

2.2 Using a pre-installed SD card

The pre-installed Raspberry Pi operating system is available on various sized microSD cards. In this section, the author used the pre-installed 32 GB microSD card supplied by Elektor. Additionally, the author used a 7-inch HDMI compatible monitor, a Raspberry Pi official keyboard, and a mouse. The author's hardware setup between the Raspberry Pi 5 and various devices is shown in Figure 2.1.

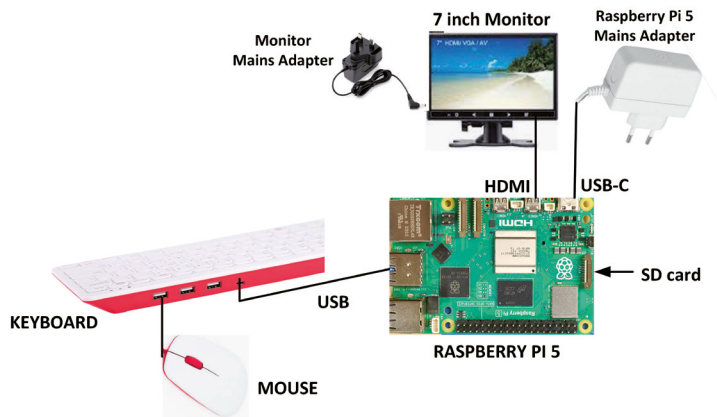


Figure 2.1 The author's hardware setup

The steps are as follows:

- Insert the pre-installed microSD card into your Raspberry Pi 5
- Connect all the devices as in Figure 2.1
- Connect the Raspberry Pi power adapter to the mains supply
- You should see the Raspberry Pi booting the first time and asking you various questions to set up the device, such as the username, password, Wi-Fi network name and password, any updates if necessary, etc. (see Figure 2.2 for some displays on the monitor). In this book, the username is set to **pi**.
- The Raspberry Pi will boot in Desktop mode and will display the default screen. You can press Ctrl+Alt+F1 at any time to change to the Console mode

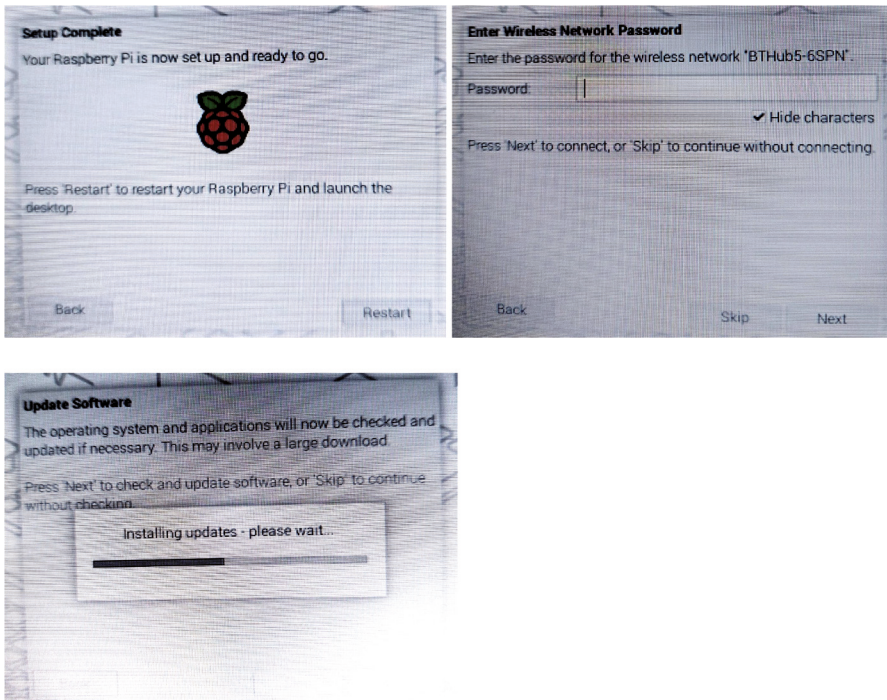


Figure 2.2 Raspberry Pi 5 booting for the first time.

