

 Super Fast Shipping  
from just £2.99

# Maker Advent Calendar Day #1: Getting Started!

By The Pi Hut • Nov 30, 2022 •  50 comments

Welcome to the first day of your [12 Projects of Codemas Advent Calendar](#). Excited? We are too!

This twelve-day advent calendar will teach you how to code with MicroPython using a Raspberry Pi Pico, using software along with a selection of fun, popular components along the way. We're going to use code to make hardware interact with the real world!

Each day we'll show you how to use the fun things you find in each box, explaining what they are, what they do, how to construct the circuit and then how to code them with MicroPython using

simple methods.

The best bit – once you've completed the 12 projects of Codemas, you'll have a Raspberry Pi Pico along with a great stash of components, with the knowledge and confidence to make your own projects with them!

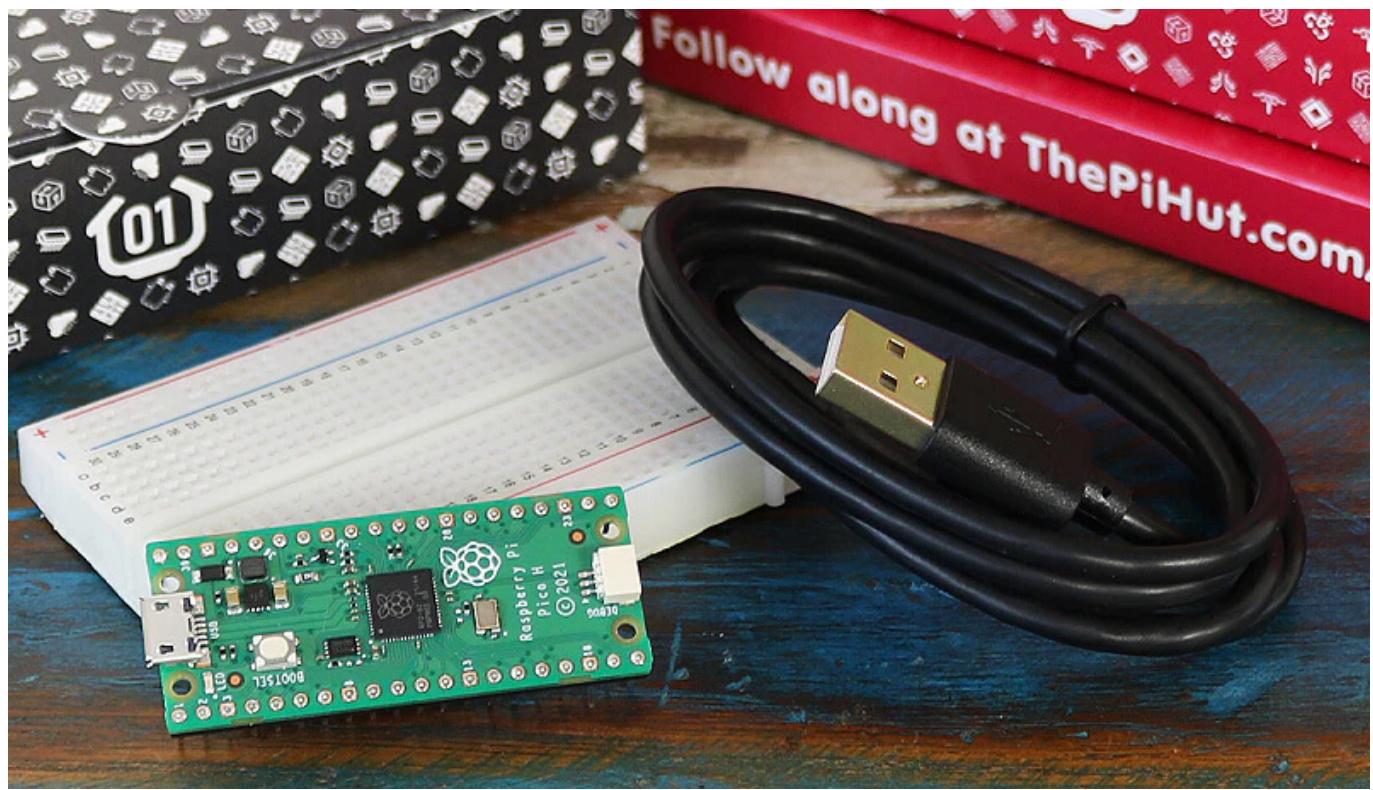
## Day #1: Getting Started

You'll have already noticed that the first box is larger than most of the others. This is to give everything you need to build the foundations for the fun projects you'll be making over the next 12 days.

### Box #1 Contents

In this box you will find:

- **1x** Raspberry Pi Pico H (with headers pre-soldered)
- **1x** 1m Micro-USB cable
- **1x** 400-point solderless breadboard



### Today's Project

Today we're going to get you set-up and familiar with the Raspberry Pi Pico, including installing the software required to program the board as well as running a couple of very easy projects to get to

grips with MicroPython.

## Introducing the Raspberry Pi Pico

The Raspberry Pi Pico is a powerful, flexible microcontroller board from, you guessed it, Raspberry Pi - the same folks who brought us the [Raspberry Pi 4](#) and [Raspberry Pi Zero](#).

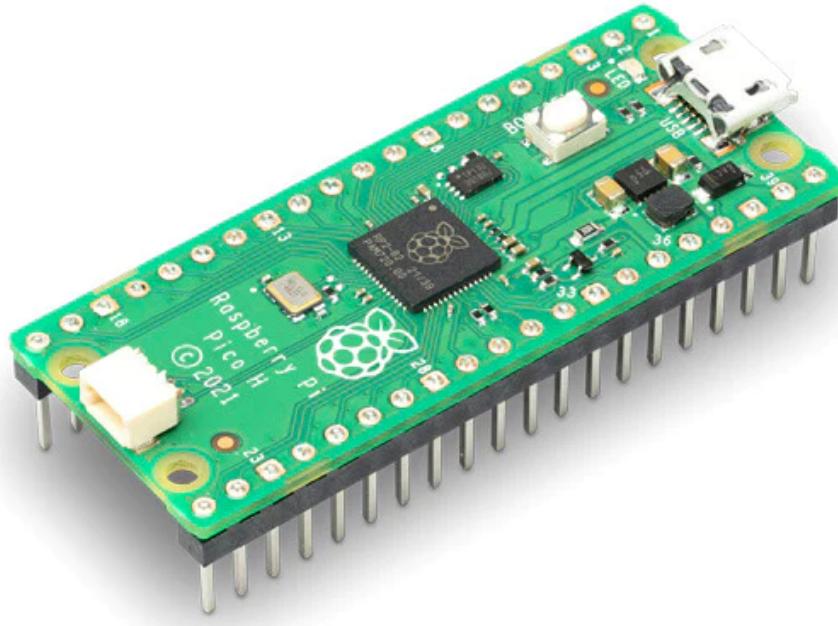
The Raspberry Pi Pico is a different kind of Raspberry Pi though - it's a **microcontroller** board, not a **microcomputer**. The brain of the Pico is the RP2040 microcontroller chip, the little black square in the middle.

Whereas you might use a Raspberry Pi 4 (*which has an operating system*) to play games, write stories and browse the web, the Raspberry Pi Pico is designed for physical computing projects where it can control components such as LEDs, buttons, sensors, motors and even other microcontrollers.

The board has lots of pins around it called GPIO (**General Purpose Input Output**) pins, which allow you to connect all sorts of components to make projects. These are controlled by code that you write. [A map of the Pico's pins is available here](#) - don't worry too much about the different pins and labels just yet, we'll be covering this over the next twelve days.

