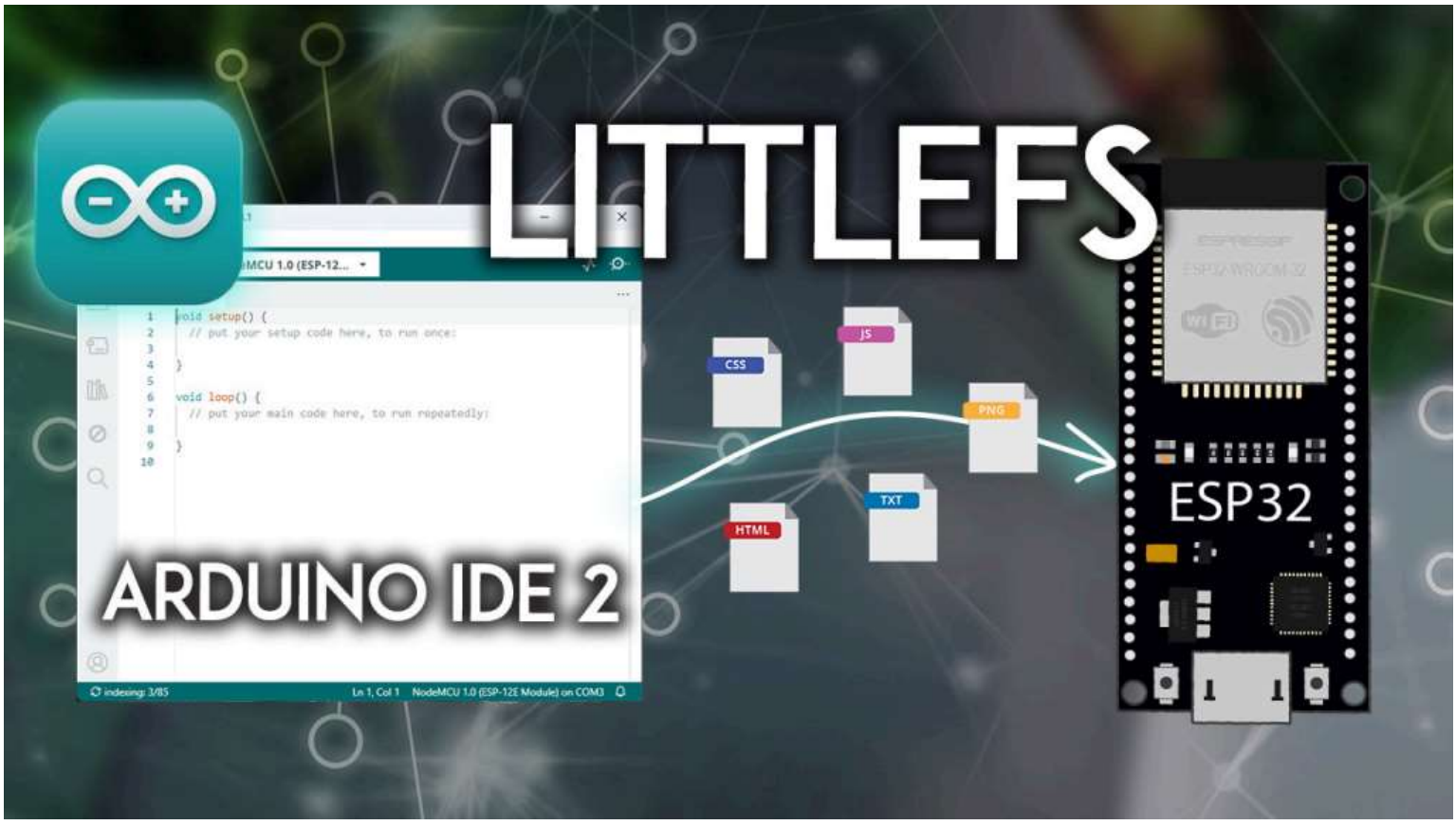
Arduino IDE 2: Instalar ESP32 LittleFS Uploader (Carregar arquivos para o sistema de arquivos)

Neste guia, você aprenderá como fazer upload de arquivos para o ESP32 LittleFS Filesystem usando o Arduino IDE 2 (2.2.1 ou uma versão superior). Instalaremos um plugin de upload que adicionará um novo menu ao Arduino IDE. Este plugin também é compatível com as placas ESP8266 e Raspberry Pi Pico.



Usando um ESP8266? Siga este tutorial em vez disso: Arduino IDE 2: Instalar ESP8266 NodeMCU LittleFS Uploader (Carregar Arquivos para o Sistema de Arquivos) .

Índice

Se você ainda estiver usando **o Arduino 1.8** , você pode seguir este tutorial: ESP32: Carregar arquivos para o LittleFS usando o Arduino IDE (antigo) .

Apresentando o LittleFS

LittleFS é um sistema de arquivos leve criado para microcontroladores que permite que você acesse a memória flash como você faz em um sistema de arquivos padrão no seu computador, mas é mais simples e mais limitado. Você pode ler, escrever, fechar e excluir arquivos. Usar LittleFS com as placas ESP32 é útil para:

- Crie arquivos de configuração com configurações;
- Salvar dados permanentemente;
- Crie arquivos para salvar pequenas quantidades de dados em vez de usar um cartão microSD;
- Salve arquivos HTML, CSS e JavaScript para construir um servidor web;
- Salvar imagens, figuras e ícones ;
- E muito mais.

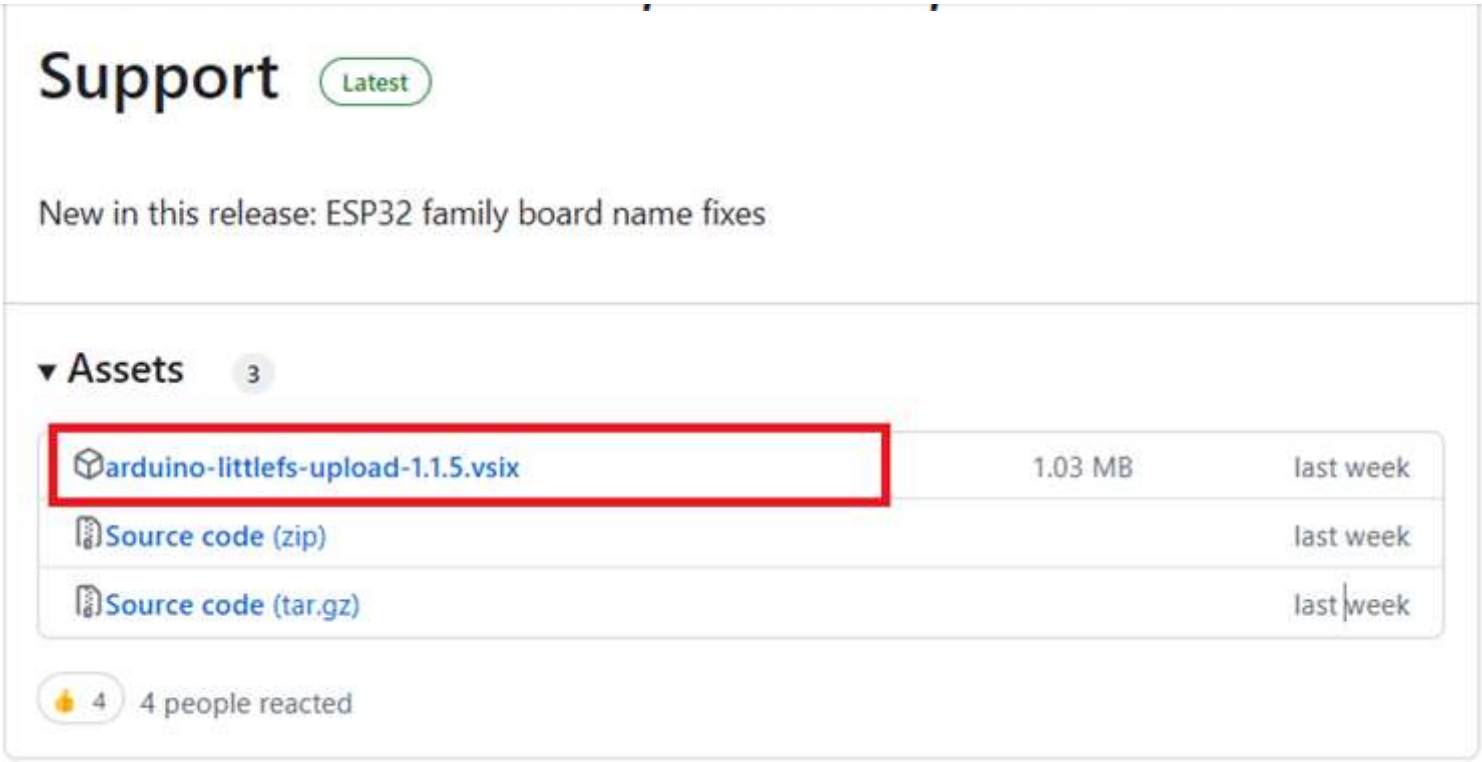
Instalando o plugin LittleFS Uploader no Arduino IDE 2

Para enviar arquivos para o ESP32 no Arduino IDE 2 , usaremos este plugin LittleFS Uploader que é compatível com o Arduino 2.2.1 ou superior e pode ser usado com as placas ESP32, ESP8266 e Raspberry Pi Pico .

