



EE 105 · Module 0

Lab: Arduino Setup

September 1, 2026

Lab Check-in



ee105.org/checkin

Scan with your phone and answer the quick questions — your responses steer today's lab.
The page is open only during lab hours.

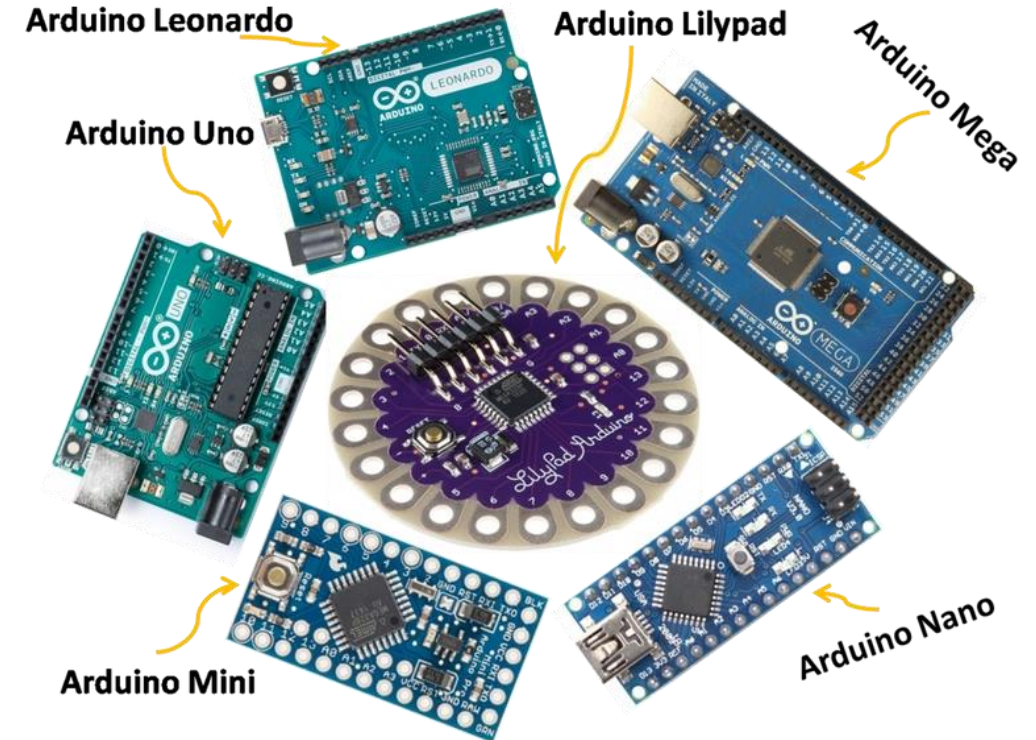


What is Arduino?

