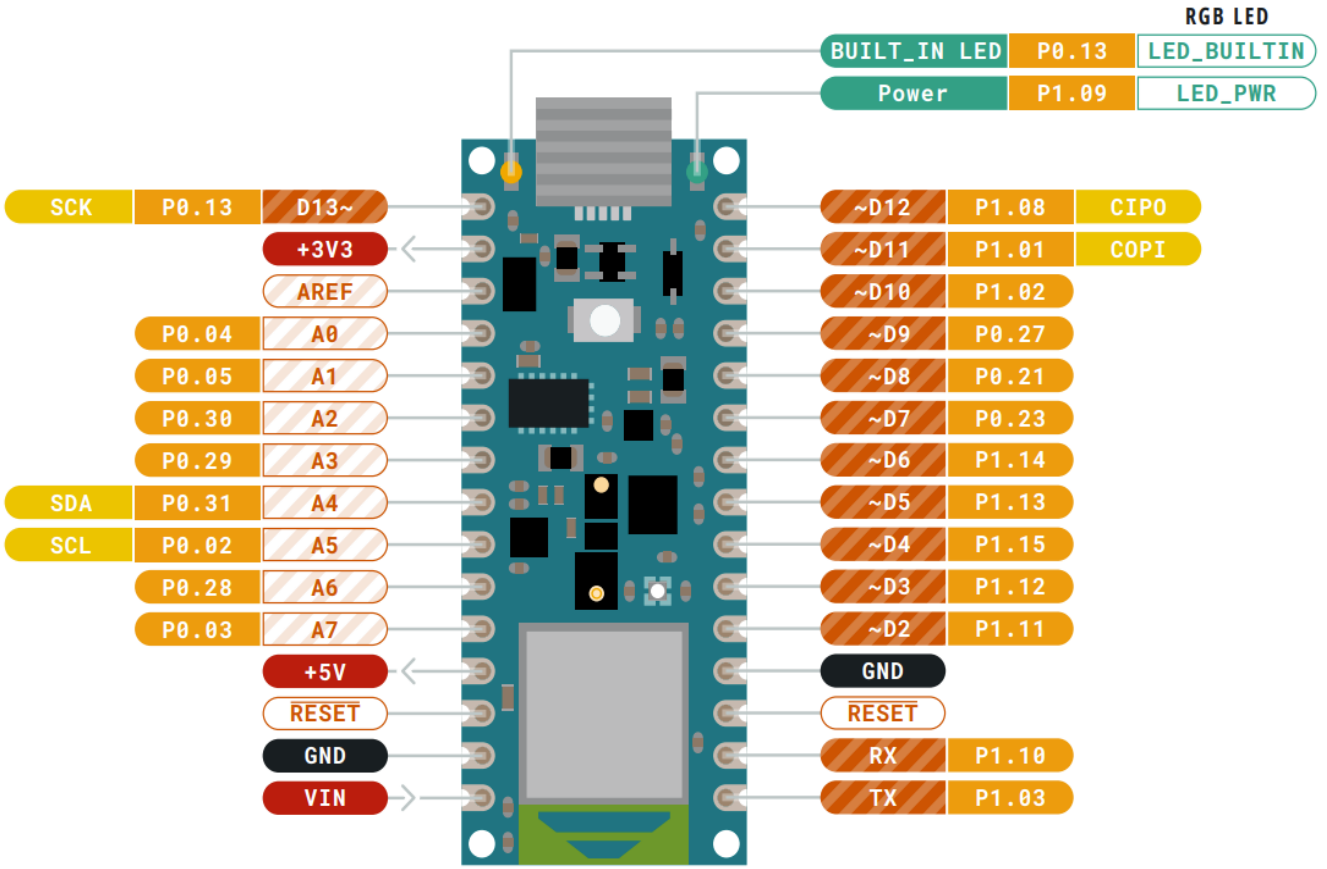
EE-105
Arduino Installation

8/29/2024

Arduino Nano 33 BLE



What is the Arduino Nano 33 BLE?

- The Arduino Nano 33 BLE is a small and powerful microcontroller board designed for building projects that need sensors, Bluetooth connectivity, and more. Think of it as a tiny computer you can program to do cool things like controlling LEDs, reading sensors, or connecting to your phone wirelessly.

- **Bluetooth Low Energy (BLE):** This allows it to communicate wirelessly with devices like your smartphone.

- **Built-in Sensors:** It comes with several sensors, including: A motion sensor to detect movement like shaking or tilting.