2.3 Larger font in Console mode

It is probably hard to see the characters on a 7-inch monitor in console mode. You can follow the steps below to increase the font size:

- Make sure you are in the Console mode
- Enter the following command:

```
pi@raspberrypi: ~ $ sudo dpkg-reconfigure console-setup
```

- Select **UTF-8** in the **Package Configuration** screen (Figure 2.3)

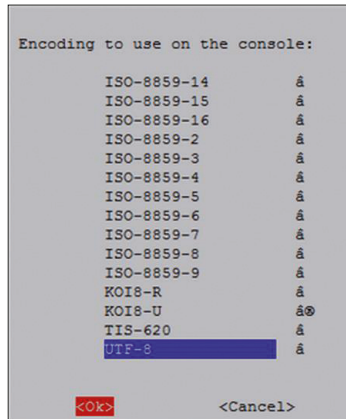


Figure 2.3 Select UTF-8

- Select **Guess optimal character set** (Figure 2.4)

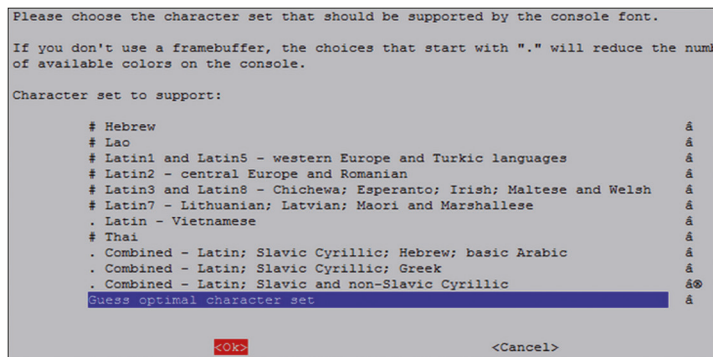


Figure 2.4 Select Guess optimal character set

- Select **Terminus** (Figure 2.5)

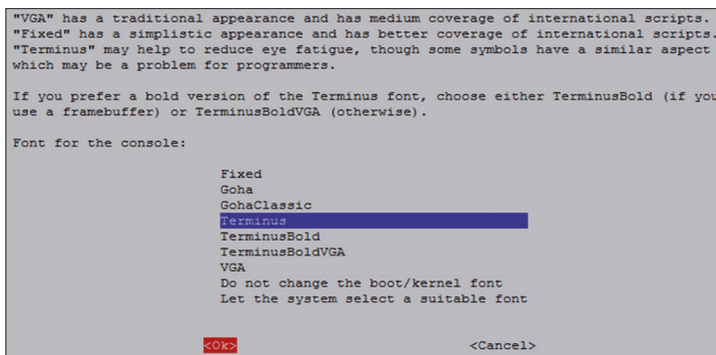


Figure 2.5 Select Terminus

- Select font **16x32** (Figure 2.6)

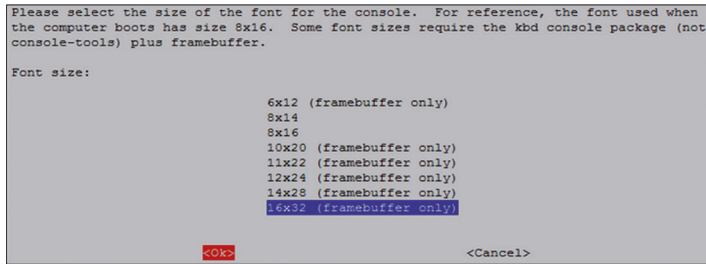


Figure 2.6 Select font 16x32

2.4 Accessing your Raspberry Pi 5 Console from your PC – the Putty program

In many applications, you may want to access your Raspberry Pi 5 from your PC. This requires enabling the SSH on your Raspberry Pi and then using a terminal emulation software on your PC. The steps to enable the SSH are as follows:

- Make sure you are in Console mode
- Type: **sudo raspi-config**
- Move down to **Interface Options**
- Highlight **SSH** and press Enter (Figure 2.7)

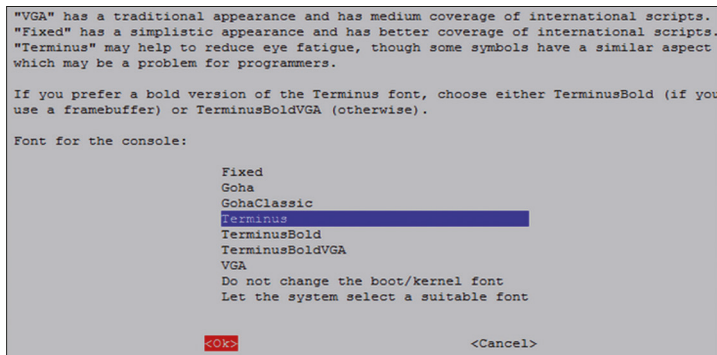


Figure 2.7 Highlight SSH

- Click **Yes** to enable SSH
- Click **OK**
- Move down and click **Finish**

You will now have to install a terminal emulation software on your PC. The one used by the author is the popular Putty. Download Putty from the following website:

<https://www.putty.org>

- Putty is a standalone program and there is no need to install it. Simply double click to run it. You should see the Putty startup screen as in Figure 2.8.

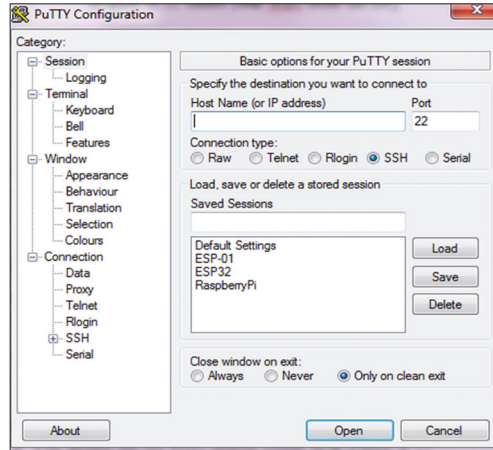


Figure 2.8 Putty startup screen

- Make sure that the Connection type is SSH and enter the IP address of your Raspberry Pi 5. You can obtain the IP address by entering the command **ifconfig** in console mode (Figure 2.9). In this example, the IP address was: **192.168.1.251** (see under **wlan0**;))

```
pi@raspberrypi:~$ ifconfig
eth0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether d8:3a:dd:77:b2:e2 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device interrupt 107

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 105 bytes 9175 (8.9 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 105 bytes 9175 (8.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.251 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 2a00:23c7:868d:7b01:1562:5802:73c0:1fff6 prefixlen 64 scopeid 0x
<global>
    inet6 fe80::966a:a2d8:912:fa6b prefixlen 64 scopeid 0x20<link>
```

Figure 2.9 Command **ifconfig**

- Click **Open** in Putty after entering the IP address and selecting **SSH**
- The first time you run Putty, you may get a security message. Click **Yes** to accept this security alert.
- You will then be prompted to enter the Raspberry Pi 5 username and password. You can now enter all Console-based commands through your PC.

