Intro to the Raspberry Pi (RPi) & The Internet of Things (IoT)

Class 2 – 2018 / 03 / 18

Raspberry Pi Name Origin

… from last class
Thanks for this question!

TS: Where does the name Raspberry Pi come from?

Raspberry is a reference to a fruit naming tradition in the old days of microcomputers. A lot of computer companies were named after fruit. There's Tangerine Computer Systems, Apricot Computers, and the old British company Acorn, which is a family of fruit.

Pi is because originally we were going to produce a computer that could only really run Python. So the Pi in there is for Python. Now you can run Python on the Raspberry Pi but the design we ended up going with is much more capable than the original we thought of, so it's kind of outlived its name a little bit.

PCWorld on „Pi Day“ 3 / 14 – New RPi Model 3 B+!

The Raspberry Pi 3 Model B+ rolls out just in time for Pi Day | PCWorld

NEWS

The Raspberry Pi 3 Model B+ rolls out just in time for Pi Day

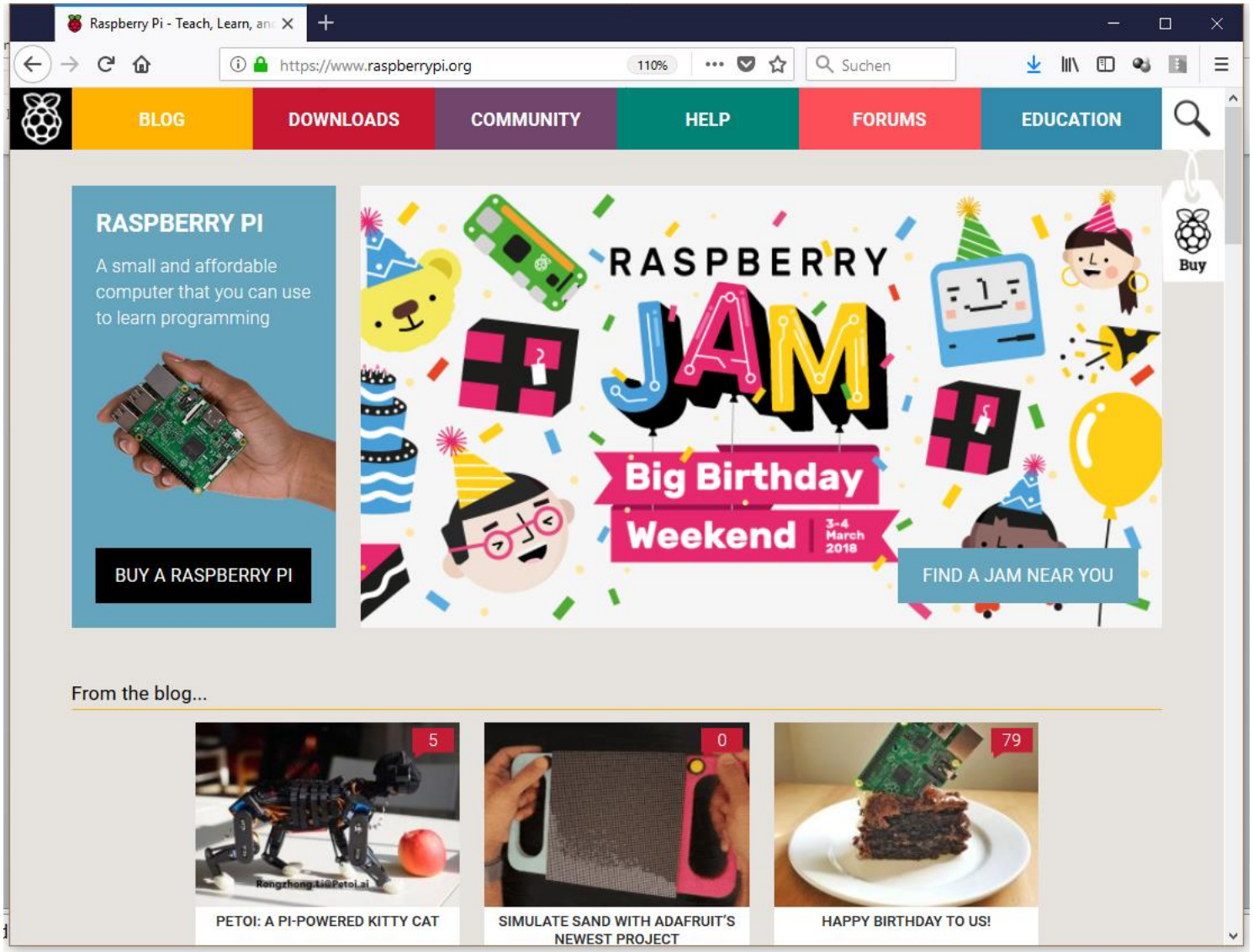
For $35 you can pick-up the Raspberry Pi 3 Model B+ with a faster processor and dual-band Wi-Fi.

By Ian Paul
Contributor, PCWorld | MAR 14, 2018 9:03 AM PT

MORE LIKE THIS

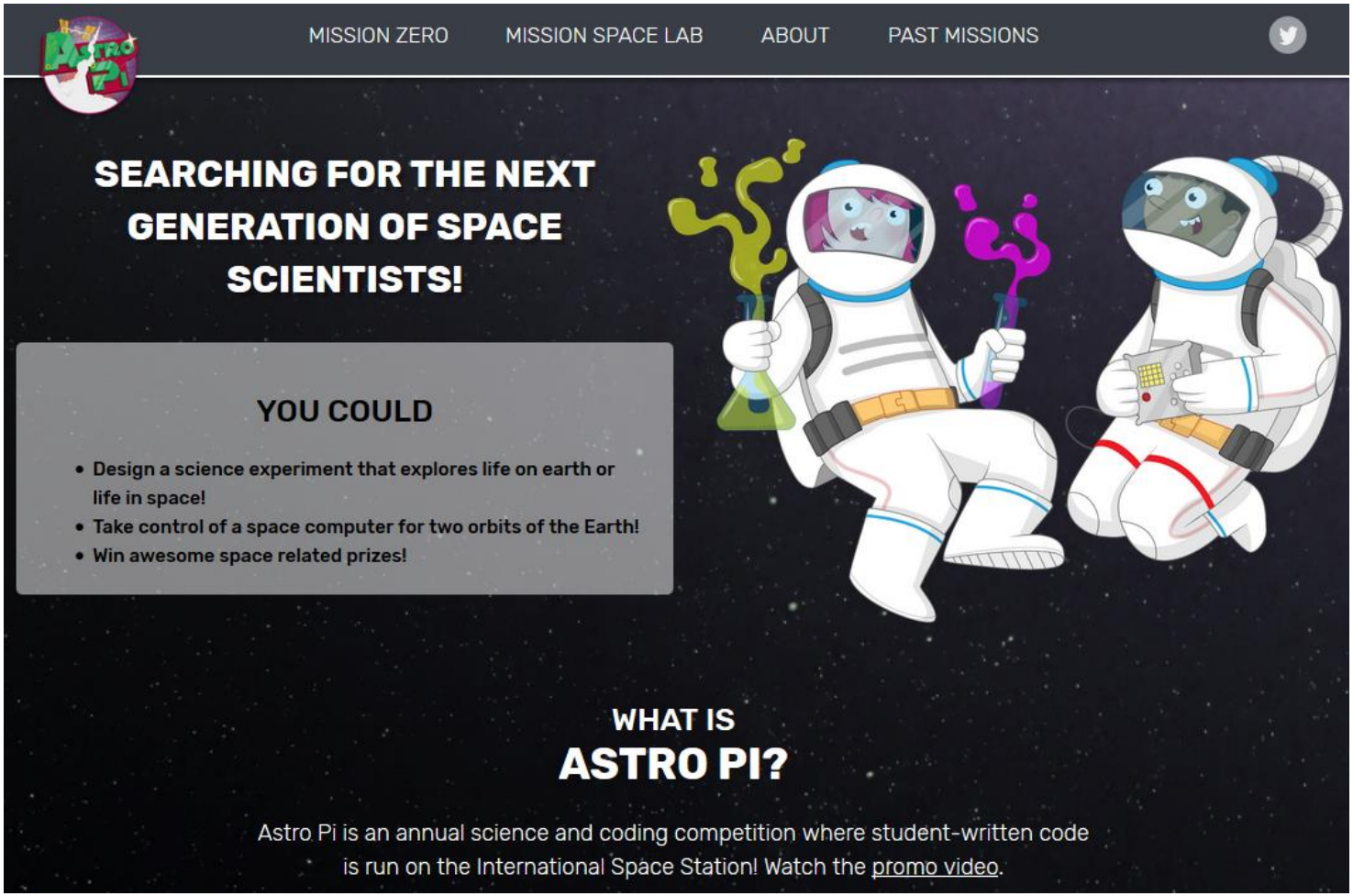
- Raspberry Pi: Everything you need to know
- Best Raspberry Pi 3 cases: 10 great options to dress up your mini PC
- Best Raspberry Pi 3 kits: 8 options for beginners and experienced makers
- VIDEO Kyle Bennett from HardOCP: What's up with Nvidia's GeForce Partner Program? |...

Check out http://www.raspberrypi.org



AstroPi

https://www.youtube.com/watch?v=yjll_4JY98g



Raspberry Pi Weather Station

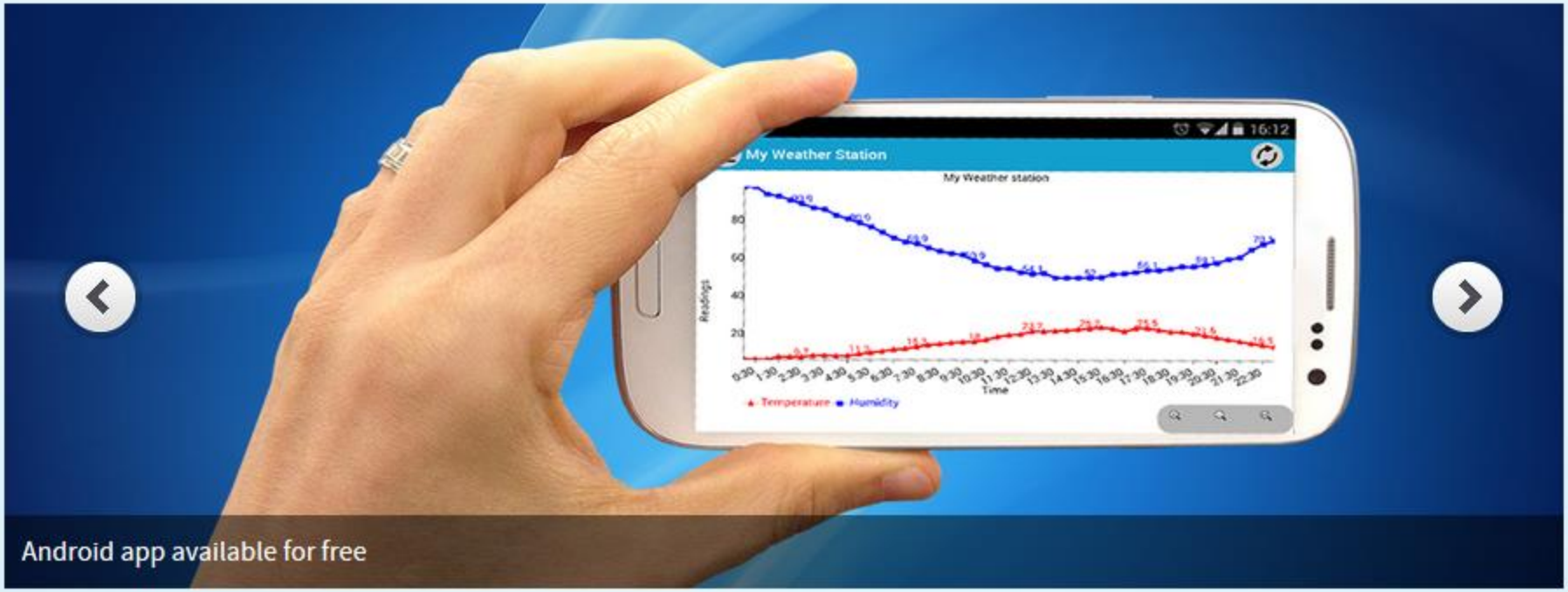
https://www.raspberryweather.com

Suchen

Raspberry Weather

search...

Home Software set up Hardware set up Optional Additional info

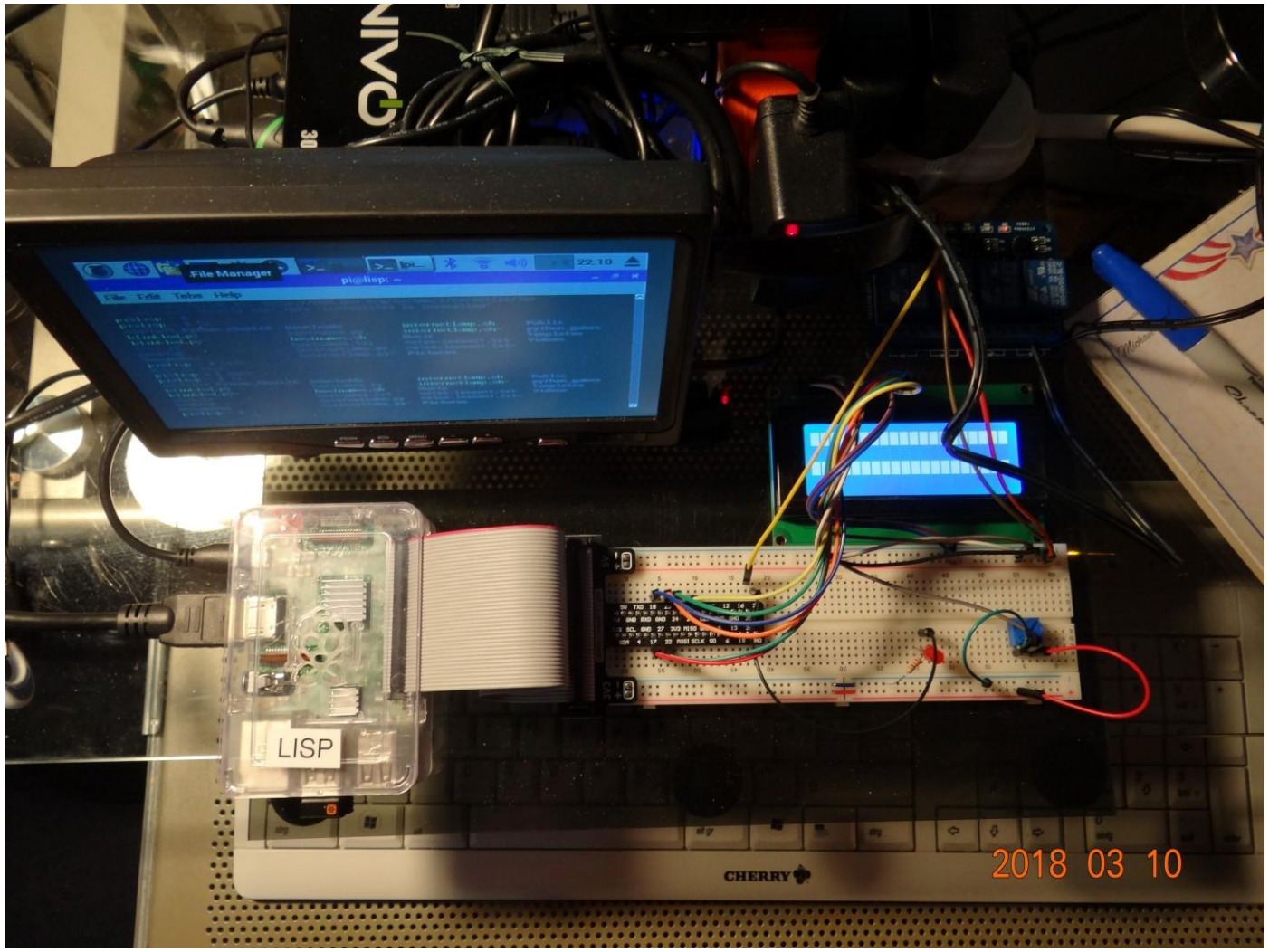


Optional - „RPi Workstation“ for Easy Transport

- Get an acrylic sheet from Osh!

- Double Sided *Really Strong* Tape, e.g., M3 Mounting Tape (not simple transparent Scotch double sided)
- Get a magnifier glass if your eyes are as bad as mine (GPIO labels are small)
- Notice that monitor has glued tape underneath already!