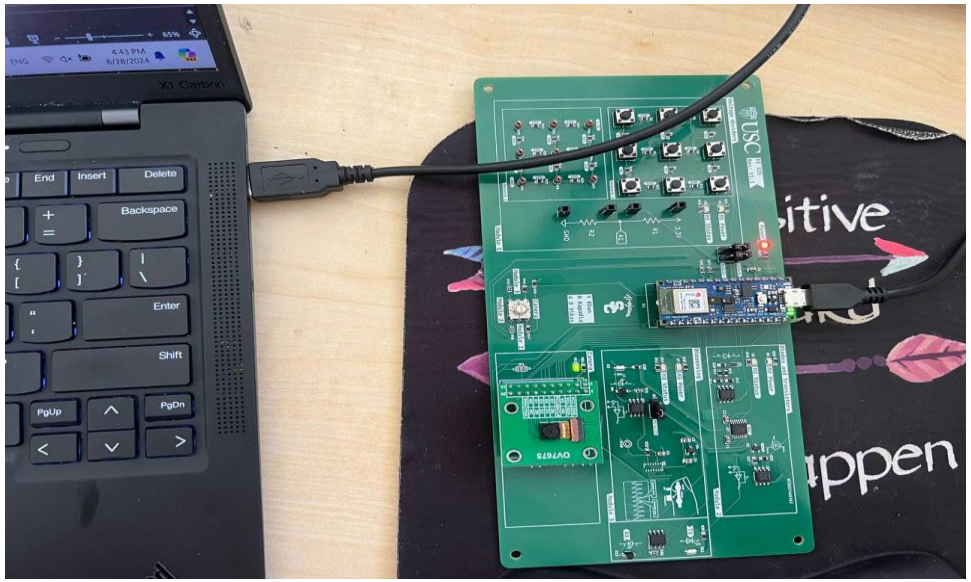
- Temperature and humidity sensors for environment monitoring.

- **Programming:** You can program it using the Arduino IDE, which is beginner-friendly and supports many examples.

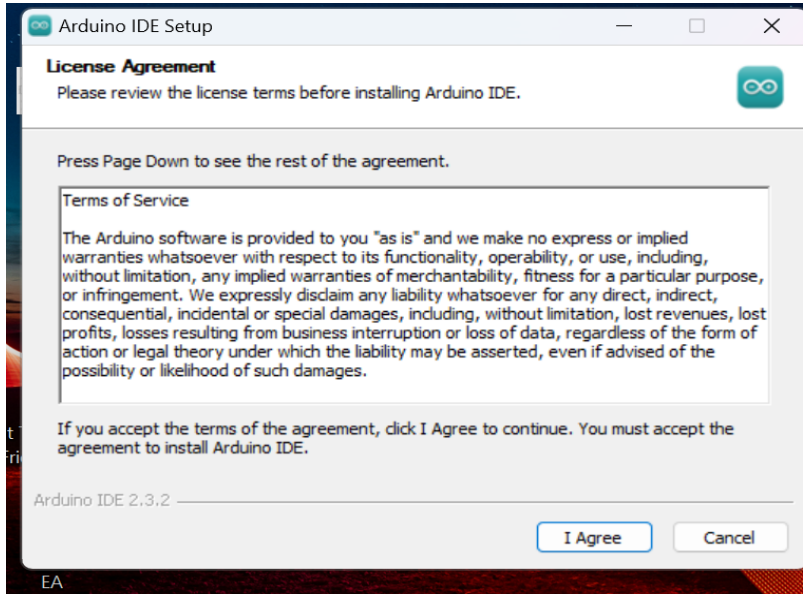
Installing the Arduino IDE

1. Connect the USB cable to Arduino

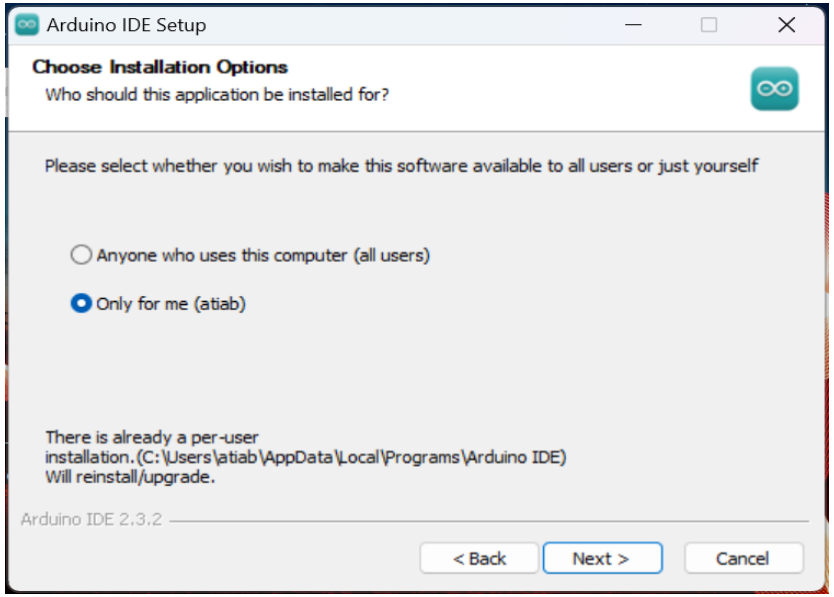
2. Download the latest release of the Arduino IDE here
https://support.arduino.cc/hc/en-us/articles/360019833020-Download-and-install-Arduino-IDE



