

Arduino CLI with Visual Studio Code on MacOS

Get setup in 5 minutes!

If you landed on this article I'm guessing one of the many shortcomings with the Arduino IDE was finally enough. Don't get me wrong, the Arduino IDE is an awesome starting point for many users. Nervertheless, I use Arduinos as prototyping platforms and things like better source control integration and project structure go a long way. For the other users like me, here is an alternate way to program arduinos.

Configuration

Here is the version of all pieces for this solution. If you're just following along and not really sure what's going on, install the exact version listed to improve your odds of success. If you're a pro, the versions may give you an insight of where the issue is in case something goes wrong.

Operating System: macOS Big Sur 11.5.2

IDE: Visual Studio Code 1.60.0

IDE Extension: Arduino 0.4.4

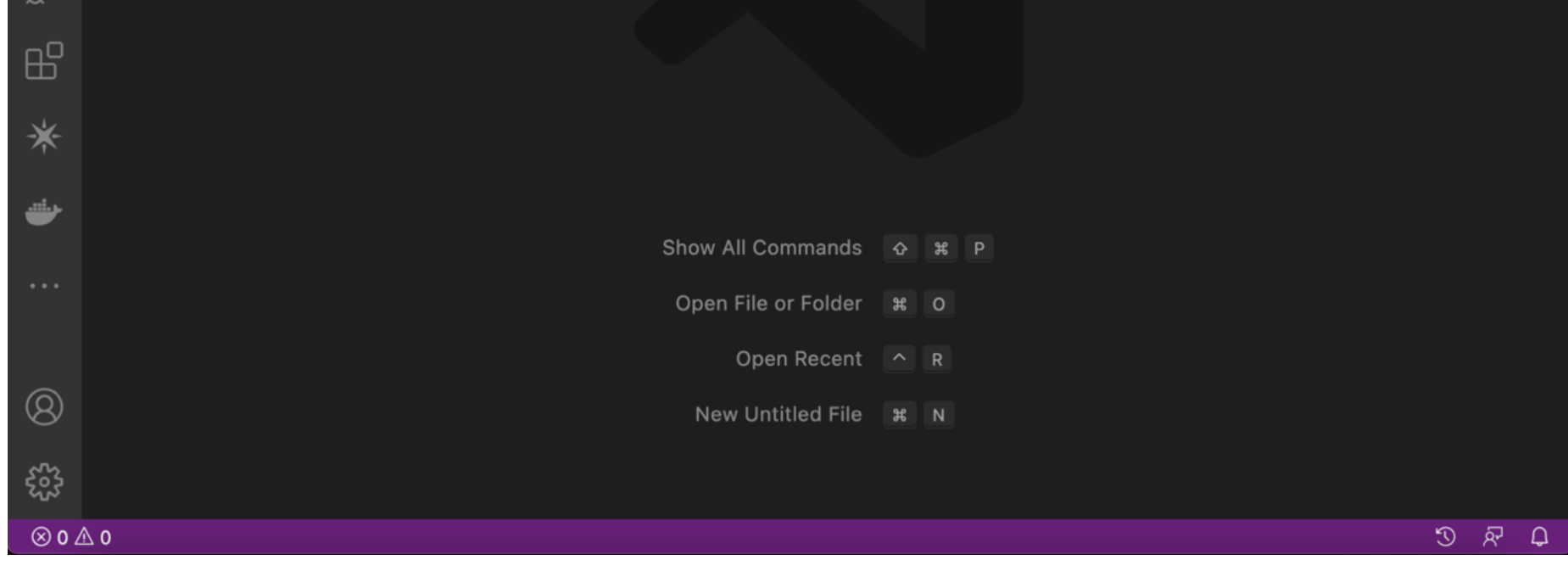
Toolchain: Arduino CLI 0.13.0

Board: Arduino Nano 33 IOT

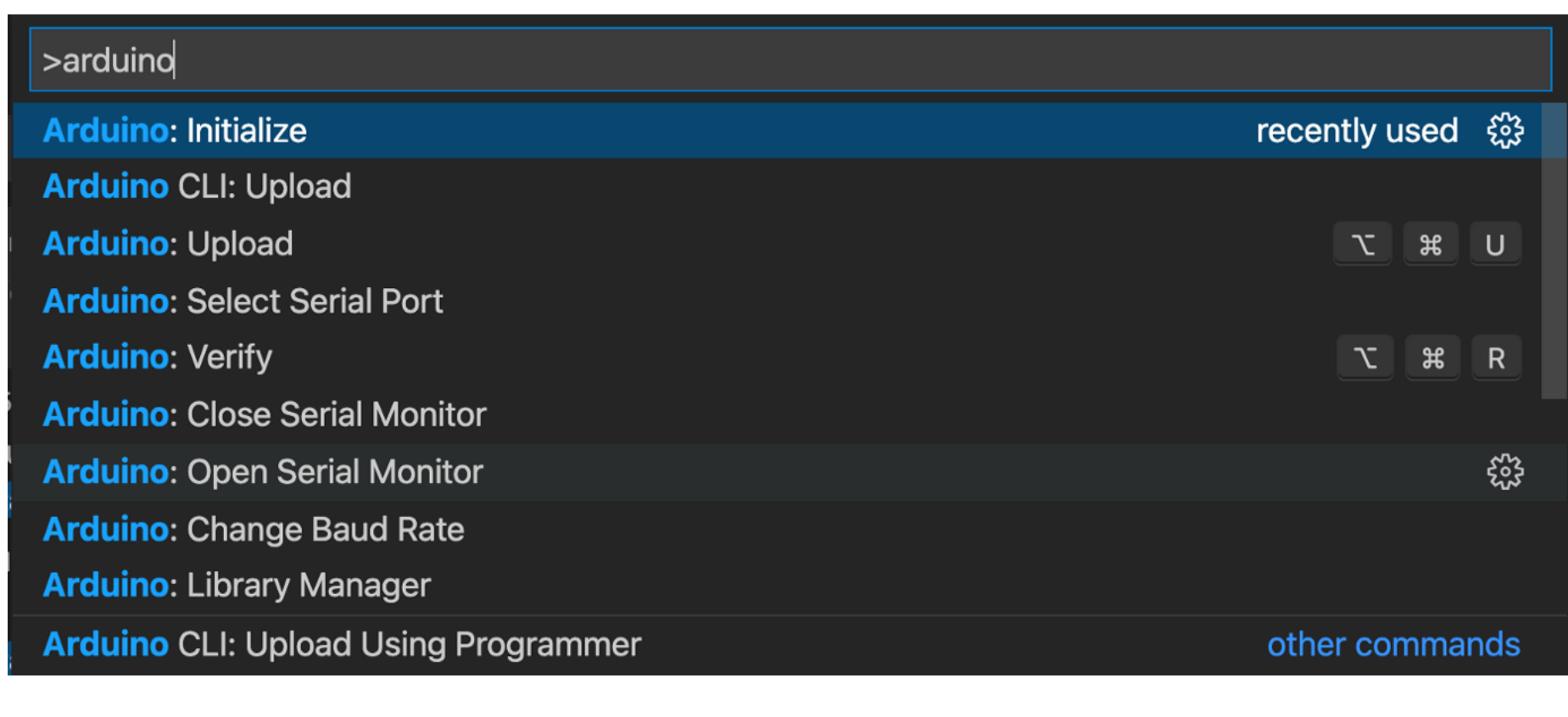
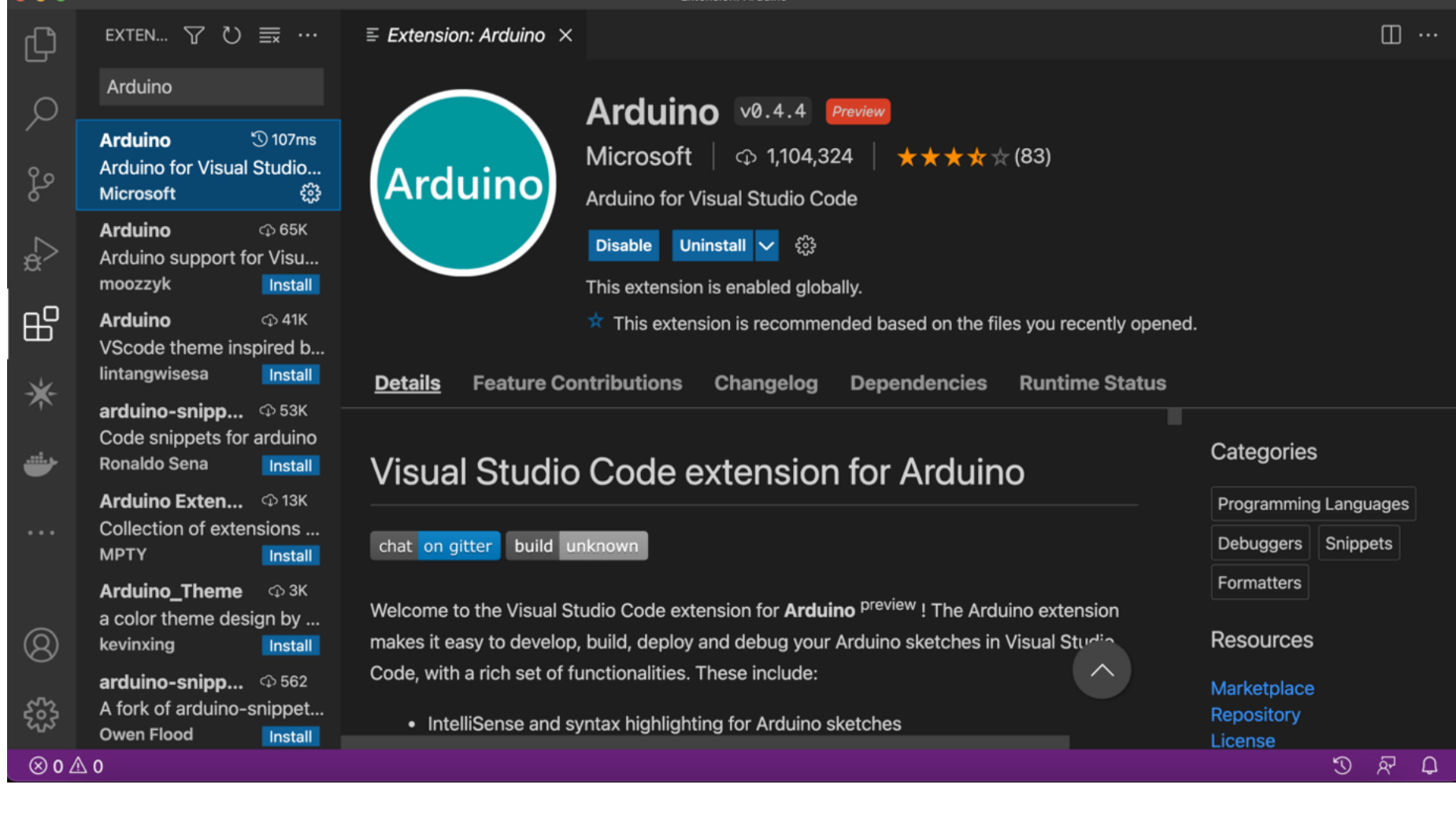
• • •