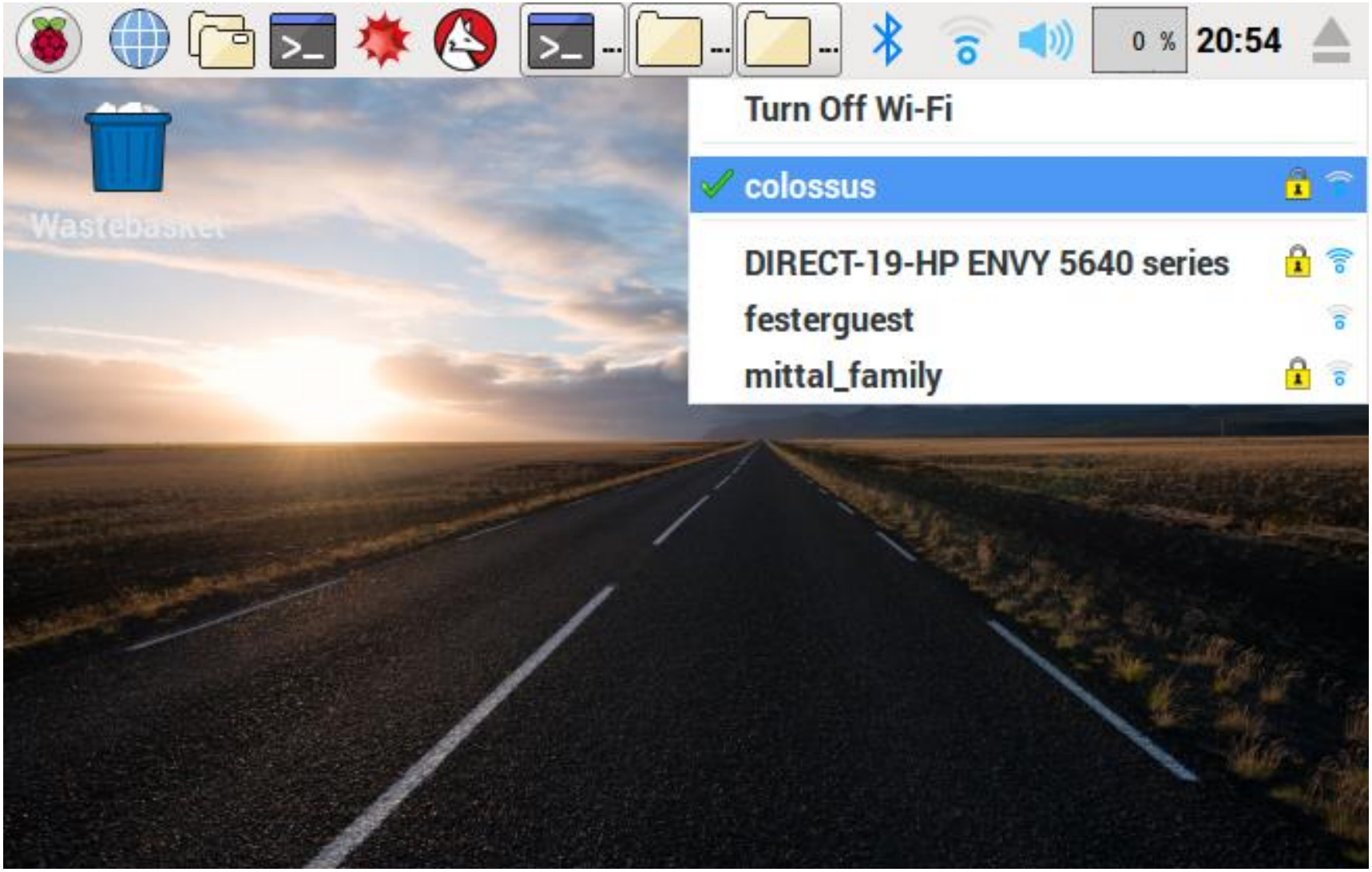
Idea for Setup



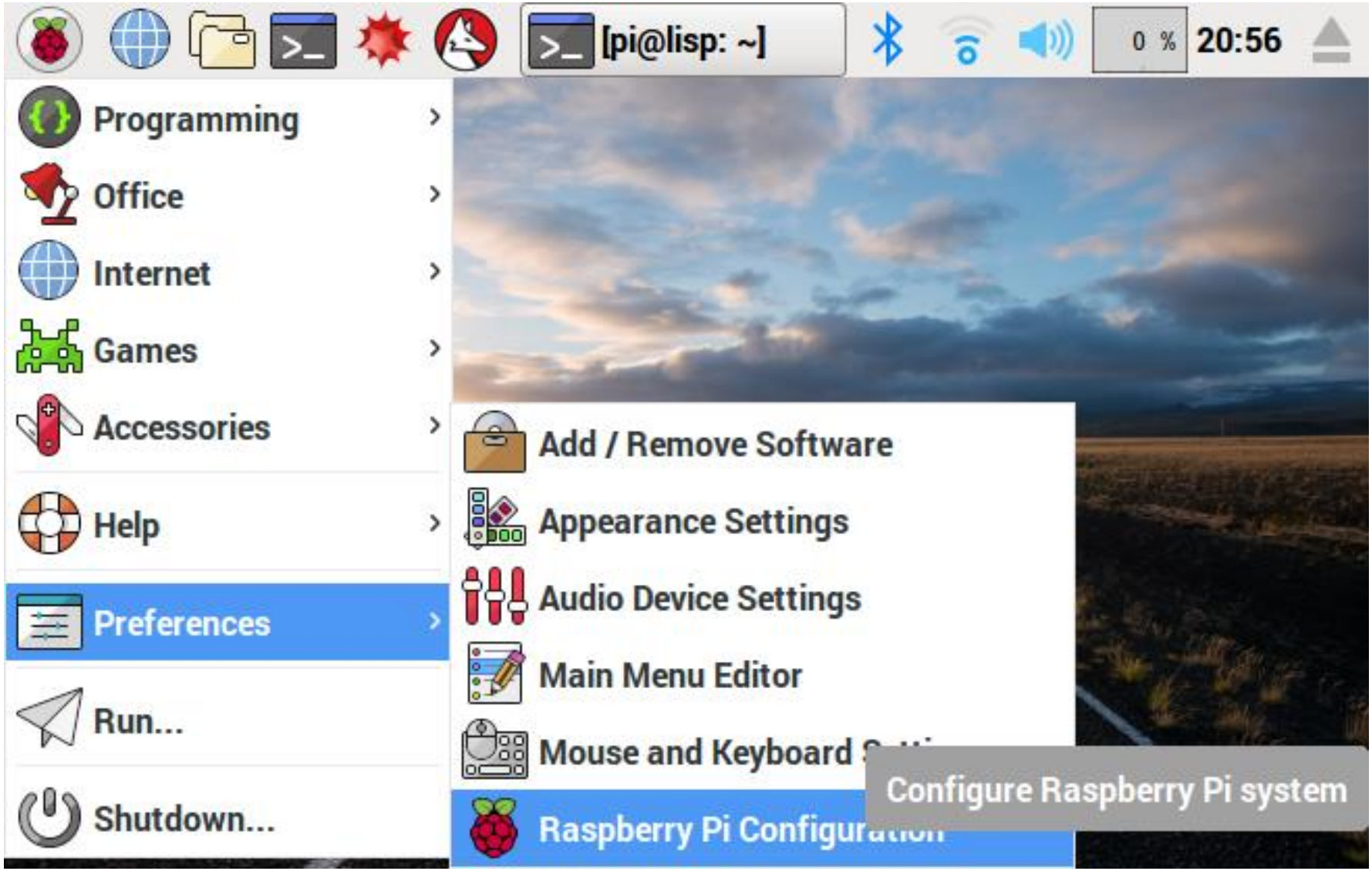
Configuration

- Screen resolution
- Internet WiFi
- Password
  - Default user: **pi** (no need to change)
  - Default password: **raspberry** – let‘s change that!
  - Super user login disabled, but **sudo**
- Set Hostname from Lottery
  - Use clear tape for stickers
  - 1 on RPi case, 1 under keyboard
  - Leave the keyboard dongles plugged in all the time to avoid confusion

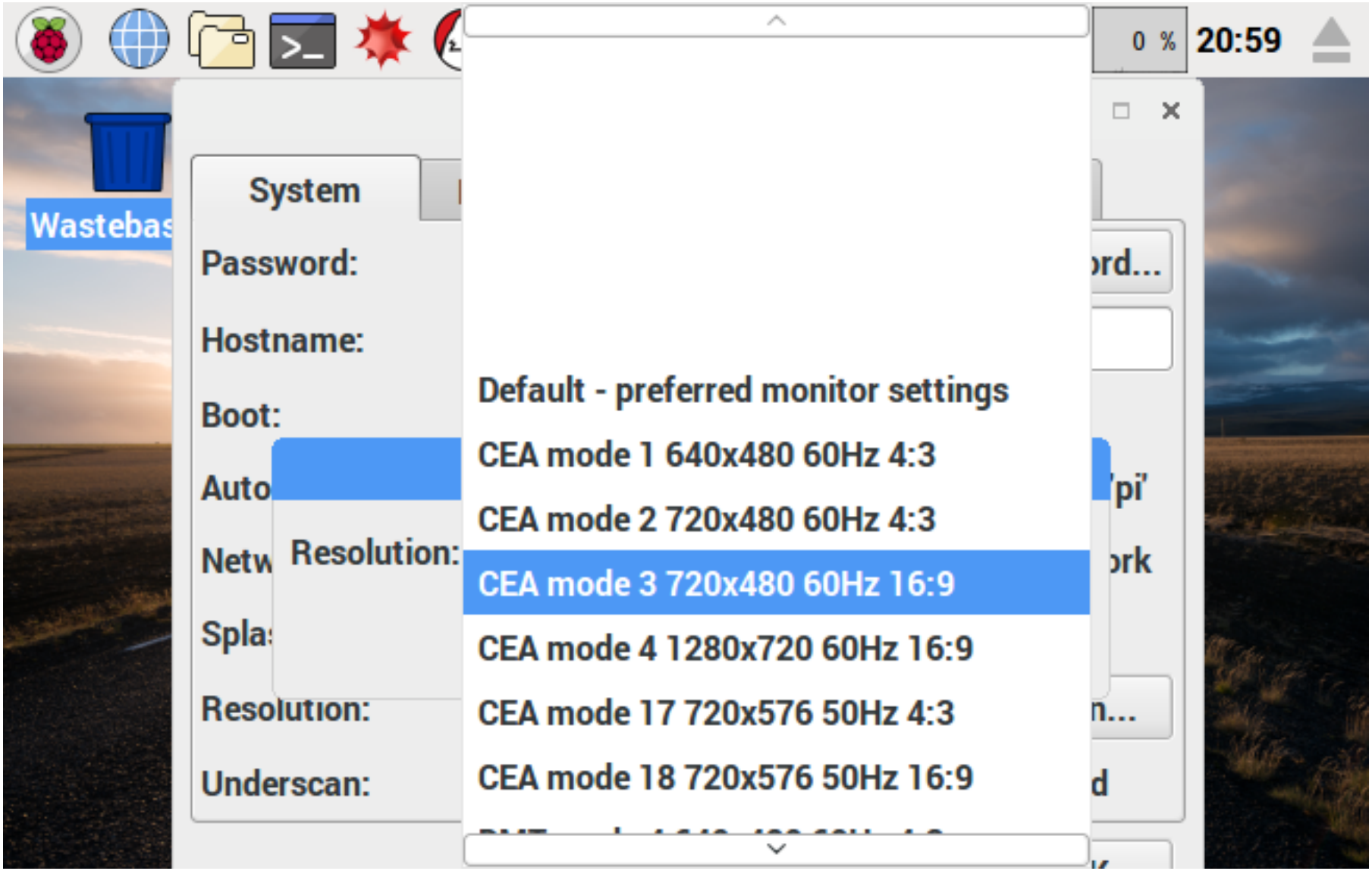
Wifi Configuration
Wifi Password is ….