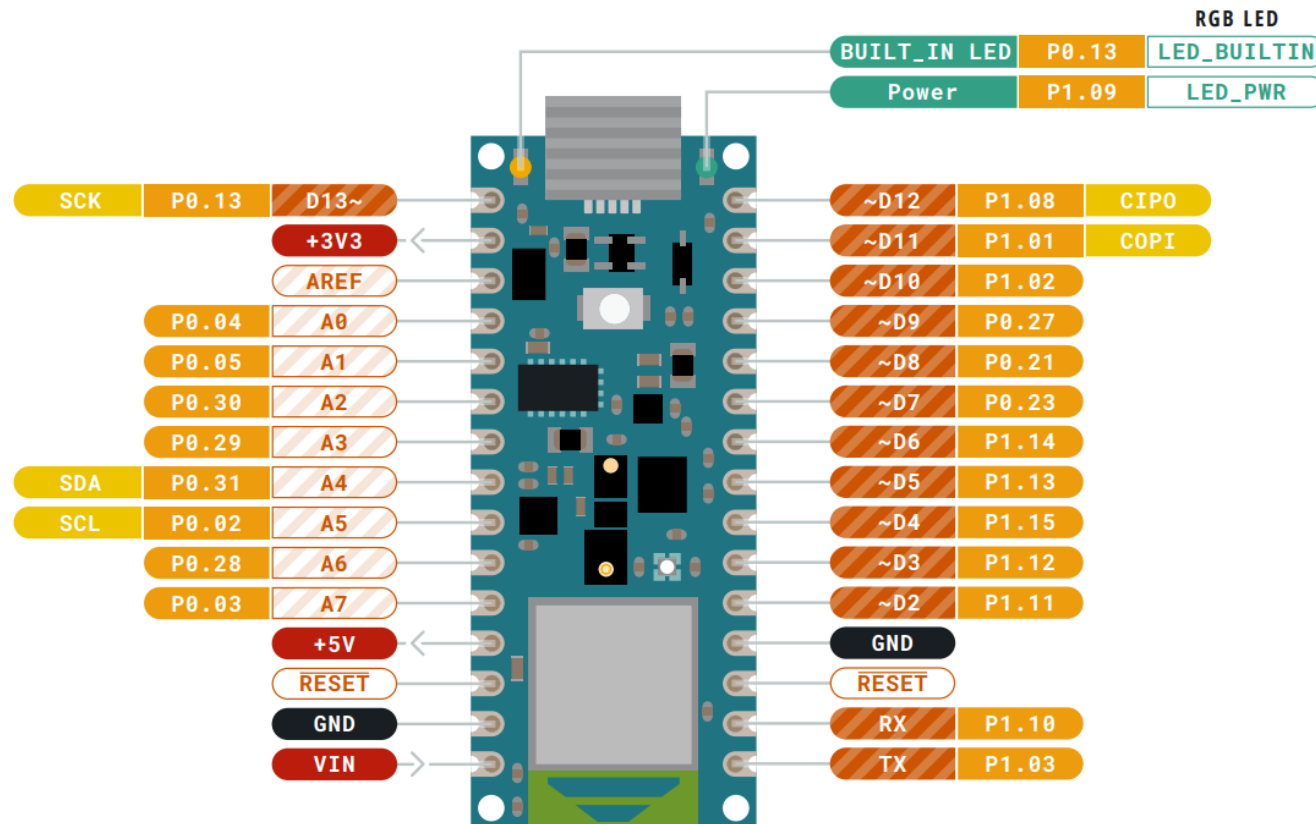
- Microcontroller Development Board
- Open-source hardware + software.
- Simple programming with Arduino IDE.
- Used for rapid prototyping.

Why use Arduino?

- Low cost and widely available.
- Large community and libraries.
- Easy to connect sensors/actuators.
- Good entry point into embedded systems.



