

# PlatformIO Nibble プログラミングガイド

## PlatformIOを使用したプログラミング

### 新しいプロジェクトの作成

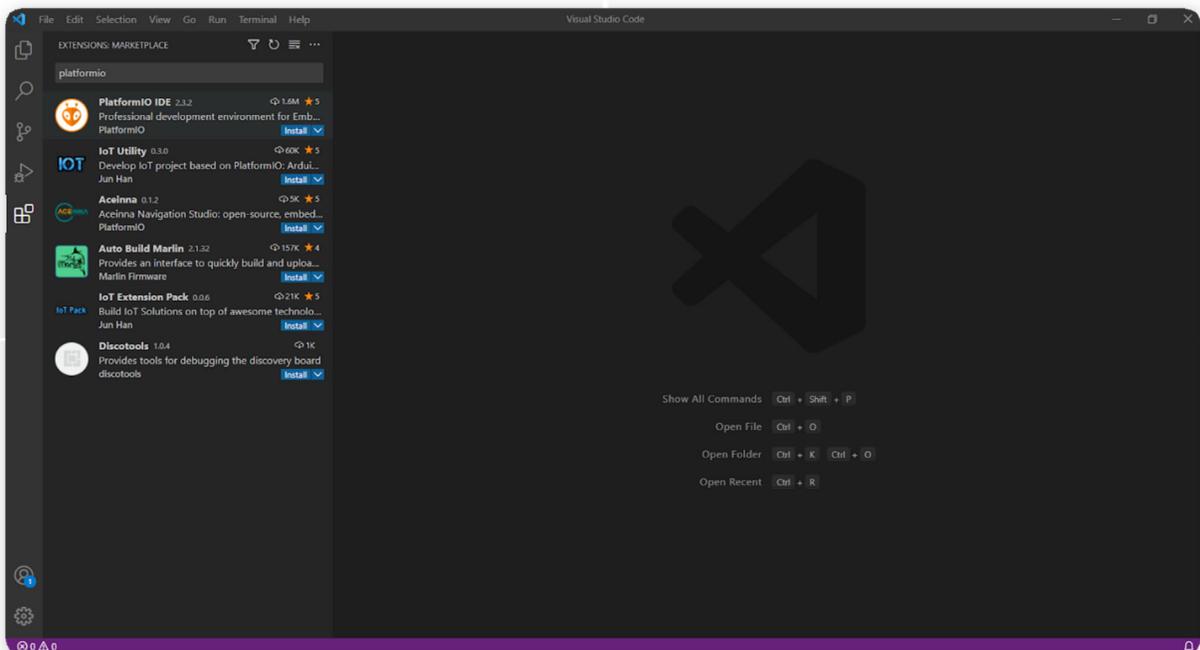
PlatformIO および VSCode Nibbleプログラミングガイドへようこそ。