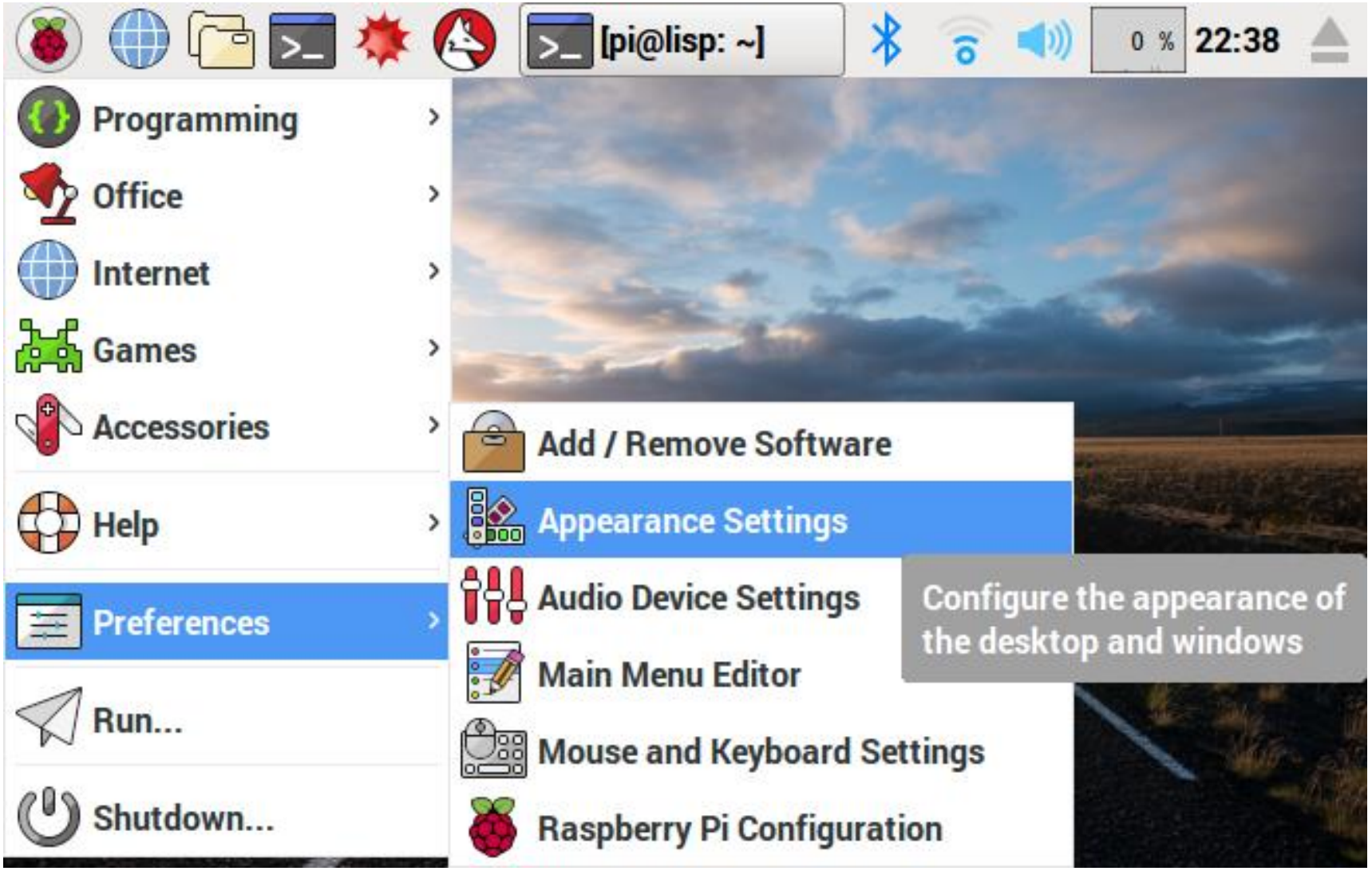
Screen Resolution



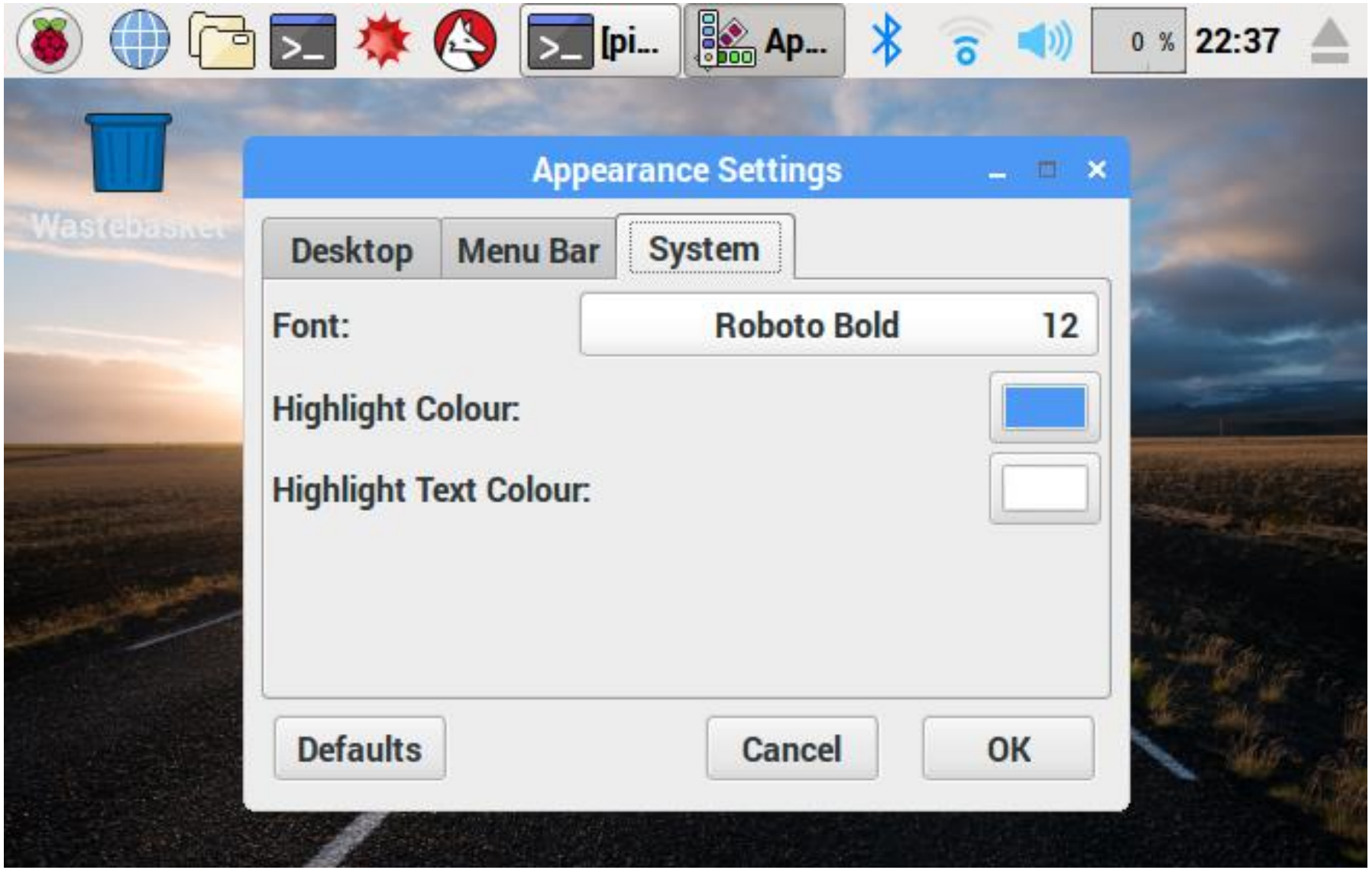
Screen Resolution



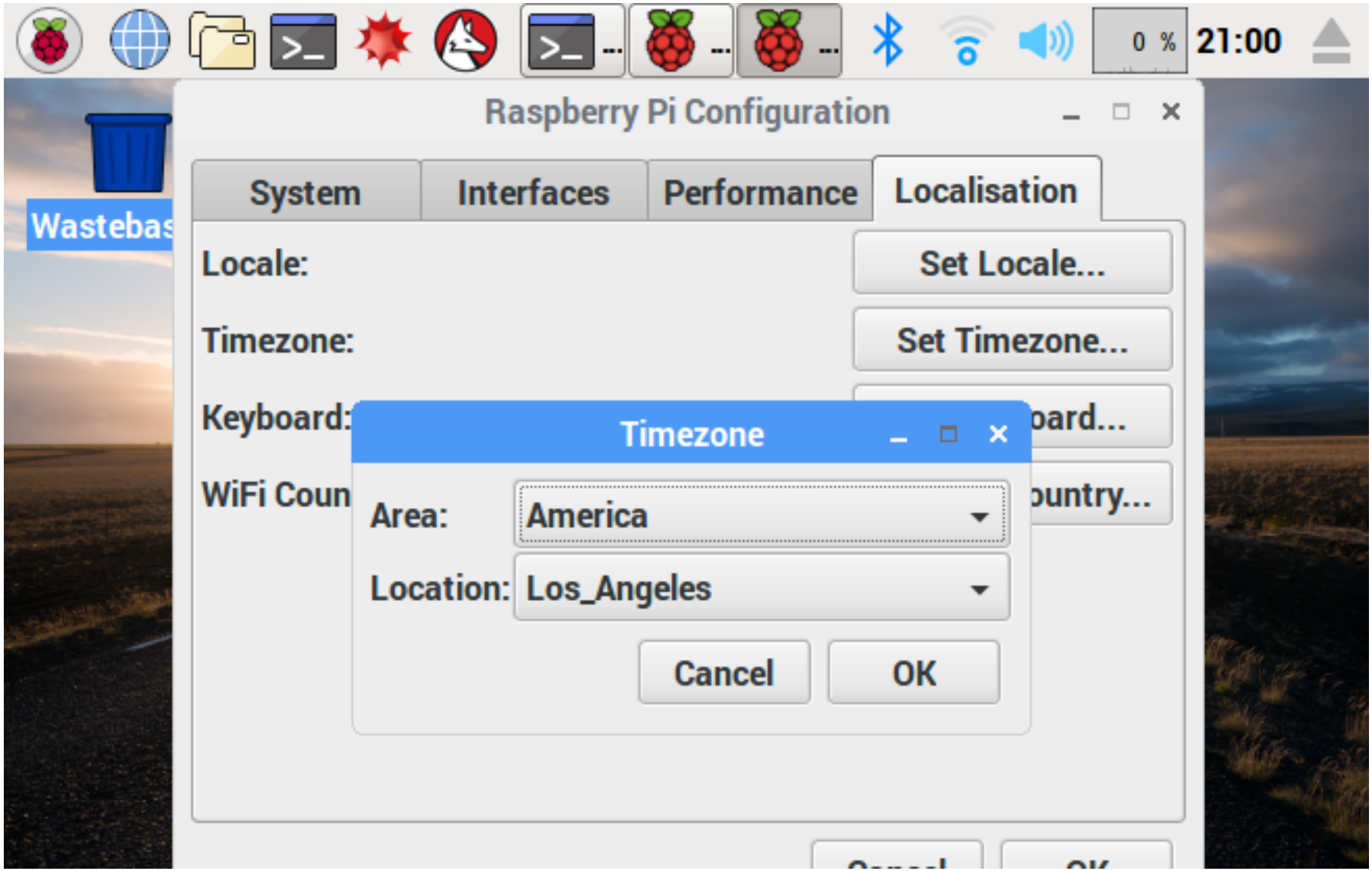
Increase Font Size



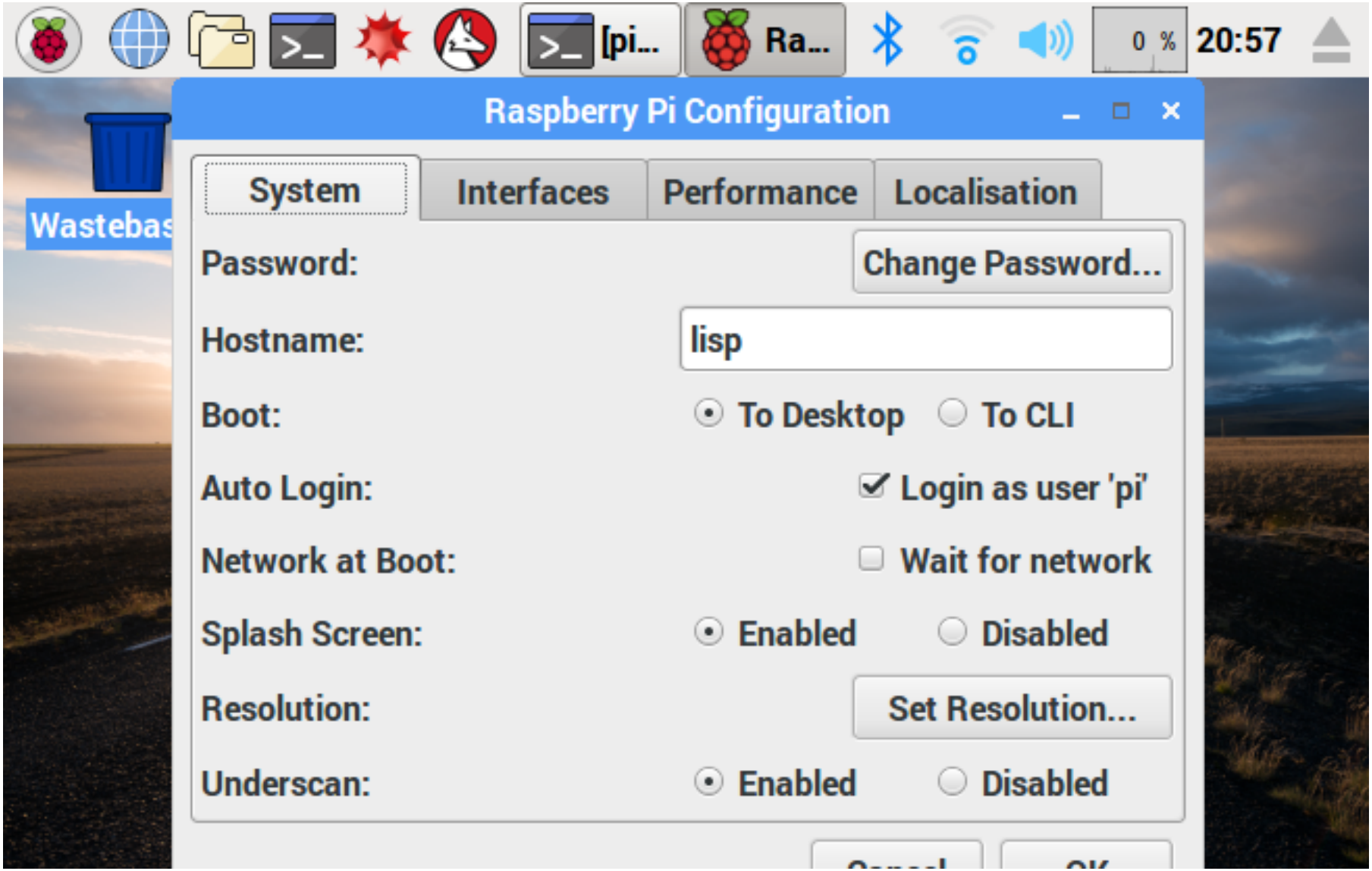
Increase Font Size – 2



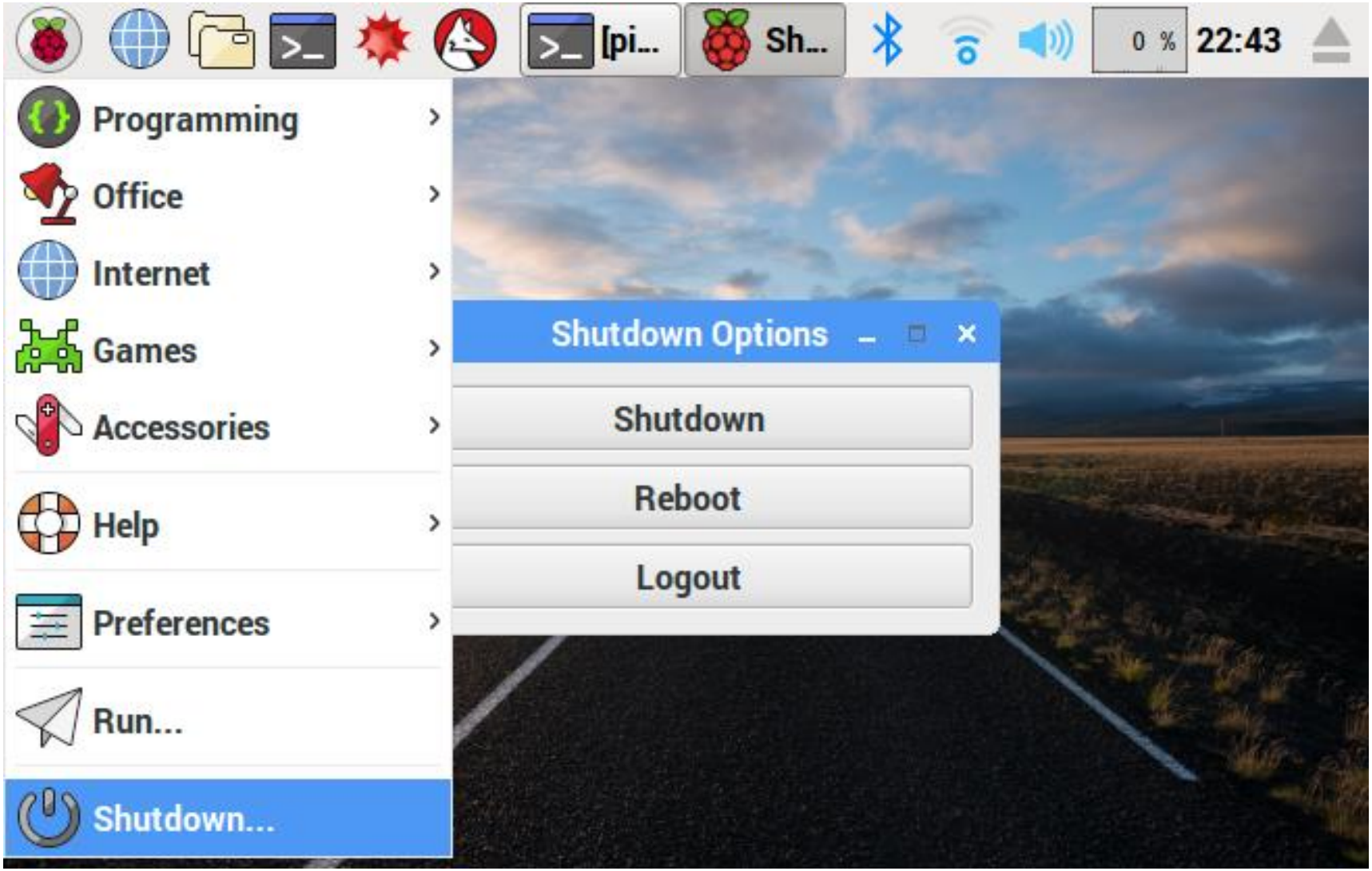
TimeZone / Localization for Clock



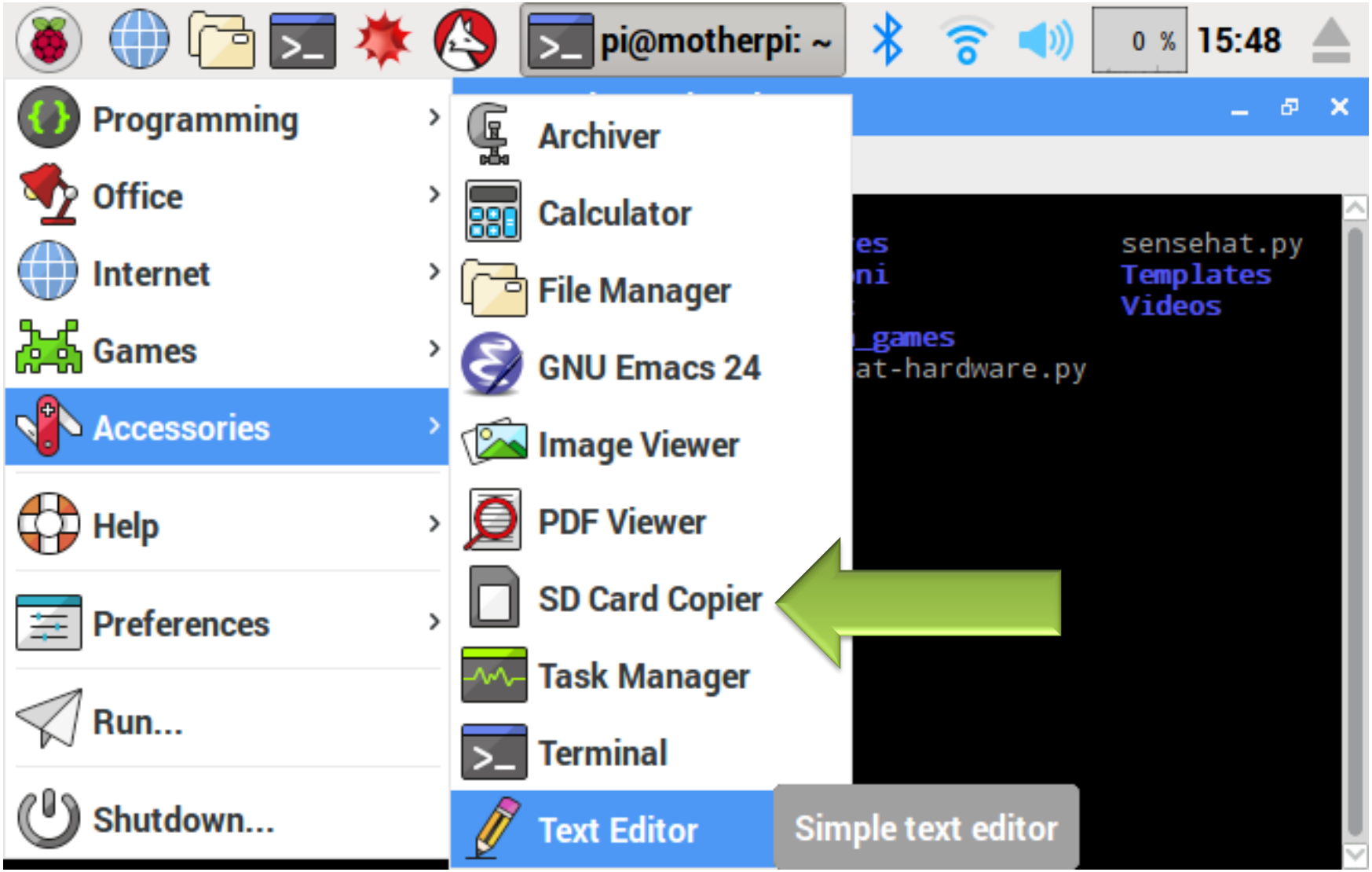
Password & Hostname from Lottery



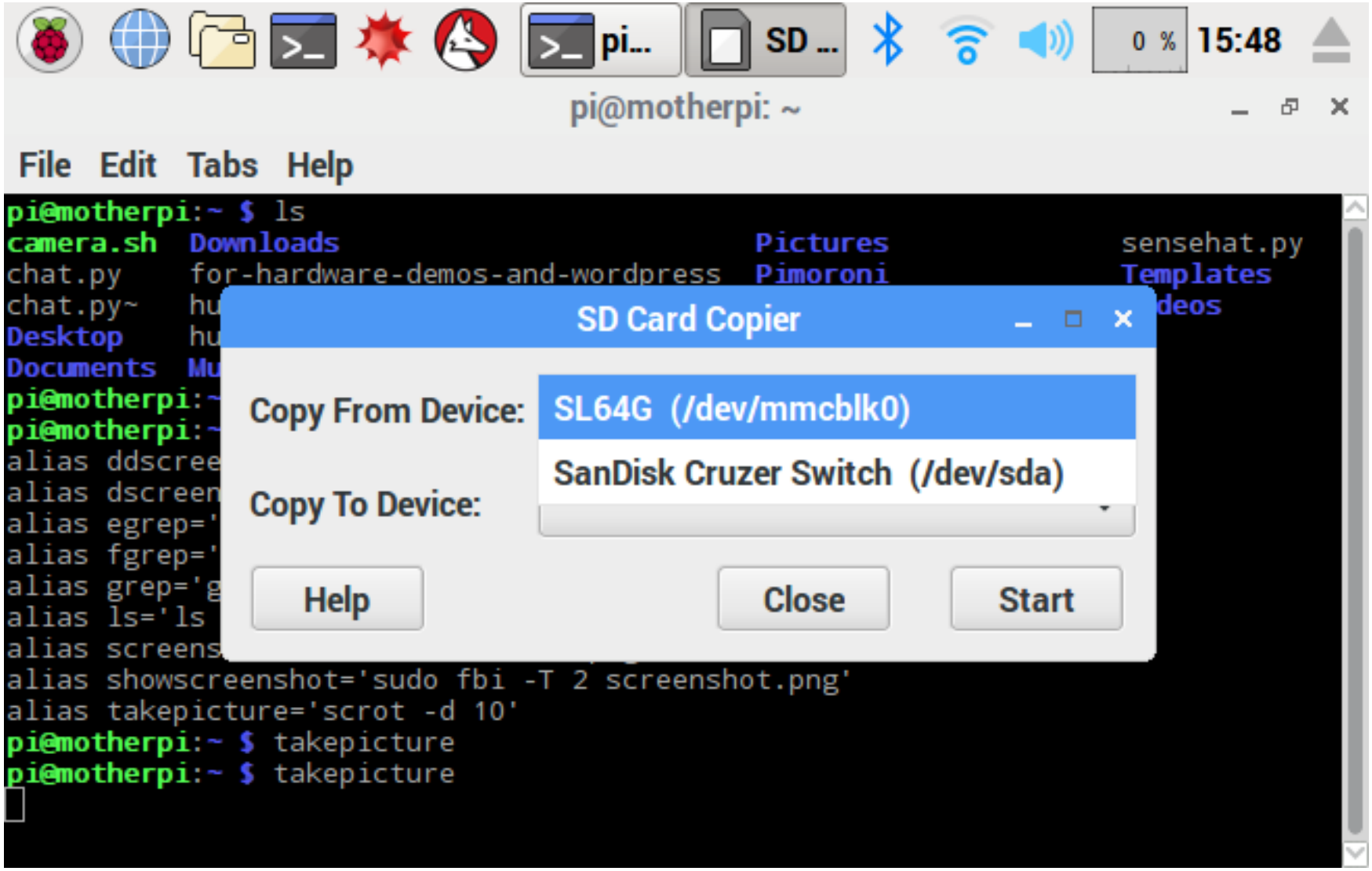
Reboot -
Never Just Unplugg Your RPi!



MAKE A BACKUP!!



Use the USB SDCard Adapter



Adapter USB – Micro SDCard

Exporing PIXEL – Taking Notes in Class

- PIXEL =

  '*Pi* Improved XWindow Environment, Lightweight‘
- Show File Browser
- **Take screenshots during class of your work**
  - **Keep a record of what you did**
  - **FN + Delete = „Print Screen“**
- TouchPad
  - Use 2 Finger Scrolling
  - Right Mouse Button for Mac Users ☺
- Browser
  - Chromium
  - Flash
  - YouTube works also!

PIXEL Keys / Taking Notes in Class

- Window not showing all content?
  Use maximize button!
  - Also minimize, close
- **Switch / cycle between windows: Alt-tab**
  - **Open the Text Editor**
  - **Take notes using Alt-tab switching!!**
  - **Between Consol / Text Editor**
- Close window Ctrl-w
- Use Ctrl-Esc for „Start Menu“

Linux Basics

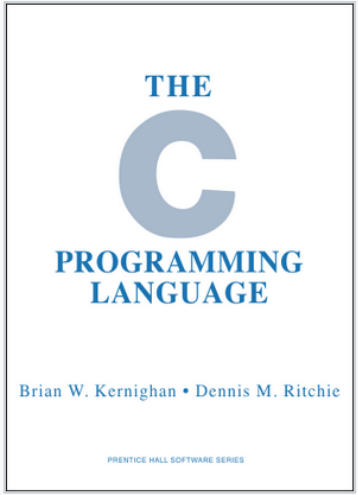
- Some History
- Command Shell / BASH / „Terminal“ / Console
  - the „command post“ of every Unix computer
- Why not do everything from the desktop?
  - The shell is much more powerful and versatile
  - Allows „scripting“
  - Most things can ONLY be done from the shell
  - True hackers don‘t use the desktop ☺

Influential Unix Hackers

Ken Thompson and Dennis Ritchie, principal developers of Research Unix

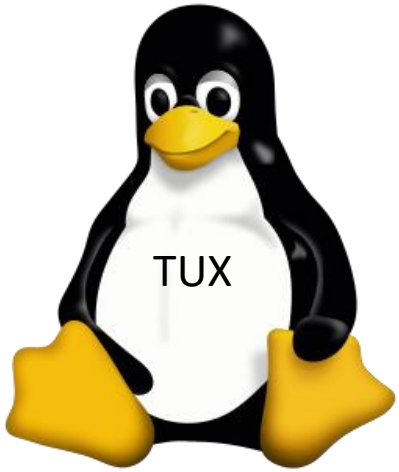
Ken Thompson (sitting) and Dennis Ritchie working together at a PDP-11

K&R C –
„The C Bible“



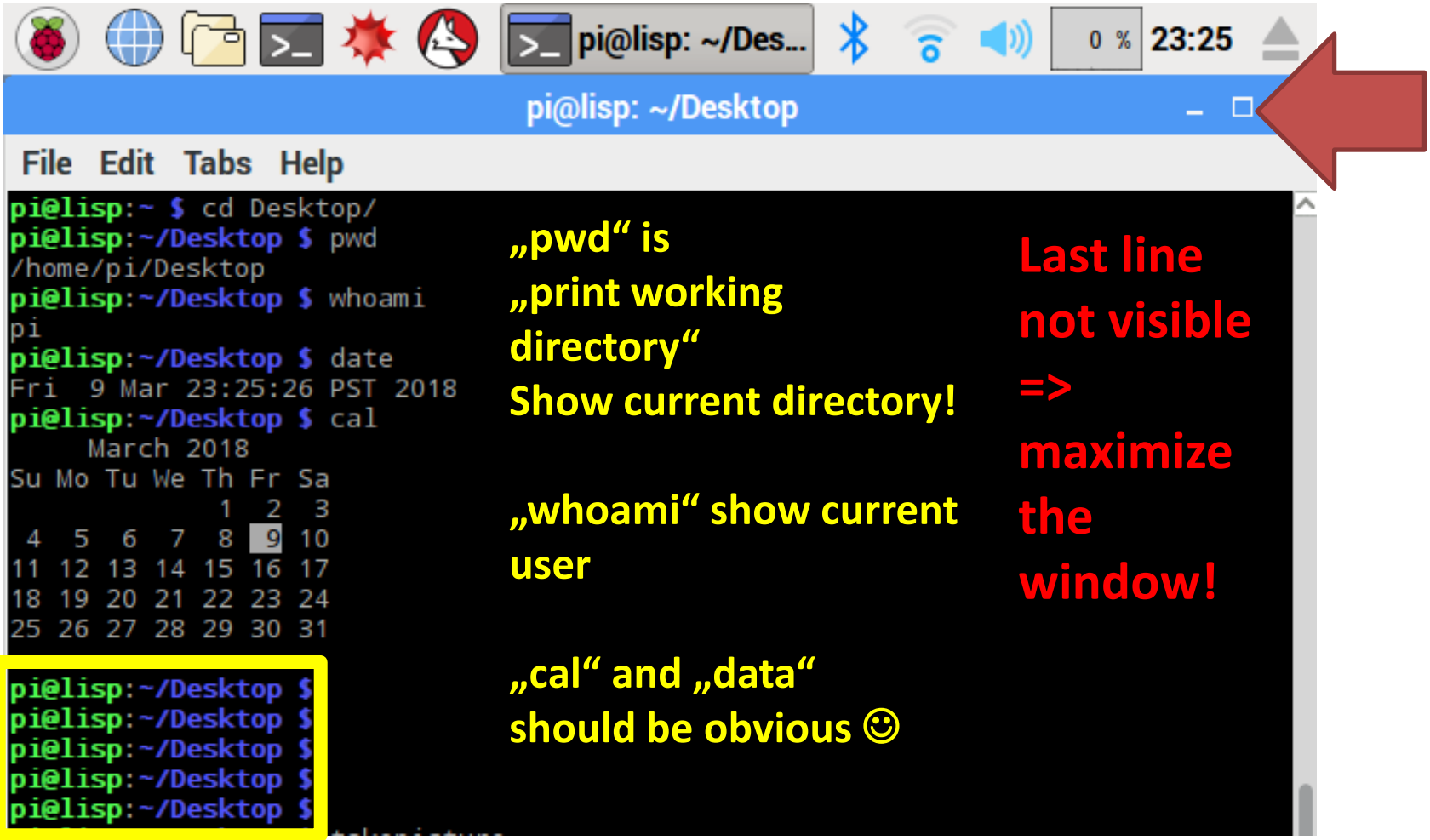
The cover of the book, *The C Programming Language*, first edition by Brian Kernighan and Dennis Ritchie