The Pico can use many different programming languages, however the most popular is MicroPython which is what we'll be using over the next 12 days.

The board has a Micro-USB port which gives it power and allows it to communicate to our computer for programming. There's a lot more to this little board, but we'll keep to the basics for now.



## Software Setup

We're going to install software called Thonny, which will allow us to program the Pico with MicroPython.

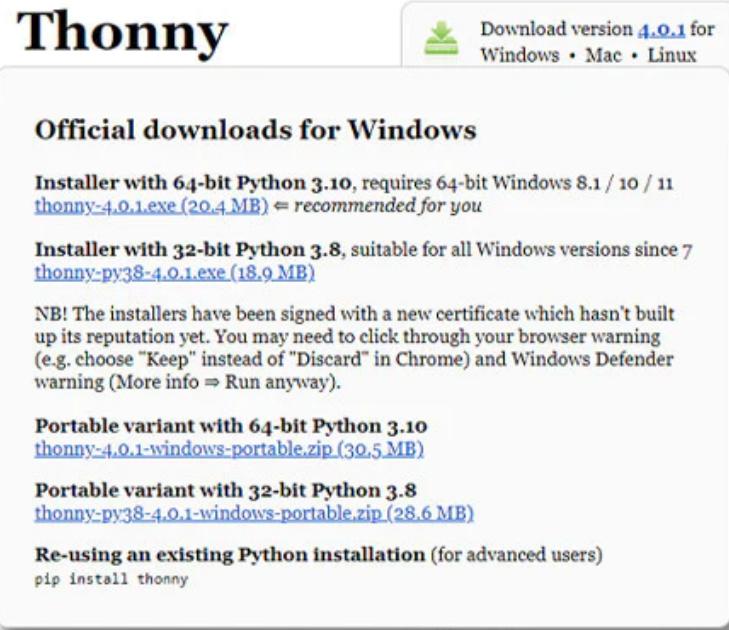
MicroPython is a smaller, more efficient version of the Python 3 programming language, designed to be used with microcontroller boards such as the Raspberry Pi Pico. If you've programmed with Python previously, MicroPython will feel very familiar - but don't worry - we're going to guide you every step of the way over the next 12 days.

Thonny is a Python IDE. An IDE is an Integrated Development Environment. IDEs help us to write code by giving us lots of helpful features to avoid errors, make our code more readable and lots of other benefits.

So, let's get Thonny! Head over to <https://thonny.org/> and look for this little box:



If you hover over the system you're using (*we're using Windows in this example*) you should see some options, with a recommended option based on the system you're using. For us, using a 64-bit version of Windows, it's recommending the 64-bit version. Your system may be different, so it's best to just go with the version the website recommends:



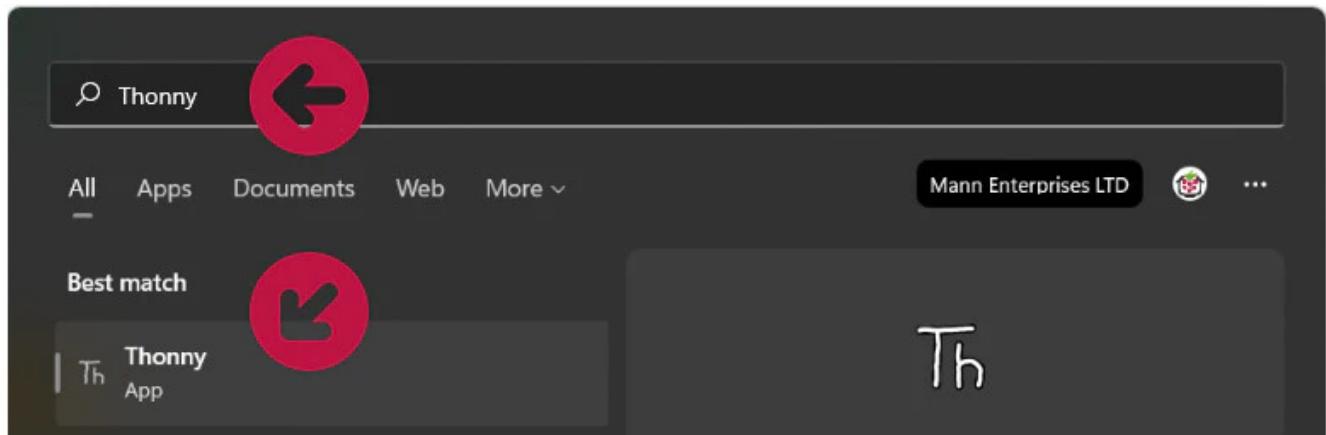
The screenshot shows the official download page for Thonny. At the top right, there's a button to "Download version 4.0.1 for Windows • Mac • Linux" with a green download icon. Below this, a section titled "Official downloads for Windows" lists several options:

- Installer with 64-bit Python 3.10**, requires 64-bit Windows 8.1 / 10 / 11. Link: [thonny-4.0.1.exe \(20.4 MB\)](#) (recommended for you)
- Installer with 32-bit Python 3.8**, suitable for all Windows versions since 7. Link: [thonny-py38-4.0.1.exe \(18.9 MB\)](#)
- NB!** The installers have been signed with a new certificate which hasn't built up its reputation yet. You may need to click through your browser warning (e.g. choose "Keep" instead of "Discard" in Chrome) and Windows Defender warning (More info => Run anyway).
- Portable variant with 64-bit Python 3.10** Link: [thonny-4.0.1-windows-portable.zip \(30.5 MB\)](#)
- Portable variant with 32-bit Python 3.8** Link: [thonny-py38-4.0.1-windows-portable.zip \(28.6 MB\)](#)
- Re-using an existing Python installation** (for advanced users) Command: `pip install thonny`

Download the correct version for your computer then open the file to start the installation.

Thonny installs like most programs, and will ask you to accept the license agreement and confirm where to install it. We kept all the options as default and it took just under a minute to install.

Once installed, select 'Finish' and then open the program. In Windows 10/11 you can select the Windows Start button and just type in 'Thonny'. The program should open and look like the example below:



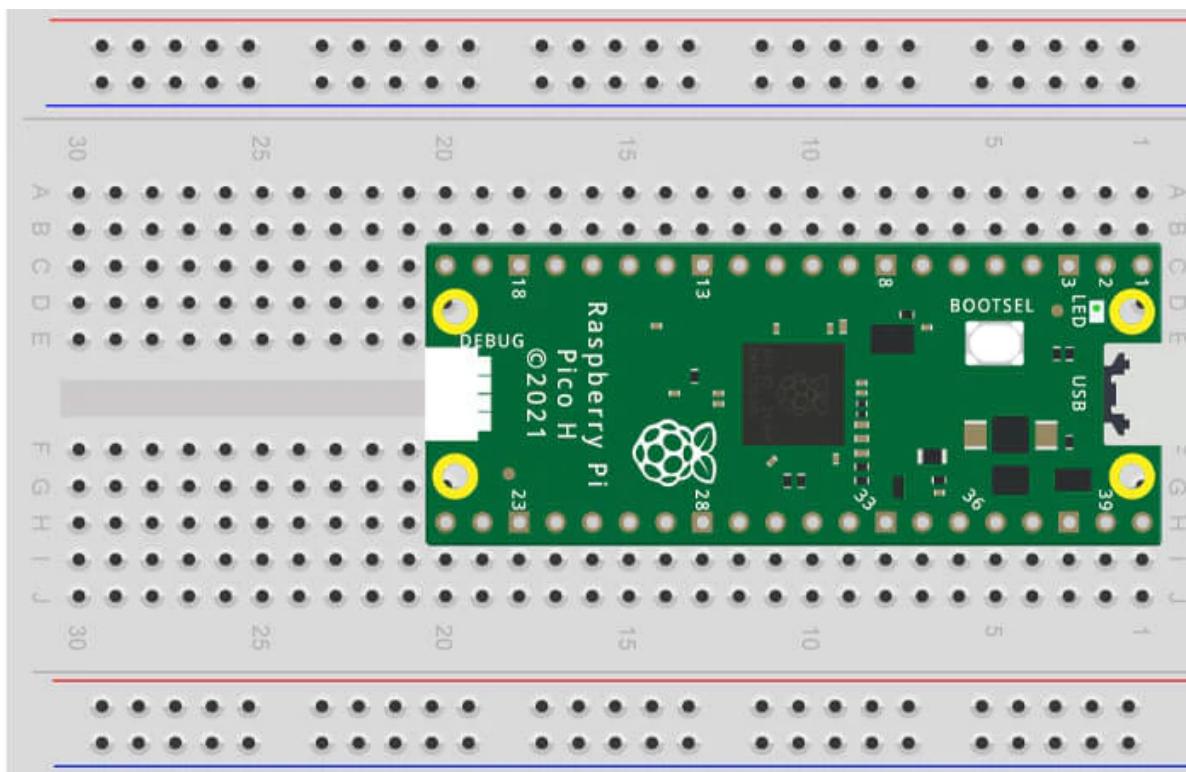
## Install MicroPython on the Raspberry Pi Pico

We now need to install the MicroPython firmware on the Raspberry Pi Pico. This gives the Pico the basic software it needs to run MicroPython code that we'll send to it later.

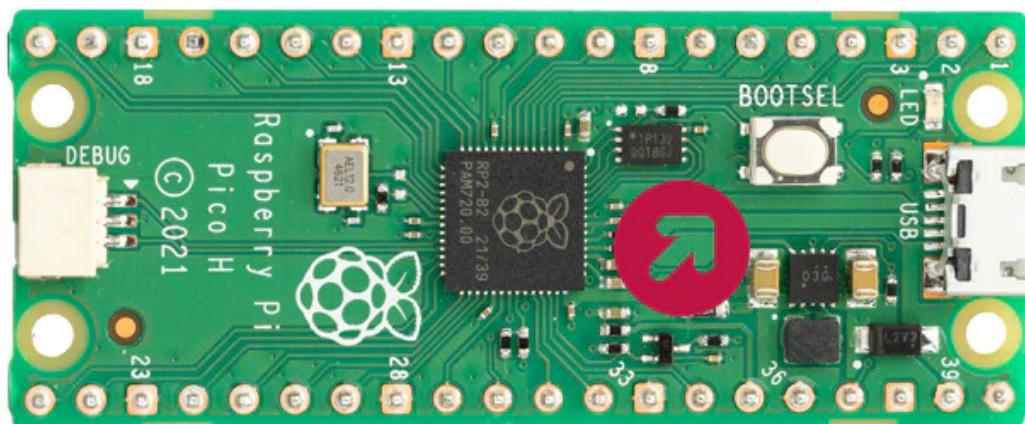
**Important Note:** A handful of customers are experiencing an issue when using the method we have suggested below (SSL errors and other unfestive nasties). Give it a try, but if you have trouble installing

MicroPython on your Pico, [try the other method here instead](#). Thanks to everyone who let us know about this issue.

Plug the Pico into the breadboard (the end where the number 1 is) then plug the Micro-USB cable into the Pico. Your Pico should be **fully inserted** so that you can't see any of the metal legs/pins. The black plastic on the underside of the Pico should sit flush with the top of the Breadboard. Apply strong pressure evenly at both ends:



Now locate the little button on top of your Pico. This is the '**BOOTSEL**' button which, when held in whilst connecting the cable, allows us to install different firmware to the Pico.

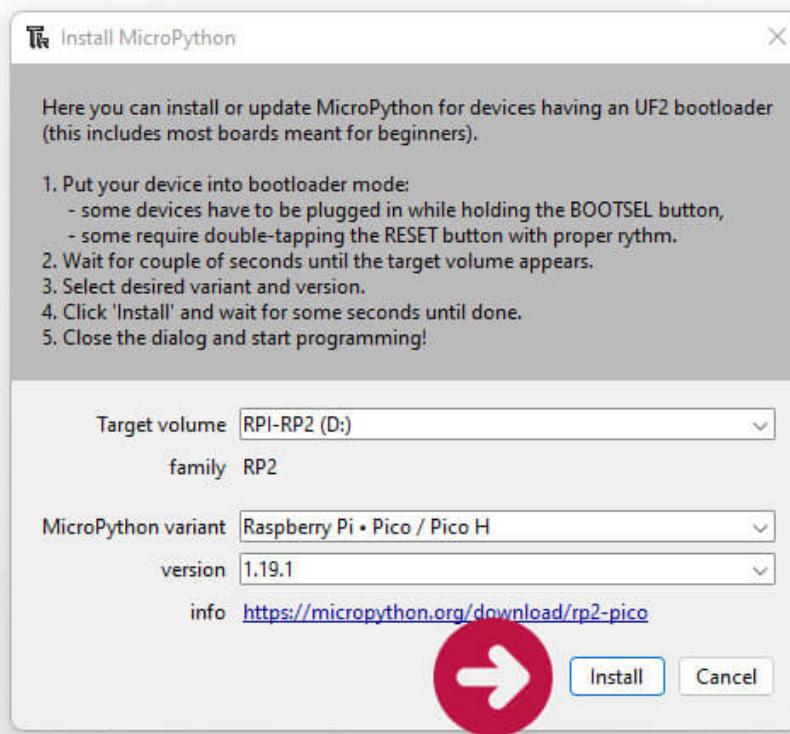


Hold down that button whilst plugging in the USB cable into your computer's USB port at the same time. You may see it pop up as a new device on your computer but just ignore that for now.

In Thonny, at the bottom-right-hand corner, you'll see something like 'MicroPython (Raspberry Pi Pico)' and a COM port, like the example below (*this may vary but it doesn't matter at this stage*). Click on this and you'll see a few options. Select 'Install MicroPython...':



You'll now be presented with some options in a new dialogue box. Use the same options as per the screenshot below, selecting the **RPI-RP2** target volume, **Raspberry Pi Pico / Pico H** variant, and whatever the latest version is available (*at the time of writing it was 1.19.1*):

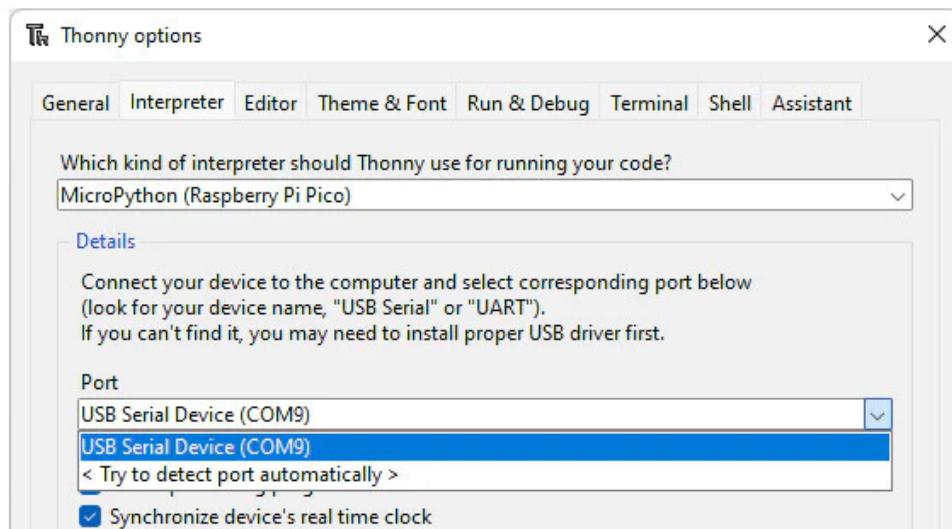


Select 'Install' and you'll see a small progress bar whilst it installs. Once complete, select 'Close'.