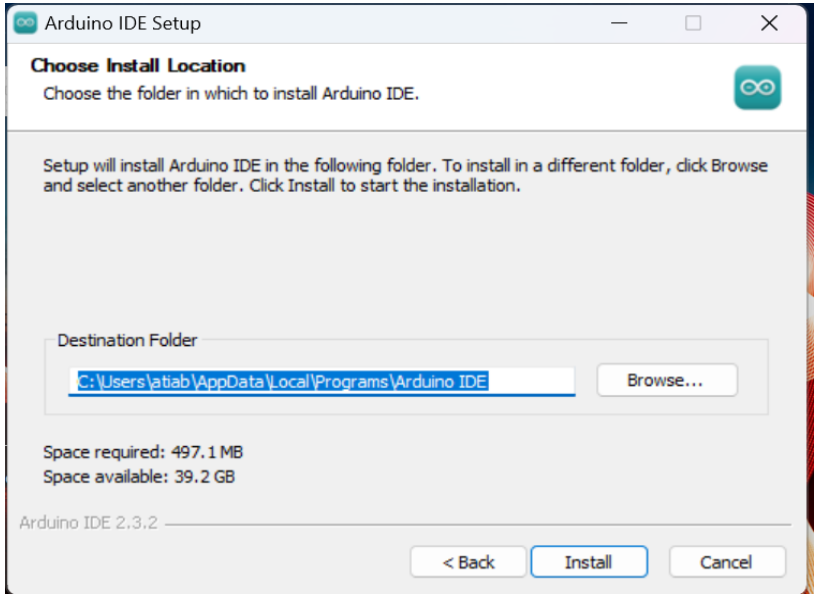
1


Arduino IDE Setup
License Agreement
Please review the license terms before installing Arduino IDE.
Press Page Down to see the rest of the agreement.
Terms of Service
The Arduino software is provided to you "as is" and we make no express or implied warranties whatsoever with respect to its functionality, operability, or use, including, without limitation, any implied warranties of merchantability, fitness for a particular purpose, or infringement. We expressly disclaim any liability whatsoever for any direct, indirect, consequential, incidental or special damages, including, without limitation, lost revenues, lost profits, losses resulting from business interruption or loss of data, regardless of the form of action or legal theory under which the liability may be asserted, even if advised of the possibility or likelihood of such damages.
If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install Arduino IDE.
Arduino IDE 2.3.2
I Agree
Cancel

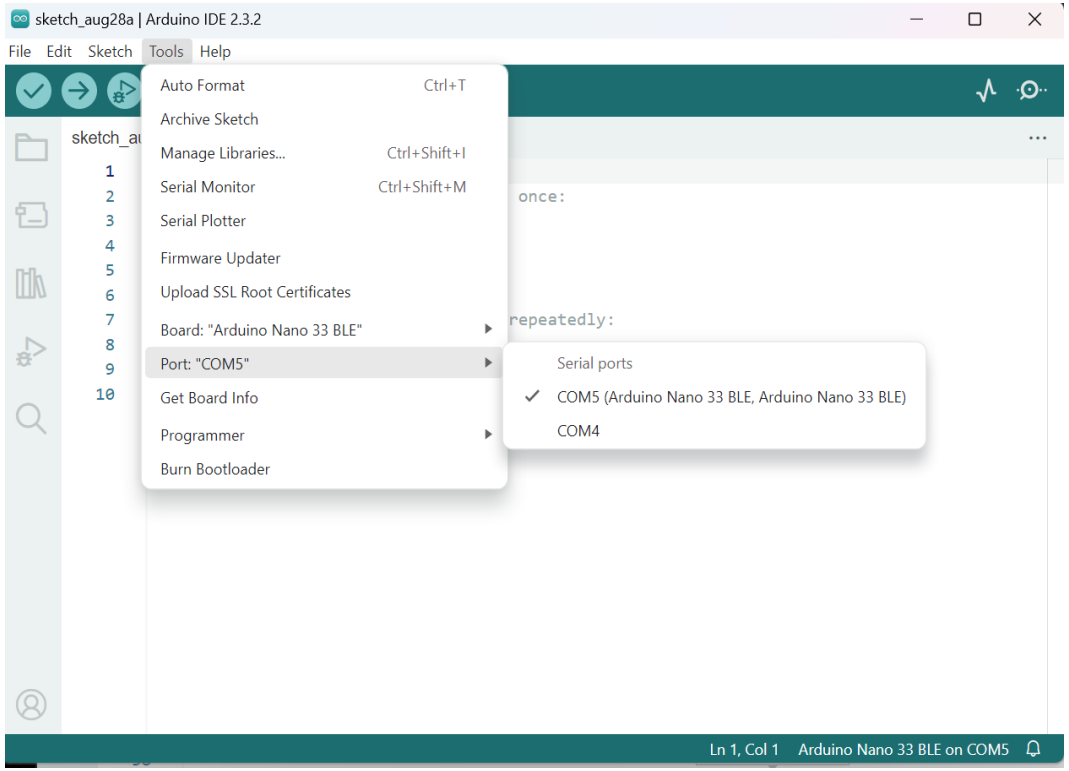
2


Arduino IDE Setup
Choose Installation Options
Who should this application be installed for?
Please select whether you wish to make this software available to all users or just yourself
Anyone who uses this computer (all users)
Only for me (atiab)
There is already a per-user installation. (C:\Users\atiab\AppData\Local\Programs\Arduino IDE)
Will reinstall/upgrade.
Arduino IDE 2.3.2
< Back
Next >
Cancel