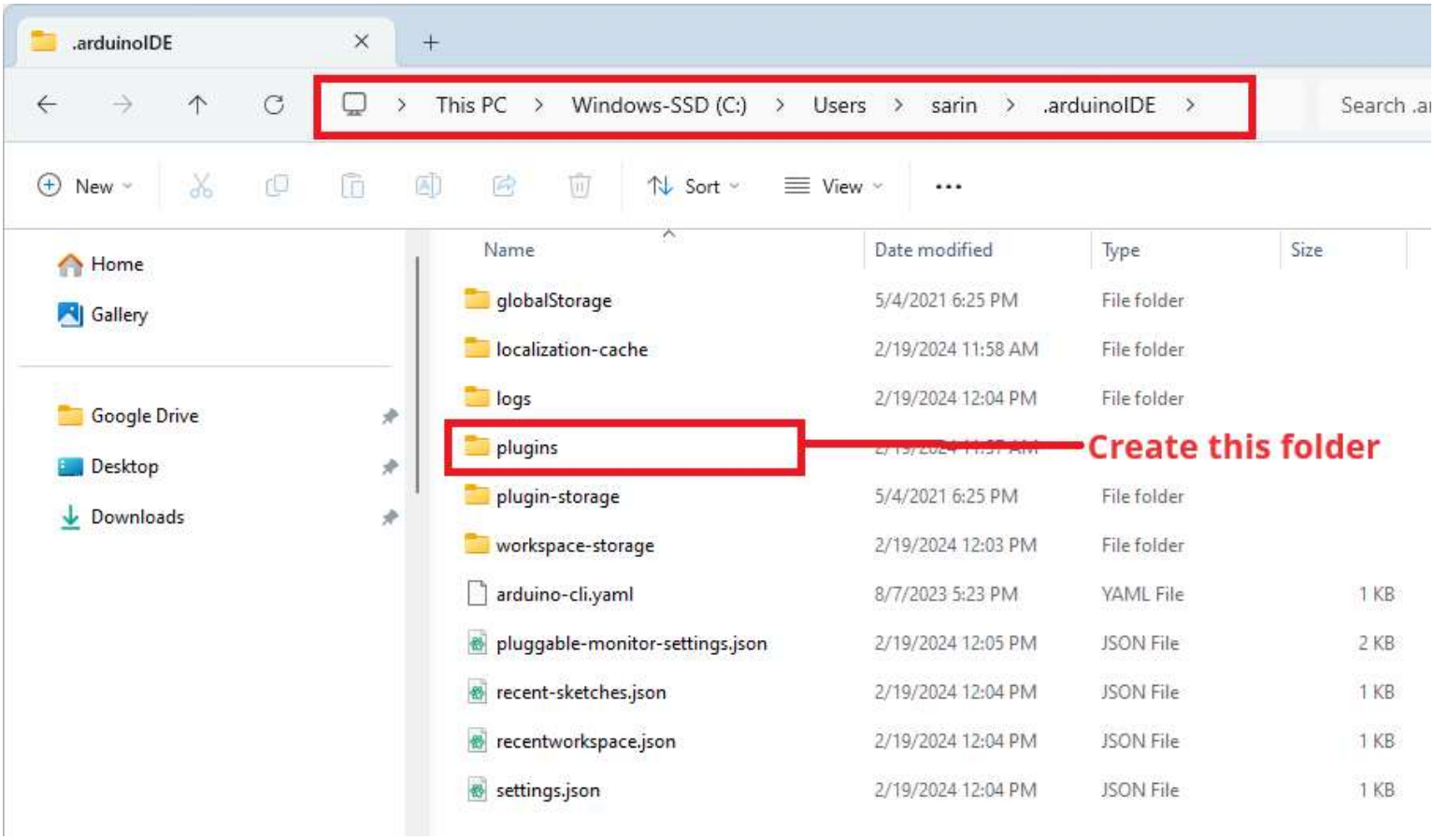
Instruções do Windows

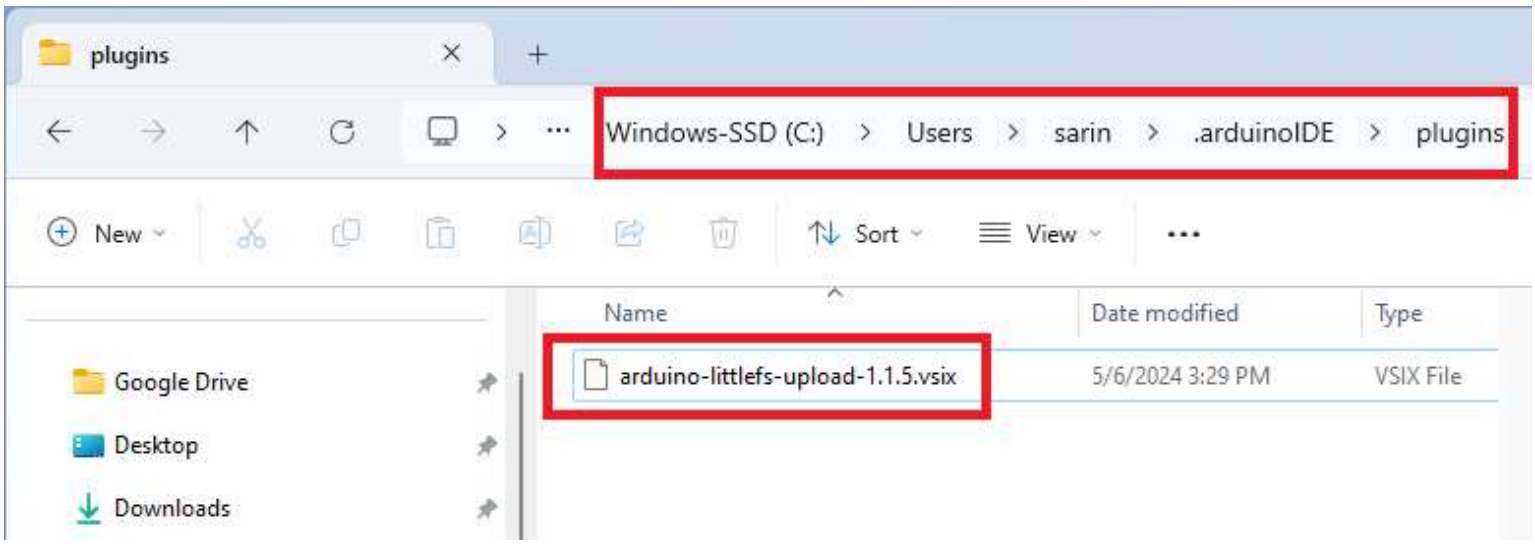
Siga os próximos passos para instalar o carregador de sistema de arquivos se estiver usando Windows (clique aqui para obter instruções sobre MacOS):

1) Vá para a página de lançamentos e clique no arquivo *.vsix* para fazer o download.