## Setting up Thonny

Thonny can be used with a range of development boards, so we need to tell Thonny that we are using a Raspberry Pi Pico.

From the top menu, select 'Run' then 'Configure interpreter':



Now just make sure that 'MicroPython (Raspberry Pi Pico)' is selected in the first drop-down box, select the available device in the 'Port' drop-down, then select 'OK'.

**Tip:** If you see multiple options in the Port drop-down box, it just means you have other devices connected to your computer. If you're not sure which one is the Pico, make a note of what COM ports are showing, disconnect the Pico from the USB port, close the Thonny options box, then try again. Whichever one is now missing was the Pico, so connect it again and select that device next time (it's a lot easier than disconnecting everything else!)

You should now see a 'Raspberry Pi Pico' panel on the left-side of Thonny, and information in the bottom-right 'Shell' panel.

**Tip:** If you don't see your Pico on the left, try selecting 'View' from the top navigation bar, and selecting 'Files'.



That's the setup out of the way, we're ready to code!

## Activity 1: Your First Program - Print

Let's make a very simple program and run it on your Pico.

We'll tell the Pico to '*print*' some words (*known as a 'string' of text*). These words/strings are then '*printed*' to the Shell window. The print function is something you'll use a LOT when programming. It's a useful tool to see what's happening with a program, what part of a program is running and can help when debugging projects.

In the top panel (which should be called <untitled> at the moment as we haven't saved it yet), copy/paste or type in enter the following line and then **select the green run icon** (it looks like a 'Play' button) from the menu.

```
print("This is my Pico talking")
```

You should now see the following in the Shell window:

The image shows two windows side-by-side. On the left is the code editor window titled '<untitled> \*'. It contains a single line of code: '1 print("This is my Pico talking")'. On the right is the shell window titled 'Shell'. It shows the command '>>> %Run -c \$EDITOR\_CONTENT' followed by the output 'This is my Pico talking' and then another prompt '>>>'. Both windows have a red circular icon with a white letter 'R' in the bottom right corner.

**You did it! Your first program!**

A few things to note:

- Print text strings always sit within inverted commas, within brackets
- Letter case is very important with MicroPython and other code languages. Our print is always 'print' and never 'Print'.

Now try changing the text in between the inverted commas - you can write anything you want... "This calendar is going to be awesome" or "I don't like pineapple on pizza"...or even the famous "Hello, World!".

You can even add multiple print lines one after the other - give it a try!

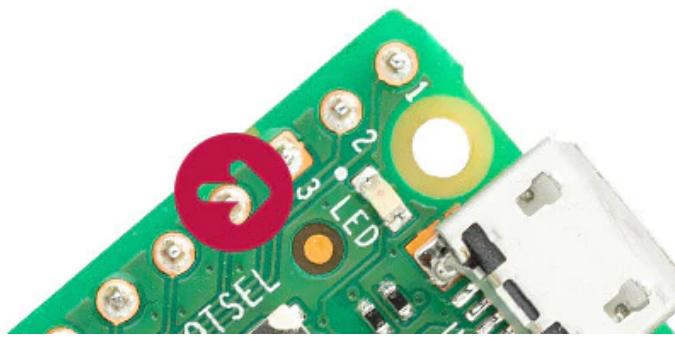
The image shows two windows side-by-side. On the left is the code editor window titled '<untitled> \*'. It contains a single line of code: '1 print("I don't like pineapple on pizza")'. On the right is the shell window titled 'Shell'. It shows the command '>>> %Run -c \$EDITOR\_CONTENT' followed by the output 'I don't like pineapple on pizza' and then another prompt '>>>'. Both windows have a red circular icon with a white arrow pointing right in the bottom right corner.

## Activity 2: Light the Onboard LED

We now know where to write code in Thonny and how to run a program. We'll be doing a lot more of that over the coming days, so for now let's jump straight into a bit of blinky!

The Pico comes with a little LED onboard, which is attached to one of the Pico's RP2040 pins, **GPIO25**. This pin isn't one of the physical pins available around the edge of the board.

We can tell our program to turn this pin on and off whenever we like, which is a nice easy first hardware project and can be handy as an indicator for your own projects later on. This is the beginning of getting your code to control physical things!



### Inputs and outputs

Our GPIO pins (the green ones from the [pin map](#)) can be set as **inputs** or **outputs**. We use **inputs** when we want something to send signals/voltage in to the Pico, like a button or a sensor, and we use **outputs** when we want the Pico to send signals/voltage out to something else, like LEDs or buzzers.

We're going to use an output here, which will send voltage to the onboard LED to light it up.

### Imports

Take a quick glance at the first line of the example code below. When we write code in Thonny, a lot of the time we need to add **imports** at the start.

Imports bring in code from other places (*called 'modules' or 'classes' - but let's not get bogged down with that for now*) that we need to make our program work. Usually these are from built-in modules.

To use the GPIO pins, we always need to import **Pin**.

### The Code

Take another look at the example below. So, what is it doing?

- The first line imports '**Pin**' from the '**machine**' module on the first line, which simply allows us to use the GPIO pins.
- The second line is setting up one of the pins to allow us to use it in our code. This sets up **GPIO25** with the name '**onboardLED**'. We use **Pin.OUT** at the end there to tell MicroPython to set this pin as an **output**.
- The third line then finally sets that pin value to '**1**' which is HIGH (on). 1 is always used for HIGH (on) and 0 is used for LOW (off).

```
from machine import Pin
onboardLED = Pin(25, Pin.OUT)
onboardLED.value(1)
```

Copy that code to your top-right panel, then hit the run button. The LED should light up – hooray – you've controlled your first physical component!

Now try changing the **1** to a **0** - run the code again and it should turn the LED off. You can also try adding a print line at the bottom to show a message at the same time.

## Day #1 Complete!

Well done fellow maker, today you have:

- Become an owner of your very own microcontroller!
- Learnt about the Raspberry Pi Pico
- Set up your Raspberry Pi Pico with Thonny
- Created your first program
- Learnt about the print function
- Controlled a physical component with code

Now, get some rest and a good night's sleep, for tomorrow we will continue the fun with day #2 (no spoilers here, and no peeking until the morning!).

---

We used [Fritzing](#) to create the breadboard wiring diagram images for this page.

## Featured Products



What are you looking for?



Currency  
GBP ▾

Login /  
Signup  
 Account

0  
Cart

Raspberry Pi ▾

Maker Store ▾

micro:bit ▾

Arduino ▾

Gifts ▾

Sale!

Tutorials      Blog

### 50 comments

**Emily**

January 11, 2023 at 12:48 pm



I got this in December but am only just finding time to start playing with things. Really clear instructions, I have successfully completed Day 1. Such a relief to find these instructions still online!

I got this in December but am only just finding time to start playing with things. Really clear instructions, I have successfully completed Day 1. Such a relief to find these instructions still online!