3

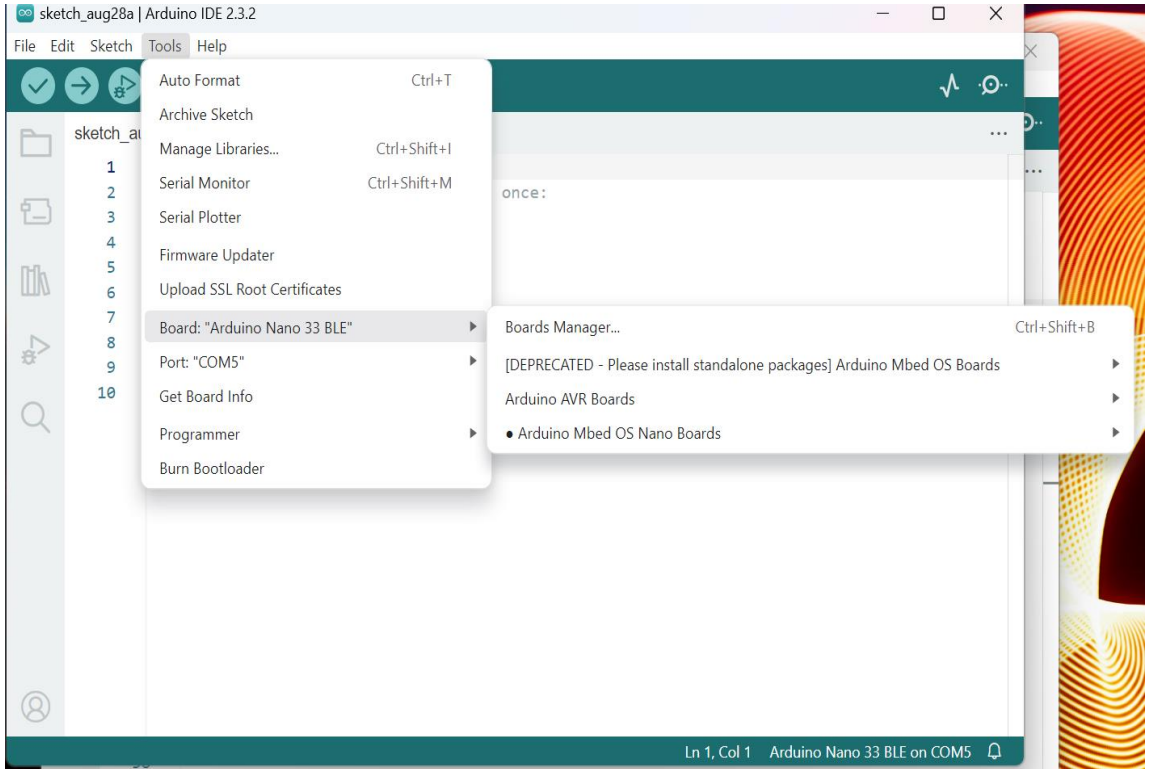

Arduino IDE Setup
Choose Install Location
Choose the folder in which to install Arduino IDE.
Setup will install Arduino IDE in the following folder. To install in a different folder, click Browse and select another folder. Click Install to start the installation.
Destination Folder
C:\Users\atiab\AppData\Local\Programs\Arduino IDE
Browse...
Space required: 497.1 MB
Space available: 39.2 GB
Arduino IDE 2.3.2
< Back
Install
Cancel