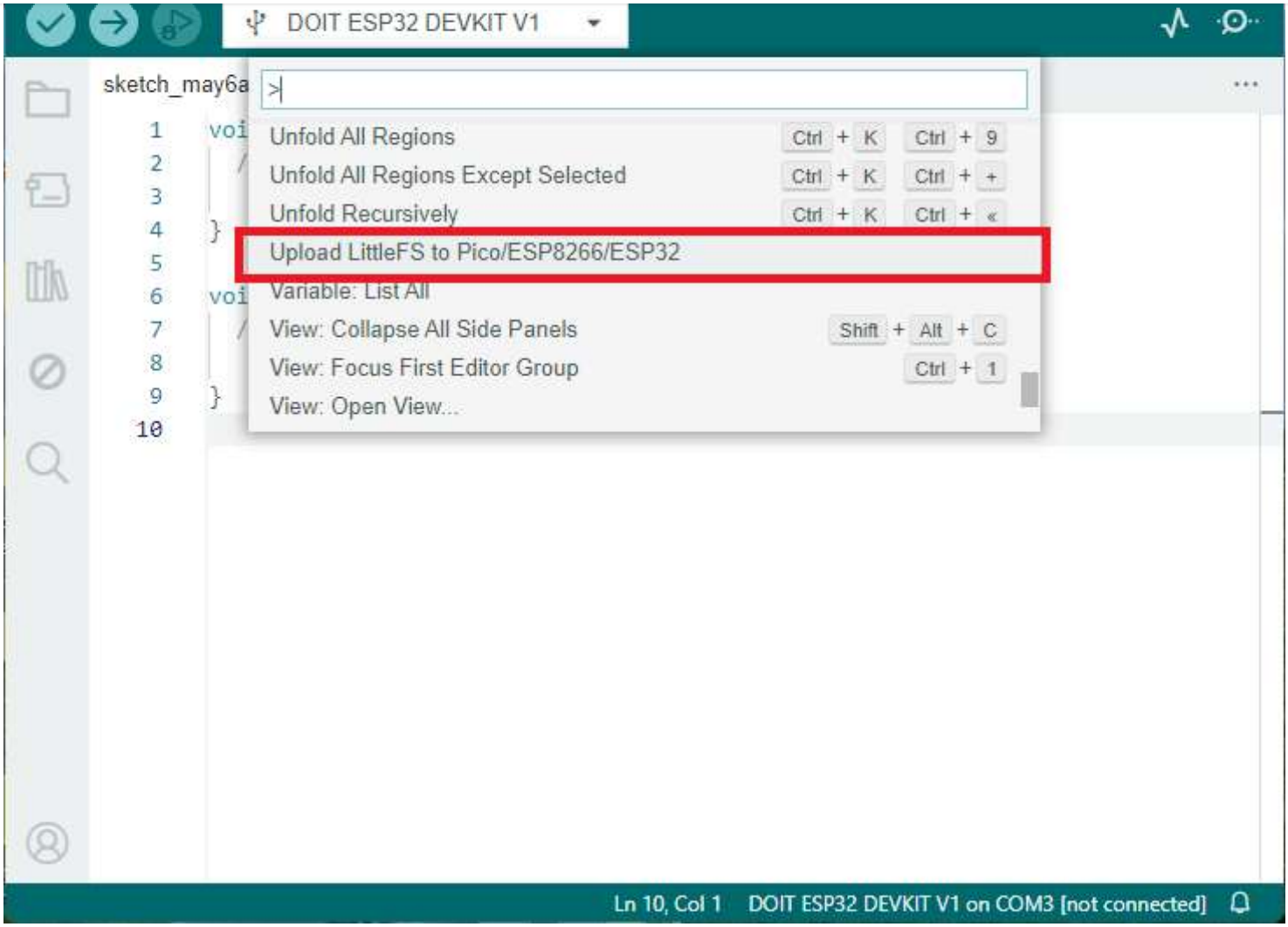


2) No seu computador, acesse o seguinte caminho: `C:\Usuários\<nome de usuário>\.arduinoIDE\`. Crie uma nova pasta chamada *plugins,* caso ainda não tenha feito isso.





4) Reinicie ou abra o Arduino IDE 2. Para verificar se o plugin foi instalado com sucesso, pressione [**Ctrl**] + [**Shift**] + [**P**] para abrir a paleta de comandos. Uma instrução chamada ' **Upload Little FS to Pico/ESP8266/ESP32** ' deve estar lá (basta rolar para baixo ou procurar o nome da instrução).

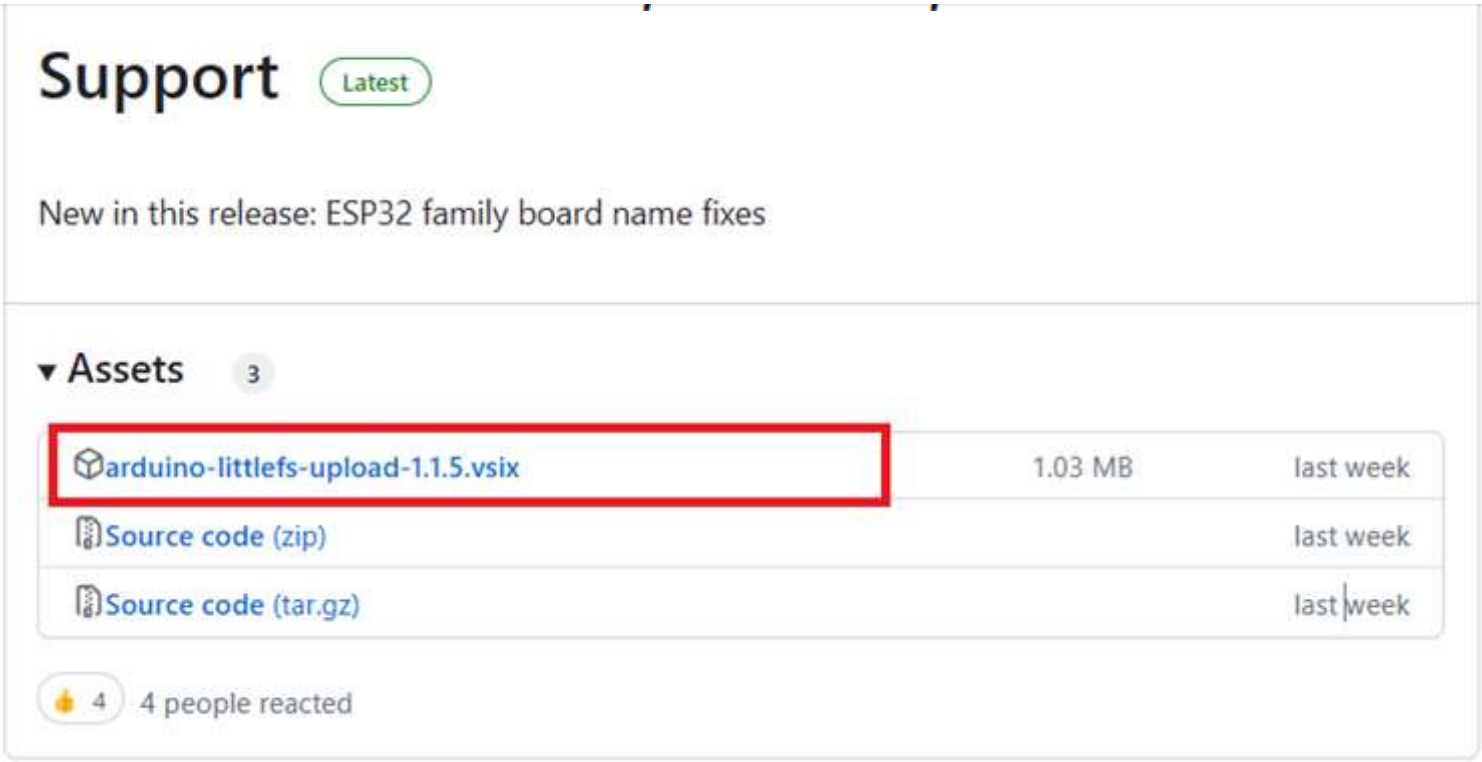


Isso significa que o plugin foi instalado com sucesso. Prossiga para esta seção para testar o plugin do carregador de sistema de arquivos.

Instruções do Mac OS X

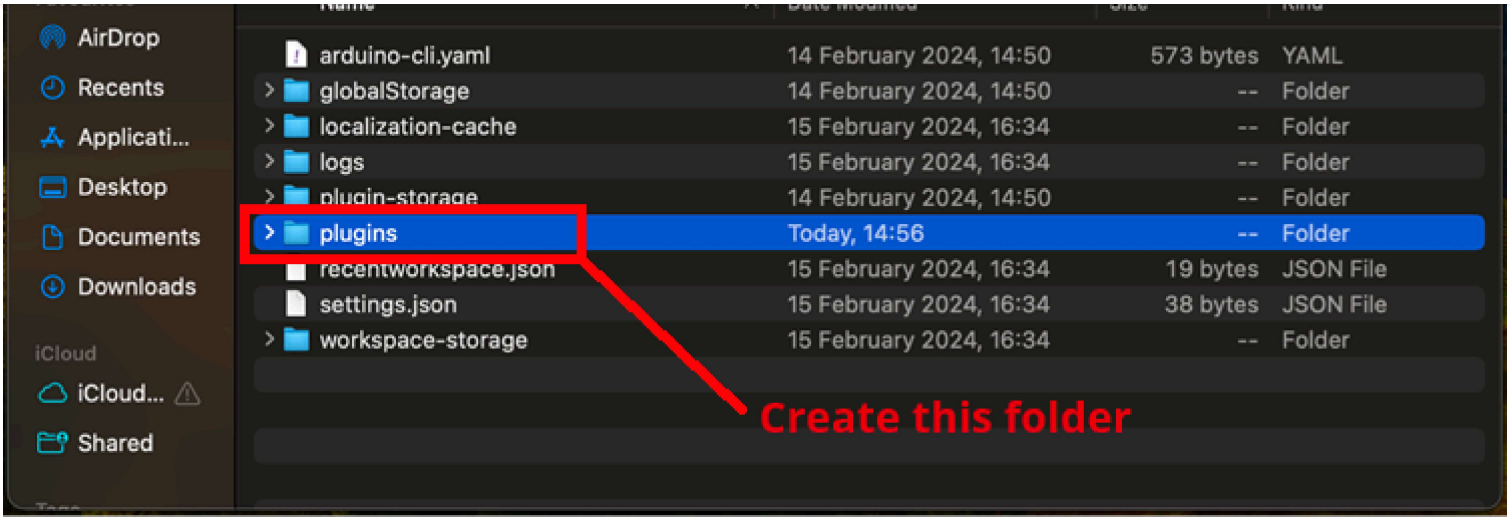
Siga os próximos passos para instalar o carregador do sistema de arquivos se estiver usando o Mac OS X:

1) Vá para a página de lançamentos e clique no arquivo *.vsix* para fazer o download.

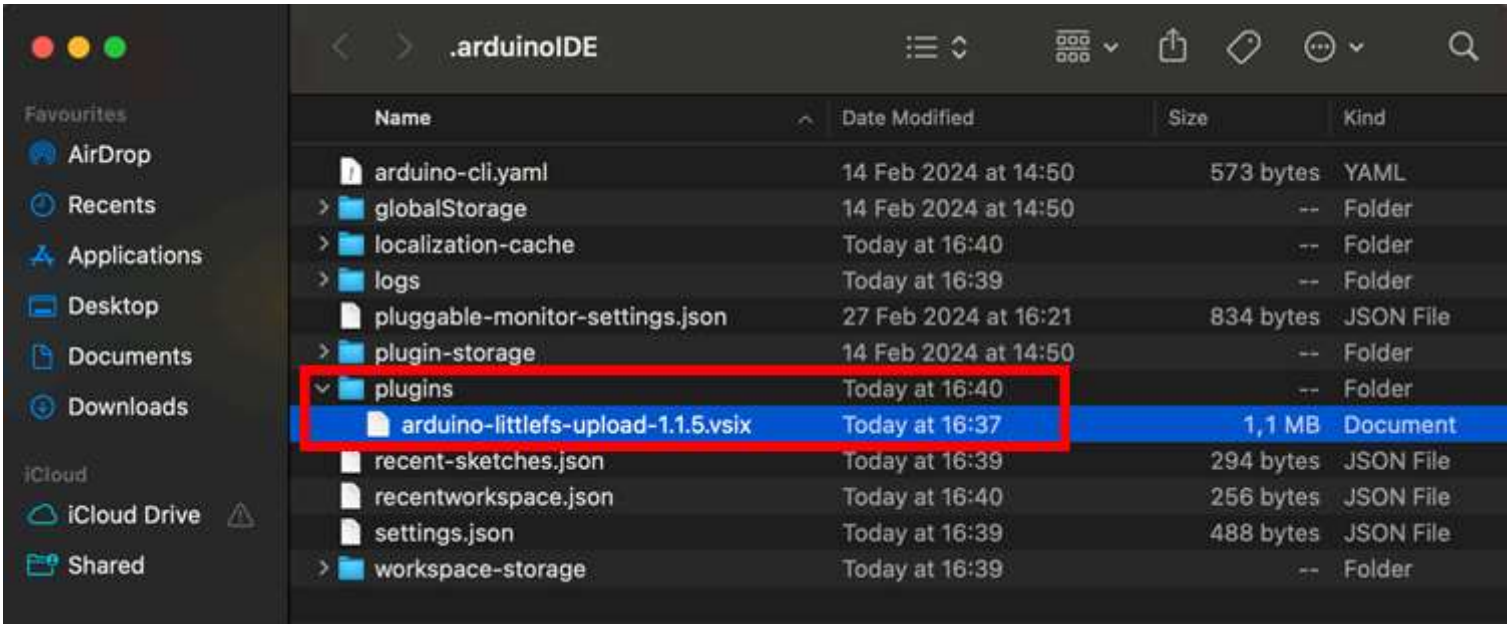


2) No Finder, digite `~/.arduinoIDE/` e abra esse diretório.

3) Crie uma nova pasta chamada *plugins* .



4) Mova o arquivo *.vsix* para a pasta *de plugins* (remova quaisquer outras versões anteriores do mesmo plugin, se for o caso).



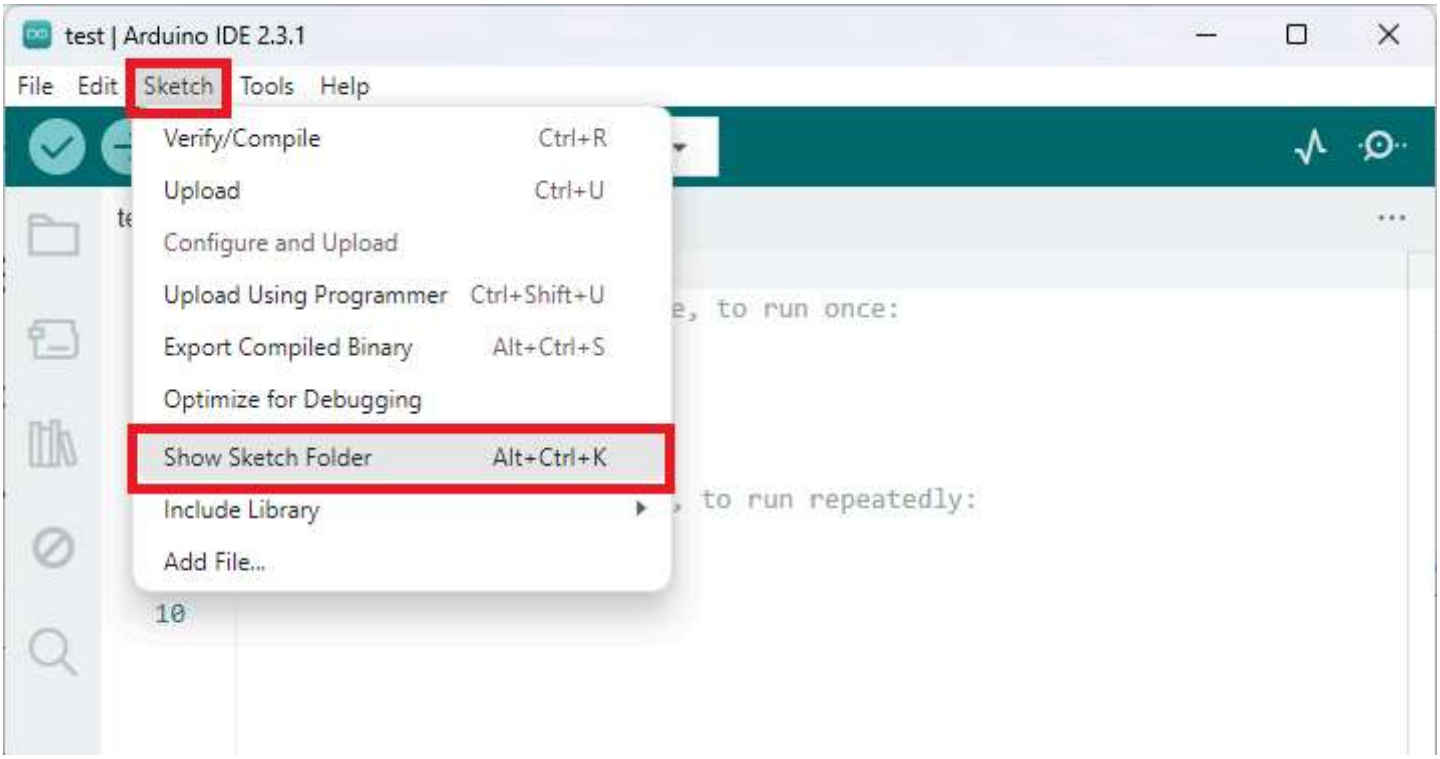
5) Reinicie ou abra o Arduino IDE 2. Para verificar se o plugin foi instalado com sucesso, pressione [⌘] + [**Shift**] + [**P**] para abrir a paleta de comandos. Uma instrução chamada ' **Upload LittleFS to Pico/ESP8266/ESP32** ' deve estar lá (basta rolar para baixo ou procurar o nome da instrução).

