

# Raspberry Pi Terminal Setup & PRATICO Instrument Guide

---

## 1. First Boot and Login

- Connect Raspberry Pi to a monitor via mini HDMI and connect a keyboard.
- Power on the Raspberry Pi.
- Default login (Raspberry Pi OS):

Username: pi  
Password: raspberry

- You will see the terminal prompt:

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

---

## 2. Configure Wi-Fi using wpa\_supplicant

1. Edit the wpa\_supplicant file:

```
sudo nano /etc/wpa_supplicant/wpa_supplicant.conf
```

2. Add your network credentials:

```
country=US
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1

network={
    ssid="YOUR_SSID"
    psk="YOUR_PASSWORD"
}
```

3. Save (CTRL+O) and exit (CTRL+X).

4. Apply changes:

```
sudo wpa_cli reconfigure
```

5. Check your IP:

```
ip a
```

---

## 3. Enable SSH for Remote Access

1. Enable and start SSH:

```
sudo systemctl enable ssh
sudo systemctl start ssh
```

2. Find Pi's IP:

```
hostname -I
```

3. Connect from another computer:

```
ssh pi@<IP_ADDRESS>
```

---

#### 4. Wiring for FWW\_user.py

- **Green LED (light source):** GPIO 26 -> Positive, GND -> Negative
- **Red LED (working/not ready):** GPIO 23 -> Positive, GND -> Negative
- **Ready LED (green):** GPIO 24 -> Positive, GND -> Negative
- **Button:** GPIO 4 -> Button -> GND
- **I2C Sensor (LTR329):** SDA -> SDA, SCL -> SCL, 3.3V -> VCC, GND -> GND

---

#### 5. Python Script Overview (FWW\_user.py and Calibration)

**Purpose:** Automates water analysis for N-NO3 and P-PO4 using LED light measurements and allows sensor calibration.

**Initialization:** - Setup GPIO, button, and I2C sensor. - Load calibration files (P\_cal.txt, N\_cal.txt).

**Key Functions:** - lux\_conversion(): Converts raw sensor data to lux. - LED\_on(): Turns green LED on and records light readings. - simple\_measure(): Measures ambient + LED light. - blank(): Blank measurement for reference. - measure(): Measures analyte sample. - concentration(): Converts absorbance to concentration. - save(): Saves results with timestamp. - **Offline storage:** Each measurement is automatically saved as a .txt file in /home/pa/ with a timestamp and analyte identifier (e.g., 2025\_10\_06\_14\_30\_NN03.txt). This allows offline access to results without network or database connection. - Nitrates() and Phosphates(): Handles analyte measurement logic. - NN03\_calibration() and PP04\_calibration(): Guides calibration using standard solutions. - linear\_fitting(): Calculates calibration slope, intercept, and R<sup>2</sup>. - main(): Controls loop and LED/button interactions.

**Workflow:** 1. Blank measurement at startup. 2. Loop: - Red LED blares. - Wait 10s for button press: - Pressed -> N-NO3 measurement. - Not pressed -> P-PO4 measurement. - Results are saved automatically offline as .txt files in /home/pa/. 3. Calibration guides user through known concentration standards and saves results to file.

---

## 6. ABC of Linux Terminal Commands

Command	Meaning
ls	List files
cd	Change directory
pwd	Print working directory
nano	Text editor
sudo	Execute as admin
chmod	Change file permissions
systemctl	Manage services
reboot	Restart Pi
ip a	Show network interfaces
ssh	Secure remote login

---

## 7. Auto-run Script at Boot (Before Login)

1. Edit rc.local:

```
sudo nano /etc/rc.local
```

2. Add above exit 0:

```
python3 /home/pi/FWW_user.py &
```

3. Save and exit. The script will run on boot before login.
- 

## 8. Python Requirements

### Requirements.txt:

```
rpi.gpio  
adafruit-circuitpython-ltr329-ltr303  
numpy  
pandas  
scikit-learn  
gpiozero
```

### Installation (system-wide, no virtual environment):

```
sudo apt update  
sudo apt install python3-pip -y  
sudo pip3 install -r /home/pi/requirements.txt
```

This will install all required libraries for both the main instrument script and calibration script.

---

### *9. Move Files In/Out using SCP*

#### **Copy file from local machine to Raspberry Pi:**

```
scp /path/to/local/file pi@<IP_ADDRESS>:/home/pi/
```

#### **Copy file from Raspberry Pi to local machine:**

```
scp pi@<IP_ADDRESS>:/home/pi/file.txt /path/to/local/destination/
```

---