- To change your password, enter the following command:

```
pi@raspberrypi: ~ $ passwd
```

- To restart the Raspberry Pi, enter the following command:

```
pi@raspberrypi: ~ $ sudo reboot
```

- To shut down the Raspberry Pi, enter the following command. Never shutdown by pulling the power cable, as this may result in the corruption or loss of files:

```
pi@raspberrypi: ~ $ sudo shutdown -h now
```

2.4.1 Configuring Putty

By default, the **Putty** screen background is black with white characters. The author prefers a white background with black characters, with the character size set to 12 points bold. You should save your settings so that they are available next time you want to use Putty. The steps to configure Putty with these settings are given below:

- Restart Putty
- Select **SSH** and enter the Raspberry Pi IP address
- Click **Colours** under **Window**
- Set the **Default Foreground** and **Default Bold Foreground** colours to black (Red:0, Green:0, Blue:0)
- Set the **Default Background** and **Default Bold Background** to white (Red:255, Green:255, Blue:255)
- Set the **Cursor Text** and **Cursor Colour** to black (Red:0, Green:0, Blue:0)
- Select **Appearance** under **Window** and click **Change** in **Font settings**. Set the font to **Bold 12**.
- Select **Session** and give a name to the session (e.g. MyZero) and click **Save**.
- Click **Open** to open **Putty** session with the saved configuration
- Next time you restart **Putty**, select the saved session and click **Load** followed by **Open** to start a session with the saved configuration

2.5 Accessing the Desktop GUI from your PC

If you are using your Raspberry Pi 5 with local keyboard, mouse, and display, you can skip this section. If, on the other hand, you want to access your Desktop remotely over the network, you will find that SSH services cannot be used. The easiest and simplest way to access your Desktop remotely from a computer is by using the VNC (Virtual Network Connection) client and server. The VNC server runs on your Pi and the VNC client runs on your computer. It is recommended to use the **tightvncserver** on your Raspberry Pi 5. The steps are:

- Enter the following command:

```
pi$raspberrypi:~ $ sudo apt-get install tightvncserver
```

- Run the **tightvncserver**:

```
pi$raspberrypi:~ $ tightvncserver
```

You will be prompted to create a password for remotely accessing the Raspberry Pi desktop. You can also set up an optional read-only password. The password should be entered every time you want to access the Desktop. Enter a password and remember your password.

- Start the VNC server after reboot by the following command:

```
pi$raspberrypi:~ $ vncserver :1
```

You can optionally specify screen pixel size and colour depth in bits as follows:

```
pi$raspberrypi:~ $ vncserver :1 -geometry 1920x1080 -depth 24
```

- We must now set up a VNC viewer on our laptop (or desktop) PC. There are many VNC clients available, but the recommended one which is compatible with **TightVNC** is **TightVNC** for the PC, which can be downloaded from the following link:

<https://www.tightvnc.com/download.php>

- Download and install the **TightVNC** software for your PC. You will have to choose a password during the installation.
- Start the **TightVNC Viewer** on your PC and enter the Raspberry Pi IP address followed by ':1'. Click **Connect** to connect to your Raspberry Pi (Figure 2.10)

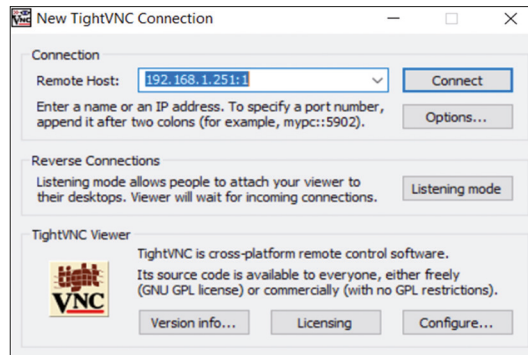


Figure 2.10 Connect to TightVNC Viewer

- Enter the password you have chosen earlier. You should now see the Raspberry Pi 5 Desktop displayed on your PC screen (Figure 2.11)

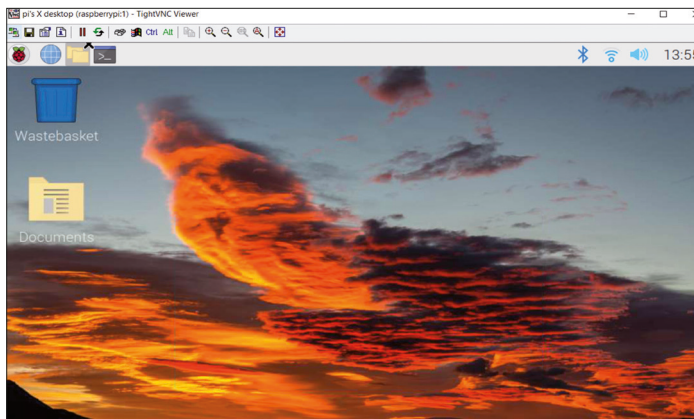


Figure 2.11 Raspberry Pi 5 Desktop

- The VNC server is now running on your Raspberry Pi 5 and you have access to the Desktop GUI.