Carregando arquivos para o ESP32 usando o Filesystem Uploader no Arduino IDE 2

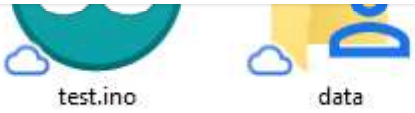
Para carregar arquivos no sistema de arquivos ESP32 LittleFS, siga as próximas instruções.

1) Crie um esboço do Arduino e salve-o. Para fins de demonstração, você pode salvar um esboço vazio.

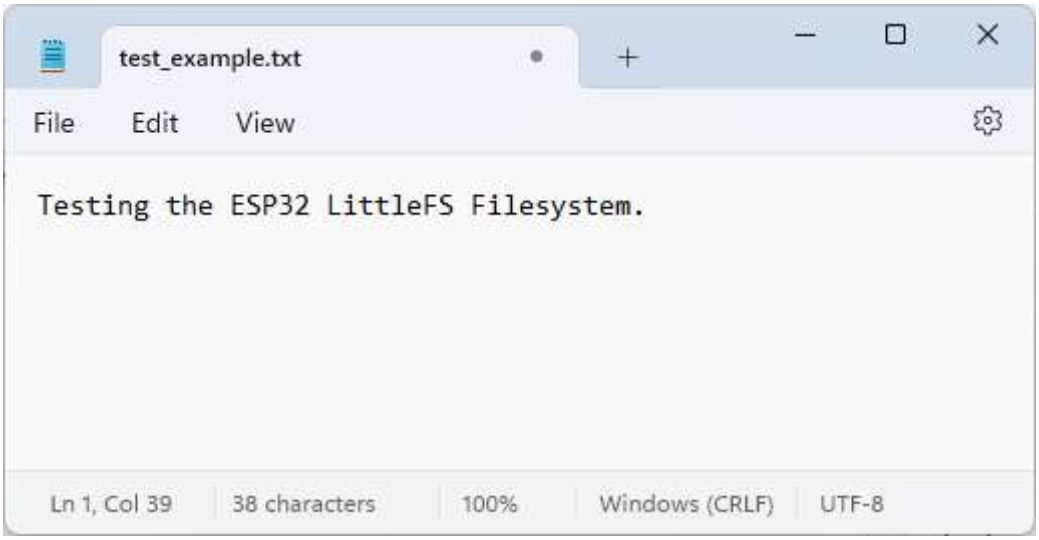
2) Então, abra a pasta de esboços. Você pode ir para **Sketch** > **Show Sketch Folder** . A pasta onde seu esboço está salvo deve abrir.



3) Dentro dessa pasta, crie uma nova pasta chamada ***dados*** .



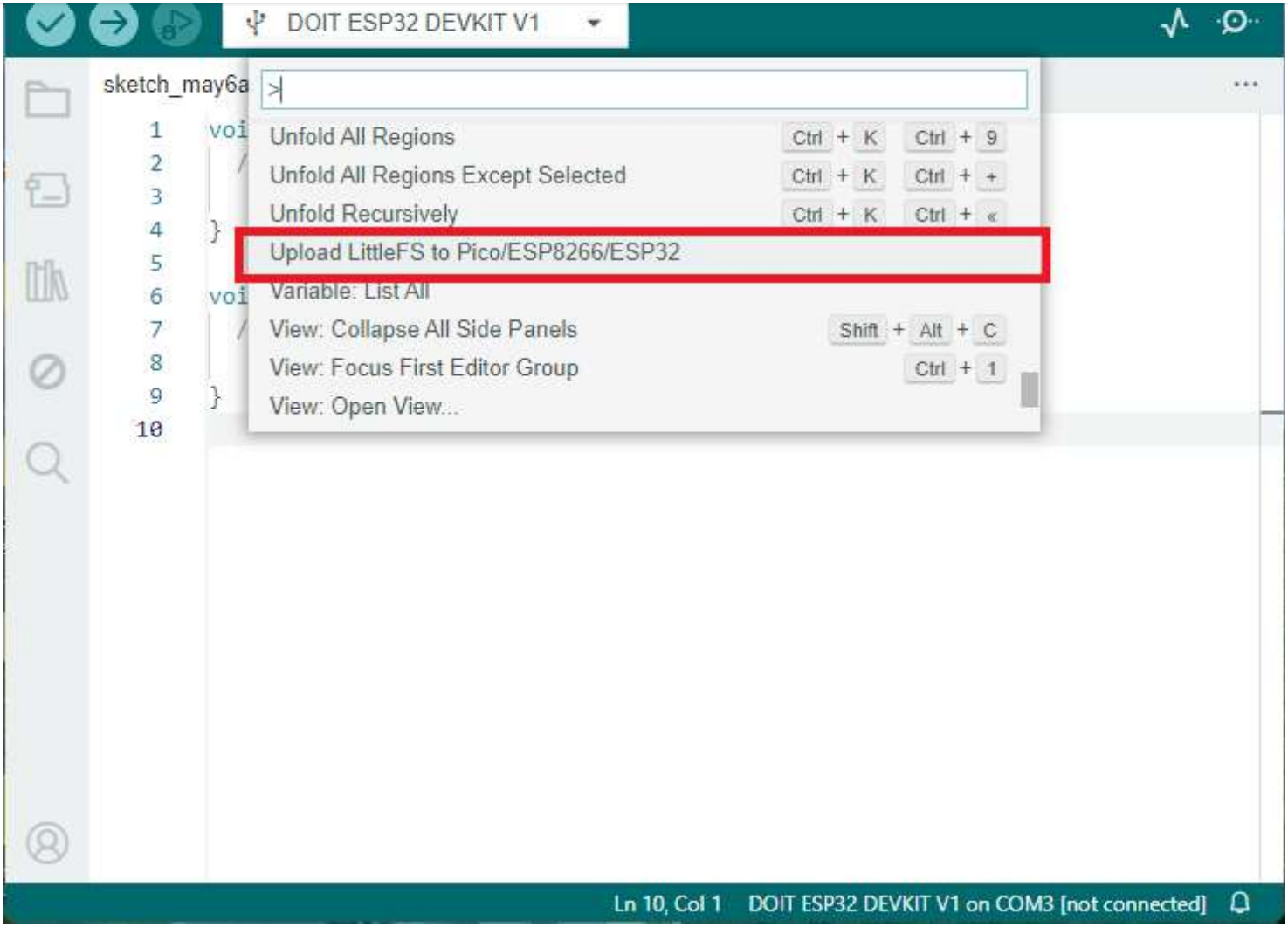
4) Dentro do ***dados*** pasta é onde você deve colocar os arquivos que deseja carregar para o sistema de arquivos ESP32. Como exemplo, crie um arquivo *.txt* com algum texto chamado *test_example* . *txt* (e salve-o dentro da pasta *data*).



5) Certifique-se de ter a placa correta (**Ferramentas** > **Placa**) e a porta COM selecionadas (**Ferramentas** > **Porta**).