Linus Torvalds
Linux = Linus‘ Unix

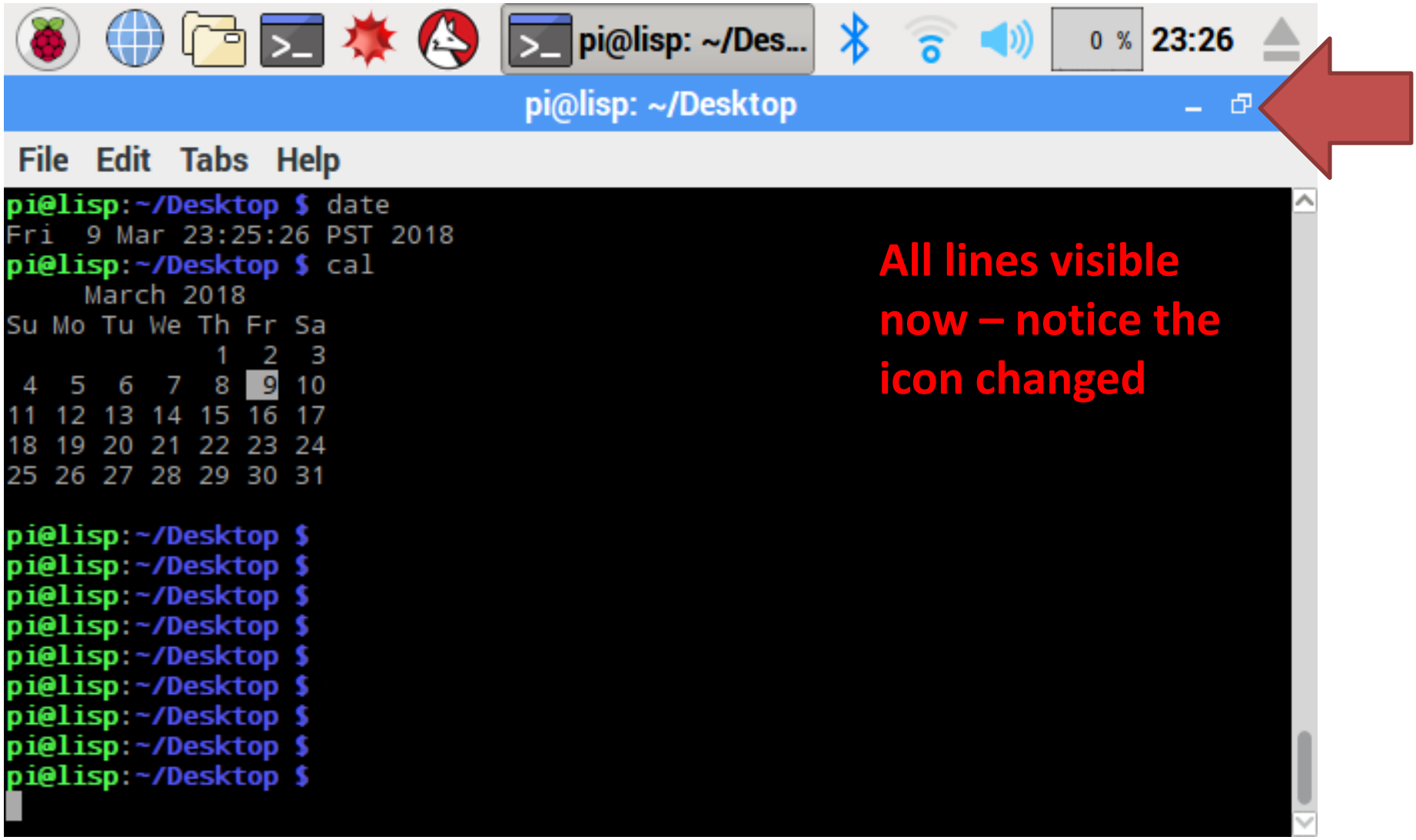


Command Shell = BASH = Terminal = Console = Command Line Interface (CLI) …

PROMPT = <username>@<hostname>: <current directory>



Windows: Minimize, Maximize, Close



BASH Basic Keyboard Shortcuts

- Use „clear“ in the console to clear content
  - However, HISTORY is still available
  - And you can also scroll back
- Make a new „tab“ with Ctrl-Shift-t
- Close a tab with with Ctrl-Shift-w
- Close a shell / BASH with Ctrl-Shift-q
- To get control back over the BASH, use Ctrl-c to interupt / cancel any currently running command
  - Show „top“ process managment program
  - „q“ or „Ctrl-c“ to quit it

BASH Navigation

- Use tab key to complete filenames in current directory!
  - type: cd ~
  - type: cd De<tab> -> completes to cd Desktop/
- Use up/down arrows for history of commands
- Use Ctrl-r for „reverse search“, enter search string
  - Use Ctrl-c to cancel
  - Enter to re-use that command from history

Files, Directories, and Trees

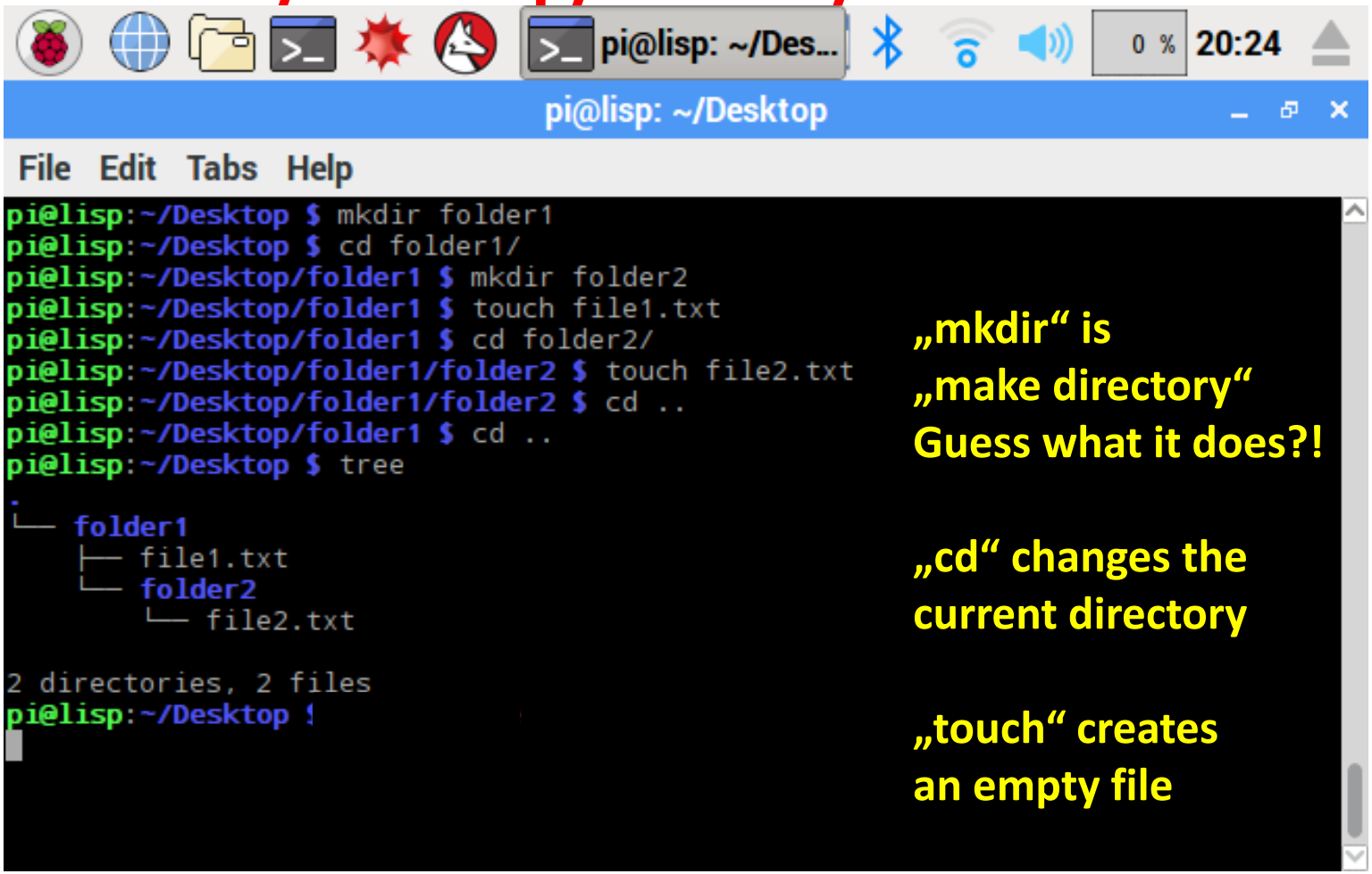
- Files = clear, right? Like in MacOS, Windows, …
  - Files can be „hidden“
    (start with a „period“, eg., „.alias“)
  - Files CAN have an extension, eg., „.txt“ for „text file“
  - Files are OWNED by a user („pi“ or „root“):
    - Privacy, Security (Access Rights)
    - You cannot delete or rename or move files from owner „root“, for examples
  - Files can be created with the File Manager
  - Or from the BASH shell!
- Directories = like folders
  - Can also be hidden, have owners, …
  - Directories can contain other directories

  -> „Tree structure“

Files, Directories, and Trees - 2

- Each file is contained in SOME directory

~/Desktop/folder1/file1.txt = FILE PATH

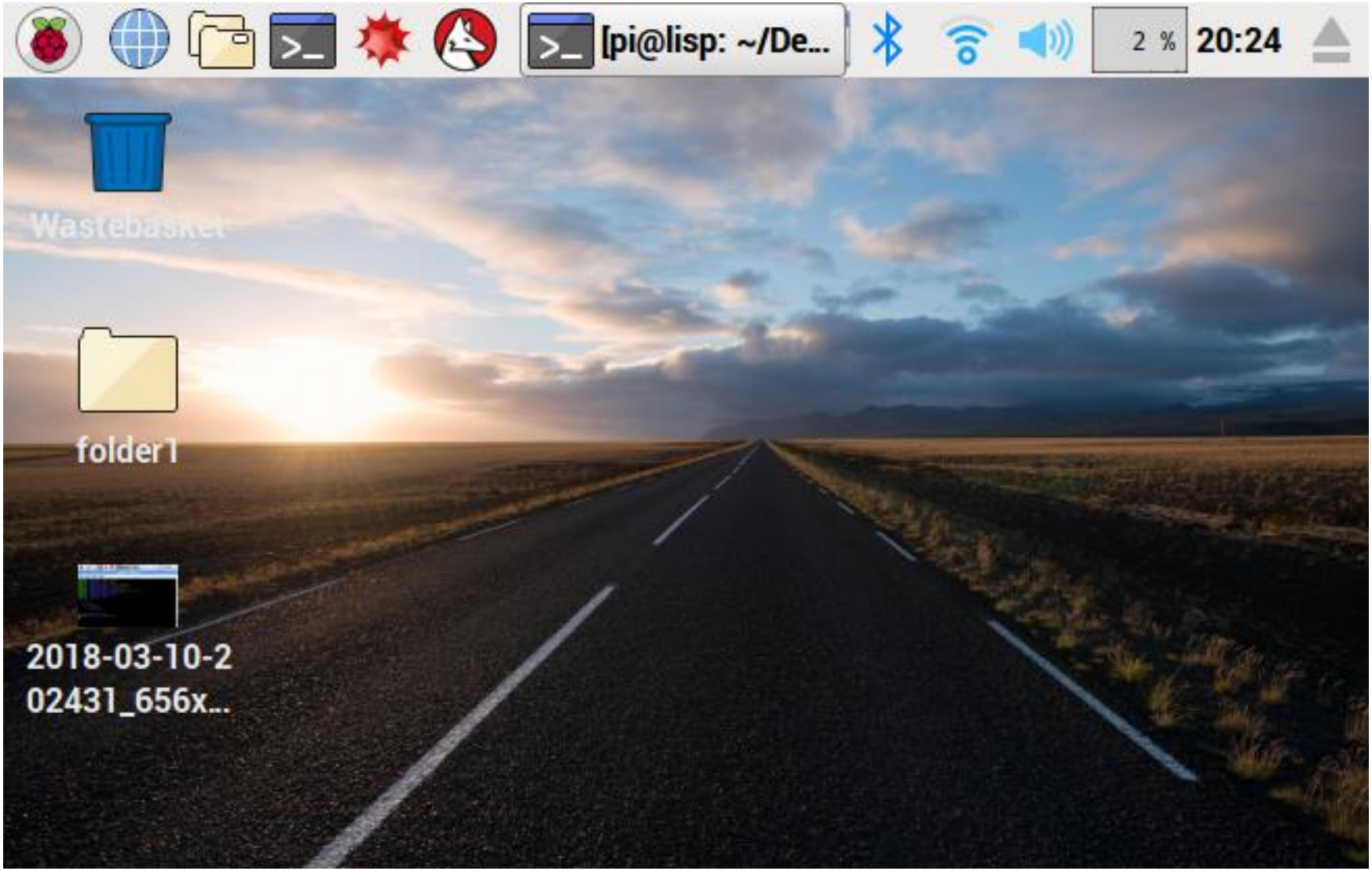


„mkdir“ is „make directory“ Guess what it does?!

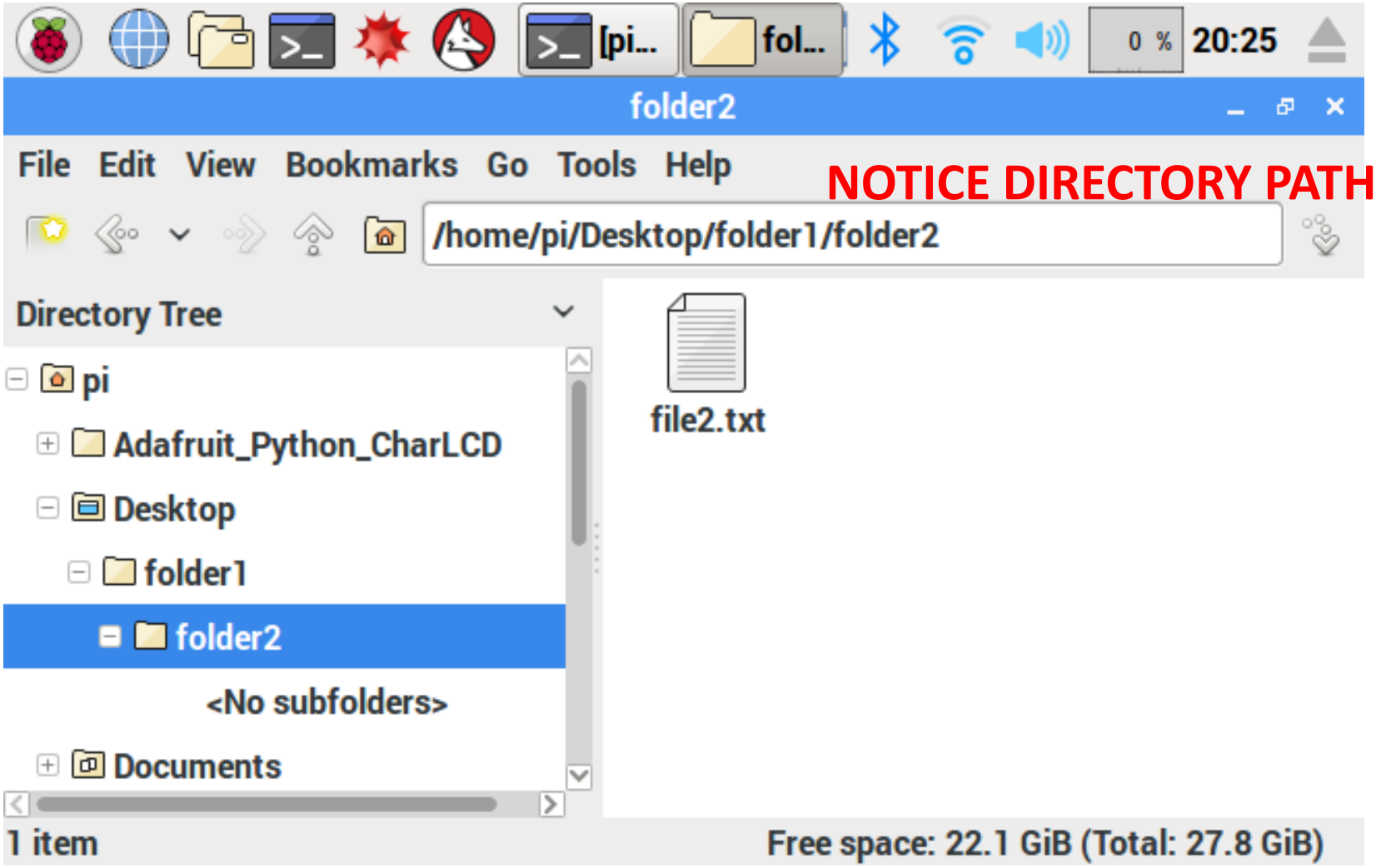
„cd“ changes the current directory

„touch“ creates an empty file

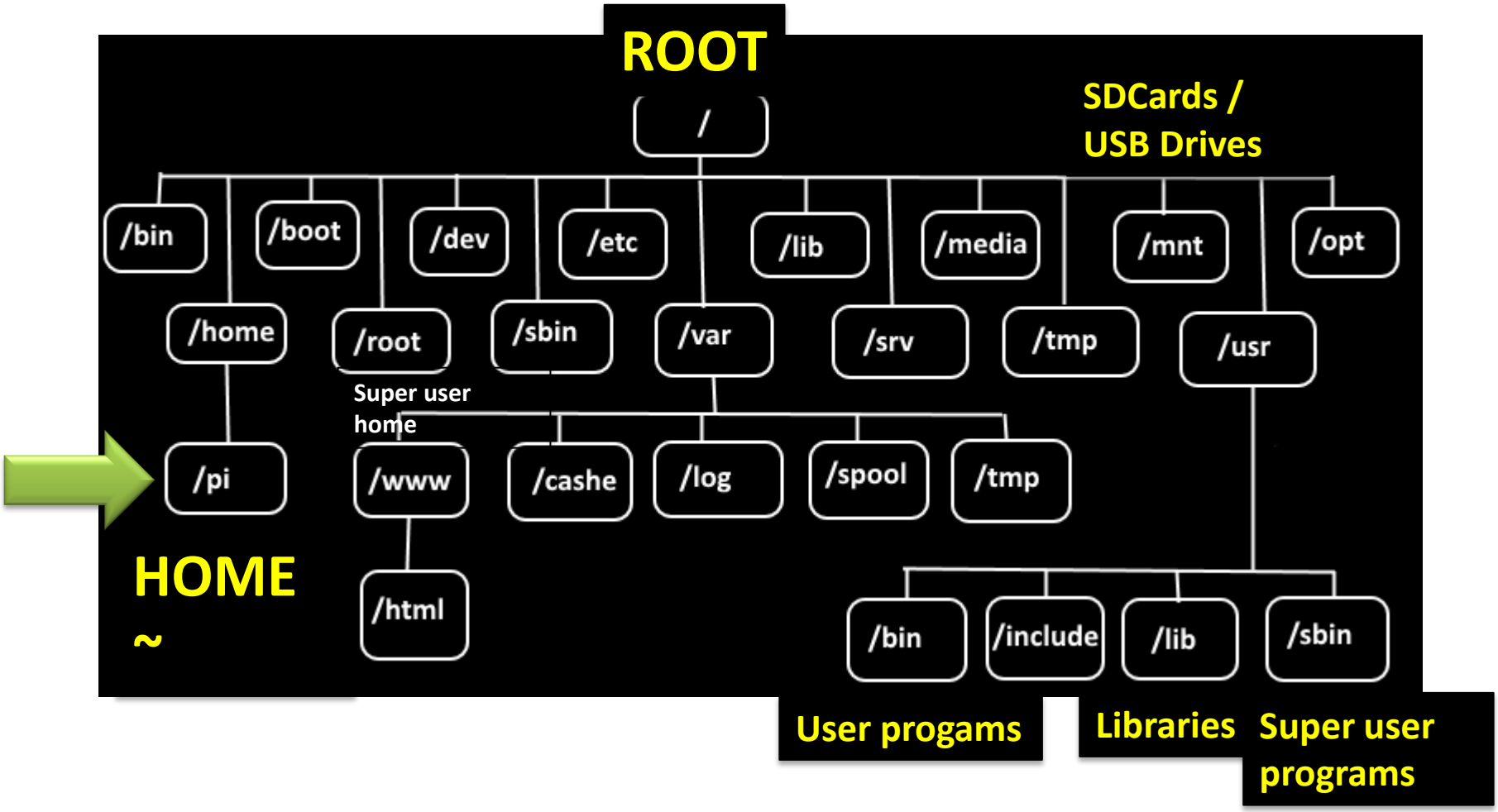
We created a folder on the desktop!