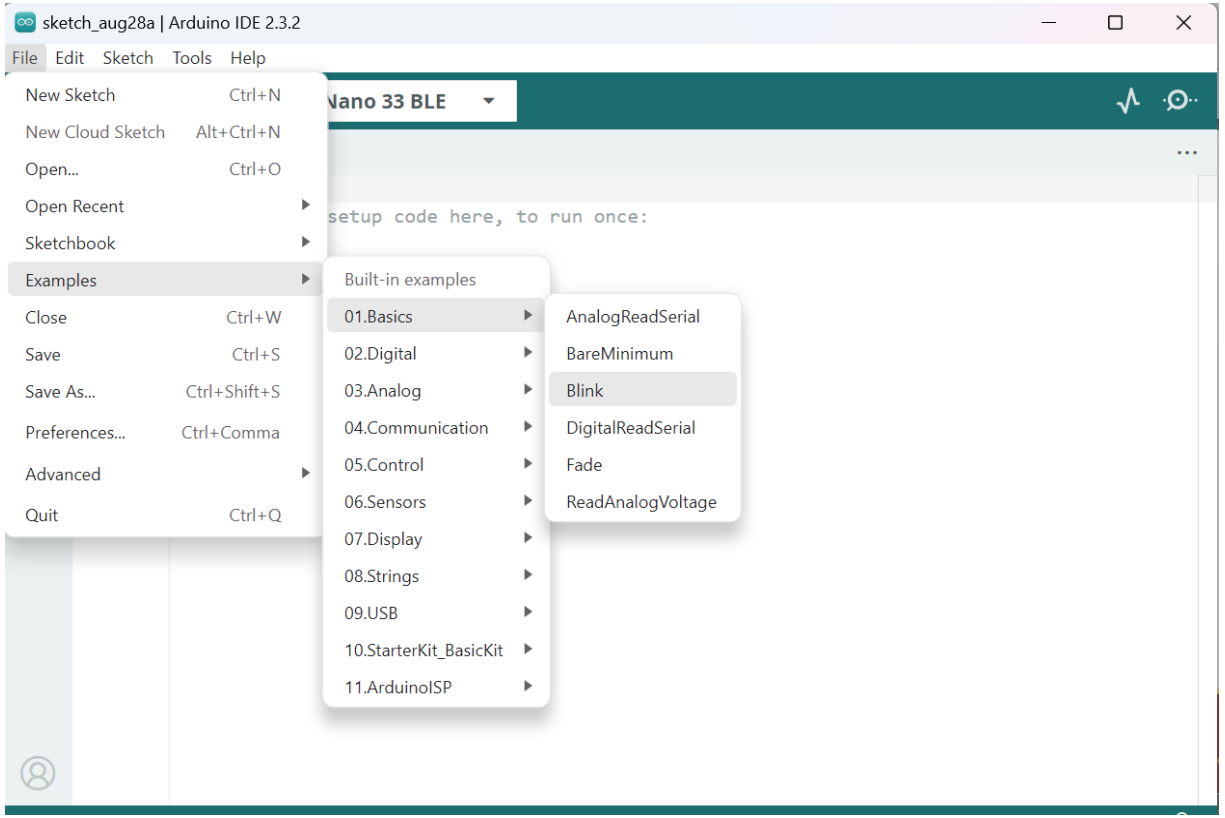
1. Identify the COM port that Arduino is connected



2.Ensure that the Mbed OS Nano Boards package is installed

Tools -> Board: -> Boards Manager... -> Install "Arduino Mbed OS Nano Boards"






```ino
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);  // turn the LED on (HIGH is the voltage level)
  delay(1000);                      // wait for a second
  digitalWrite(LED_BUILTIN, LOW);   // turn the LED off by making the voltage LOW
  delay(1000);                      // wait for a second
}
```

Another way of coding

```
// Pin number for the built-in LED
const int ledPin = 13;

void setup() {
  // initialize digital pin 13 as an output.
  pinMode(ledPin, OUTPUT);
}

void loop() {
  digitalWrite(ledPin, HIGH);  // turn the LED
on (HIGH is the voltage level)
  delay(1000);                 // wait for a
second
  digitalWrite(ledPin, LOW);   // turn the LED
off by making the voltage LOW
  delay(1000);                 // wait for a
second
}
```

JupyterLab Installation:

1. Open a terminal and check if Python is installed on your system or not, if python is installed this command will print the version.

python3 --version

2. If python is not installed, use this link below and install python on your system.

https://www.python.org.

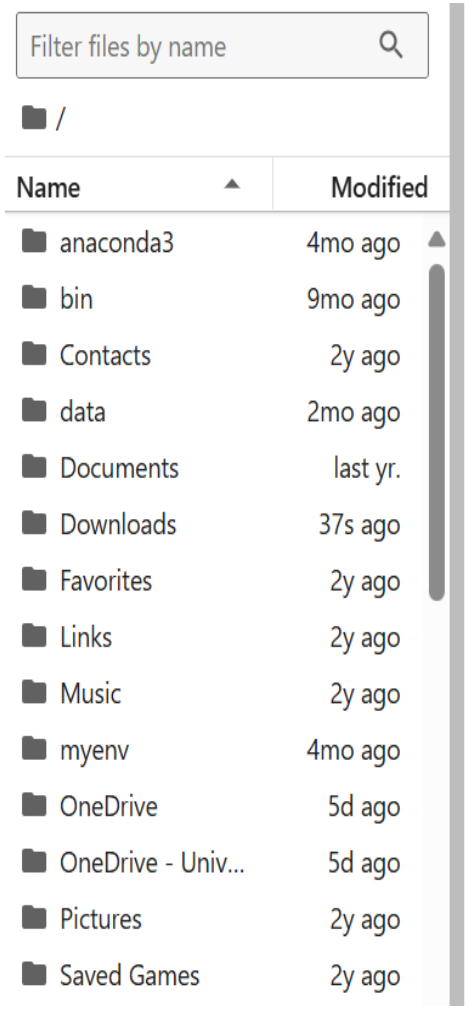
3. In the terminal, type the following command to install JupyterLab:

pip install JupyterLab

4. After finishing the installation, type this command to open the JupyterLab:

jupyter lab

JupyterLab window



You might get this error:

1. pip Command Not Found, solution is to use this command instead:

python3 -m pip install jupyterlab

2. If this still fails, reinstall pip:

python3 -m ensurepip --upgrade
python3 -m pip install --upgrade pip

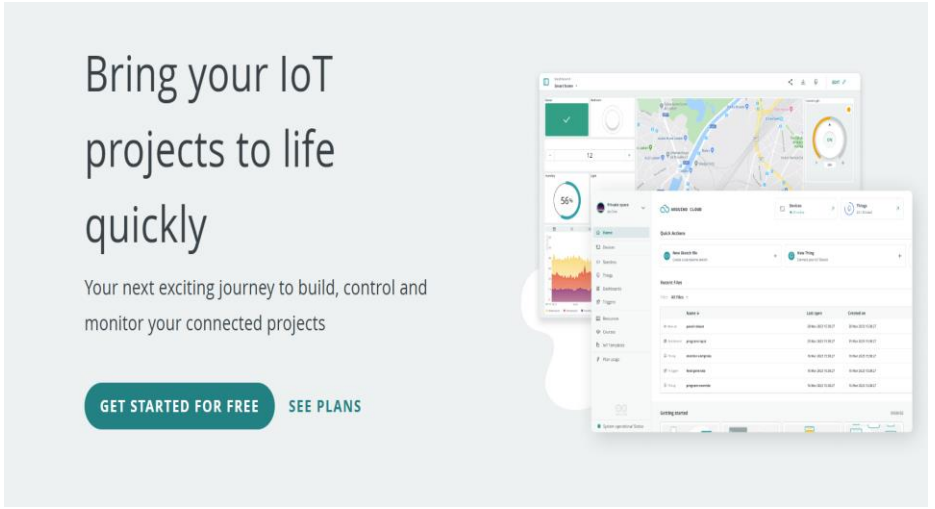
Streaming Sensor Data to a Web App

1. Let's start using Arduino Cloud
   https://cloud.arduino.cc/?gad_source=1&gbraid=0AAAACbEa84ofzTu6ZqKTNrvfC00iYeOK&gclid=CjwKCAjwlbu2BhA3EiwA3yXyuyRs52Pa6IMOnlhxourjC6ZlovP8tXgo2qfPTzjIE3rKl_pymK4kmxoCDSMQAvD_BwE

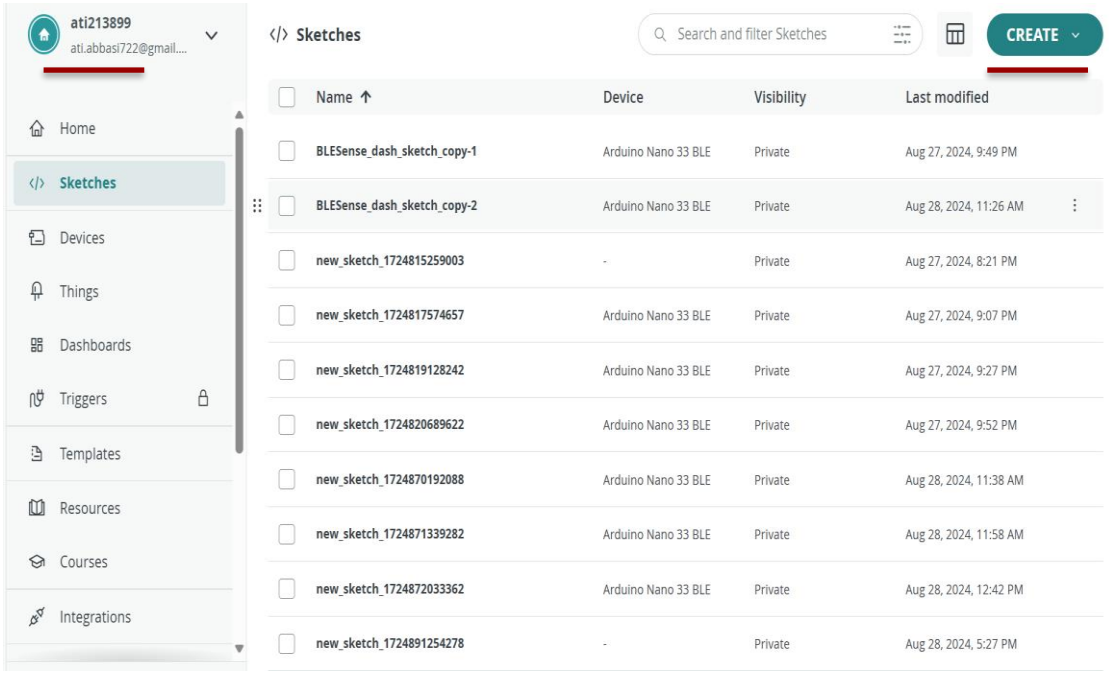
2. You need to install Arduino Cloud Agent:

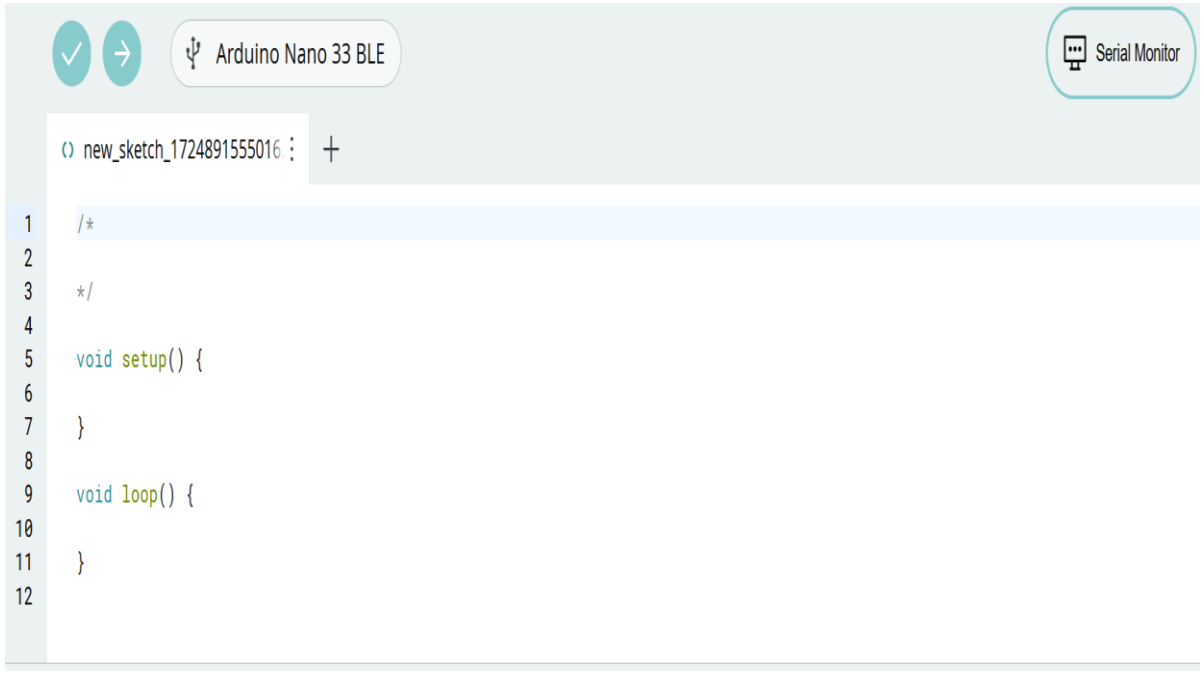
You need Arduino cloud agent to be able to detect or interact with your Arduino board, because web browsers don't have direct access to your computer's hardware.

https://support.arduino.cc/hc/en-us/articles/360014869820-Install-the-Arduino-Create-Agent



Sketches → Create





How can we monitor the streamed data?

- Webapp link: https://arduino.github.io/ArduinoAI/BLESense-test-dashboard/

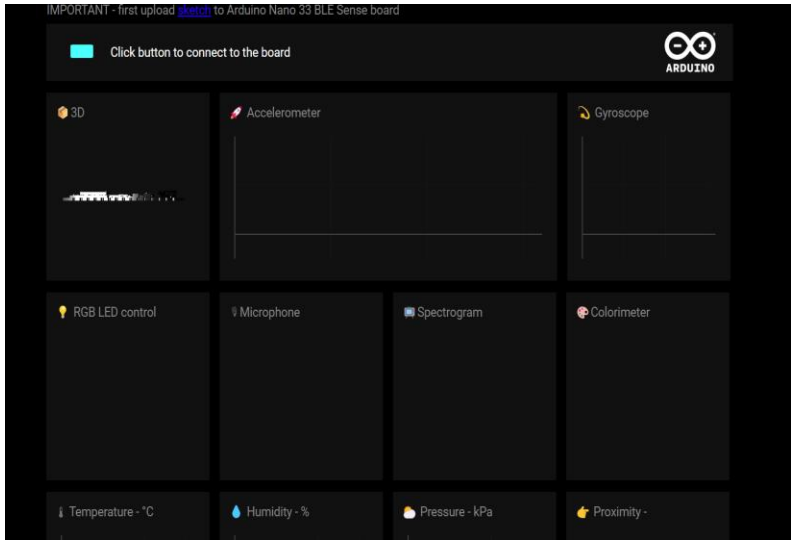
- Upload this code to Arduino nano 33 ble to be able to stream the sensor data to a webapp

https://app.arduino.cc/sketches/71cdba81-1380-4aee-9cb1-14cde2da7607?view-mode=preview