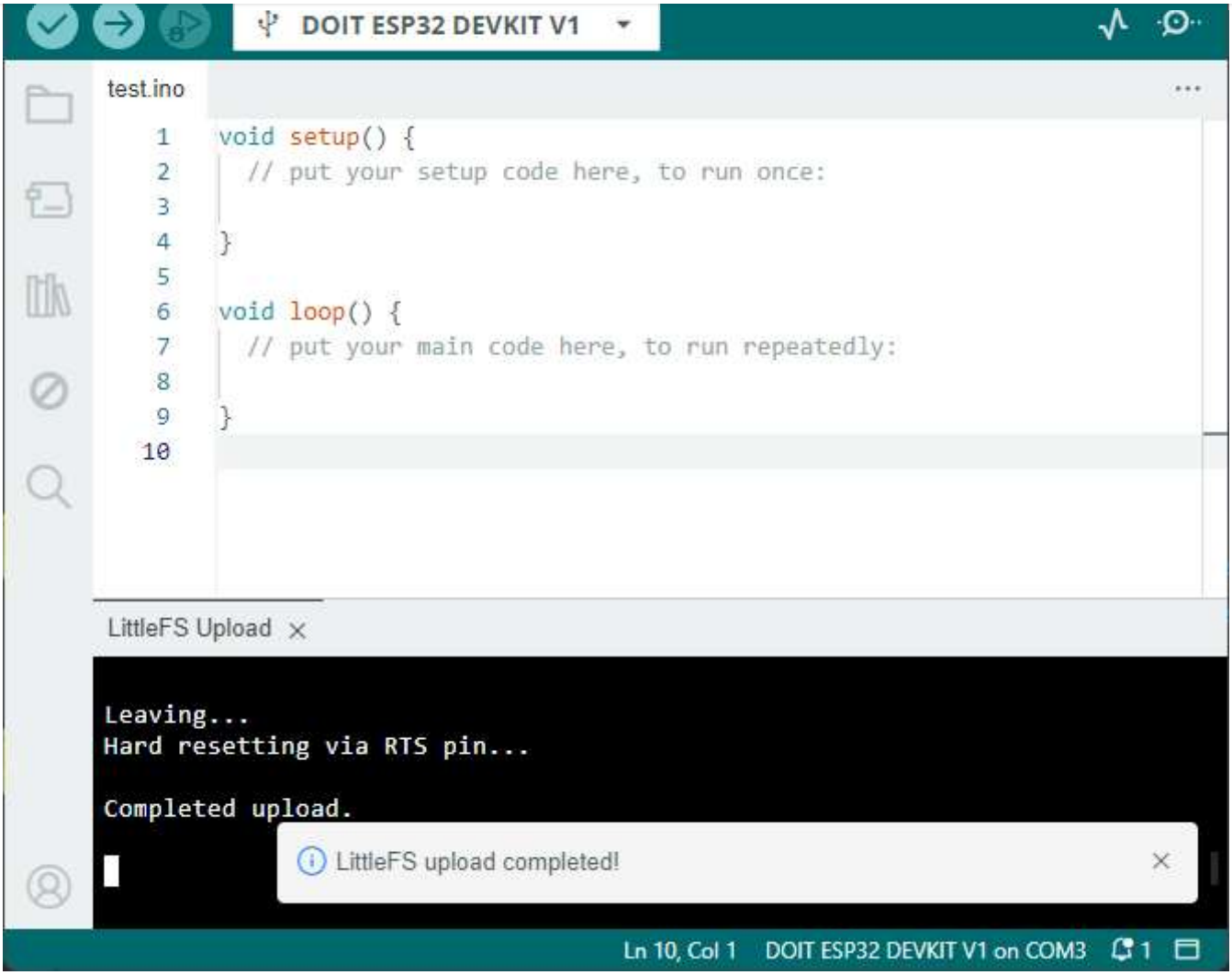
6) Depending on the ESP32 board selected, you may need to select the desired flash size (some boards don't have that option, don't worry). In the Arduino IDE, in **Tools** > **Flash size**, select the desired flash size (this will depend on the size of your files).

7) Then, upload the files to the ESP32 board. Press [**Ctrl**] + [**Shift**] + [**P**] on Windows or [⌘] + [**Shift**] + [**P**] on MacOS to open the command palette. Search for the **Upload LittleFS to Pico/ESP8266/ESP32** command and click on it.



Important: ensure the Serial Monitor is closed. Otherwise, the upload will fail.

After a few seconds, you should get the message "**Completed upload.** ". The files were successfully uploaded to the ESP32 filesystem.



Troubleshooting

If you get the following error message “ERROR: No port specified, check IDE menus“, restart the Arduino IDE, and try again.

Testing the ESP32 LittleFS Uploader

Now, let’s check if the file was saved into the ESP32 filesystem. Upload the following code to your ESP32 board. This code will read the contents of the *.txt* file you saved previously on LittleFS.

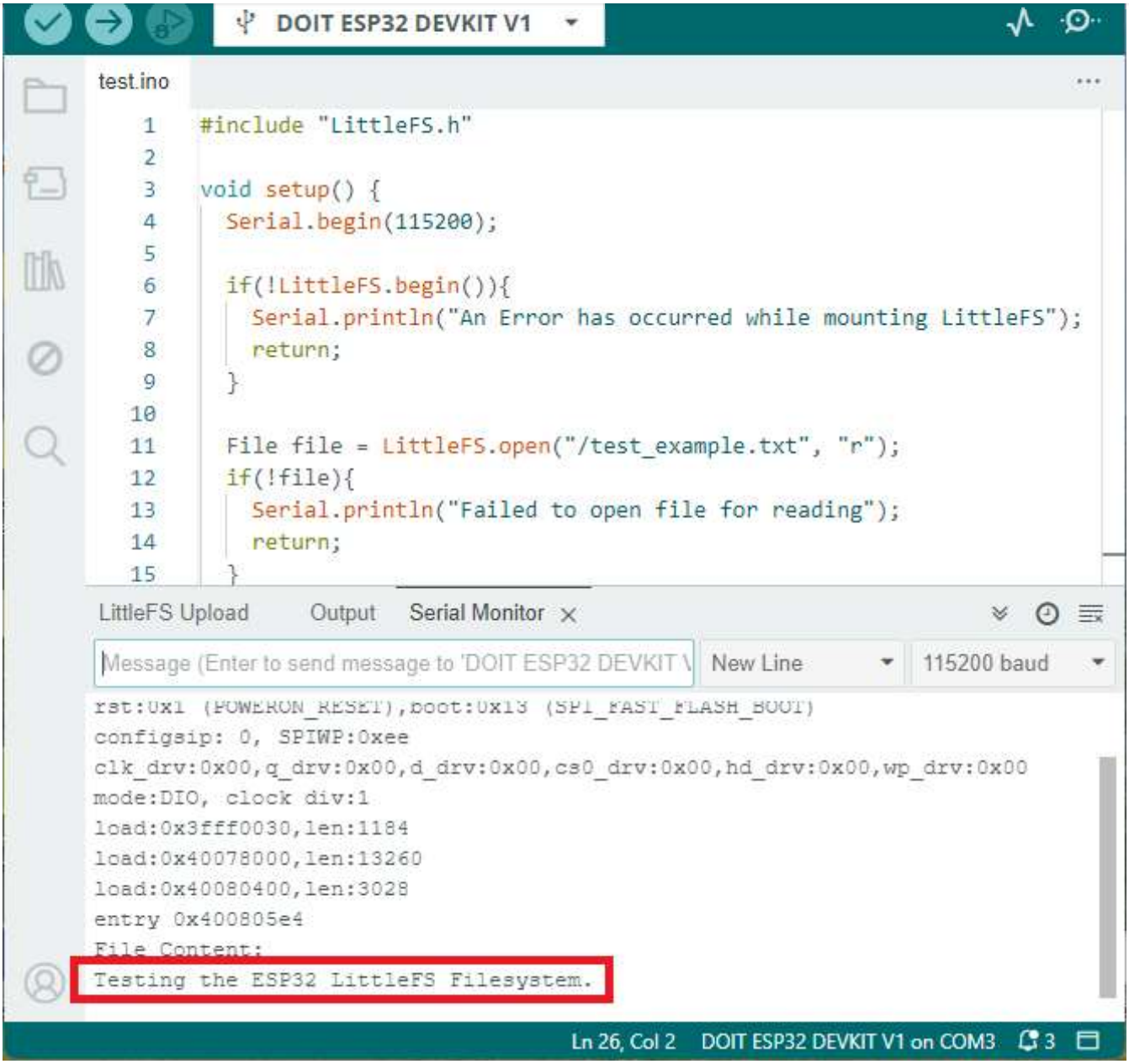
```
#include "LittleFS.h"

void setup() {
  Serial.begin(115200);
```

```
    return;
  }

  File file = LittleFS.open("/test_example.txt", "r");
  if(!file){
    Serial.println("Failed to open file for reading");
    return;
  }

  Serial.println("File Content:");
  while(file.available()){
    Serial.write(file.read());
  }
  file.close();
}

void loop() {

}
```

View raw code

After uploading, open the Serial Monitor at a baud rate of 115200.

Press the ESP32 on-board “RST” button. It should print the content of your *.txt* file in the Serial Monitor.



You've successfully uploaded files to the ESP32 filesystem using the plugin.

Wrapping Up

In this tutorial, we've shown you how to upload files to the ESP32 LittleFS filesystem on Arduino IDE 2 using an uploader plugin.

We've shown you how to upload a *.txt* file, but you can upload other file formats like HTML, CSS, and Javascript files to build a web server, images, or small icons, save configuration files, etc.

To learn more about the ESP32, check our resources:

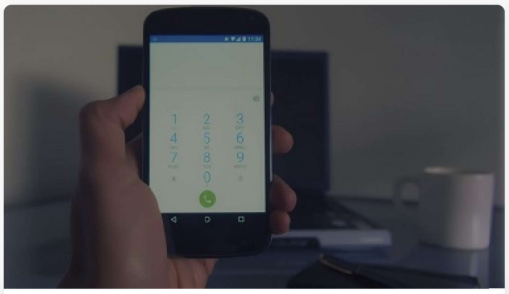
- Build Web Servers with ESP32 and ESP8266
- Firebase Web App with ESP32 and ESP8266
- More ESP32 Tutorials and Guides

Thanks for reading.