Different types of Arduino Boards



Installing the Arduino IDE

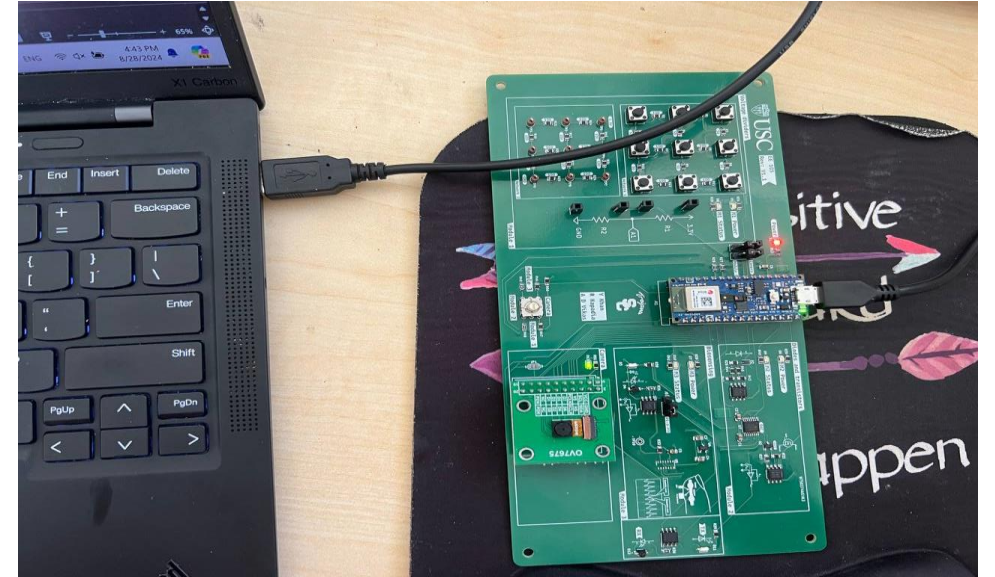


1. Connect the USB cable to Arduino
2. Download the latest release of the Arduino IDE here

support.arduino.cc → Download and install
[Arduino IDE](#)