2.6 Assigning a static IP address to your Raspberry Pi 5

When you try to access your Raspberry Pi 5 remotely over your local network, it is possible that the IP address given by your Wi-Fi router can change from time to time. This is annoying as you have to find out the new IP address allocated to your Raspberry Pi. Without knowledge of the IP address, you cannot log in using SSH or VNC.

In this section, you will learn how to fix your IP address so that it does not change between reboots. The steps are as follows:

- Log in to your Raspberry Pi 5 via Putty

- Check whether DHCP is active on your Raspberry Pi (it should normally be active):

```
pi@raspberrypi:~ $ sudo service dhcpcd status
```

If DHCP is not active, activate it by entering the following commands:

```
pi@raspberrypi:~ $ sudo service dhcpcd start
pi@raspberrypi:~ $ sudo systemctl enable dhcpcd
```

- Find the IP address currently allocated to you by entering the command **ifconfig** or **hostname -I** (Figure 2.12). In this example, the IP address was: 192.168.1.251. We can use this IP address as our fixed address, since no other device on the network is currently using it.

```
pi@raspberrypi:~ $ hostname -I
192.168.1.251 2a00:23c7:868d:7b01:1562:5802:73c0:1ff6
pi@raspberrypi:~ $
```

Figure 2.12 Find the IP address using the command `hostname -I`

- Find the IP address of your router by entering the command **ip r** (Figure 2.13). In this example, the IP address was: 192.168.1.254

```
pi@raspberrypi:~ $ ip r
default via 192.168.1.254 dev wlan0 proto dhcp src 192.168.1.251 metric 600
192.168.1.0/24 dev wlan0 proto kernel scope link src 192.168.1.251 metric 600
pi@raspberrypi:~ $
```

Figure 2.13 Find the IP address of your router.

- Find the IP address of your DNS by entering the following command (Figure 2.14). This is usually the same as your router address:

```
pi@raspberrypi:~ $ grep "nameserver" /etc/resolv.conf
```

```
pi@raspberrypi:~ $ grep "nameserver" /etc/resolv.conf
nameserver 192.168.1.254
nameserver fe80::4e1b:86ff:feb5:ba79%wlan0
pi@raspberrypi:~ $
```

Figure 2.14 Find the DNS address.

- Edit file **/etc/dhcpcd.conf** by entering the command:

```
pi@raspberrypi:~ $ nano /etc/dhcpcd.conf
```

- Add the following lines to the bottom of the file (these will be different for your router). If these lines already exist, remove the comment character '#' at the beginning of the lines and change the lines as follows (you may notice that **eth0** for Ethernet is listed):

```
interface wlan0
static_routers=192.168.1.254
static domain_name_servers=192.168.1.254
static ip_address=192.168.1.251/24
```

- Save the file by entering **CTRL + X** followed by **Y** and reboot your Raspberry Pi
- In this example, the Raspberry Pi should reboot with the static IP address:
192.168.1.251

2.7 Enabling Bluetooth

In this section, you will see how to enable the Bluetooth on your Raspberry Pi 5 so that it can communicate with other Bluetooth devices. The steps are given below:

- Enable the Bluetooth on your other device
- Click on the Bluetooth icon on your Raspberry Pi 5 at the top right-hand side, and select **Make Discoverable**. You should see the Bluetooth icon flashing
- Select 'raspberrypi' in the Bluetooth menu on your other device
- Accept the pairing request on your Raspberry Pi 5
- You should now see the message **Connected Successfully** on your Raspberry Pi 5 and you can exchange files between your other device and the Raspberry Pi computer.