PlatformIOとVSCodeを使って、Nibbleにコードをアップロードする方法を説明します。さっそく始めましょう。

### インストール

始める前に、**Visual Studio Code（無料）**をダウンロードしてインストールしてください。ダウンロードは[こちら](#)から。

[Extensions（拡張機能）] タブに移動し、「PlatformIO」と入力します。次に、[Install（インストール）] をクリックします。



#### PlatformIO IDE

PlatformIO ↓ 3.5M  
platformio.org

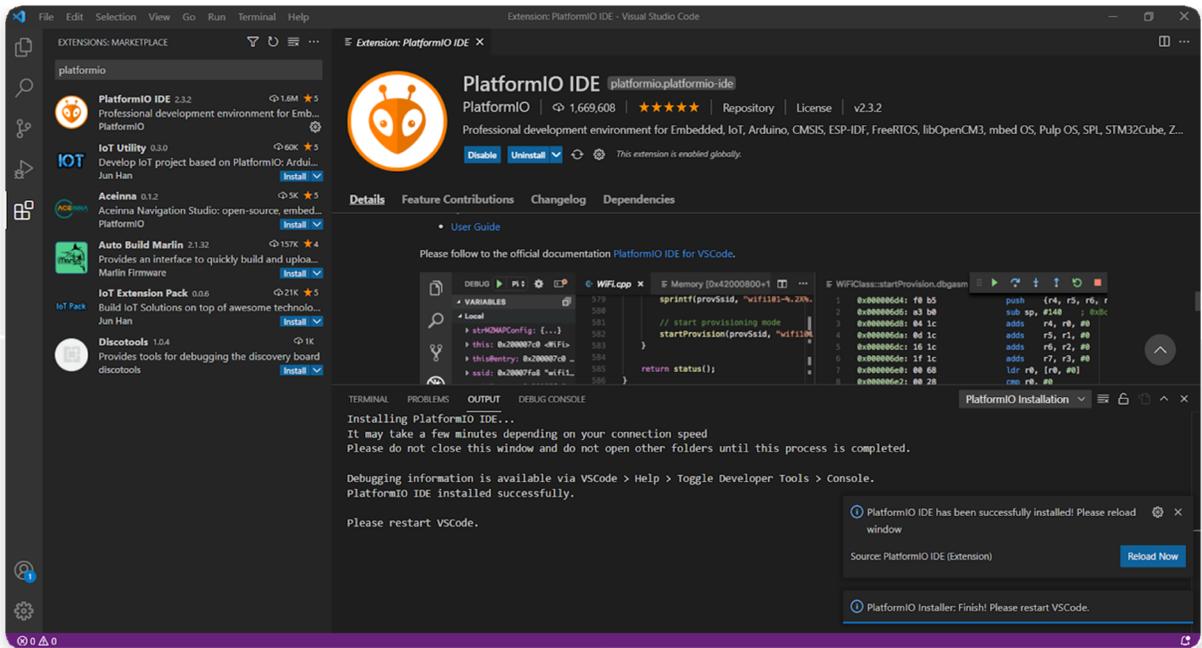
Professional development environment for Embedded, IoT, Arduino, CMSIS, ESP-ID...