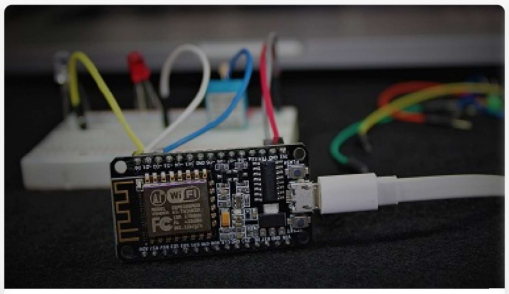
SMART HOME with Raspberry Pi, ESP32, ESP8266 [eBook]

Learn how to build a home automation system and we'll cover the following main subjects: Node-RED, Node-RED Dashboard, Raspberry Pi, ESP32, ESP8266, MQTT, and InfluxDB database **DOWNLOAD »**

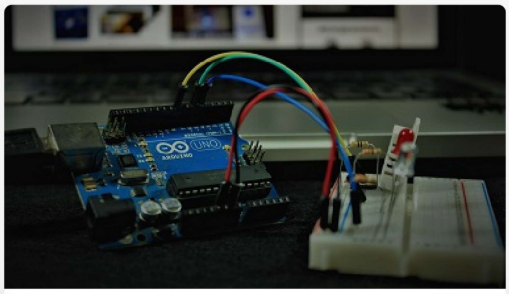
Recommended Resources



Build a Home Automation System from Scratch



Home Automation using ESP8266 eBook and video

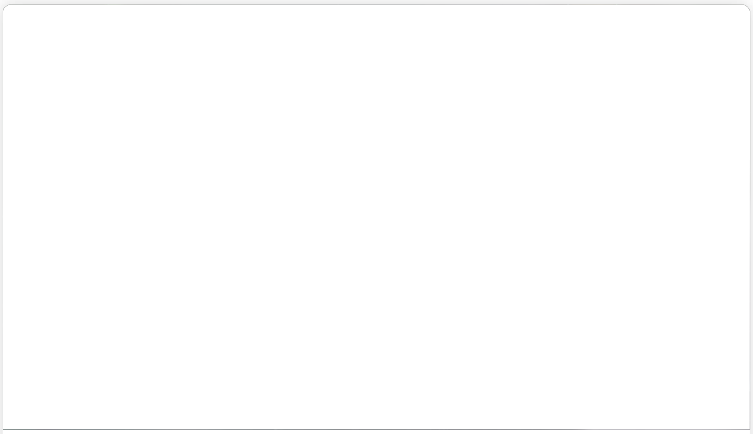


Arduino Step-by-Step Projects » Build 25 Arduino

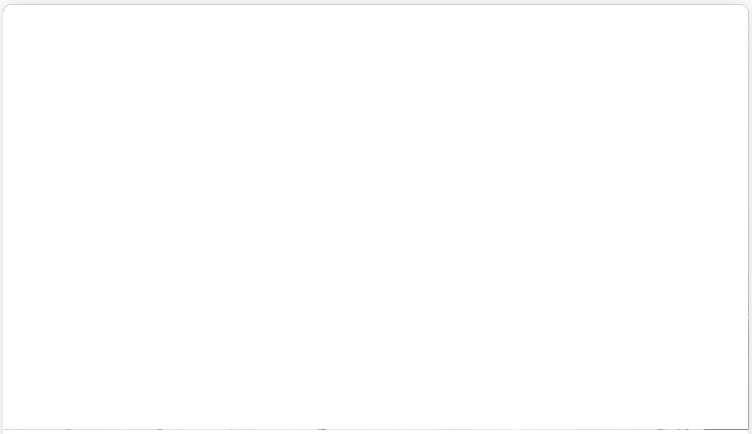
Node-RED.

experience!

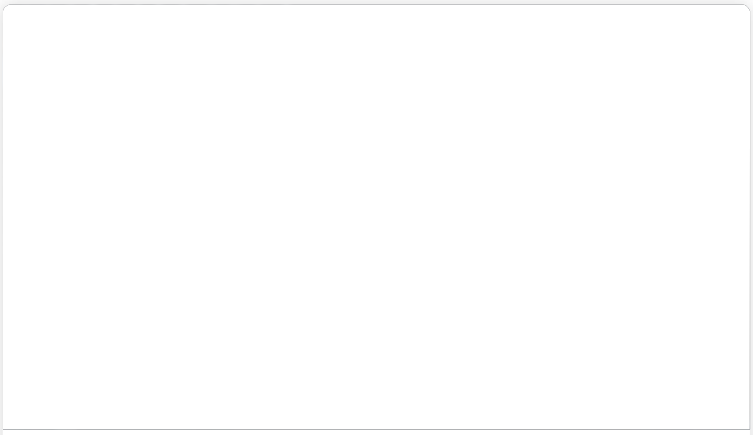
What to Read Next…



ESP8266 NodeMCU: Getting Started with InfluxDB



ESP32 MQTT – Publish DHT11/DHT22 Temperature and Humidity Readings (Arduino IDE)



Programming ESP32 with Atom Text Editor and PlatformIO IDE

ESP32 LittleFS Uploader (Upload Files to the Filesystem)”

Geralt Griffiths
May 6, 2024 at 4:57 pm

Thank you so much for this!

As someone who used LittleIFS for a couple of projects I did way-back, I found it so frustrating to read messages online that said how difficult it would be to support LittleFS in IDE2, if – indeed – it would be possible at all.

I will read your article with interest and make rude cyber-gestures at the nay-sayers!

Reply

Sara Santos
May 7, 2024 at 8:56 am

Support for the ESP32 was added very recently to the Arduino IDE 2.
Regards,
Sara

Reply

Mark D
May 6, 2024 at 5:48 pm

As usual a great tutorial which worked floorlessly and has answered alot of questions that I had about sketches that needed the data uploaded to the board.
It took me a long time to dicover that older older arduino Ide supported this function and now you have explained wonderfully how to upload LittleFs in latest Arduino IDE.
I thankyou and the team and look forward to your tutorials and the information from your site.
Thanks from the UK
Mark D

Reply

Eddie
May 6, 2024 at 10:22 pm

Wow, finally the LittleFS is available on Arduino IDE 2.0! Excelent post, by the way, the quality of the tutorials remains the same when I found this site, 5 yars ago: excelent!

Reply

Sara Santos
May 7, 2024 at 9:07 am

Thank you 😀
Regards,
Sara

Reply

meebox
May 10, 2024 at 4:10 am

Reply

Sara Santos
May 10, 2024 at 9:53 am

For spiffs, you have to use the old Arduino IDE.
https://randomnerdtutorials.com/install-esp32-filesystem-uploader-arduino-ide/
Regards,
Sara

Reply

Martin Maly
May 12, 2024 at 4:31 pm

See here: https://github.com/espx-cz/arduino-spiffs-upload

Reply

meebox
May 13, 2024 at 12:21 pm

cool.

Reply

May 13, 2024 at 6:45 am

Use this for Arduino IDE V2: https://github.com/espx-cz/arduino-spiffs-upload

Reply

Jop van der Werff
May 10, 2024 at 7:22 am

Hi Sara and Rui,

I was very happy to see your new tutorial about upload LittleFS now is implemented in the Arduino IDE 2.

I'm working on MAC and stuck on one problem. My upload to LitteFS ends with this error:

A fatal error occurred: Could not open /dev/cu.usbserial-1420, the port doesn't exist
ERROR: Upload failed, error code: 2

Maybe it has to do with an open Serial-monitor connection. In the old IDE 1, you had to close this connection before starting upload of data and this was done very simple by just closing the (distinct) window were Serial-monitor was dumping it's data.
In the IDE 2 it's not very clear how to close Serial-monitor. As Serial monitor is in the bottom panel of my sketch-window.

Here the complete log output of my uploading to LittleFS:

Using partition: default
Building LittleFS filesystem
/Users/jop/Library/Arduino15/packages/esp32/tools/mklittlefs/3.0.0-gnu12-dc7f933/mklittlefs -c /Users/jop/STACK/ArduinoSketches/@KENNIS/LittleFS/TestLittleFS/data -p 256 -b 4096 -s 1441792 /var/folders/vd/rhwhbr1s5j77gkn7yq_qfmlm0000gn/T/tmp-7661-

Uploading LittleFS filesystem
/Users/jop/Library/Arduino15/packages/esp32/tools/esptool_py/4.5.1/esptool –chip esp32 –port /dev/cu.usbserial-1420 –baud 921600 –before default_reset –after hard_reset write_flash -z –flash_mode dio –flash_freq 80m –flash_size detect 2686976 /var/folders/vd/rhwhbr1s5j77gkn7yq_qfmlm0000gn/T/tmp-7661-q2cK0GA8qGrt-.littlefs.bin
esptool.py v4.5.1
Serial port /dev/cu.usbserial-1420

A fatal error occurred: Could not open /dev/cu.usbserial-1420, the port doesn't exist

ERROR: Upload failed, error code: 2

The port cu.usbserial-1420 does exist, as I used it to upload my test sketch.
My macOS is Sonoma.