Installation

- Follow instructions at <https://code.visualstudio.com> to install Visual Studio Code. Verify your installation by launching it.

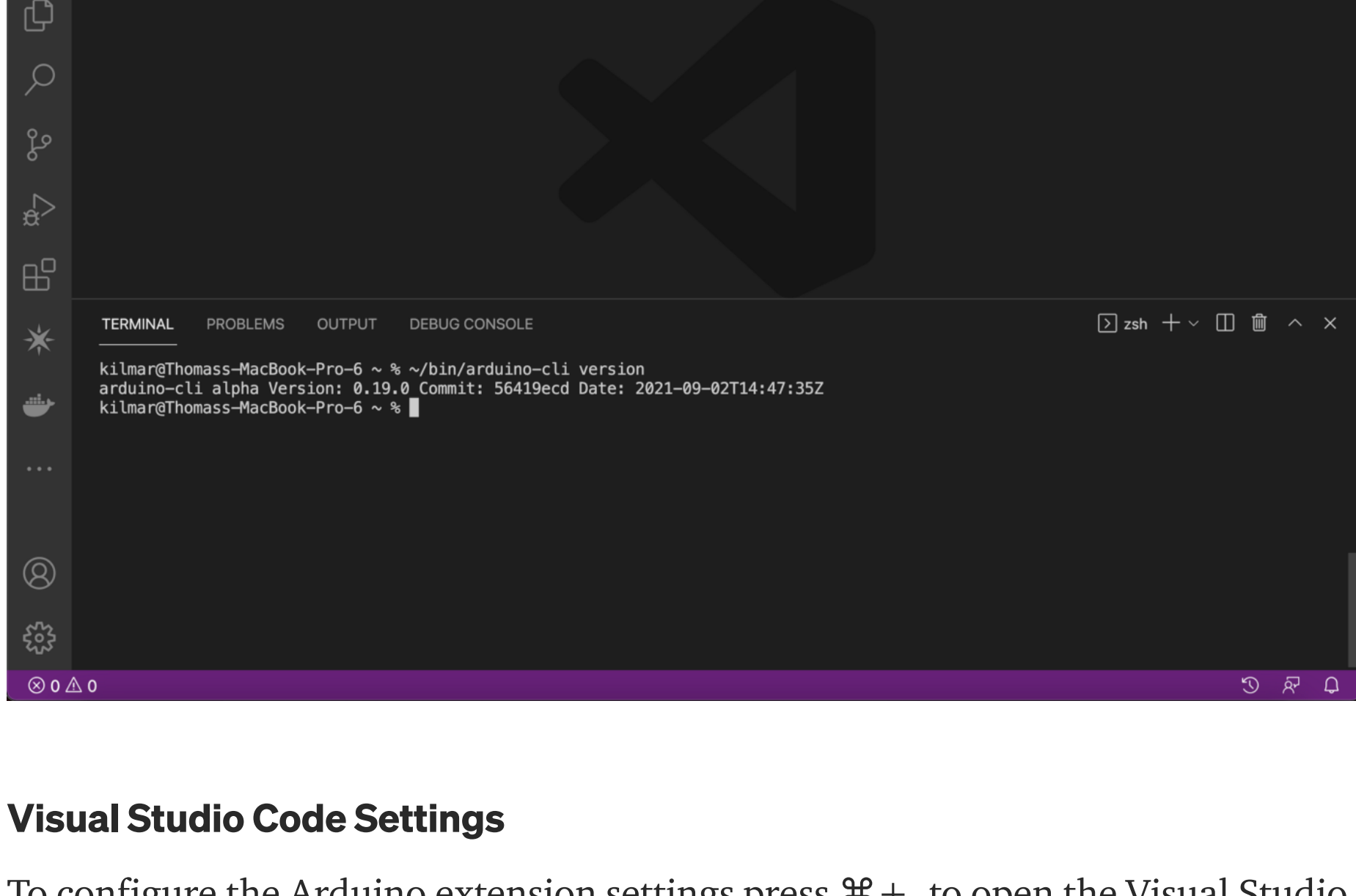


- On the left tool bar, click on the 4 blocks icon to install the Arduino extension by Microsoft. Verify your installation by pressing `⌘+shift+P` and typing Arduino, you should see available extension commands.



- Install Arduino CLI by following the instructions here <https://arduino.github.io/arduino-cli/0.19/installation/>, I used the install script. Make note of the installation location, in my case it is `~/bin/arduino-cli`.

Verify installation of Arduino CLI by running `~/bin/arduino-cli` version on the Terminal.



Visual Studio Code Settings

To configure the Arduino extension settings press `⌘+,` to open the Visual Studio Code settings page. Search for Arduino and set the following:

Arduino: Command Path: `arduino-cli`

Arduino: Path: `~/bin/`

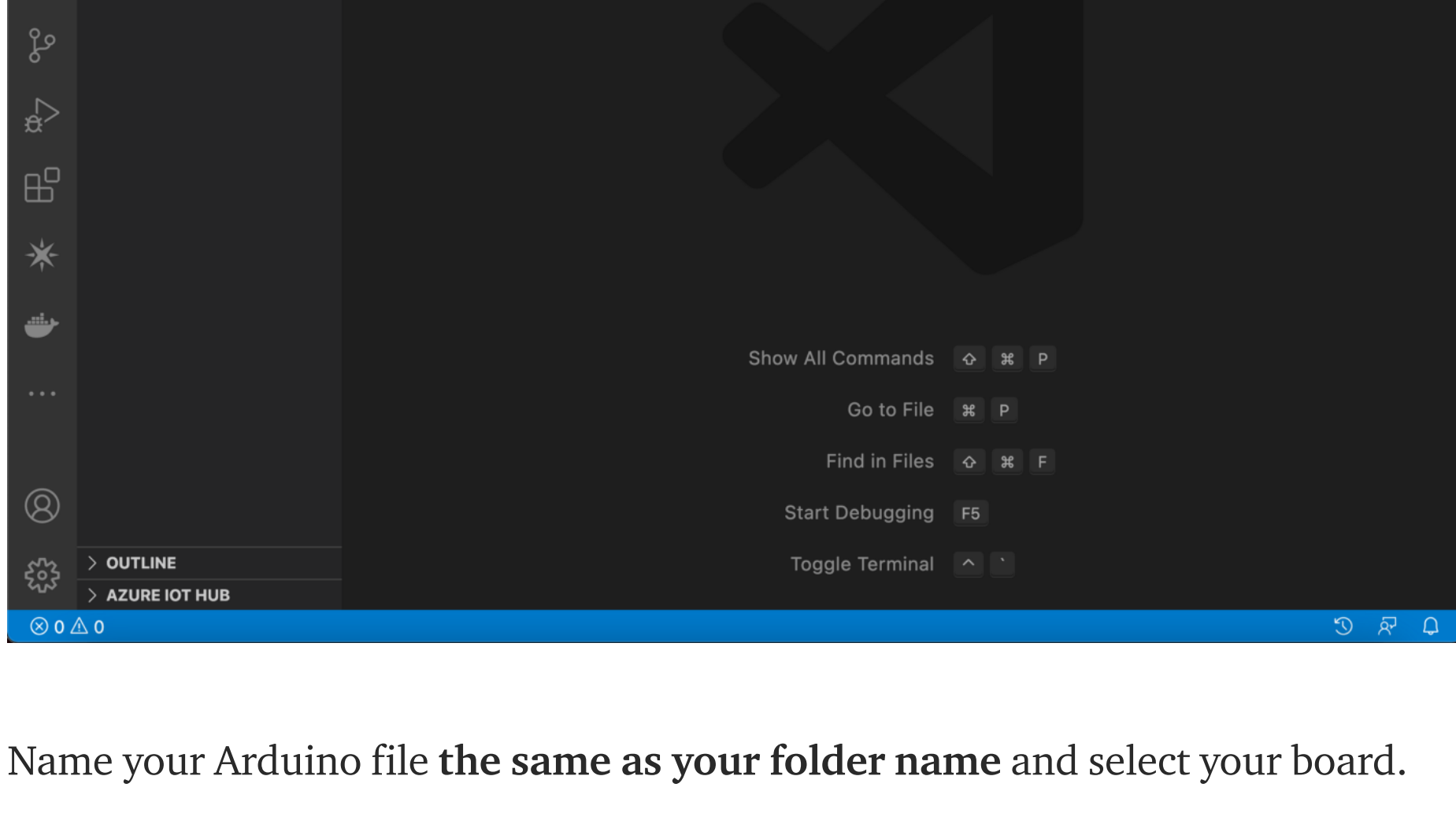
Arduino: Use Arduino CLI: checked

****Important!** Restart Visual Studio Code

• • •

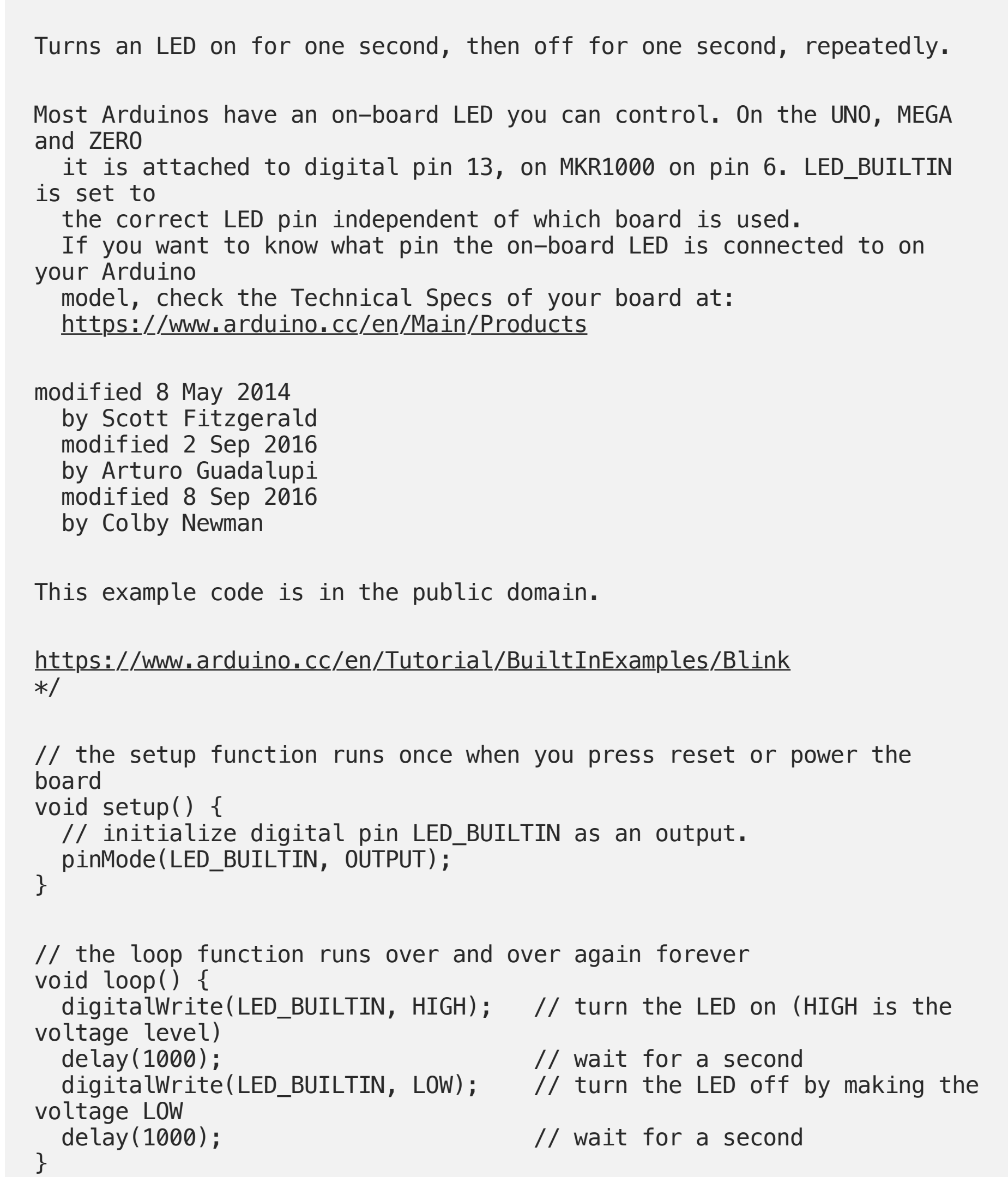
Create a Project

Now that all is setup, open your project folder in Visual Studio Code and initialize the project with the command Arduino: Initialize (press `⌘+shift+P` to search for commands).