**The Pi Hut**

December 28, 2022 at 1:22 pm



@Keith A few other customers had this issue. If you scroll up, we've added a little section about how to install MicroPython on the Pico a different way, which links here:  
<https://www.raspberrypi.com/documentation/microcontrollers/micropython.html#drag-and-drop-micropython>

@Keith A few other customers had this issue. If you scroll up, we've added a little section about how to install MicroPython on the Pico a different way, which links here:  
<https://www.raspberrypi.com/documentation/microcontrollers/micropython.html#drag-and-drop-micropython>

**Keith Lomax**

December 28, 2022 at 1:21 pm

I followed the instructions. When I got to installing the micro python variant, it said:



```
raise URLError(err)
```

```
urllib.error.URLError: <urlopen error [SSL: CERTIFICATE_VERIFY_FAILED]  
certificate verify failed: certificate has expired (_ssl.c:997)>
```

There was no option to bypass this error???

I followed the instructions. When I got to installing the micro python variant, it said:

```
raise URLError(err)
```

```
urllib.error.URLError: <urlopen error [SSL: CERTIFICATE_VERIFY_FAILED] certificate  
verify failed: certificate has expired (_ssl.c:997)>
```

There was no option to bypass this error???

### Paul Kent

December 28, 2022 at 1:16 pm

Hi,

We've had a bit of fun working out our own implementation of SOS in morse code , but run in to a problem (see the end)

```
import machine  
import time  
led = machine.Pin("LED",machine.Pin.OUT)  
  
def off(t):  
  
    led.off() time.sleep(t)  
  
def on(t):  
  
    led.on() time.sleep(t)
```

```
def short():
    off(0.3)
    on(0.3)
    off(0)
```



```
def long():
    off(0.3)
    on(0.6)
    off(0)
```

```
short()
short()
short()
long()
long()
long()
short()
short()
short()
```

```
led.off()
```

But we've run in to some issues where, when the pico is not given an instruction for a short while (a minute or two) it creates an error.

PermissionError(13, 'a device attached to the system is not functioning.' None, 31)

If you have a serial connection from a another program , then disconnect it there first.

Process ended with exit code 1.

Hi,

We've had a bit of fun working out our own implementation of SOS in morse code , but run in to a problem (see the end)

```
import machine
import time
led = machine.Pin("LED",machine.Pin.OUT)
```

```
def off(t):
```

```
led.off() time.sleep(t)
```

```
def on(t):
```

```
led.on() time.sleep(t)
```

```
def short():
```

```
off(0.3)
```

```
on(0.3)
```

```
off(0)
```

```
def long():
```

```
off(0.3)
```

```
on(0.6)
```

```
off(0)
```

```
short()
```

```
short()
```

```
short()
```

```
long()
```

```
long()
```

```
long()
```

```
short()
```

```
short()
```

```
short()
```

```
led.off()
```

But we've run in to some issues where, when the pico is not given an instruction for a short while (a minute or two) it creates an error.

PermissionError(13, 'a device attached to the system is not functioning' None, 31)

If you have a serial connection from another program , then disconnect it there first .

Process ended with exit code 1.

**Enda**

December 28, 2022 at 1:16 pm



Finally got it working on Ubuntu & Thonny ! It seems using Ubuntu is life on hard mode. Dial out was an issue, so watch for the error messages .

Finally got it working on Ubuntu & Thonny ! It seems using Ubuntu is life on hard mode. Dial out was an issue, so watch for the error messages .

### Ruth

December 19, 2022 at 10:59 am

 Is this guide available as a PDF? I find it easier to follow instructions on paper than scrolling on a screen (but obviously using the screen to click on links etc)

Is this guide available as a PDF? I find it easier to follow instructions on paper than scrolling on a screen (but obviously using the screen to click on links etc)

### DAR

December 19, 2022 at 10:57 am

 Yeah I had to use the drag python onto the pico, but every time I unplug the pico the python program is no longer there and I have to drag the python installer onto the pico again, is this normal?

Yeah I had to use the drag python onto the pico, but every time I unplug the pico the python program is no longer there and I have to drag the python installer onto the pico again, is this normal?

### The Pi Hut

December 15, 2022 at 12:13 pm

 @Helen Sounds like it's not running the code on the Pico (or the Pico is no longer attached or recognised). When you run the code, it should ask if you want to run it on the computer or the Pico, you need to select the Pico. Might be best to go back through the setup steps making sure the Pico (and COM port) are showing and selected, then try again.

@Helen Sounds like it's not running the code on the Pico (or the Pico is no longer attached or recognised). When you run the code, it should ask if you want to run it on the computer

or the Pico, you need to select the Pico. Might be best to go back through the setup steps making sure the Pico (and COM port) are showing and selected, then try again.

### Helen

December 15, 2022 at 12:11 pm



I'm on day 1 and Activity 1 was fine, but when I run activity 2 (LED) I get an error message saying  
`ModuleNotFoundError: No module named 'machine'`

I'm on day 1 and Activity 1 was fine, but when I run activity 2 (LED) I get an error message saying

`ModuleNotFoundError: No module named 'machine'`

### The Pi Hut

December 12, 2022 at 10:19 am



@James Bayliss-Smith – It looks like you're trying to run the code on your Mac rather than the Pico (Thonny should give you the option when you click to run the code). The machine module wouldn't work unless you run it on your Pico, so this might explain why you see that error. Try disconnecting and re-connecting the Pico, check it's showing as connected in Thonny with a COM port (some other mac users gave some tips on that in the comments) then try again.

@James Bayliss-Smith – It looks like you're trying to run the code on your Mac rather than the Pico (Thonny should give you the option when you click to run the code). The machine module wouldn't work unless you run it on your Pico, so this might explain why you see that error. Try disconnecting and re-connecting the Pico, check it's showing as connected in Thonny with a COM port (some other mac users gave some tips on that in the comments) then try again.

### James Bayliss-Smith

December 12, 2022 at 10:15 am

Hi there,



got this calendar for my daughter. We got almost all the way on day 1 but are getting an error when we try to run the last bit of code. Any ideas? Thanks. We're on macOS Big Sur 11.7

```
>>> %Run -c $EDITOR_CONTENT
Traceback (most recent call last):
File "", line 1, in
ModuleNotFoundError: No module named 'machine'
```

Hi there,

got this calendar for my daughter. We got almost all the way on day 1 but are getting an error when we try to run the last bit of code. Any ideas? Thanks. We're on macOS Big Sur 11.7

```
>>> %Run -c $EDITOR_CONTENT
Traceback (most recent call last):
File "", line 1, in
ModuleNotFoundError: No module named 'machine'
```

### Nick Woodruffe

December 10, 2022 at 10:00 pm

look good so far. Thanks for the kit.

Day 1 on a loop....

```
from machine import Pin
import time
true = 1
```

```
onboardLED = Pin(25, Pin.OUT)
```

```
while true:
    onboardLED.value(1)
    time.sleep(1)
    onboardLED.value(0)
    time.sleep(1)
```

look good so far. Thanks for the kit.

## Day 1 on a loop....

```
from machine import Pin  
import time  
true = 1  
  
onboardLED = Pin(25, Pin.OUT)  
  
while true:  
    onboardLED.value(1)  
    time.sleep(1)  
    onboardLED.value(0)  
    time.sleep(1)
```

### **Stacy**

December 12, 2022 at 11:16 am



Nice little project, im folowing this course as well as the one being done in C++ via OldSkhoolCoders streams via Twitch <https://www.twitch.tv/videos/1669495599>

Nice little project, im folowing this course as well as the one being done in C++ via OldSkhoolCoders streams via Twitch <https://www.twitch.tv/videos/1669495599>

### **The Pi Hut**

December 9, 2022 at 12:56 am



@Gill – You may need to go back to the step where you select the COM port and see if it's showing under a different number this time (COM ports do that sometimes), then select that one instead. Also double-check the USB cable is connected to the Pico properly, and try other USB ports too. If still no joy, pop us a quick message via [support.thepihut.com](http://support.thepihut.com) and we'll help.