★★★★★

FREE

PlatformIO IDEの横にある [Install] をクリック

PlatformIOからプロンプトが表示されたら、VSCodeを再起動します。



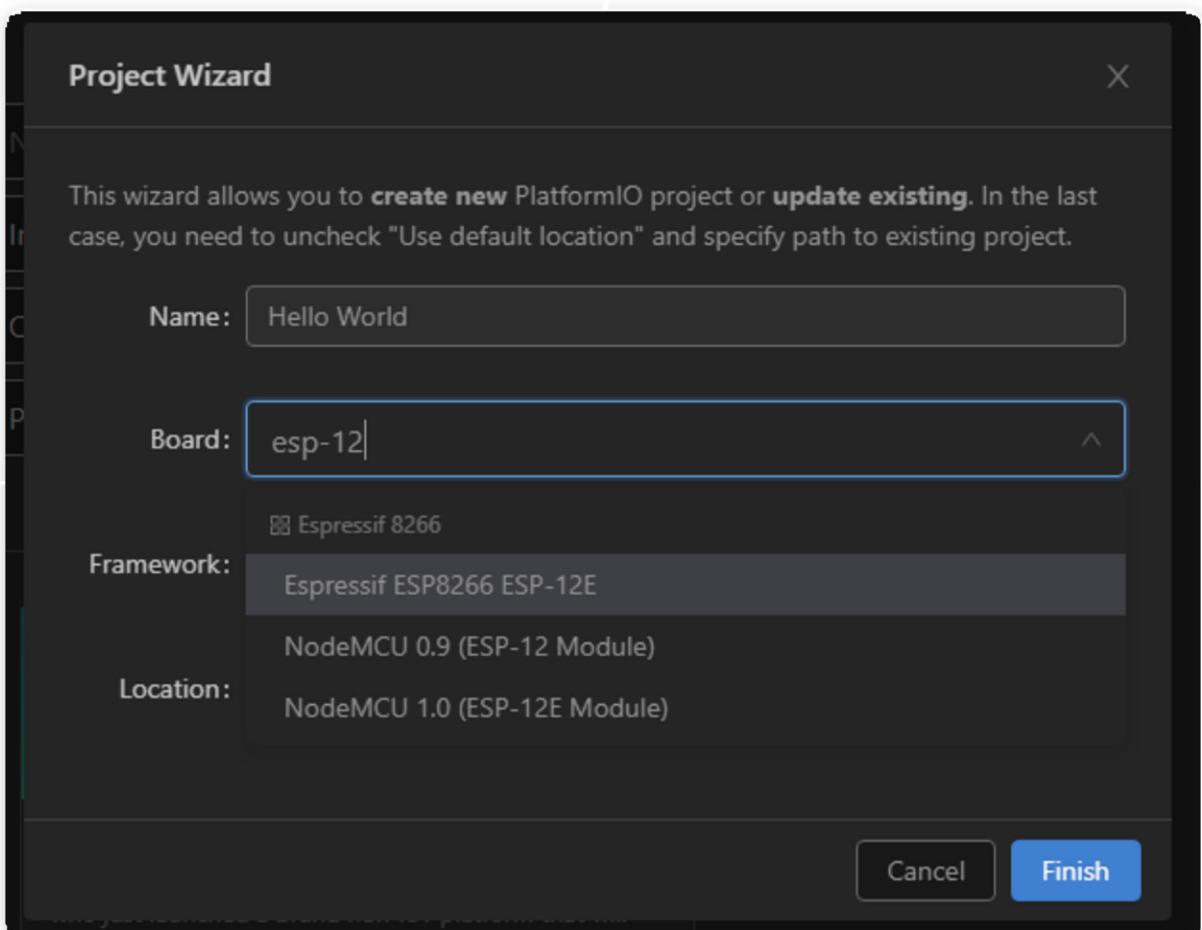
## VSCoideの再起動

## 新しいプロジェクトの作成

PlatformIOのホームページに移動したら、[New Project (新しいプロジェクト)] を選択します。その後、プロジェクトの名前を決める画面が表示されますので、クリエイティブな名前をつけてください。簡単なHello Worldから始めます。

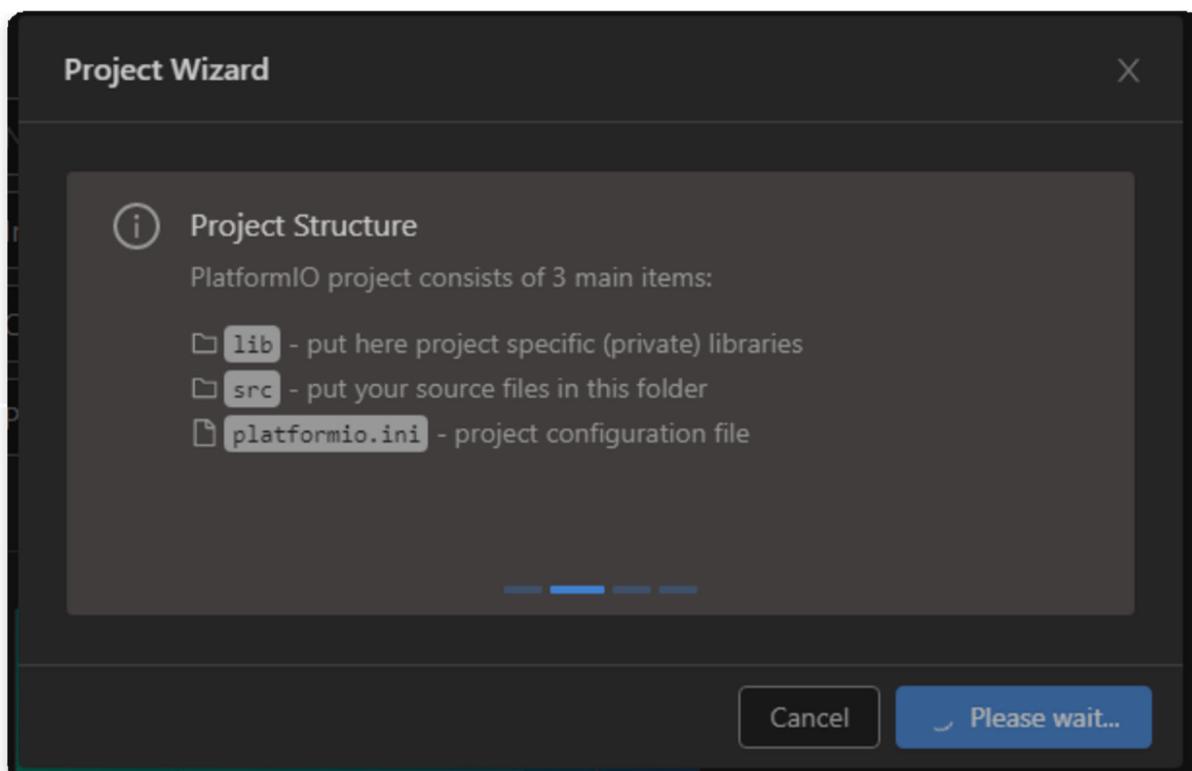
「esp-12」と入力し、ボードのドロップダウンメニューから[Espressif ESP8266 ESP-12E]を選択します。