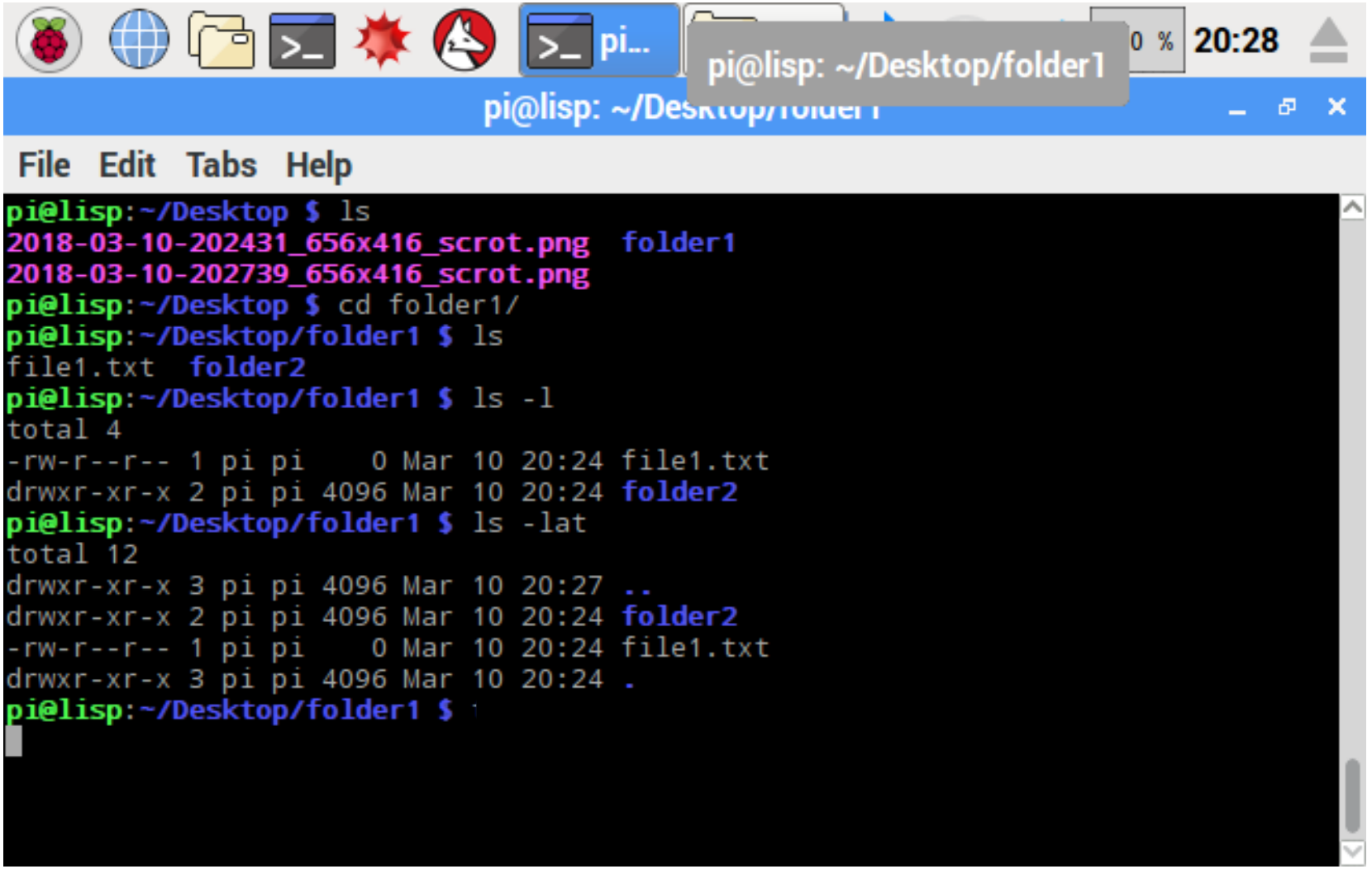
Let‘s look inside with the File Manager



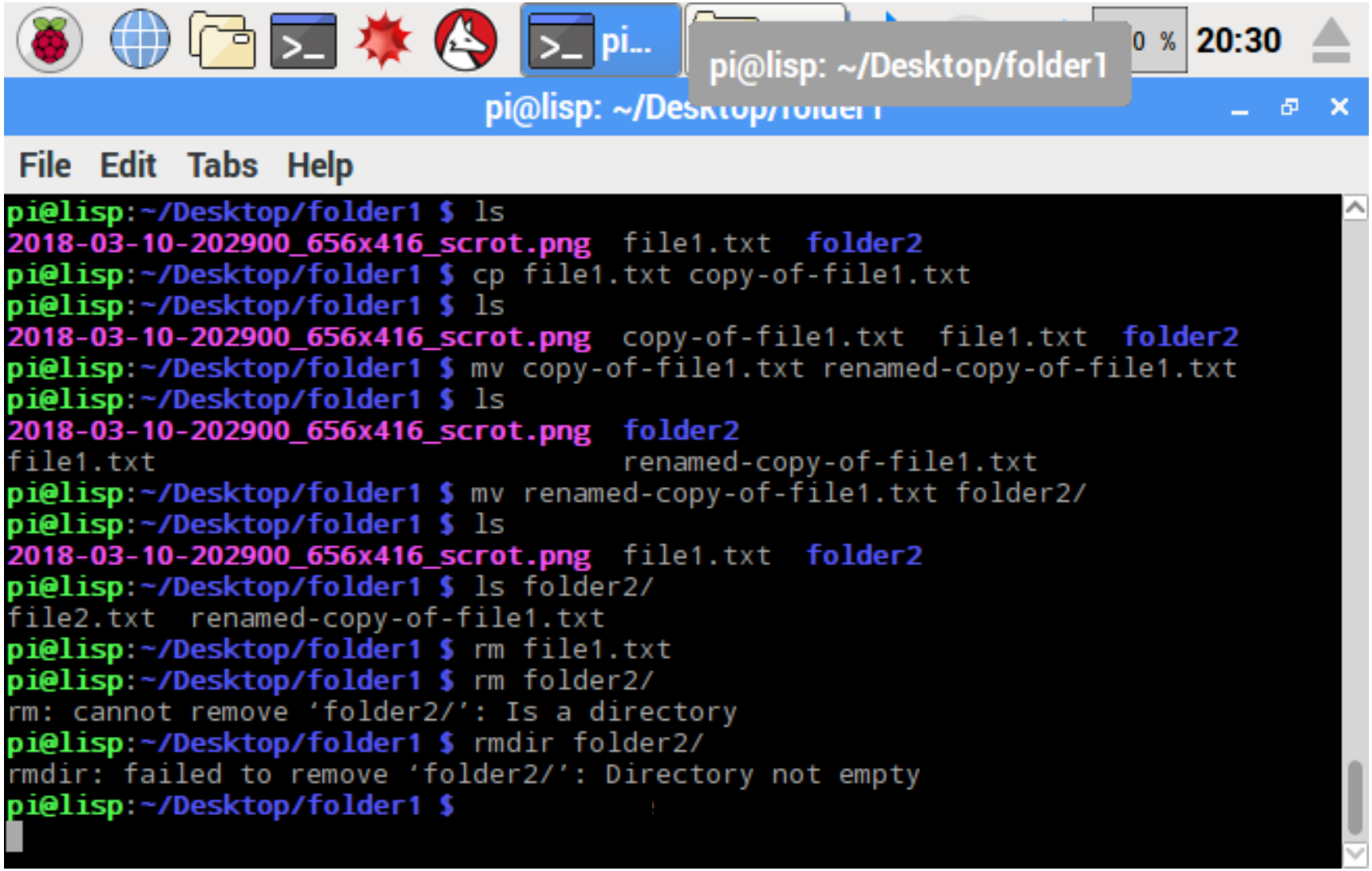
Raspbian Filesystem Tree



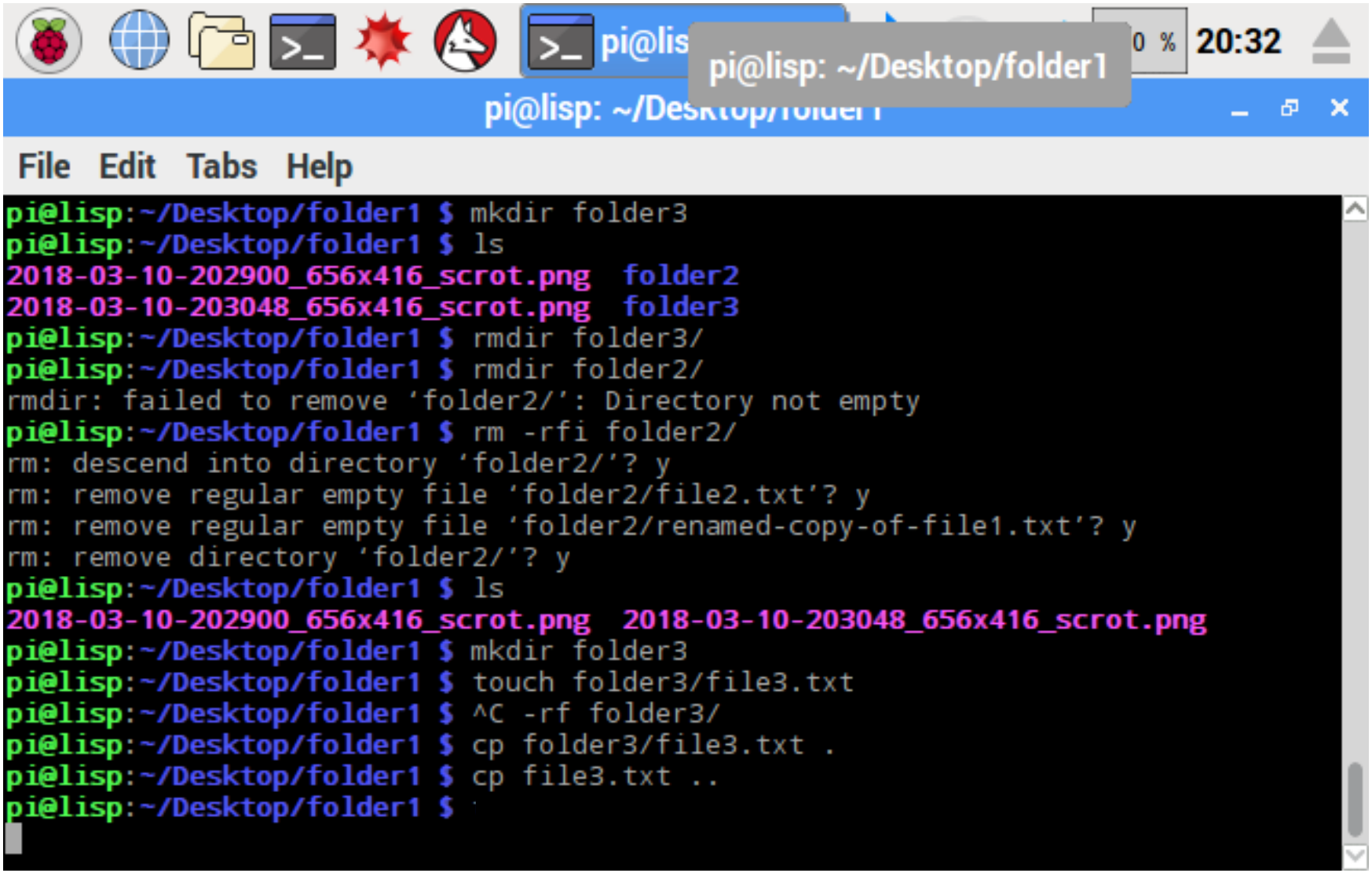
What‘s in a directory?
„ls“ to „list files“ – Files have Attributes



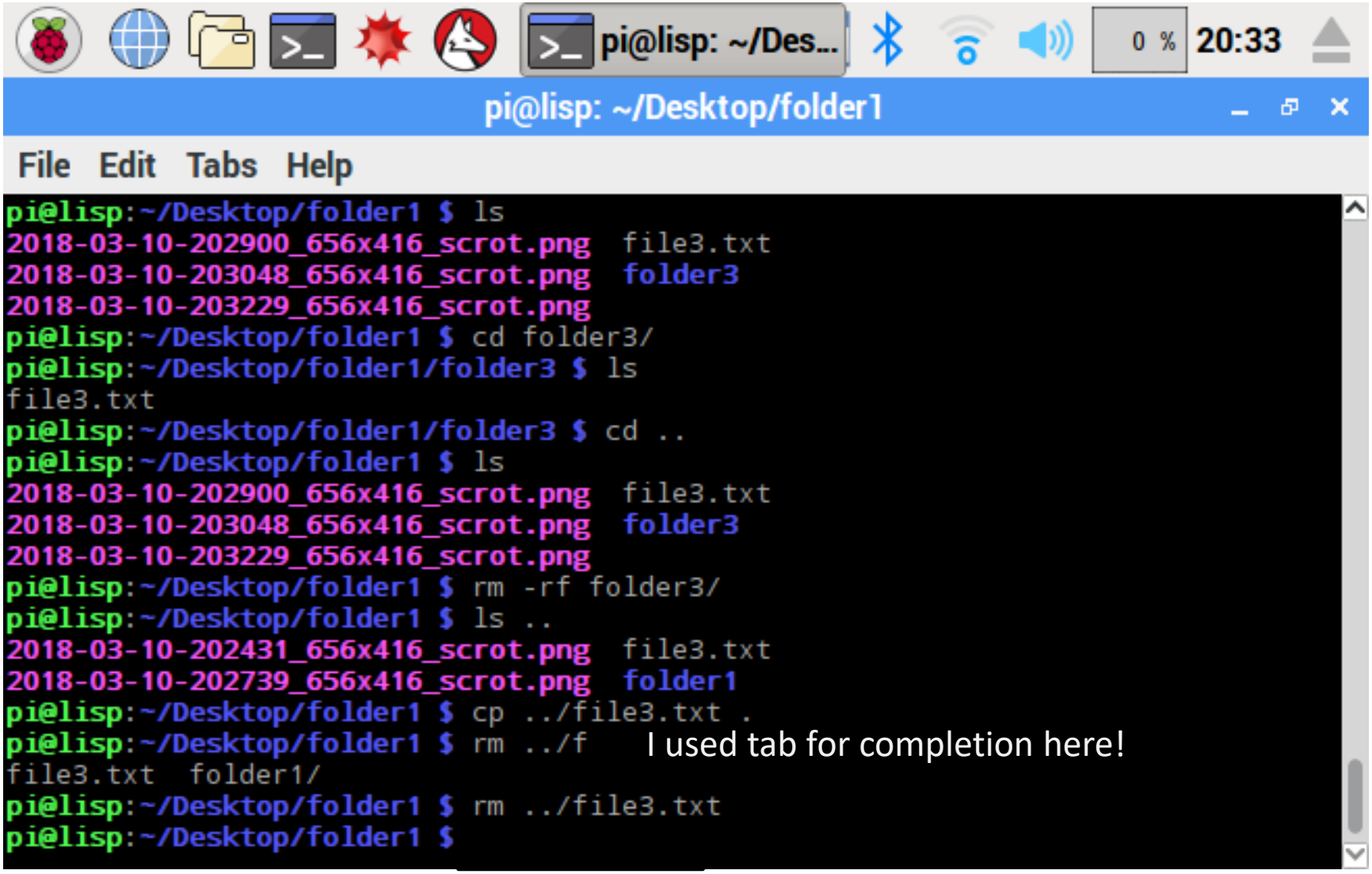
copy (cp), move (mv), remove (rm), remove directory (rmdir)



Deleting directories – „rmdir“ and „rm –rfi“

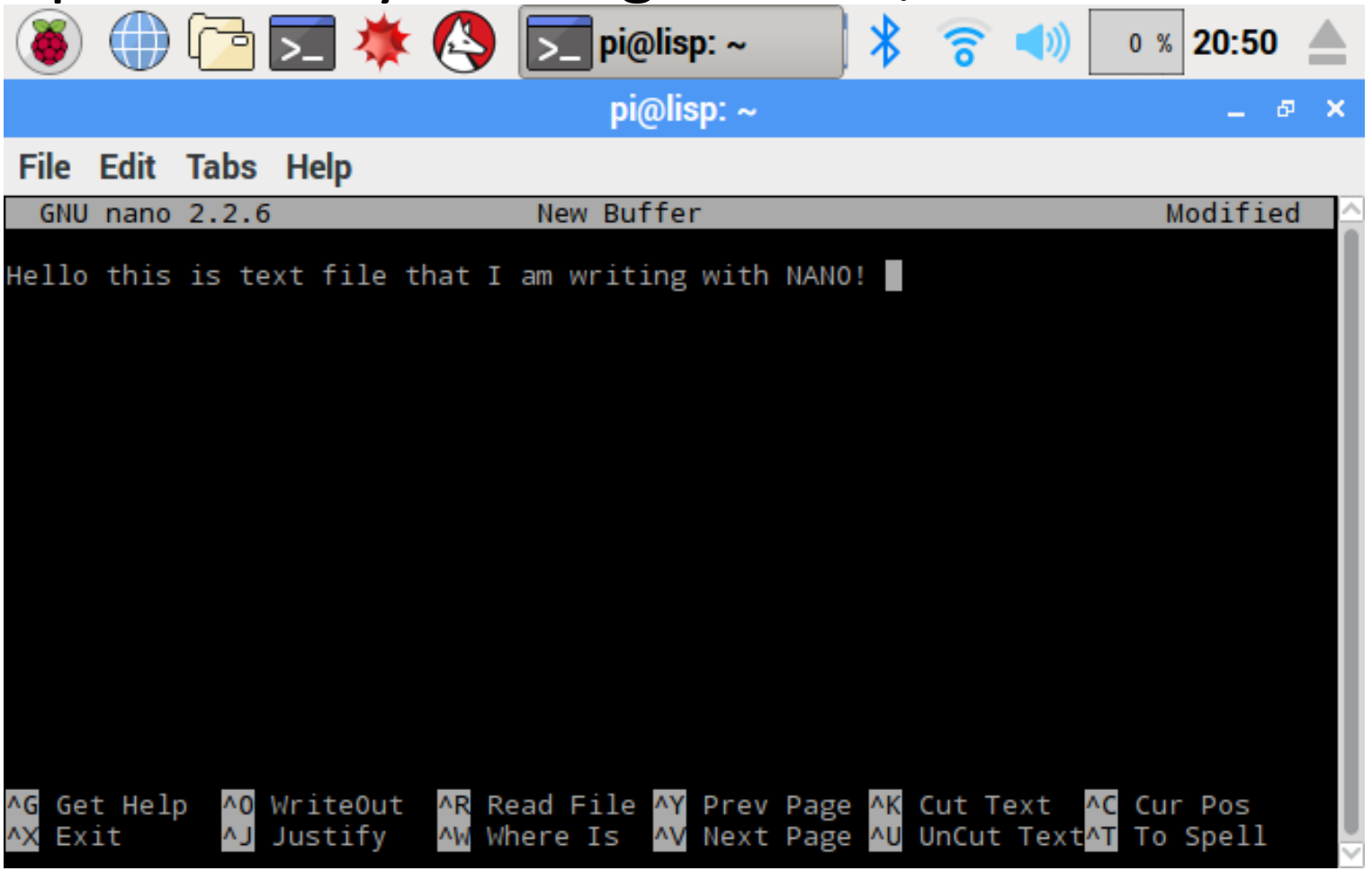


Using „.“ and „..“ in commands



NANO Text Editor

- Start with „nano“ – from the Shell
- Important key bindings: Ctrl-X, etc. See below:

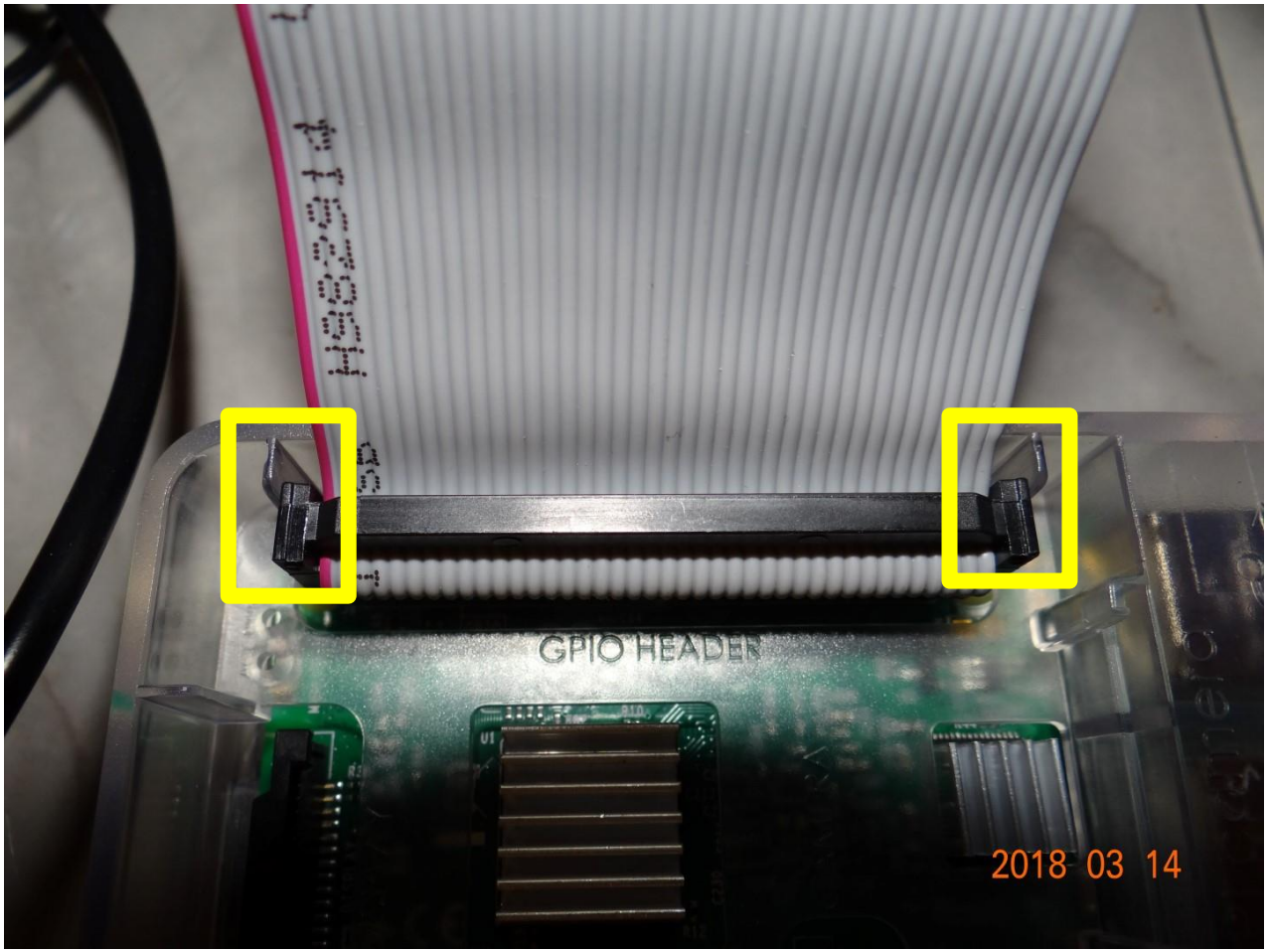


Other Useful Commands

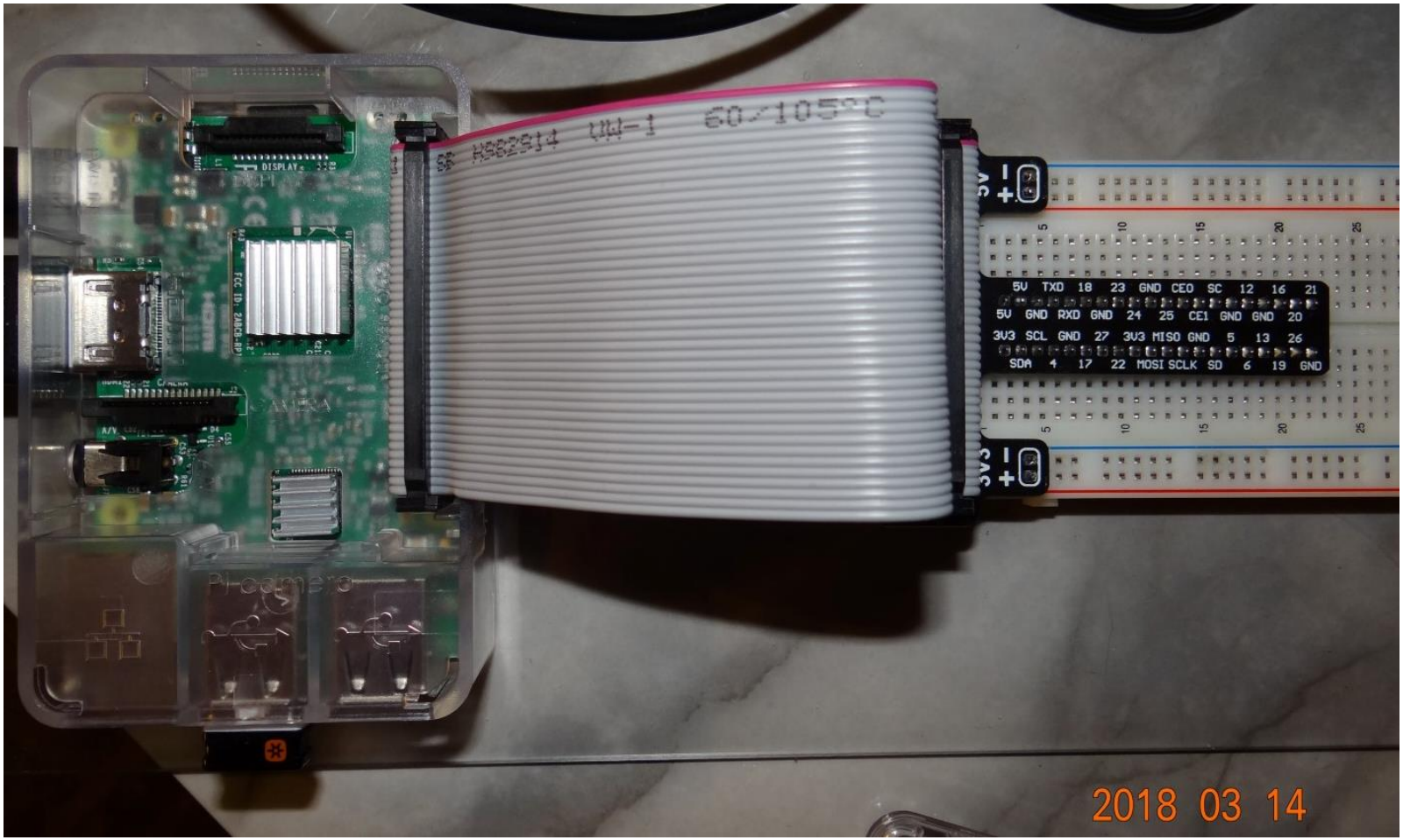
- Space free on disk:
  df –h
- Look at a text file:
  cat file1.txt
  Better: use **more** program if more than one screen! Use „q“ to quit **more**
- Run a program as „super user = root“:
  sudo <command here>
  E.g., to reboot or halt RPi:
  sudo restart, sudo halt
- **sudo** also needed for Raspberry Pi updates!

Plug in Breadboard Cobbler

Notice !

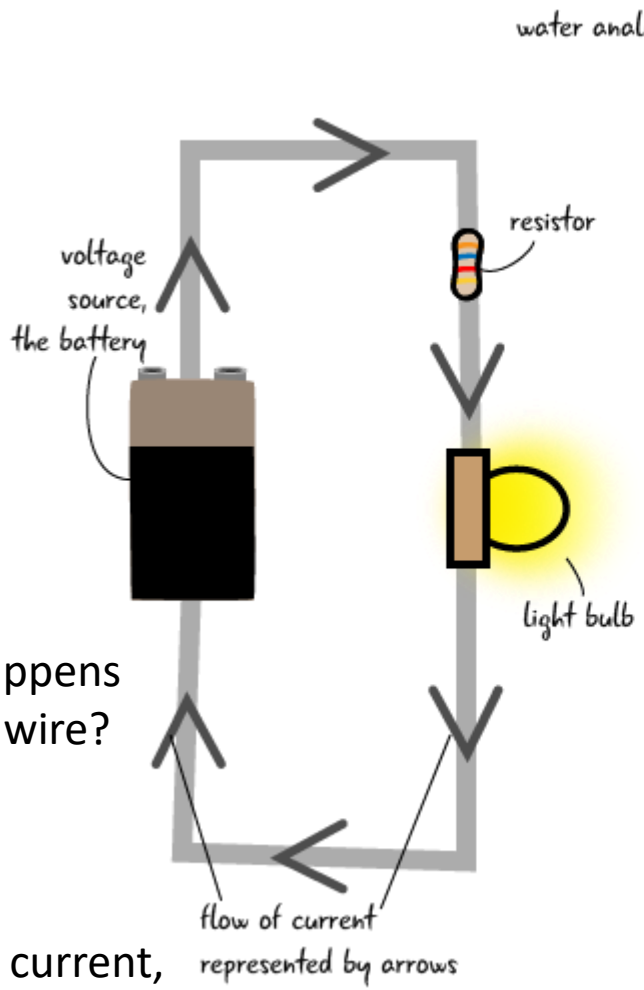


Plug in Breadboard Cobbler



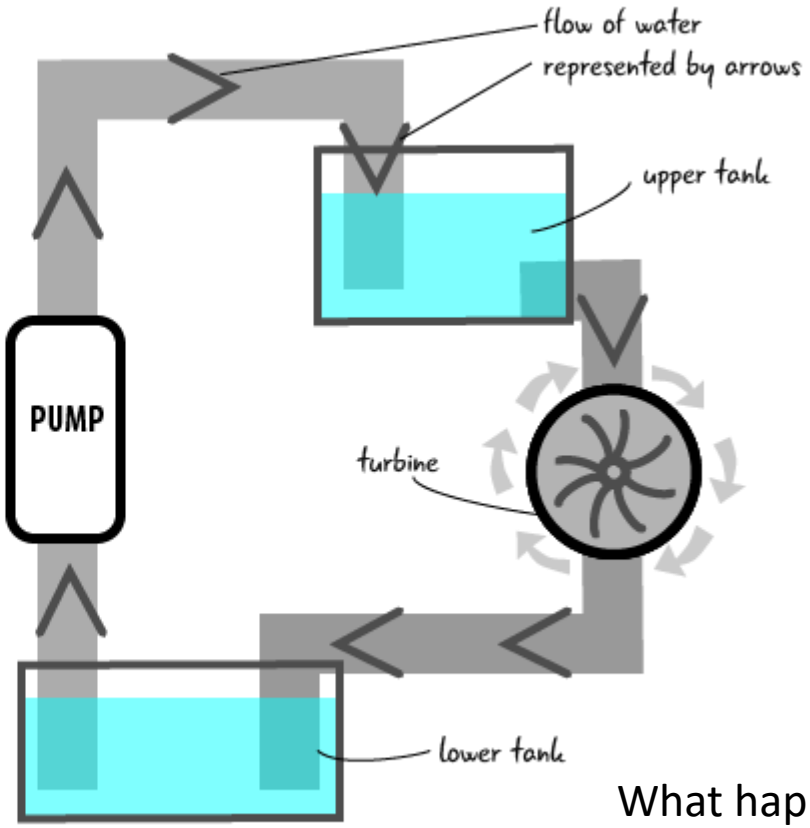
Lid on Top Fits - Gap for Cable

Electric Circuit & Water Analogy



What happens if I cut a wire?

flow of current, flow of charges, flow of electrons caused by battery = voltage source

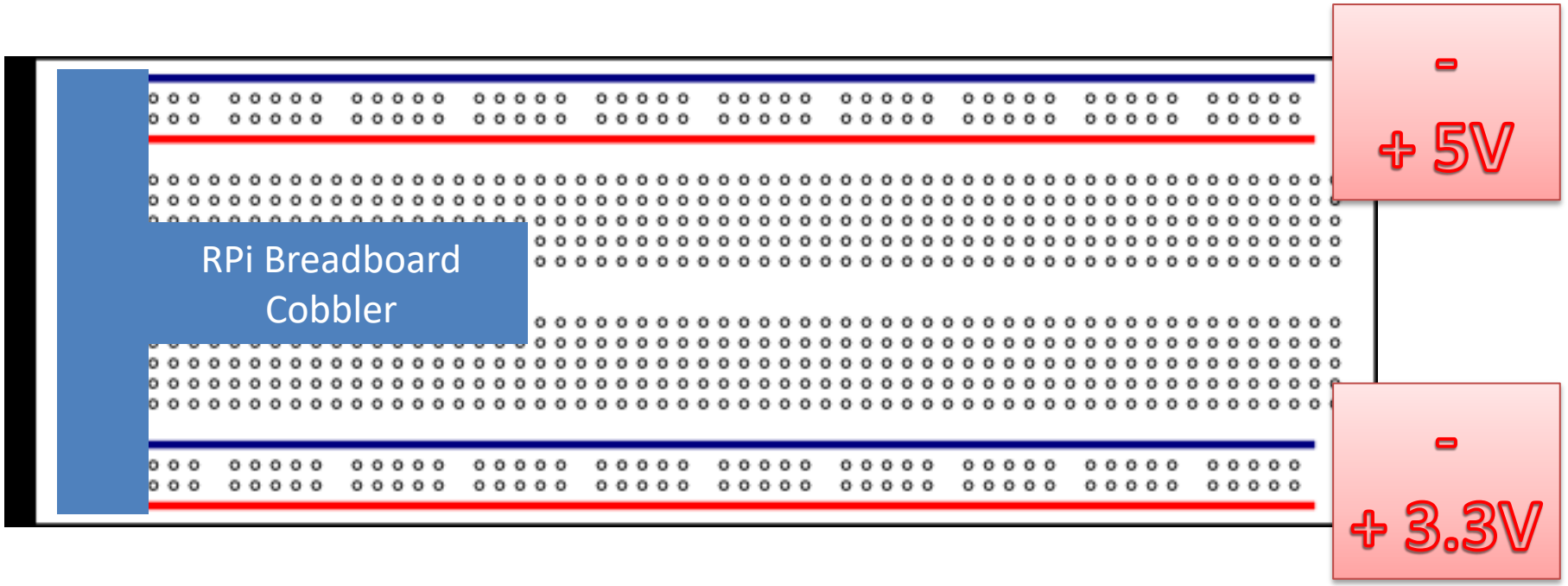


What happens if I block the pipe?

flow of water caused by pump

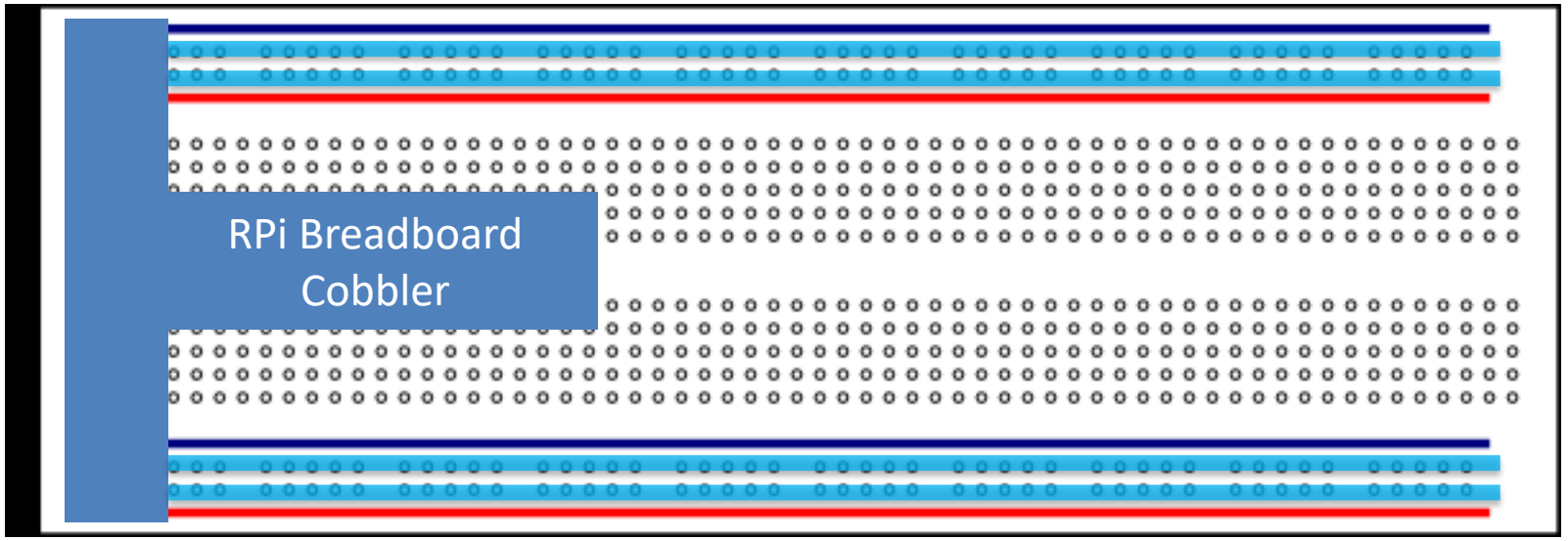
Raspberry Pi

- BLUE = „MINUS“ or „GROUND“ (GND), 0V
- RED = „PLUS“, 3.3 V or 5.0 V



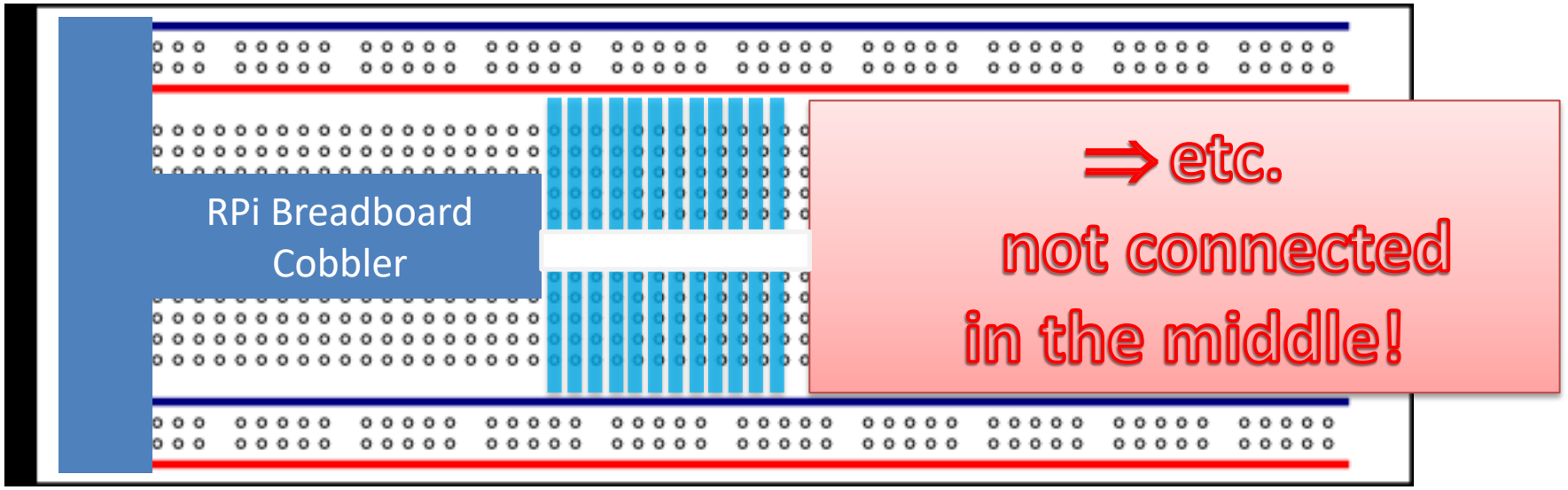
Row Connections

- The holes on the power / voltage lines are connected horizontally as shown



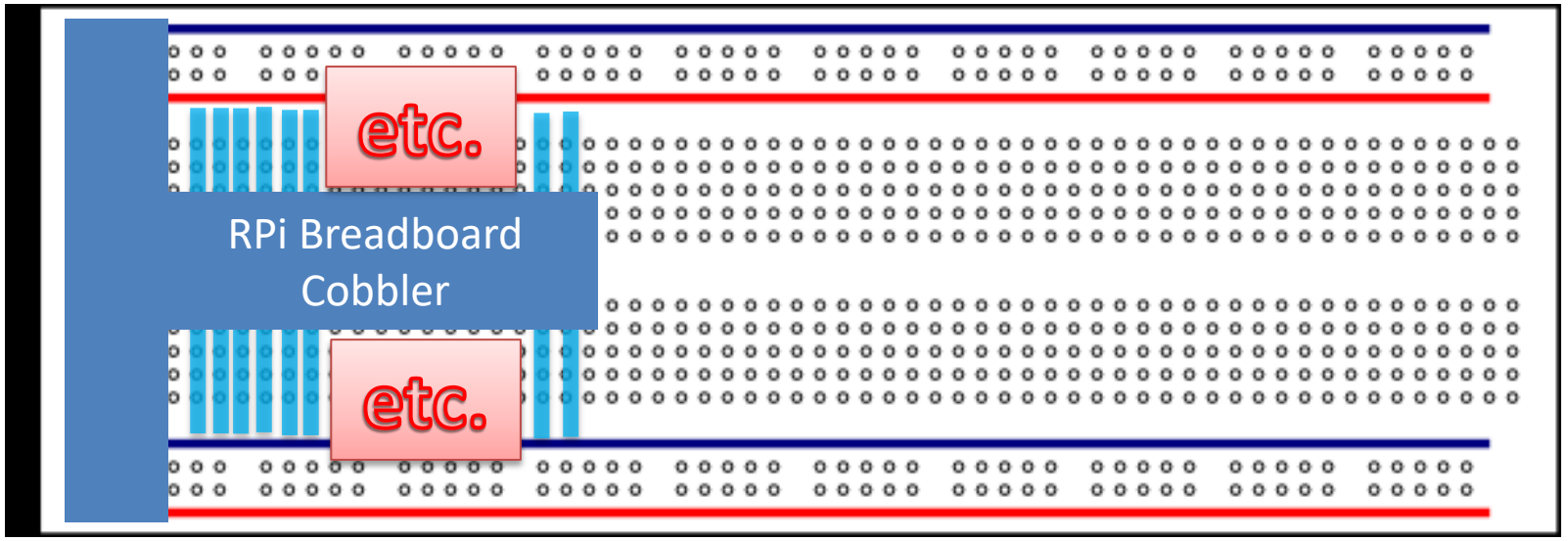
Column Connections

- The other holes are connected vertically as shown, throughout the board



Connections to Rpi GPIOs – Use Later

- This is how we connect to the GPIOs of the Rpi
- For now, we only use the RPi as our „battery“



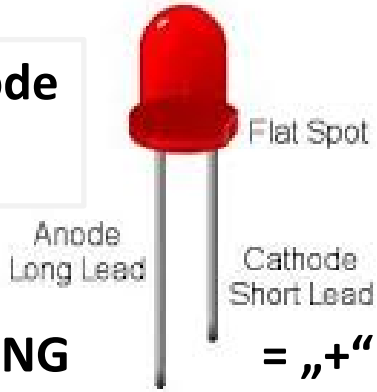
Careful with GPIO Ports !