@Gill – You may need to go back to the step where you select the COM port and see if it's showing under a different number this time (COM ports do that sometimes), then select that one instead. Also double-check the USB cable is connected to the Pico properly, and try other USB ports too. If still no joy, pop us a quick message via [support.thepihut.com](http://support.thepihut.com) and we'll help.

**Gill**

December 9, 2022 at 12:53 am



We managed to complete day 1 with no problems. Day 2 we cannot complete – have the error “Unable to connect to COM3: port not found” so we cannot run the code in Thonny. Help please! Thank you.

We managed to complete day 1 with no problems. Day 2 we cannot complete – have the error “Unable to connect to COM3: port not found” so we cannot run the code in Thonny. Help please! Thank you.

**giles**

December 8, 2022 at 1:53 am

@Steve – re using the speaker for morse code – I can't believe I didn't think of that!  
Do you know how to do a Pull Request on Github? It would be great to get some collaborators on the pico-xmas repo :-)

@Bazza, the VSCode integration is pretty sweet, I've added a few more trouble shooting tips at the top of the README <https://github.com/gilesknap/pico-xmas>

@Steve – re using the speaker for morse code – I can't believe I didn't think of that! Do you know how to do a Pull Request on Github? It would be great to get some collaborators on the pico-xmas repo :-)

@Bazza, the VSCode integration is pretty sweet, I've added a few more trouble shooting tips at the top of the README <https://github.com/gilesknap/pico-xmas>

**Leanne Borland**

December 7, 2022 at 1:27 am



This advent calendar is an ingenious idea PiHut! As a 40 year old mother of two boys, I'm loving the opportunity to learn something new and techy with my kids this year, and the sense of achievement from lighting that little LED is way greater than you could imagine. Looking forward to the other days :-)

This advent calendar is an ingenious idea PiHut! As a 40 year old mother of two boys, I'm loving the opportunity to learn something new and techy with my kids this year, and the

sense of achievement from lighting that little LED is way greater than you could imagine.  
Looking forward to the other days :-)

**Dmitry**

December 6, 2022 at 3:17 pm

Very pleased so far with the clear instructions. They do anticipate the issues a complete novice like myself would find. Tried both a Windows PC and the Pi 400, both worked well. Proof that even in your 50s you can still enjoy learning something new! Great all good.

Thank you PHut!

Very pleased so far with the clear instructions. They do anticipate the issues a complete novice like myself would find. Tried both a Windows PC and the Pi 400, both worked well. Proof that even in your 50s you can still enjoy learning something new! Great all good.

Thank you PHut!

**PaulH**

December 6, 2022 at 1:50 am

Package waiting by my front door when I got home today (Delivered Saturday so thanks for that)

Windows 11 user here – I found that I needed to do everything from the bottom right of the IDE window.

If you go into ‘configure interpreter’ and select the ‘general’ tab – you can change the UI to regular and then you get a normal set of menus across the top fo the window – which may make following the tutorials easier.

In any case – Day 1 accomplished – LED on, LED off.

Package waiting by my front door when I got home today (Delivered Saturday so thanks for that)

Windows 11 user here – I found that I needed to do everything from the bottom right of the IDE window.

If you go into ‘configure interpreter’ and select the ‘general’ tab – you can change the UI to regular and then you get a normal set of menus across the top fo the window – which may

make following the tutorials easier.

In any case – Day 1 accomplished – LED on, LED off.

### The Pi Hut

December 5, 2022 at 12:57 pm



@Darren Hill – The online tutorials will be online forever, but we'll look at creating some PDFs to attach to each day for printing. It's a nice idea :)

@Darren Hill – The online tutorials will be online forever, but we'll look at creating some PDFs to attach to each day for printing. It's a nice idea :)

### Darren Hill

December 5, 2022 at 12:52 pm

Are these instructions going to be available for download in pdf format? Either daily or as a complete set at the end?



Got my calendar but probably won't have time to do much with it until after Christmas, and it would be nice to have everything to hand. Also to keep for the future to help pass on the content to the kids so they can have a play too without having to keep referring back to the website.

Are these instructions going to be available for download in pdf format? Either daily or as a complete set at the end?

Got my calendar but probably won't have time to do much with it until after Christmas, and it would be nice to have everything to hand. Also to keep for the future to help pass on the content to the kids so they can have a play too without having to keep referring back to the website.

### Bazza

December 5, 2022 at 12:37 pm

@giles

Nice one.

Just got my calendar delivered today. I use VS code all day at work and will be having a go as setting up to connect to the Pico later.

@giles

Nice one.

Just got my calendar delivered today. I use VS code all day at work and will be having a go as setting up to connect to the Pico later.

## The Pi Hut



December 4, 2022 at 10:48 pm

@Steve ...you'll just have to wait and see ;) #NoPeeping

@Steve ...you'll just have to wait and see ;) #NoPeeping

## Paula

December 4, 2022 at 10:50 pm

```
from machine import Pin  
import time
```

```
onboardLED = Pin(25, Pin.OUT)
```

```
for x in range (1,10):
```

```
    print(x)
```

```
    if (x % 2) == 0:
```

```
        onboardLED.value(1)
```

```
    else:
```

```
        onboardLED.value(0)
```

```
    time.sleep(1)
```

```
from machine import Pin
```

```
import time
```

```
onboardLED = Pin(25, Pin.OUT)
```

```
for x in range (1,10):
```

```
    print(x)
```

```
    if (x % 2) == 0:
```

```
        onboardLED.value(1)
```

```
    else:
```

```
onboardLED.value(0)  
time.sleep(1)
```

**Steve**

December 4, 2022 at 10:47 pm

 @giles, nice idea on the Morse Code and thanks for sharing the code on GitHub, here's hoping we get a little buzzer or speaker in one of the days to expand on it :)

@giles, nice idea on the Morse Code and thanks for sharing the code on GitHub, here's hoping we get a little buzzer or speaker in one of the days to expand on it :)

**Richard Lees**

December 4, 2022 at 12:07 am

 Really enjoying this! I've been buying bits Pi-raphernalia for 18 months but have been too timid to take the plunge and have ago. This has been a revelation. What a great way to introduce making. Thank you.

Really enjoying this! I've been buying bits Pi-raphernalia for 18 months but have been too timid to take the plunge and have ago. This has been a revelation. What a great way to introduce making. Thank you.

**The Pi Hut**

December 3, 2022 at 9:48 pm

 @Kristian Oh no, sorry about that. Looks like the packing team mixed up the LEDs in your kit. Please drop us a quick message via support.thepihut.com and Billie will have a replacement pack sent out to you ASAP. You can still use the LEDs you have for the upcoming days and activities until the new ones arrive.

@Kristian Oh no, sorry about that. Looks like the packing team mixed up the LEDs in your kit. Please drop us a quick message via support.thepihut.com and Billie will have a replacement pack sent out to you ASAP. You can still use the LEDs you have for the upcoming days and activities until the new ones arrive.

**The Pi Hut**

December 3, 2022 at 9:38 pm



@Keith Glad your first day ran smoothly :) The final screenshot may look slightly different depending on the computer (operating system) you're using – you may need to select 'view > files' to see the same panels.