2.8 Connecting the Raspberry Pi 5 to a wired network

You may want to connect your Raspberry Pi 5 to a network through an Ethernet cable. The steps are as follows:

Step 1: Connect a network cable between your Raspberry Pi 5 and your Wi-Fi router.

Step 2: Connect the keyboard, mouse and monitor to your Raspberry Pi and power up as normal

Step 3: Log in to the system by entering your username and password

Step 4: Providing your network hub supports DHCP (nearly all network routers support DHCP), you will be connected automatically to the network and will be assigned a unique IP address within your network. Note that DHCP assigns IP addresses to newly connected devices.

Step 5: Check to find out the IP address assigned to your Raspberry Pi 5 by the network router. Enter the command **ifconfig** as described earlier

2.8.1 Unable to connect to a wired network

If you find out that you are not assigned an IP address by the DHCP server, possible causes are:

- Your network cable is faulty
- The network hub does not support DHCP
- DHCP is not enabled on your Raspberry Pi, i.e. it may have been configured for a fixed IP address

In most cases, it is very unlikely that the network cable is faulty. Also, most network hubs support the DHCP protocol. If you are having problems with the network, it is possible that your Raspberry Pi is not configured to accept DHCP issued addresses. The Raspberry Pi is normally configured to accept DHCP addresses, but it is possible that you have changed the configuration somehow.

To resolve the wired network connectivity problem, follow the steps given below:

Step 1: find out whether your Raspberry Pi is configured for DHCP or fixed IP addresses. Enter the following command:

```
pi@raspberrypi ~$ cat /etc/network/interfaces
```

If your Raspberry Pi is configured to use the DHCP protocol (which is normally the default configuration), the word **dhcp** should appear at the end of the following line:

```
iface eth0 inet dhcp
```

If, on the other hand, your Raspberry Pi is configured to use static addresses, then you should see the word **static** at the end of the following line:

```
iface eth0 inet static
```

Step 2: To use the DHCP protocol, edit file **interfaces** (e.g. using the **nano** text editor) and change the word **static** to **dhcp**. It is recommended to make a backup copy of the file **interfaces** before you change it:

```
pi@raspberrypi ~$ sudo cp /etc/network/interfaces /etc/network/int.bac
```

You should now restart your Raspberry Pi and an IP address will probably be assigned to your device.

Step 3: To use static addressing, make sure that the word **static** appears as shown above. If not, edit file **interfaces** and change **dhcp** to **static**

Step 4: You need to edit and add the required unique IP address, subnet mask and gateway addresses to file **interfaces** as in the following example (this example assumes that

the required fixed IP address is 192.168.1.251, the subnet mask used in the network is 255.255.255.0, and the gateway address is 192.168.1.1):

```
iface eth0 inet static
address 192.168.1.251
netmask 255.255.255.0
gateway 192.168.1.1
```

Save the changes and exit the editor. If you are using the **nano** editor, exit by pressing Ctrl+X, then enter Y to save the changes, and enter the filename to write to as **/etc/network/interfaces**.

Restart your Raspberry Pi 5.

2.9 Installing the Raspberry Pi 5 Bookworm operating system on a blank microSD card

If you have a pre-installed Raspberry Pi operating system Bookworm on a microSD card, then you can start using it as described earlier in this chapter. In this section, you will learn how to install the latest Bookworm operating system on a microSD card if you do not have a pre-installed card.