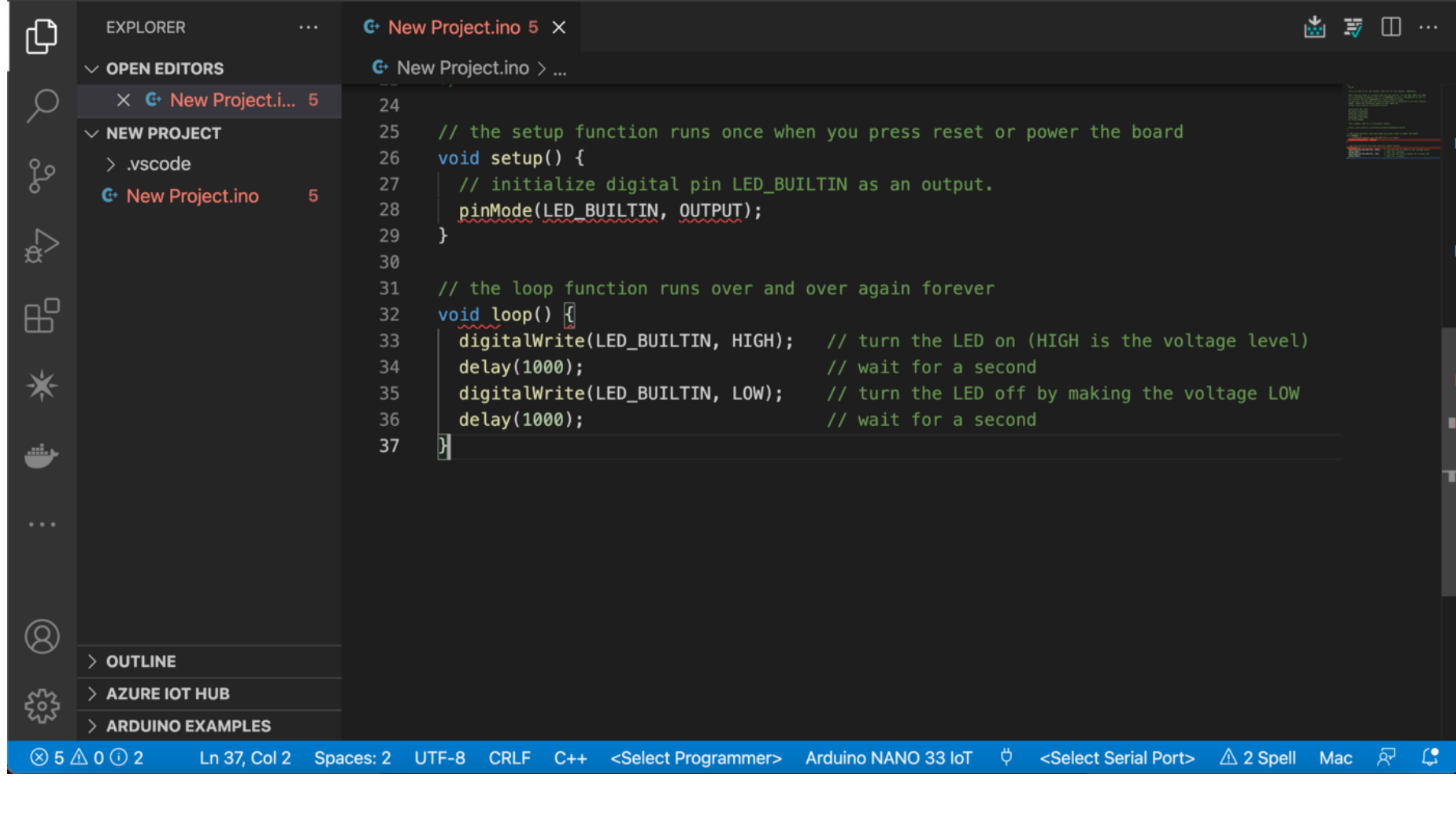


Name your Arduino file **the same as your folder name** and select your board.

As an example here is the blink code from <https://arduino.cc>



Once you type this in your .ino file you will get a few errors. This is because the IDE is not configured to include default Arduino file. Just click on the verify icon (check mark) on the top right and the Arduino Extension will take care of that by creating the file `c_cpp_properties.json` under `.vscode` directory.

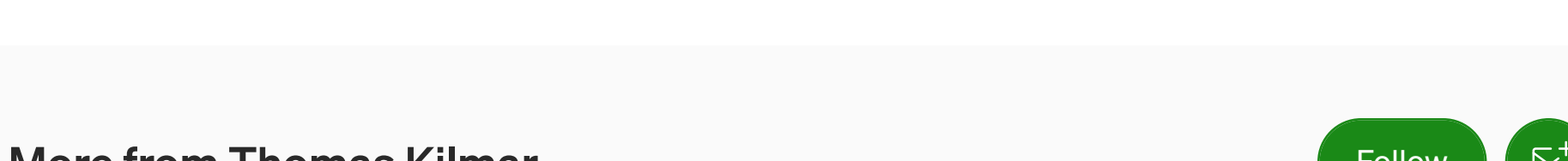


• • •

Use the Select Serial Port on the bottom right and click on the upload button to program your Arduino.

• • •

Code away!



Love podcasts or audiobooks? Learn on the go with our new app. [Try Knowable](#)

More from Medium

- Rebalancing- What the fuss is all about?

Sync/Async/Multi-thread

You have three ways to code your applicatio...

My Python learning experience

Maths is something I've always struggled wit...

Garuda Linux Provides a Hassle-free Arch

Experience With a Beautiful Neon Look

[Review with Video]

Day 2/30
Learning components of SQL, and data types

PACT—Enabling Contract Test Automation
In my last blog on contract testing fundame...

Consistency & Concurrency: A study of Oracle and SQL Server

Functional Language Codegen
I am a software engineer who started working in a C99 c...

Search

Thomas Kilmar

1 Follower

Build things, travel places

Follow

Related

Best CSS Text Animations from Terminal

CLI to create header images for your article

Run your C program on Apple M1 (2021)

The problem

How to Make Your Terminal Looks Better in

15min

VSCoide on steroids

How to setup and use VSCoide in a Spacema...

Help

Status

Writers

Blog

Careers

Privacy

Terms

About

Knowable

67 1

• • •