@Keith Glad your first day ran smoothly :) The final screenshot may look slightly different depending on the computer (operating system) you're using – you may need to select 'view > files' to see the same panels.

**Keith**

December 3, 2022 at 9:35 pm



Thanks – this worked flawlessly for me and only took about ten minutes. The only comment I would make is that the screen capture at the end of the Software Setup section is not correct.

Looking forward to day two!

Thanks – this worked flawlessly for me and only took about ten minutes. The only comment I would make is that the screen capture at the end of the Software Setup section is not correct.

Looking forward to day two!

**Kristian**

December 3, 2022 at 9:46 pm



It seems you sent me 2 green LEDs and 1 red LED or have I messed something up in the process of building the circuit? I haven't changed the code on activity one and when they light up there is no amber light.

It seems you sent me 2 green LEDs and 1 red LED or have I messed something up in the process of building the circuit? I haven't changed the code on activity one and when they light up there is no amber light.

**giles**

December 2, 2022 at 6:30 pm

Love the advent calendar. Thanks for nice instructions for the initial day!



Someone gave me the idea of Saying “Happy Christmas” in Morse Code. Here is some (rather over engineered) Micro Python to do just that!! :-)

<https://github.com/gilesknap/pico-xmas>

Love the advent calendar. Thanks for nice instructions for the initial day!

Someone gave me the idea of Saying “Happy Christmas” in Morse Code. Here is some (rather over engineered) Micro Python to do just that!! :-)

<https://github.com/gilesknap/pico-xmas>

### Bob

December 2, 2022 at 2:32 pm

I'm pleased others have set off before me, they can hammer out the bugs, because my old brain isn't ready for that just yet 😂

I'm pleased others have set off before me, they can hammer out the bugs, because my old brain isn't ready for that just yet 😂

### Mike G

December 2, 2022 at 11:39 am

- Jim P

Hopefully you sorted your “SSL Certificate expired” error.

I also get this yesterday, following the instructions. Didn’t find a solution to that error – so I just downloaded the .uf2 file and installed it manually. (You can find instructions here <https://micropython.org/download/rp2-pico/>)

- Jim P

Hopefully you sorted your “SSL Certificate expired” error.

I also get this yesterday, following the instructions. Didn’t find a solution to that error – so I just downloaded the .uf2 file and installed it manually. (You can find instructions here <https://micropython.org/download/rp2-pico/>)

**Richard Goodhand**

December 2, 2022 at 11:39 am

Hi all, I struggled for over an hour to get the Pico talking to Thonny. It ended up being that I'm using Windows 8.1 and a driver needed to be installed for them to communicate over the USB. If you are finding a similar issue check Device Manager in Control Panel. Zadig was the answer to my prayers.

<https://github.com/raspberrypi/pico-feedback/issues/118>

I hope that helps others.

Hi all, I struggled for over an hour to get the Pico talking to Thonny. It ended up being that I'm using Windows 8.1 and a driver needed to be installed for them to communicate over the USB. If you are finding a similar issue check Device Manager in Control Panel. Zadig was the answer to my prayers. <https://github.com/raspberrypi/pico-feedback/issues/118>  
I hope that helps others.

**Jim Patience**

December 2, 2022 at 11:39 am

@The Pi Hut

All sorted now. Figured out to download the uf2 file and drop it on to the “disk” of the Pico. Hey presto! Loving this little course on the basics and looking forward to a lot more.

Have used the Raspberry Pi for years but never the Pico before so this is fresh new territory for me, so even though this was a gift to myself, thank you!

@The Pi Hut

All sorted now. Figured out to download the uf2 file and drop it on to the “disk” of the Pico. Hey presto! Loving this little course on the basics and looking forward to a lot more.

Have used the Raspberry Pi for years but never the Pico before so this is fresh new territory for me, so even though this was a gift to myself, thank you!

**Shaun**

December 2, 2022 at 11:39 am

I made the light on the board come on. Then I did a google and found some code to make the light flash. I haven't had such a stupid grin for years and the kids have no idea what's wrong with me. Thank you Pie Hut!

I made the light on the board come on. Then I did a google and found some code to make the light flash. I haven't had such a stupid grin for years and the kids have no idea what's wrong with me. Thank you Pie Hut!

### The Pi Hut

December 1, 2022 at 8:21 pm



Thanks to everyone who let us know about the odd SSL errors when installing MicroPython, it seems to only be impacting a handful of customers (and we can't seem to re-create it) but it's a hiccup all the same! We have updated the page to point to the alternative method for installing MicroPython. Love this community! See you all tomorrow for day #2 :D

Thanks to everyone who let us know about the odd SSL errors when installing MicroPython, it seems to only be impacting a handful of customers (and we can't seem to re-create it) but it's a hiccup all the same! We have updated the page to point to the alternative method for installing MicroPython. Love this community! See you all tomorrow for day #2 :D

### Korhm

December 1, 2022 at 8:10 pm

I used to do some MicroPython, but you made me discover ThonnyIDE. It's easier with this IDE :) Thank you !

Is there a Raspberry Pi Pico WH planned ? I'm no bad on soldering, espacialy this tiny PINs...

I used to do some MicroPython, but you made me discover ThonnyIDE. It's easier with this IDE :) Thank you !

Is there a Raspberry Pi Pico WH planned ? I'm no bad on soldering, espacialy this tiny PINs...

### Joe

December 1, 2022 at 8:10 pm



Hi all,

If others are having issues with installing Micropython like we where this evening, SSL Error.

You can get around this by visiting the micropython site:

<https://micropython.org/download/rp2-pico/> and downloading the .uf2 file in the firmware section and following the install instructions.

Basically, just drop the file in to the usb drive that shows up when you plug the pico in, this will install and then restart the pico

Thonny should then work fine and you can follow the steps above

Hi all,

If others are having issues with installing Micropython like we where this evening, SSL Error.

You can get around this by visiting the micropython site:

<https://micropython.org/download/rp2-pico/> and downloading the .uf2 file in the firmware section and following the install instructions.

Basically, just drop the file in to the usb drive that shows up when you plug the pico in, this will install and then restart the pico

Thonny should then work fine and you can follow the steps above

**Mel**

December 1, 2022 at 8:10 pm

 Couldn't get Thonny to recognise the pico at first, so I used the UF2 method instead:

<https://www.raspberrypi.com/documentation/microcontrollers/micropython.html> , skipped the 'Install Micropython' step, and all worked fine after that. Just in case anyone else has issues. (This was on Windows 11)

Couldn't get Thonny to recognise the pico at first, so I used the UF2 method instead:

<https://www.raspberrypi.com/documentation/microcontrollers/micropython.html> , skipped the 'Install Micropython' step, and all worked fine after that. Just in case anyone else has issues. (This was on Windows 11)

**David Ian Bartlett**

December 1, 2022 at 8:10 pm



I had a frustrating afternoon! Fell down a hole somewhere between installing the ide and downloading the firmware to the pico. Could not find the com port which eventually materialised! All good now Thanks.

I had a frustrating afternoon! Fell down a hole somewhere between installing the ide and downloading the firmware to the pico. Could not find the com port which eventually materialised! All good now Thanks.

**mistie710**

December 1, 2022 at 8:27 pm



Finished today's stuff although Thonny didn't quite look the way I thought it would ("You should now see a 'Raspberry Pi Pico' panel on the left-side of Thonny, and information in the bottom-right 'Shell' panel." Well I had the bottom shell display but the rest appeared to be missing). Looking forward to the rest of it!