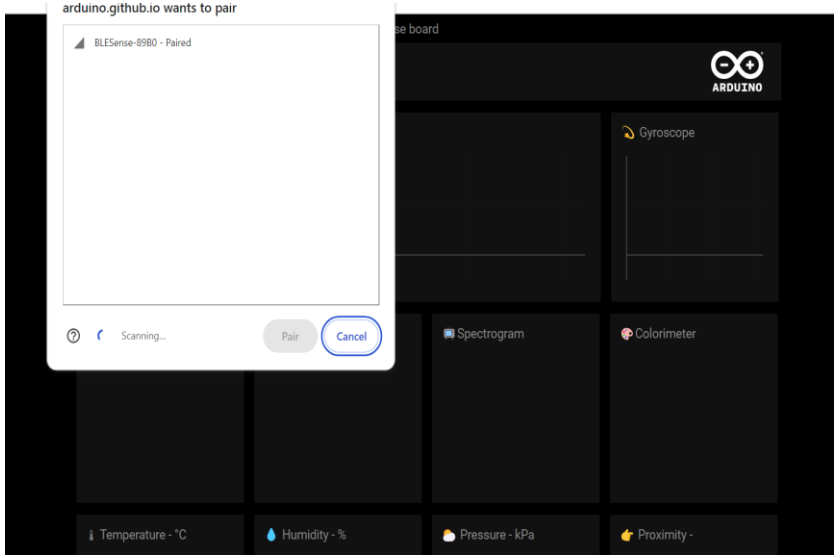
- Testing pressure sensor

https://app.arduino.cc/sketches/4573b54c-d240-41b0-94a4-4ac0dca97e66?view-mode=preview

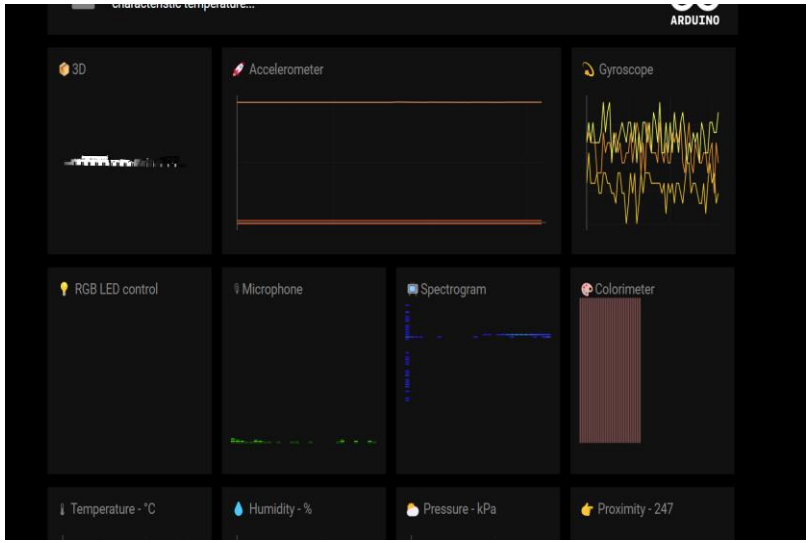
1.After uploading the code, click on the green button



2. Now your Arduino BLE will pair with the web app



3. The sensor data is streaming to the web app



The Arduino Nano 33 BLE is equipped with several built-in sensors, making it a powerful tool for various applications without needing additional hardware.

- **APDS9960**: Detects ambient light, color, proximity, and gestures.
- **LPS22HB**: Measures atmospheric pressure
- **LSM9DS1**: Tracks acceleration, rotation, and magnetic field orientation.
- **PDM Microphone**: Captures and analyzes sound signals.
- **RGB LED**: Provides visual feedback with controllable colors.

- **Color Detection**: The Colorimeter function in the web app is designed to detect and analyze the color of objects in the sensor's view. The APDS9960 sensor can measure the intensity of red, green, and blue light (RGB values) that is reflected from an object or present in the environment.

- **Proximity Sensor**: This sensor can detect objects within a certain distance, ranging from a few centimeters up to about 10 centimeters. It is commonly used in gesture control applications, where you can interact with devices without physical contact.

- **Gesture Detection**: The APDS9960 can recognize simple gestures like up, down, left, and right swipes. This feature enables touchless interaction with devices, such as controlling media playback or navigating menus with hand gestures.

•**Accelerometer**: This sensor measures acceleration forces along the X, Y, and Z axes. It is used to detect movement, orientation, and vibration. Applications include motion tracking, tilt sensing, and detecting free fall in devices like smartphones and wearables.

•**Gyroscope**: The gyroscope measures angular velocity (rotation) around the X, Y, and Z axes. It is used to detect and control the orientation of an object in 3D space. Common applications include stabilization in drones, gesture recognition, and gaming controllers.

•**Microphone**: The PDM microphone captures audio signals by converting sound waves into electrical signals. The microphone's data is processed using an FFT (Fast Fourier Transform) to analyze the frequency components of the sound.

•Spectrogram is a visual representation of the spectrum of frequencies in a signal as it varies with time. In this case, the code is generating a spectrogram from audio data captured by a microphone, likely using the PDM (Pulse Density Modulation) microphone on the Arduino Nano 33 BLE Sense