- **Never**
  - **Connect 5 V or 3.3 V directly to MINUS (GND)**
    Connect PLUS directly to MINUS (GND)

    -> SHORT CIRCUIT
  - Do not connect output directly to 5 V or 3.3 V
  - Do not „draw“ more than 50 mA current
    (eg, do not try to connect a motor to a GPIO pin!)
  - A GPIO pin can ONLY power a LED or relay or the like, not much more
- And careful with static electricity / discharge!

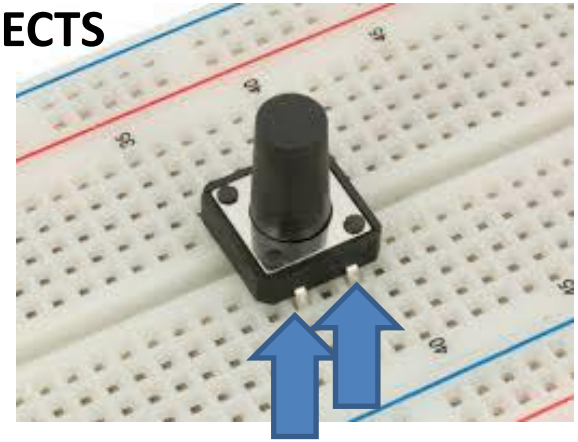
Let‘s Make a Circuit – LED & Button Components Needed

LED = Light Emitting Diode



ANODE = „+“ PIN = LONG = „+“ = MORE
CATHODE = „-“ PIN = SHORT = „-“ = LESS

„MOMENTARY PUSH BUTTON“
WHEN PUSHED, CONNECTS THE TWO PINS



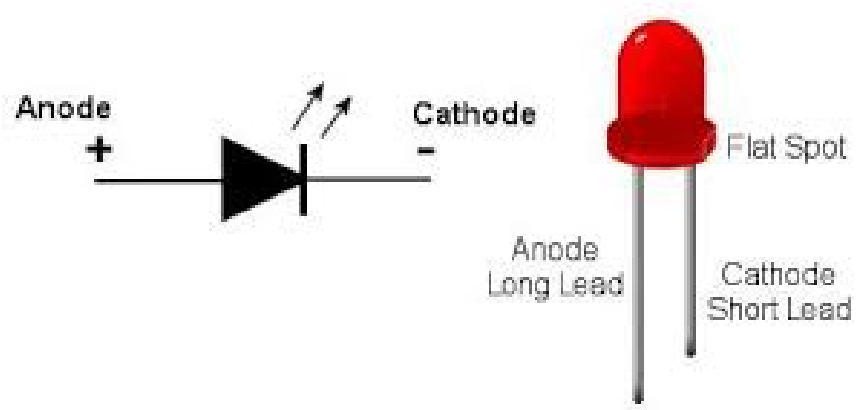
RESISTOR = LIMITS CURRENT

PROTECTS OUR LED !
LED NEED TO HAVE ~ 1.3 V
OTHERWISE IT BURNS OUT IMMEDIATELY

We need one from the 10 Resistor set! (220 Ohm)

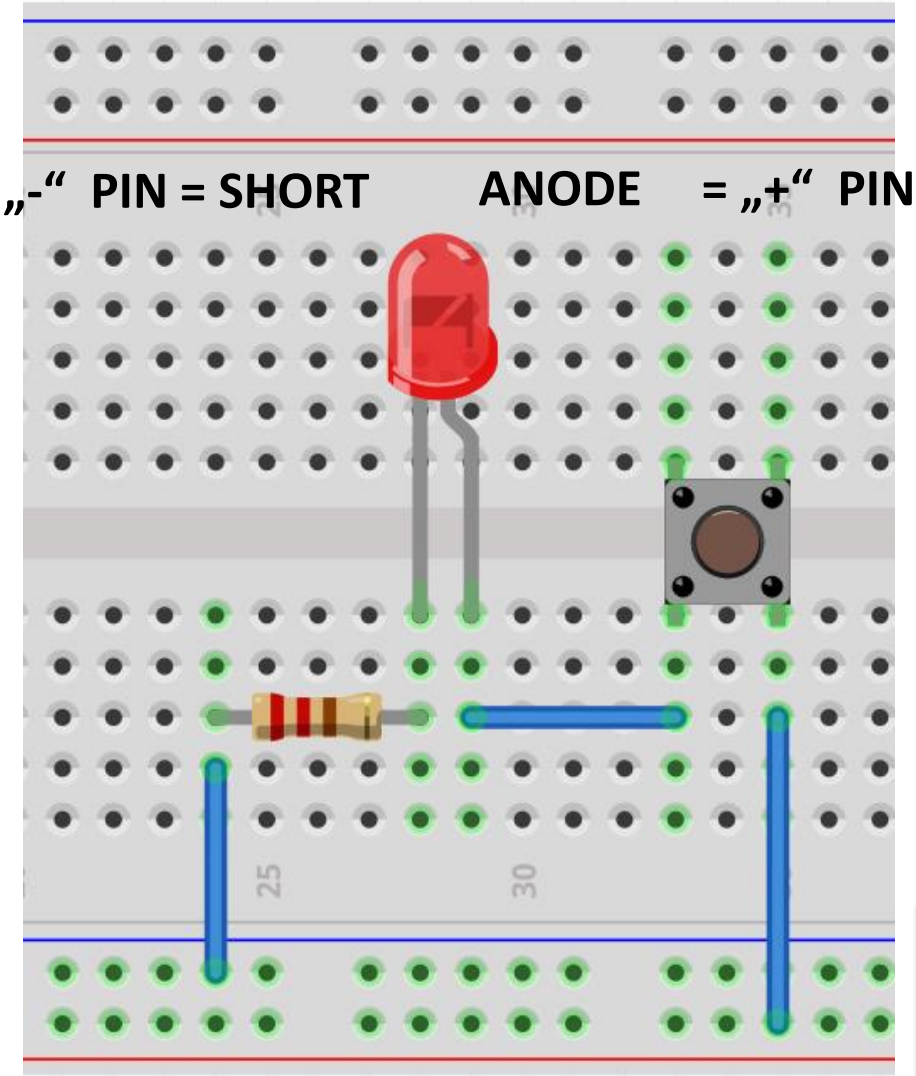
The LED is a „One Way Road“…

- … for electricity / electric current / charges / electrons !
  - Current flows from ANODE to CATHODE
  - In direction of the ARROW
  - From „PLUS“ (= LONG) to „MINUS“ (= SHORT)



Solution

CATHODE = „-“ PIN = SHORT ANODE = „+“ PIN = LONG



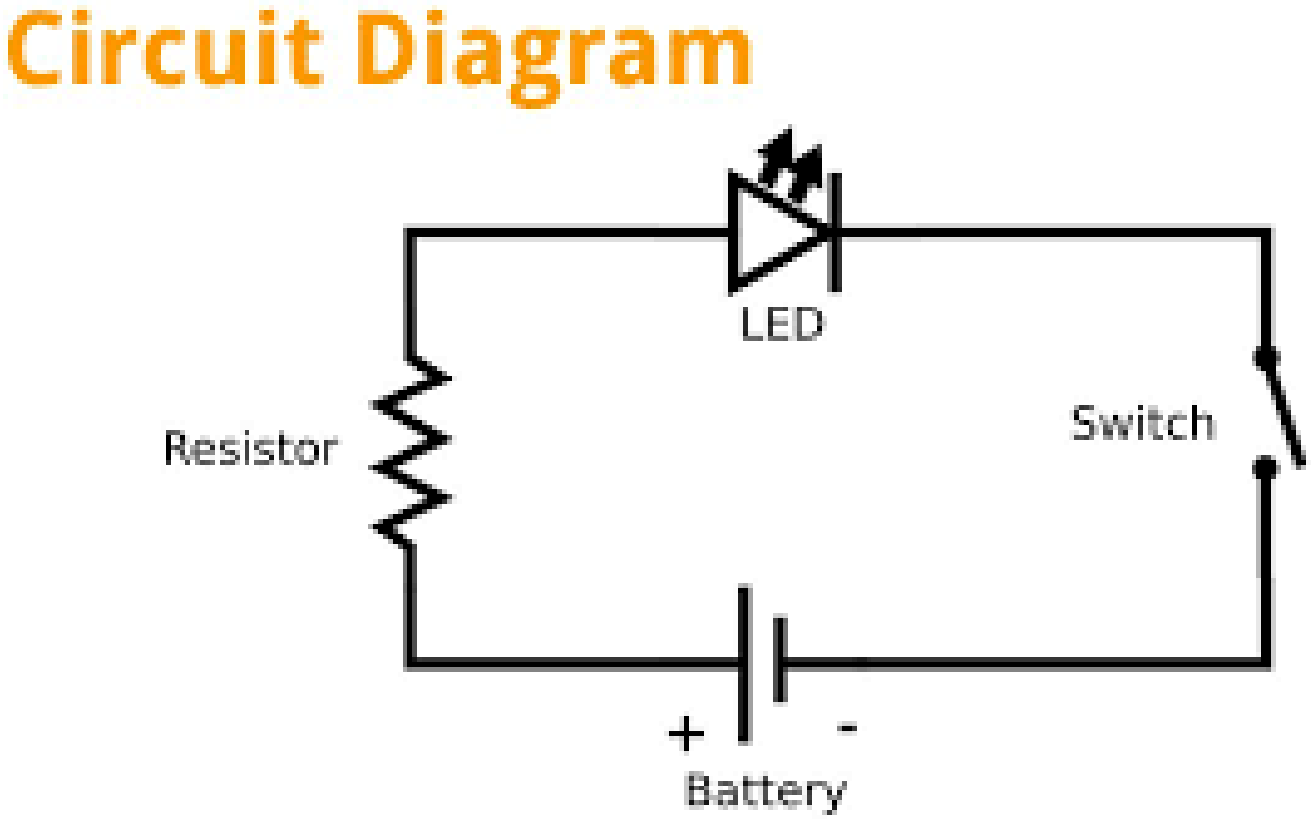
MINUS = BLUE = GND

PLUS = RED = 3.3 Volt

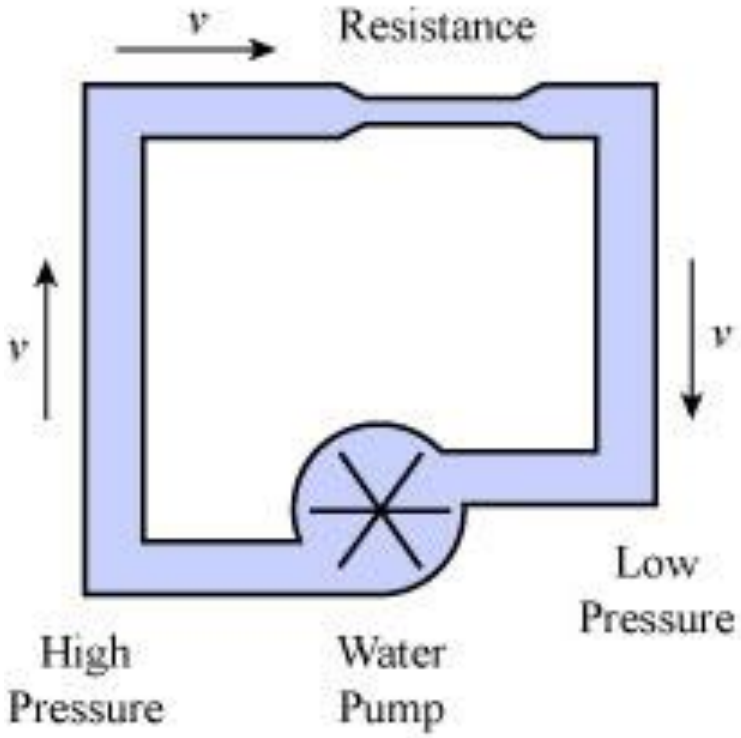
-

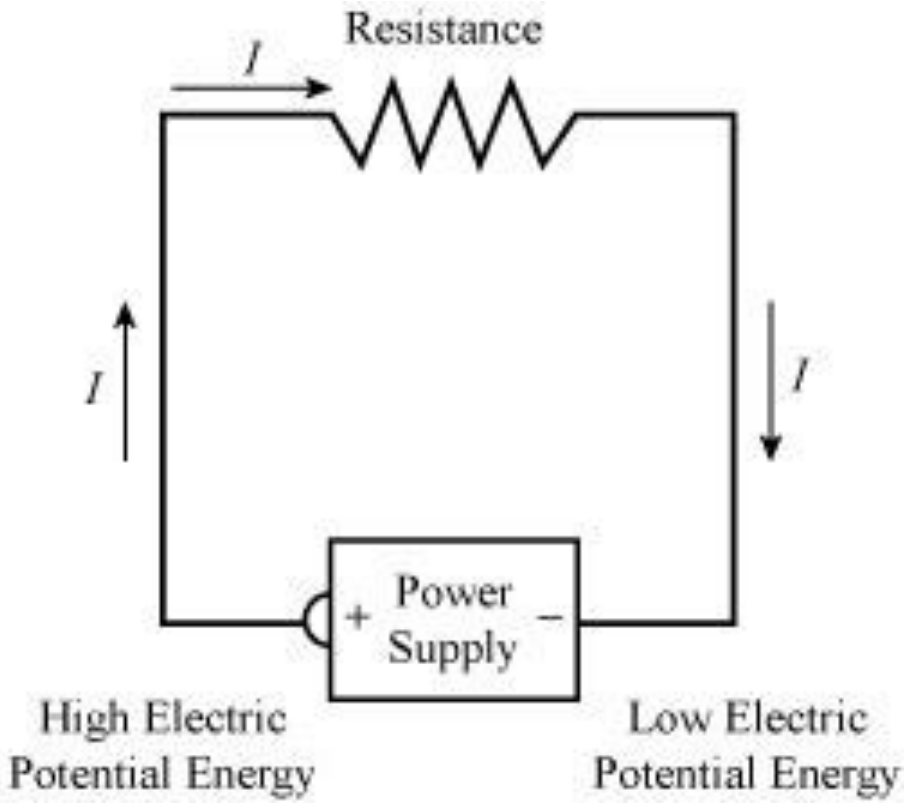
+ 3.3V

What‘s different in ours?
Does it matter?

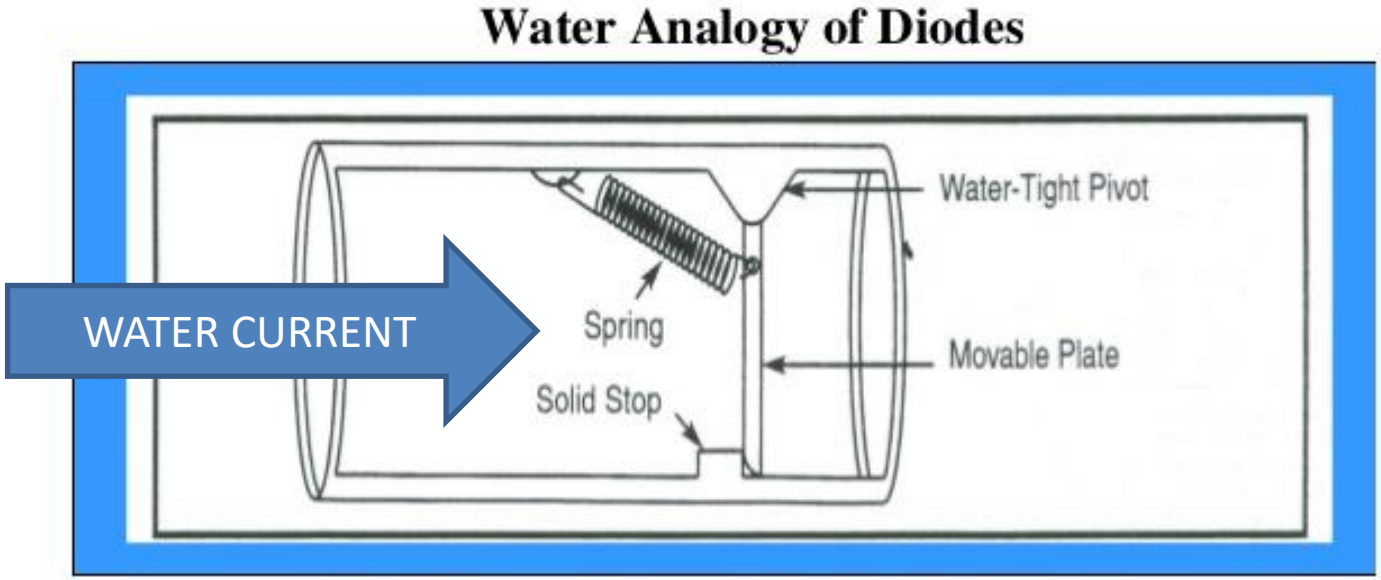


Water Analog for Resistor





Water Analogy for Diode



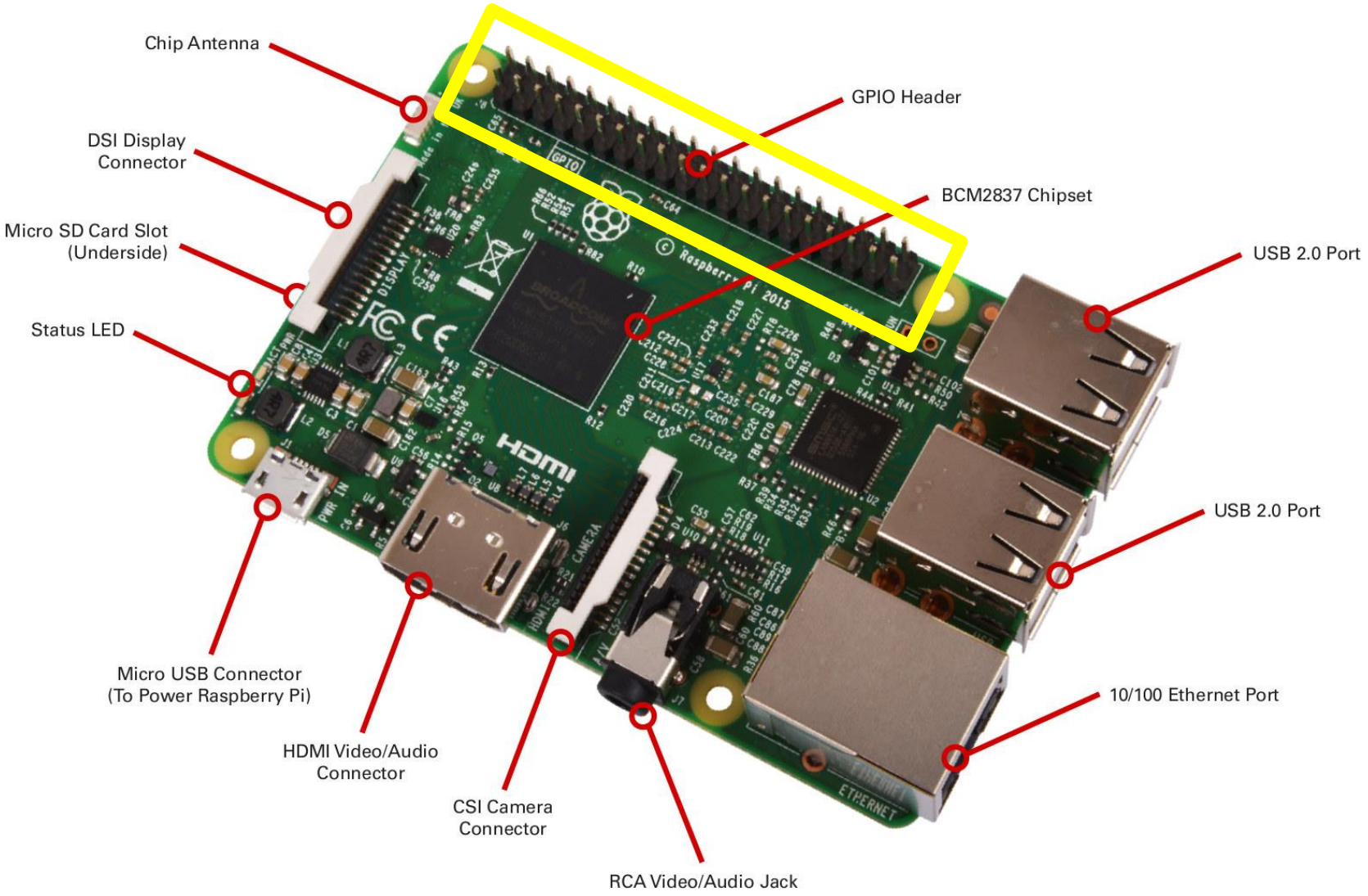
- When water pressure on left overcomes the restoring force of spring, the gate is opened and water is allowed to flow →.
- When water pressure is from right to left, the gate is pressed against the solid stop and no water is allowed to flow.
- Spring restoring force is analogous to 0.6V needed to forward bias a Si diode.