Finished today's stuff although Thonny didn't quite look the way I thought it would ("You should now see a 'Raspberry Pi Pico' panel on the left-side of Thonny, and information in the bottom-right 'Shell' panel." Well I had the bottom shell display but the rest appeared to be missing). Looking forward to the rest of it!

**The Pi Hut**

December 1, 2022 at 4:02 pm

@Paul Watts – thanks for the tip, no doubt will help others :)

@Paul Watts – thanks for the tip, no doubt will help others :)

**Paul Watts**

December 1, 2022 at 4:00 pm

Sorry, me again :)



Also note that the account that Thonny is running in needs to be a member of the 'dialout' group or you'll get 'permission denied' errors. You could run Thonny as root, but .....

Sorry, me again :)

Also note that the account that Thonny is running in needs to be a member of the 'dialout' group or you'll get 'permission denied' errors. You could run Thonny as root, but .....

**Paul Watts**

December 1, 2022 at 4:00 pm



Just to note – if you are using Thonny on Linux, the first time you try to 'run' anything it will ask you to save the text file first. You can get around this by entering the '%Run ...' command directly to the CLI, but to be honest its easier just to save the file.

Just to note – if you are using Thonny on Linux, the first time you try to 'run' anything it will ask you to save the text file first. You can get around this by entering the '%Run ...' command directly to the CLI, but to be honest its easier just to save the file.

**Misel**

December 1, 2022 at 4:00 pm



Thank you. Worked flawlessly and my sons and I created a little SOS blinking pattern :D

Thank you. Worked flawlessly and my sons and I created a little SOS blinking pattern :D

**The Pi Hut**

December 1, 2022 at 11:47 am



@Jim Hmm, it might have been a temporary blip (technical eh?!) with the MicroPython servers when you were trying to download. Can you try again? We've tested again here and everything is working as expected. Also just make sure everything is up to date on your PC (including date/time).

@Jim Hmm, it might have been a temporary blip (technical eh?!) with the MicroPython servers when you were trying to download. Can you try again? We've tested again here and everything is working as expected. Also just make sure everything is up to date on your PC (including date/time).

David

December 1, 2022 at 11:40 am

Yes Mike,

I started work with microcomputers almost before the home computer was thought of. I had to assemble and solder memory boards. I remember that they stored 256 bytes!

I then went to work in education and the (University) students were excited about being able to programme and use a computer.

When the IBM PC arrived everything changed. Students knew how to use "Applications" already, and learning to programme the machines took learning and practice. I had a student tell me "You know why we don't like programming? It's because we have to think.

Thanks to Eben and the team for encouraging us all to think.

Mike,

You mention Basic. Some of the hardest teaching we had to do in the early days was to re-educate students who had learned basic. The habits developed using basid were not conducive to writing reliable code.

What a perfect recipe. Pi, Linux and Python.

Cheers All

David

Yes Mike,

I started work with microcomputers almost before the home computer was thought of. I had to assemble and solder memory boards. I remember that they stored 256 bytes!

I then went to work in education and the (University) students were excited about being able to programme and use a computer.

When the IBM PC arrived everything changed. Students knew how to use "Applications" already, and learning to programme the machines took learning and practice. I had a student tell me "You know why we don't like programming? It's because we have to think.

Thanks to Eben and the team for encouraging us all to think.

Mike,

You mention Basic. Some of the hardest teaching we had to do in the early days was to re-educate students who had learned basic. The habits developed using basid were not conducive to writing reliable code.

What a perfect recipe. Pi, Linux and Python.

Cheers All

David

**Jim P**

December 1, 2022 at 11:41 am



Anyone else getting a “SSL Certificate expired” when trying to MicroPython? I’ve tried everything I think I can, but it just keeps giving me the error message. Also, I’ve noticed that it doesn’t seem to matter if I press the button on the Pico or not, Windows sees it just the same with the same 2 files in there.

Anyone else getting a “SSL Certificate expired” when trying to MicroPython? I’ve tried everything I think I can, but it just keeps giving me the error message. Also, I’ve noticed that it doesn’t seem to matter if I press the button on the Pico or not, Windows sees it just the same with the same 2 files in there.

**Mike**

December 1, 2022 at 10:48 am



What a fantastically well-written introductory article! Really clear steps and hitting just the right balance in tone between technical and friendly.

I can’t help but reflect on how far our industry has stepped backwards from the simple and approachable BASIC prompts of the 1980s to the IDEs and language complexities of more modern computing. We still have very far to go (back to) but I applaud The PiHut for initiatives such as this to make computing appeal again to our younger generation.

Looking forward to the forthcoming projects and hope my young teens feel the same way.

What a fantastically well-written introductory article! Really clear steps and hitting just the right balance in tone between technical and friendly.

I can’t help but reflect on how far our industry has stepped backwards from the simple and approachable BASIC prompts of the 1980s to the IDEs and language complexities of more modern computing. We still have very far to go (back to) but I applaud The PiHut for initiatives such as this to make computing appeal again to our younger generation.

Looking forward to the forthcoming projects and hope my young teens feel the same way.

## Leave a comment

All comments are moderated before being published.

This site is protected by reCAPTCHA and the Google [Privacy Policy](#) and [Terms of Service](#) apply.

## Related Posts



**Maker Advent Calendar Day #4: Amazing Analogue!**

The Pi Hut • Dec 4, 2022

Welcome to day four of your 12 Projects of Codemas Advent Calendar. Today we'll be playing with analogue inputs using a potentiometer and combining this with some of the components we've...

[Read more](#)



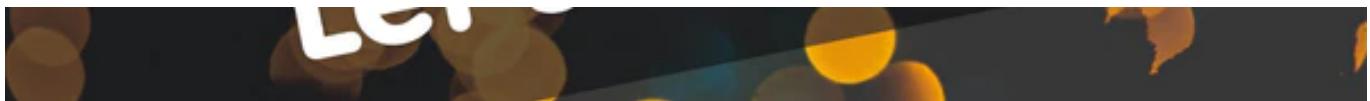
### Maker Advent Calendar Day #3: Bashing Buttons!

The Pi Hut • Dec 2, 2022

Welcome to day three of your 12 Projects of Codemas Advent Calendar. Today we'll be adding physical inputs to your program using buttons! So far we've relied purely on code...

[Read more](#)





## Maker Advent Calendar Day #2: Let's Get Blinky!

The Pi Hut • Dec 1, 2022

Welcome to day two of your 12 Projects of Codemas Advent Calendar. Today we'll be playing with one of the most popular components for makers - LEDs! LEDs (Light Emitting...

[Read more](#)

### Handy Links

[All Products](#)

[FAQs](#)

[Popular Searches](#)

[Search](#)

[Site Reviews](#)

### Got any questions?

[Contact Us / Support Portal](#)

[Can I Cancel My Order?](#)

[Has My Order Shipped Yet?](#)

[Where Is My Order?](#)

[Do You Ship To {insert country name}](#)

[How Much Is Shipping?](#)

### Terms & Conditions

[Delivery](#)

[Lithium Shipping](#)

[Pre-Orders](#)

[Privacy Statement](#)

[Policies](#)

[Terms of Service](#)

[Company Info](#)

[FAQ](#)

[Klarna FAQ](#)

[Quick Start Guide](#)

### Our Store Sections

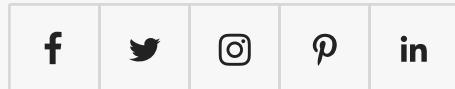
[Raspberry Pi](#)

[Maker Store](#)

[micro:bit](#)

[Arduino](#)

[Gifts](#)

[Search](#)[Support Portal](#)**Follow us**

We accept



© The Pi Hut 2023