With kind regards,

Jop

Reply

Martin Maly
May 12, 2024 at 4:30 pm

Close Serial terminal first, then try it again.

Reply

Martin Maly
May 12, 2024 at 2:04 pm

Reply

Sara Santos
May 13, 2024 at 9:49 am

Great!
Thanks for sharing.
Regards,
Sara

Reply

meebox
November 22, 2024 at 3:13 am

With both the ESP32 and ESP8266 package installed on the same IDE, the plugin always finds the mkspiffs tool in the ESP32 package. This would be a problem while uploading SPIFFS to ESP8266 since the mkspffs tool is not compatible with ESP8266. I've made a PR to fix it.

Reply

Doug Leppard
May 25, 2024 at 6:52 pm

FYI, I got this to work in the sample with a XIAO ESP32C3.
Thanks

Kevin B
May 31, 2024 at 5:03 pm

I have Arduino 2.3.2 on Windows. I followed the instructions and reopened the IDE, did the Fn+CTRL+P and the file uploader is not on the list!

First I put the latest release (arduino-littlefs-upload-1.1.7.vsix) in the plugins folder, with the following path: C:\Users\kevin.arduinoIDE\plugins. When it didn't work, I deleted that file and replaced it with the version mentioned in the tutorial (arduino-littlefs-upload-1.1.5.vsix) but it still did not work. The board I selected is ESP32 Dev Module. Any ideas?

Reply

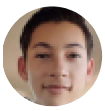
Freddy
August 30, 2024 at 9:50 am

i have the same problem like you. Sometimes i get the message "Upload LittleFS to Pico/ESP8266/esp32" on the list, but only once. When i try a second time, the message is no more there.

Reply

Michel
June 8, 2024 at 5:14 pm

Hello.
Here is a list created by the community containing plugins compatible with the Arduino IDE 2.x: github.com/MicSG-dev/list-of-compatible-plugins-with-arduino-ide-2-x



Michel
June 20, 2024 at 9:33 am

Link: github.com/MicSG-dev/list-of-compatible-plugins-with-arduino-ide-2-x/

Reply

qualia universe
June 20, 2024 at 5:09 am

Only site that gives actual solution.

Reply

Al
June 20, 2024 at 2:12 pm

Hello,
Thanks for the useful tutorials you keep sharing 🙂

I used the tool. Upload went well but it erased (overwote ?) the NVS preferences on the board.
Do you know how to upload /data to littleFS and keep “preferences” untouched ?
Thanks

Reply

June 20, 2024 at 2:33 pm

Hi.
Thanks for your feedback.
I didn't know about that.
I searched for a while and found people with the same issue, but no answers… I don't know how to fix that issue.
If you find a solution, please share.
Regards,
Sara

Reply

Al
June 23, 2024 at 11:18 am

I tried on another board (Mini D1 ESP32) and this time upload did not erase preferences. Same Arduino IDE and tool version.
I could not find a systematic way to reproduce the problem ! It is strange.

Reply

RU183
July 7, 2024 at 12:50 am

Activating extension 'arduino-littlefs-upload' failed: Cannot find module 'c:\Users……arduinoIDE\plugins\arduino-littlefs-upload-1.1.8\out\extension.js' Require stack: – C:\Users…….\AppData\Local\Programs\arduino-ide\resources\app\lib\backend\plugin-host.js

ARDUINO IDE 2.3.2
esptool 4.5.1
ESP32 S3 WROOM
For Gyver project WIFI panel matrix on WS2812b 1024 LEDS

Reply

Fu
August 1, 2024 at 5:44 am

The previous steps were executed according to your steps, but an error message appeared :
ERROR: Partition entry not found in csv file!
How to fix it?

Reply

Sara Santos
August 6, 2024 at 8:41 am

Hi.
When do you get that error?
What board are you selecting in Tools > Board?

Regards,
Sara

Reply

August 11, 2024 at 8:38 am

Try replacing partitions.csv with this one: github.com/espressif/arduino-esp32/blob/master/tools/partitions/default.csv

Reply

hans

August 5, 2024 at 5:55 pm

As always, this place is my first rescue whenever I run into an Arduino related problem and once again I found a crystal clear tutorial that works. Thank you so much!

Reply

Edzman

August 29, 2024 at 9:24 pm

I have a question about Step #7. Before asking, I would like to say this website has helped me a lot when building my projects. Thanks in advance for your help.

QUESTION:

What would happen if we DON’t perform step #7:

“Then, upload the files to the ESP32 board. Press [Ctrl] + [Shift] + [P] on Windows or [⌘] + [Shift] + [P] on MacOS to open the command palette. Search for the Upload LittleFS to Pico/ESP8266/ESP32 command and click on it.”

…and just programatically write:

then we serialize some JSON data to the file "example_data.json"

and finally we close the file using “file.close();”

Wouldnt that technically CREATE the file if it doesnt exist?

Or its mandatory to ALWAYS perform manually step#7 in ArduinoIDE 2.0?

I mean, I know we need to create a “data” folder at the root of our sketch, but do we also NEED to manually create the file we want to manipulate and ALSO perform step #7?

Can step #7 be performed programatically inside the sketch?

Thank you!!

Reply

Jorge Vila
September 9, 2024 at 7:52 pm

I have a problem
The plugin doesn’t work
Wisdows 10 (PT_BR) + Arduino IDE 2.3.2 + arduino-littlefs-upload-1.1.8.vsix
When I press SHIFT+CTRL+P some options appear, but not “Upload Little FS to Pico/ESP8266/ESP32”
Any tips to solve the problem?
or
Any other way to send data to spiffs

Reply

Sara Santos
September 9, 2024 at 9:34 pm

Make sure you're placing the file on the right folder.
It works well for us.
If it doesn't work, it's better to post an issue in the developer github page:
https://github.com/earlephilhower/arduino-littlefs-upload/issues
Regards,
Sara

Reply

Ramon
September 19, 2024 at 10:22 am

I have the same problem, I have checked the folder and its destination and the file.
but the option does not appear.
did you solve the problem??
Arduino 2.3.2

Reply

flat four
January 10, 2025 at 12:16 pm

I have the same problem. Has it been solved?
I have followed the instructions carefully and placed the "arduino-littlefs-upload-1.5.3.vsix" in the .arduinoIDE/plugins directory. restarted the IDE, hit CTRL/SHIFT/P and cannot see the upload function in the palette.

Reply

October 22, 2024 at 5:35 am

Hello, excellent tutorial that has helped me a lot to solve doubts and problems that I had, I have managed to load the files into the esp32 memory but I cannot read them. I can read the files that I create directly from the sketch, but not the imported ones. Any advice?
All the best

Reply

Yves
November 21, 2024 at 5:04 pm

Hello, i have installed the plugin on arduino ide 2.3 (windows11). When i use the command “upload little fs” i get an error about the baud rate option :

esptool: error: argument –baud/-b: invalid arg_auto_int value: ‘NaN’
ERROR: Upload failed, error code: 2

Any idea ?
Best Regards
Yves

Reply

Leave a Comment

Nome *

E-mail *

Site

Notify me of follow-up comments by email.

Notify me of new posts by email.

Post Comment

Divulgação de Afiliados: *Random Nerd Tutorials é um participante de programas de publicidade de afiliados projetados para fornecer um meio para ganharmos taxas ao vincular a Amazon, eBay, AliExpress e outros sites. Podemos ser compensados por encaminhar tráfego e negócios para essas empresas.*

Aprenda ESP32 com o Arduino IDE eBook » Guia completo para programar o ESP32 com o Arduino IDE!

CASA INTELIGENTE com Raspberry Pi, ESP32 e ESP8266 » aprenda a construir um sistema completo de automação residencial.

Aprenda Raspberry Pi Pico/Pico W com MicroPython » O guia completo de introdução para aproveitar ao máximo a placa microcontroladora Raspberry Pi Pico/Pico W (RP2040) usando a linguagem de programação MicroPython.

🔥**Aprenda LVGL: Crie GUIs para projetos ESP32 »** Aprenda a criar interfaces gráficas de usuário (GUIs) para projetos ESP32 usando LVGL (Light Versatile Graphics Library) com o Arduino IDE.