フレームワークのオプションはArduinoのままにしておきます。WindowsのデフォルトのプロジェクトパスはDocuments/PlatformIO/Projectsですが、好きな場所に保存してください。



## 新しいプロジェクトフォームに記入

[Finish (完了)] をクリックした後、ボードファイルがダウンロードされるまで少し待ちます。



お待ちください

## 簡単なプログラムの作成

空のPlatformIOプロジェクトを新規に作成したら、次のステップに進みます。

### 準備

PlatformIOのプロジェクト構成は以下の通りです。

- platformio.ini (プロジェクト構成ファイル)
- ソースコードを配置する src ディレクトリ (\*.h, \*.c, \*.cpp, \*.S, \*.ino など)
- lib ディレクトリは、プロジェクト固有の (プライベート) ライブラリに使用できます。

ここで、必要なライブラリや設定ファイルをGitHubのリポジトリからダウンロードし、プロジェクトディレクトリに配置する必要があります。

### 簡単なプログラムの作成

簡単なプログラムを書いてみましょう。簡単な例を次に示します。

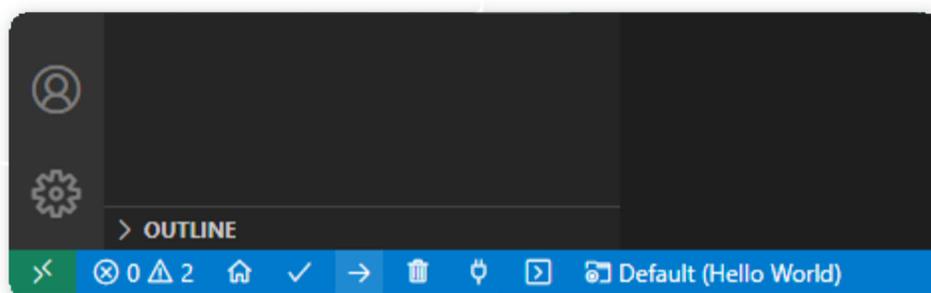
```
1 #include
2 #include
3
4 Display* display;
5 Sprite* sprite;
6
7 void setup()
8 { Nibble.begin();
9   display = Nibble.getDisplay();
10  sprite = display->getBaseSprite();
11
12  sprite->clear(TFT_BLACK);
13  sprite->setTextColor(TFT_WHITE);
14  sprite->setTextFont(2);
```

```
15  sprite-&gt;setCursor(0,0);
16  sprite-&gt;print("Hello World!");
17  display-&gt;commit();
18
19  }
20
21  void loop() {
22    delay(1000);
23  }
```

## 次のステップ

Nibbleの電源を入れ、USBケーブルでコンピューターに接続します。

画面左下のアップロードボタン（右向きの矢印）をクリックします。



右向きの矢印をクリック

Nibbleにコードをアップロードすることができました！