The steps are as follows:

- Insert a microSD card into your PC. You may need to use an SD card adapter
- Go to the website: <https://www.raspberrypi.com/software/>
- Click to download the **Raspberry Pi Imager**. At the time of writing this book, this file was called: **imager_1.7.5.exe**
- Double click to start the imager program and click to install it
- Click **Finish** to run the imager
- Click **Operating System** and select the operating system at the top of the list as: **Raspberry Pi OS (64-bit)**. See Figure 2.15

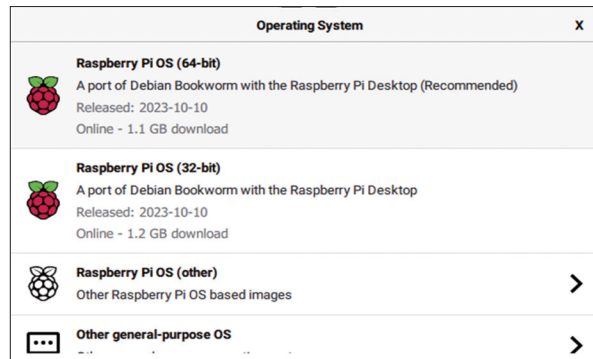


Figure 2.15 Select the operating system

- Click **Storage** and select the SD card storage
- Click to open the settings (gear shape)
- Click to enable SSH
- Click to enable password authentication
- Set username and password
- Click to **Configure wireless LAN**
- Click **Save**
- Click **Write** to write the operating system to the microSD card
- Wait until writing and verifying are finished (Figure 2.16)



- Remove the microSD card and insert into your Raspberry Pi 5

If you have a monitor and keyboard, you can log in to your Raspberry Pi 5 directly and start using it. Otherwise, find the IP address of your Raspberry Pi 5 (e.g. from your router, or there are many apps for smartphones, such as **who's on my wifi** that shows all the devices connected to your router with their IP addresses). Then log in to your Raspberry Pi 5 and start using it.

Chapter 3 • Using The Console Commands

3.1 Overview

Raspberry Pi is based on a version of the Linux operating system. Linux is one of the most popular operating systems in use today. Linux is very similar to other operating systems, such as Windows and UNIX. Linux is an open operating system based on UNIX and has been developed collaboratively by many companies since 1991. In general, Linux is harder to manage than some other operating systems like Windows, but offers more flexibility and configuration options. There are several popular versions of the Linux operating system, such as Debian, Ubuntu, Red Hat, Fedora and so on.

Linux commands are text-based. In this chapter, you will be looking at some of the useful Linux commands and see how you can manage your Raspberry Pi using these commands.

When you apply power to your Raspberry Pi 5, the Linux command line (or the Linux shell, or Console commands) is the first thing you see, and it is where you can enter operating system commands.

3.2 The Command Prompt

Assuming your username is **pi**, after you log in to Raspberry Pi 5, you will see the following prompt displayed where the system waits for you to enter a command:

```
pi@raspberrypi: ~$
```

Here, **pi** is the name of the user who is logged in.

raspberrypi is the name of the computer, used to identify it when connecting over the network.

~ character indicates that you are currently in your default directory.

3.3 Useful Console commands

In this section, you will be learning some of the useful Console commands, where examples will be given for each command. **In this chapter, commands entered by the user are shown in bold for clarity.** Also, it is important to remind you that all the commands must be terminated by the Enter key.

3.3.1 System and user information

These commands are useful as they tell you information about the system. Command **cat /proc/cpuinfo** displays information about the processor (command **cat** displays the contents of a file. In this example, the contents of file **/proc/cpuinfo** is displayed). Since there are four cores in the Raspberry Pi 5, the display is in four sections. Figure 3.1 shows an example display, where only part of the display is shown here.

```

pi@raspberrypi:~ $ cat /proc/cpuinfo
processor       : 0
BogoMIPS      : 108.00
Features      : fp asimd evtstrm aes pmull sha1 sha2 crc32 atomics fphp
p cquid asimdrrm lrcpc dcpop asimddp
CPU implementer : 0x41
CPU architecture: 8
CPU variant   : 0x4
CPU part      : 0xd0b
CPU revision  : 1

processor       : 1
BogoMIPS      : 108.00
Features      : fp asimd evtstrm aes pmull sha1 sha2 crc32 atomics fphp
p cquid asimdrrm lrcpc dcpop asimddp
CPU implementer : 0x41
CPU architecture: 8
CPU variant   : 0x4
CPU part      : 0xd0b
CPU revision  : 1

processor       : 2
BogoMIPS      : 108.00