Experiment… What happens when we use 5 V instead of 3.3 V ?

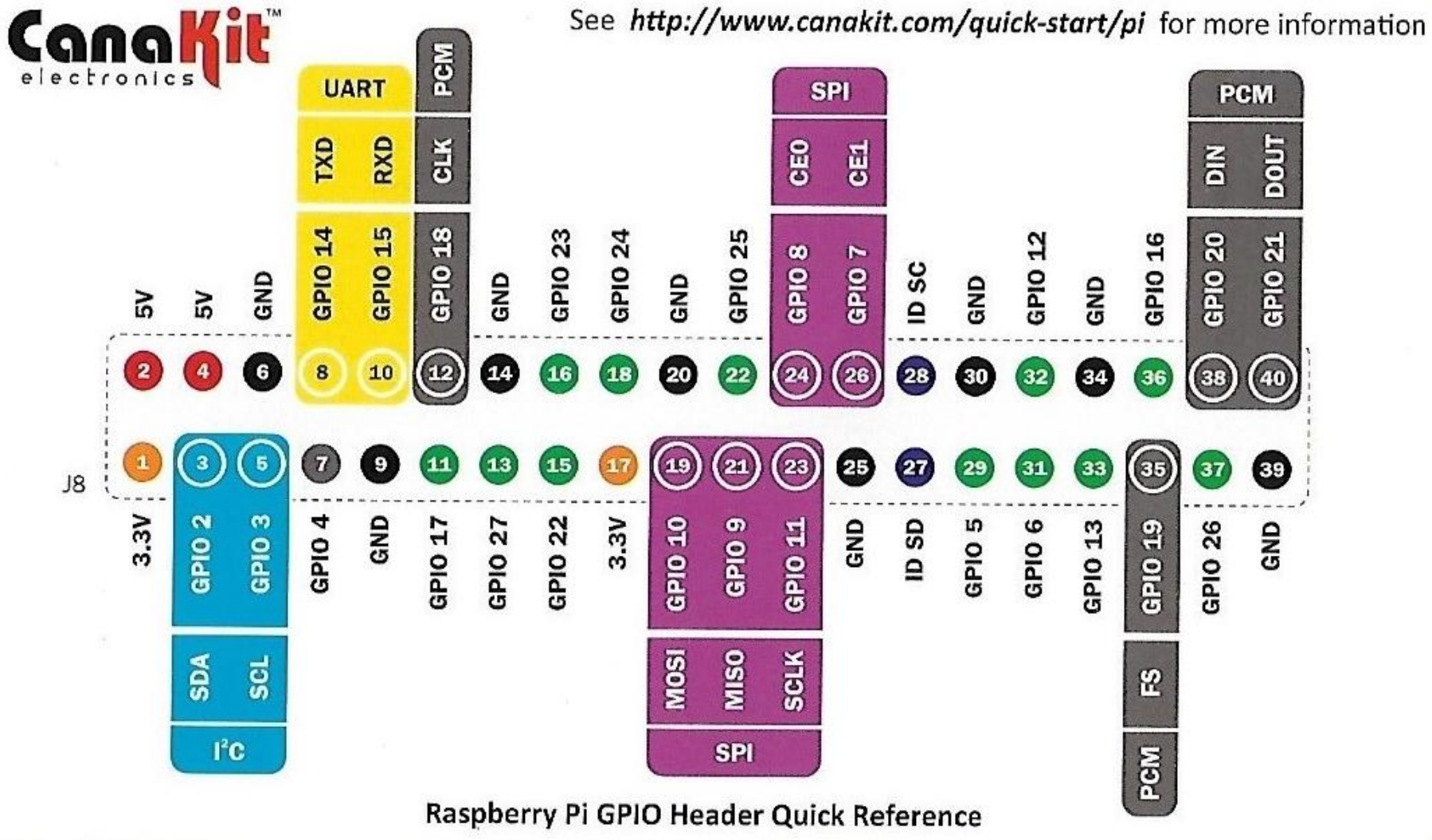
Answer

- Nothing!
  - There is no change in brightness of the LED
  - Unlike a light bulb
  - LEDs cannot be „dimmed“ liked that
    - But there is a way…
      We will learn later how to dimm LEDs…
- LEDs are very much „binary“ = on / off = 1 / 0 = high / low devices!
  - A traditional lightbulb is more „analog“
    Here, Brightness can be changed by changing Voltage / Current
  - Not with LEDs (or only very slightly)

General Purpose Input Output Ports (GPIO)



Functions of GPIO Pins
Also see https://pinout.xyz/#



Raspberry Pi GPIO Header Quick Reference

Key Features

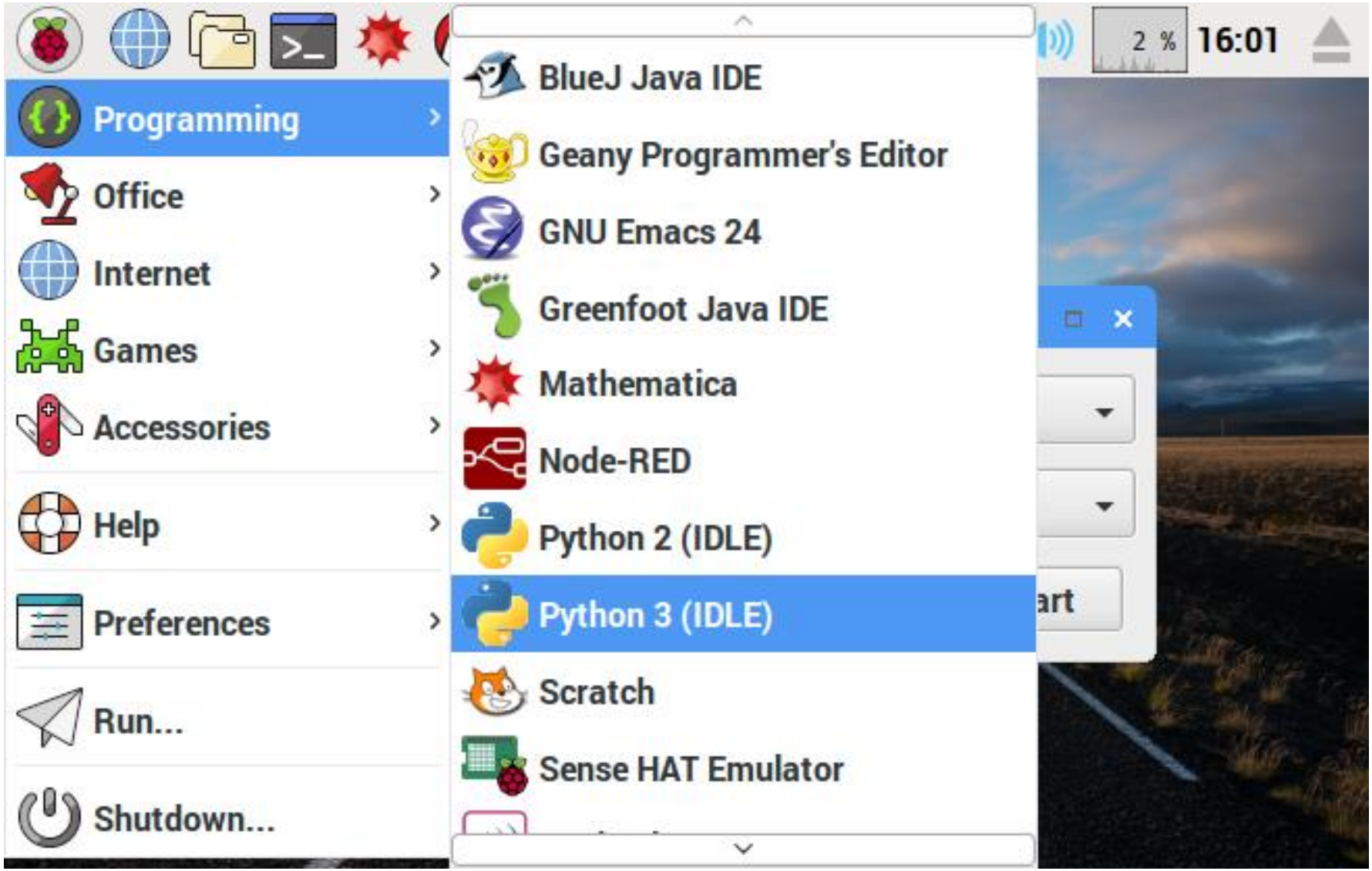
- **GPIO Port = „General Purpose Input Output“ Port**
  - **40 pin „header“**
  - **Digital inputs and outputs (direction per pin configurable)**
  - **PWM = „Pulse Width Modulation“**
  - **SPI, I2C: Indutry Standards Serial Protocols**
    - **Serial Peripheral Interface (SPI) – for Sensors, …**
      - **3 or 4 wires**
      - **Multiple devices with „select device“ signal**
      - **One master only**
      - **Full duplex**
      - **Longer ranges**
    - **I2C = I Squared C = I Two C = IIC = I^2C (Inter-Integrated Circuit)**
      - **2 wires only**
      - **Multiple devices on same bus, with chip addresing**
      - **Multiple masters possible**
      - **Cheaper**
      - **Half duplex**
  - **I2C more flexible, SPI faster**
  - **GPIOs need to be treated with care though; can be damaged by static electricity!**
- **However, no ANALOG inputs / outputs (unlike Arduino)**

Careful with GPIO Ports !

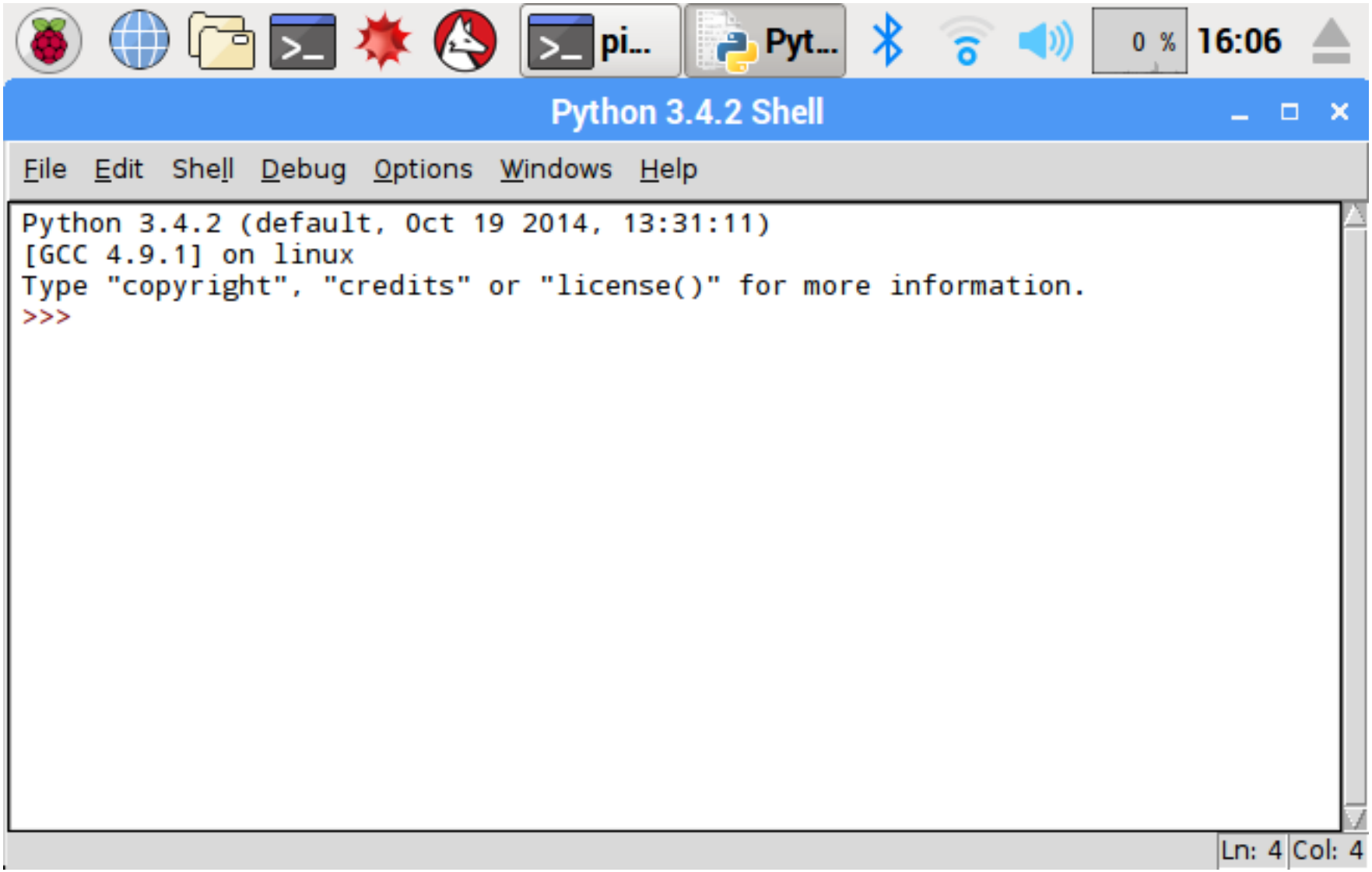
- **Never:**
  - **Connect 5 V or 3.3 V directly to MINUS (GND)**
    Connect PLUS directly to MINUS (GND)

    -> SHORT CIRCUIT
  - **Connect GPIO PIN in OUTPUT MODE directly to MINUS (GND) – if the PIN = HIGH = 3.3 V, then this is a SHORT CIRCUIT, too**
    - **It is OK if pin is set to INPUT mode, though**
    - **For OUTPUT mode, one always needs a „consumer“ – i.e., and LED and a resistor, otherwise it is a short circuit**

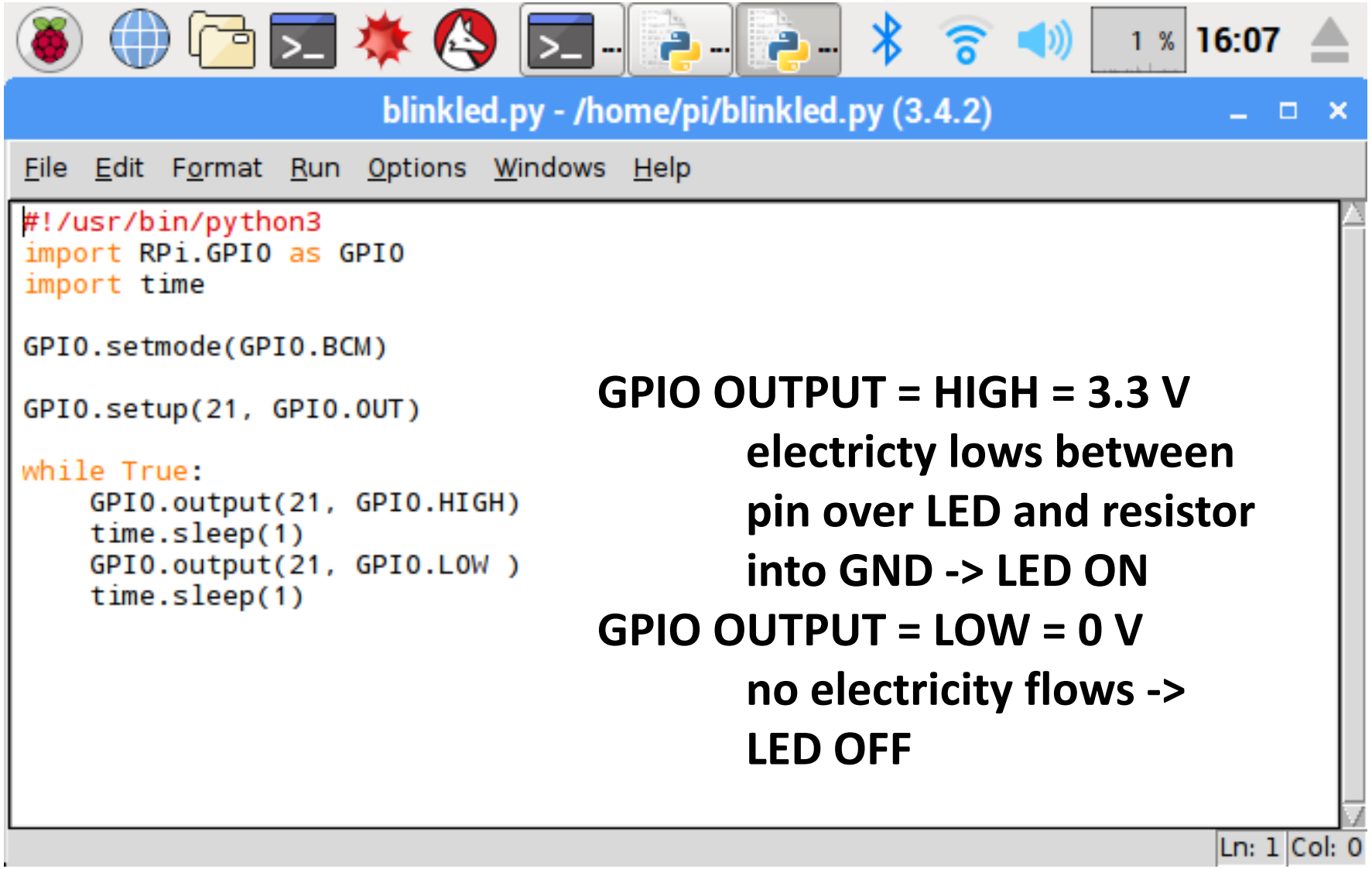
… on to Python GPIO Programming



Python Shell / Console / …



Editor – Use Function-F5 / Ctrl-c



GPIO OUTPUT = HIGH = 3.3 V
electricty lows between pin over LED and resistor into GND -> LED ON
GPIO OUTPUT = LOW = 0 V
no electricity flows -> LED OFF

Improvements

- Clean up source code
  - Write a couple of functions, „initialize“, „led_on“, „led_off“, „main“
  - Write a „main“ function that takes an argument
    - Time delay in milli seconds
    - Read in delay from keyboard using: **delay = int(input(„Delay?“))**
- Also, use GPIO.cleanup() at the end

```
try:
    <code>
finally:
        GPIO.cleanup()
```