```

Figure 3.1 Command: `cat /proc/cpuinfo`

Command **uname -s** displays the operating system kernel name, which is Linux. Command **uname -a** displays complete detailed information about the kernel and the operating system. An example is shown in Figure 3.2.

```

pi@raspberrypi:~ $ uname -a
Linux raspberrypi 6.1.0-rpi4-rpi-2712 #1 SMP PREEMPT Debian 1:6.1.54-1+rpt1
3-09-27) aarch64 GNU/Linux
pi@raspberrypi:~ $ █

```

Figure 3.2 Command: `uname -a`

Command **cat /proc/meminfo** displays information about the memory on your Raspberry Pi. Information such as the total memory and free memory at the time of issuing the command are displayed. Figure 3.3 shows an example, where only part of the display is shown here.

```

pi@raspberrypi:~ $ cat /proc/meminfo
MemTotal:      8246848 kB
MemFree:       7792320 kB
MemAvailable:  7993952 kB
Buffers:       21552 kB
Cached:        246240 kB
SwapCached:    0 kB
Active:        280096 kB
Inactive:      64848 kB
Active(anon):  77008 kB
Inactive(anon): 4112 kB
Active(file):  203088 kB
Inactive(file): 60736 kB
Unevictable:   0 kB
Mlocked:      0 kB
SwapTotal:    102368 kB
SwapFree:     102368 kB
Zswap:        0 kB
Zswapped:     0 kB
Dirty:        0 kB
Writeback:    0 kB
AnonPages:    77232 kB
Mapped:       70880 kB

```

Figure 3.3 Command: `cat /proc/meminfo`

Command **whoami** displays the name of the current user. In this case, **pi** is displayed as the current user.

A new user can be added to your Raspberry Pi 5 using the command **useradd**. In the example in Figure 3.5, a user called **John** is added. A password for the new user can be added using the **passwd** command followed by the username. In Figure 3.4, the password for user John is set to **mypassword** (not displayed for security reasons). Notice that both the **useradd** and **passwd** are privileged commands, and the keyword **sudo** must be entered before these commands. Notice that the **-m** option creates a home directory for the new user.

```
pi@raspberrypi:~ $ sudo useradd -m John
pi@raspberrypi:~ $ sudo passwd John
New password:
Retype new password:
passwd: password updated successfully
pi@raspberrypi:~ $ █
```

Figure 3.4 Commands: *useradd* and *passwd*

You can log in to the new user account by specifying the username and the password as shown in Figure 3.5. You can type command **exit** to log out from the new account.

```
pi@raspberrypi:~ $ su John
Password:
John@raspberrypi:/home/pi $ exit
exit
pi@raspberrypi:~ $ █
```

Figure 3.5 Logging into a new account

Command **sudo apt-get upgrade** is used to upgrade all the software packages on the system.

3.3.2 The Raspberry Pi 5 directory structure

The Raspberry Pi 5 directory structure consists of a single root directory, with directories and subdirectories under the root. Different types of operating system programs and application programs are stored in different directories and subdirectories.

Figure 3.6 shows part of the Raspberry Pi 5 directory structure. Notice that the root directory is identified by the **/** symbol. Under the root we have directories named such as **bin**, **boot**, **dev**, etc, **home**, **lib**, **lost+found**, **media**, **mnt**, **opt**, **proc**, and many more. The important directory as far as the users are concerned is the **home** directory. The **home** directory contains subdirectories for each user of the system. In the example in Figure 3.7, **pi** is the subdirectory for user **pi**. In a new system, this subdirectory contains two subdirectories called **Desktop** and **python_games**.