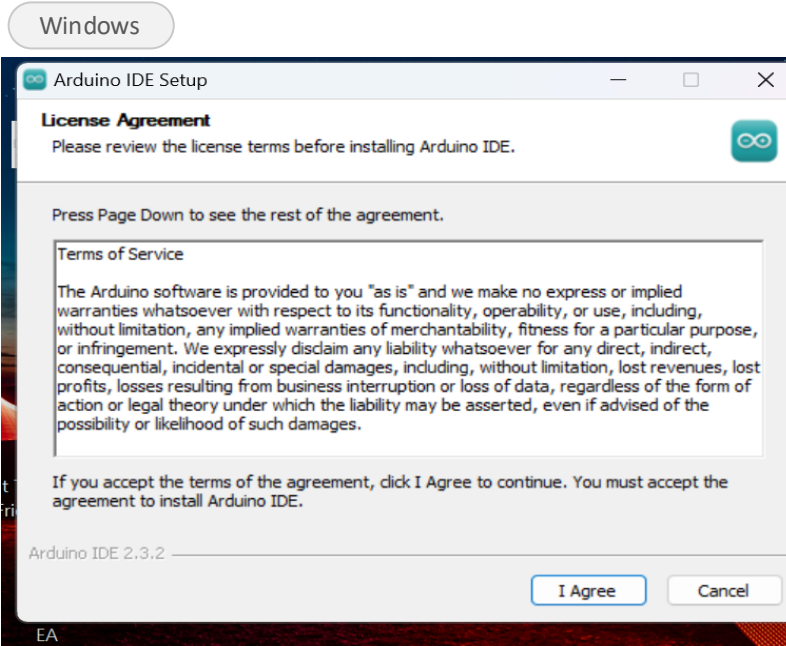
Scan to download



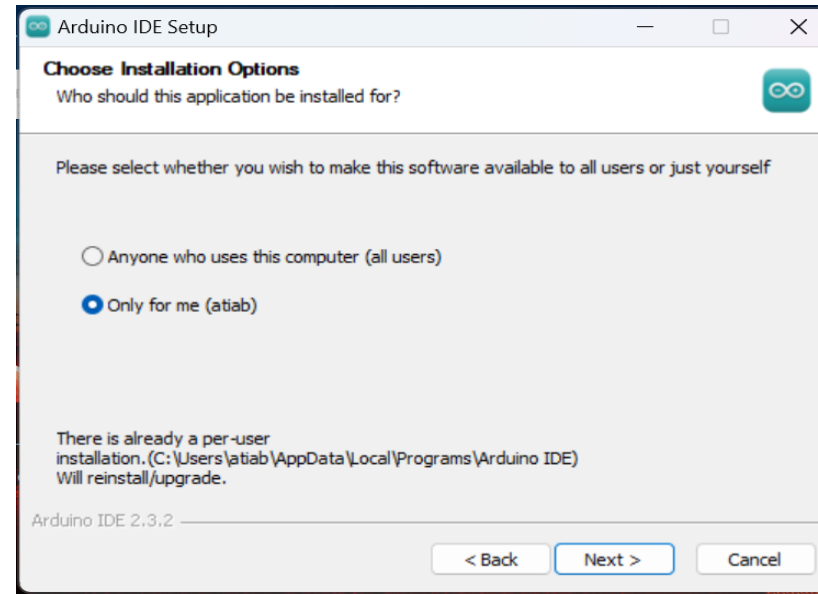
Run the Installer



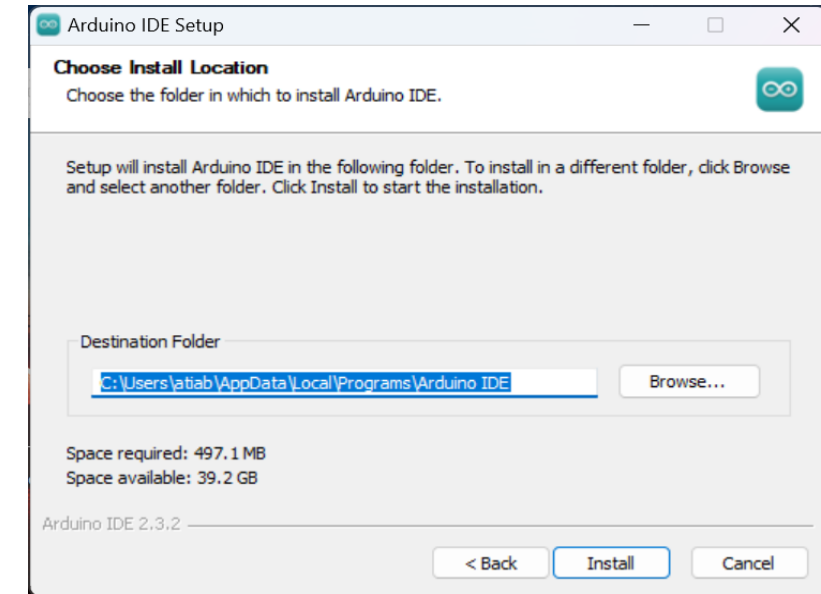
1



2



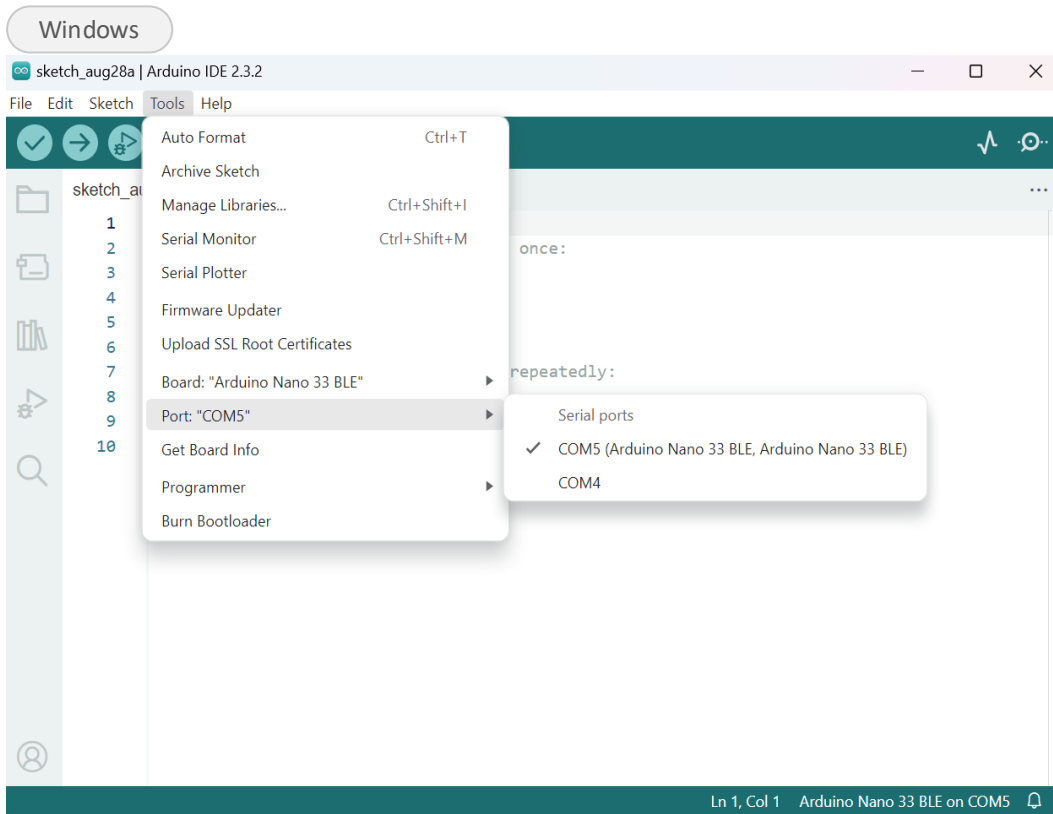
3



Select Board and Port

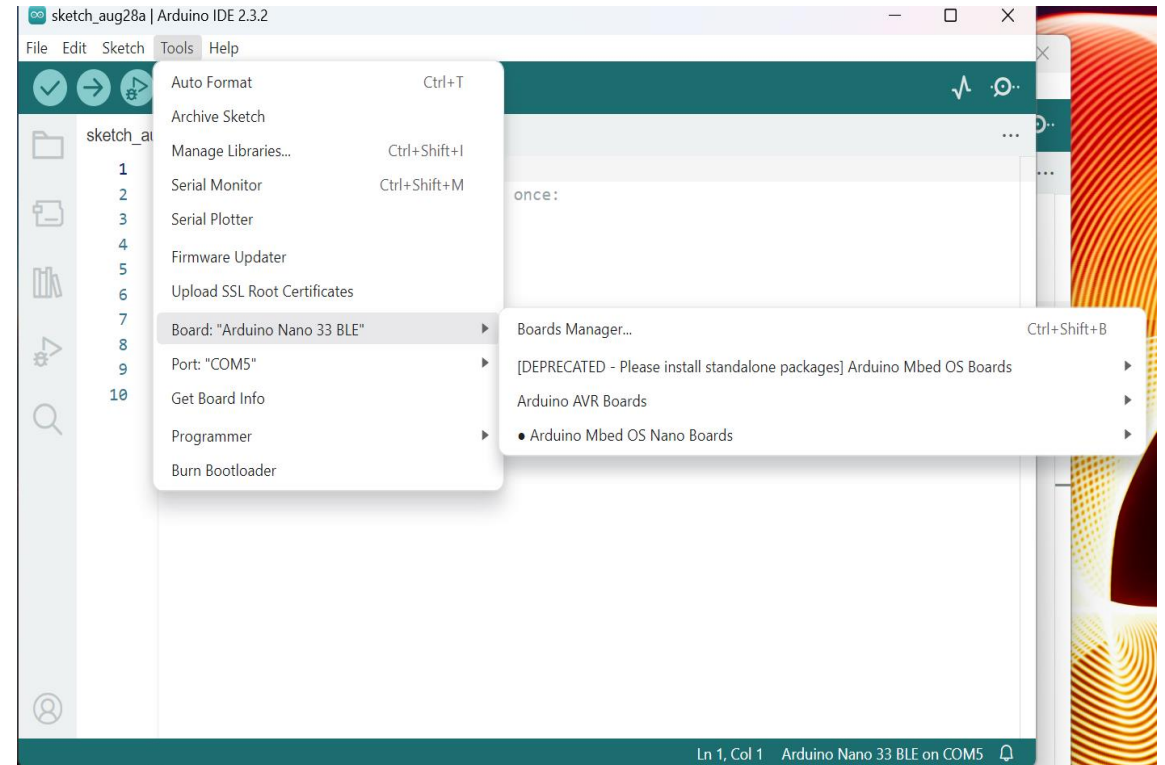


1. Identify the COM port that Arduino is connected

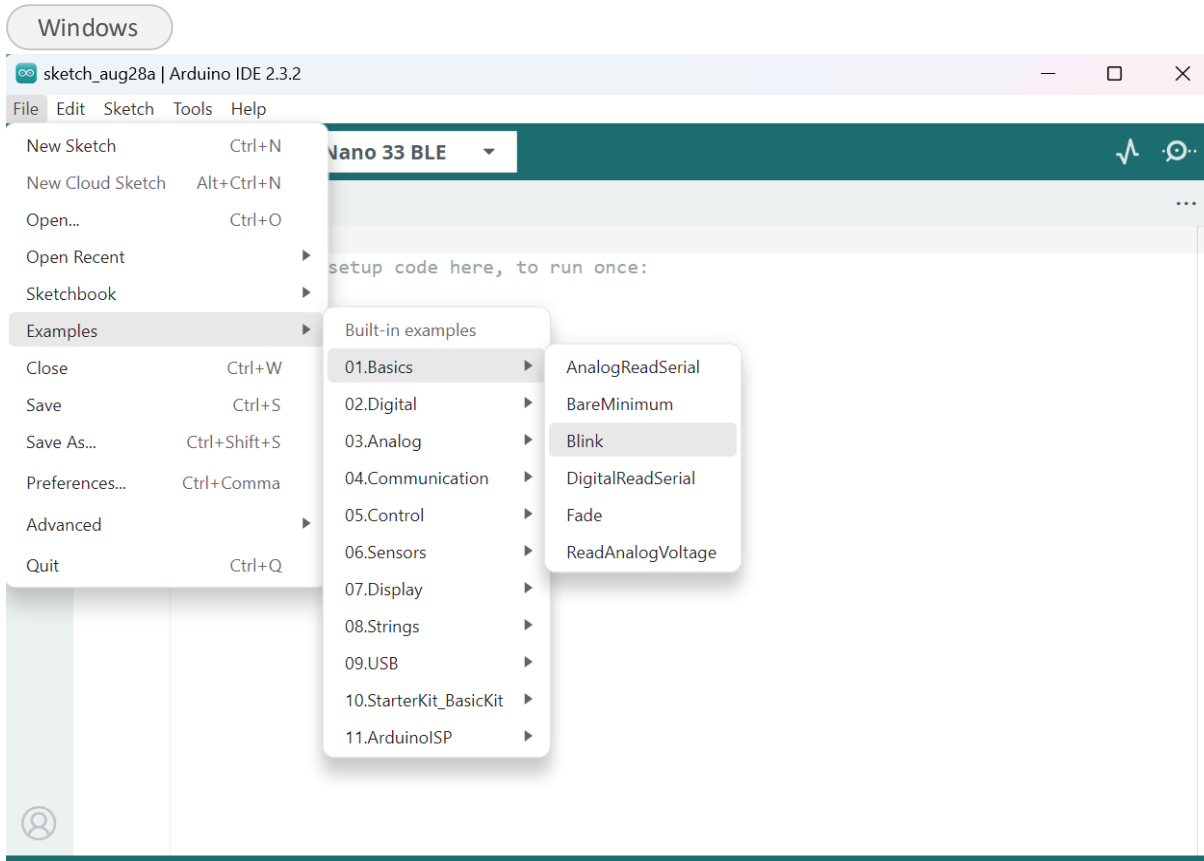


2. Ensure the Mbed OS Nano Boards package is installed

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



Your First Sketch: Blink



```
void setup() {  
    // initialize digital pin LED_BUILTIN as an  
    output.  
    pinMode(LED_BUILTIN, OUTPUT);  
}  
  
// loop() repeats forever  
void loop() {  
    digitalWrite(LED_BUILTIN, HIGH); // LED on  
    delay(1000); // wait for a second  
    digitalWrite(LED_BUILTIN, LOW); // LED off  
    delay(1000); // wait for a second  
}
```

Blink, Refactored



Another way of coding

Notice: the frequency of the LED is **0.5 Hz** — the two highlighted `delay(1000)` calls set the 2-second period.

```
// 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
    delay(1000);                // wait for a second
    digitalWrite(ledPin, LOW);  // turn the LED off
    delay(1000);                // wait for a second
}
```

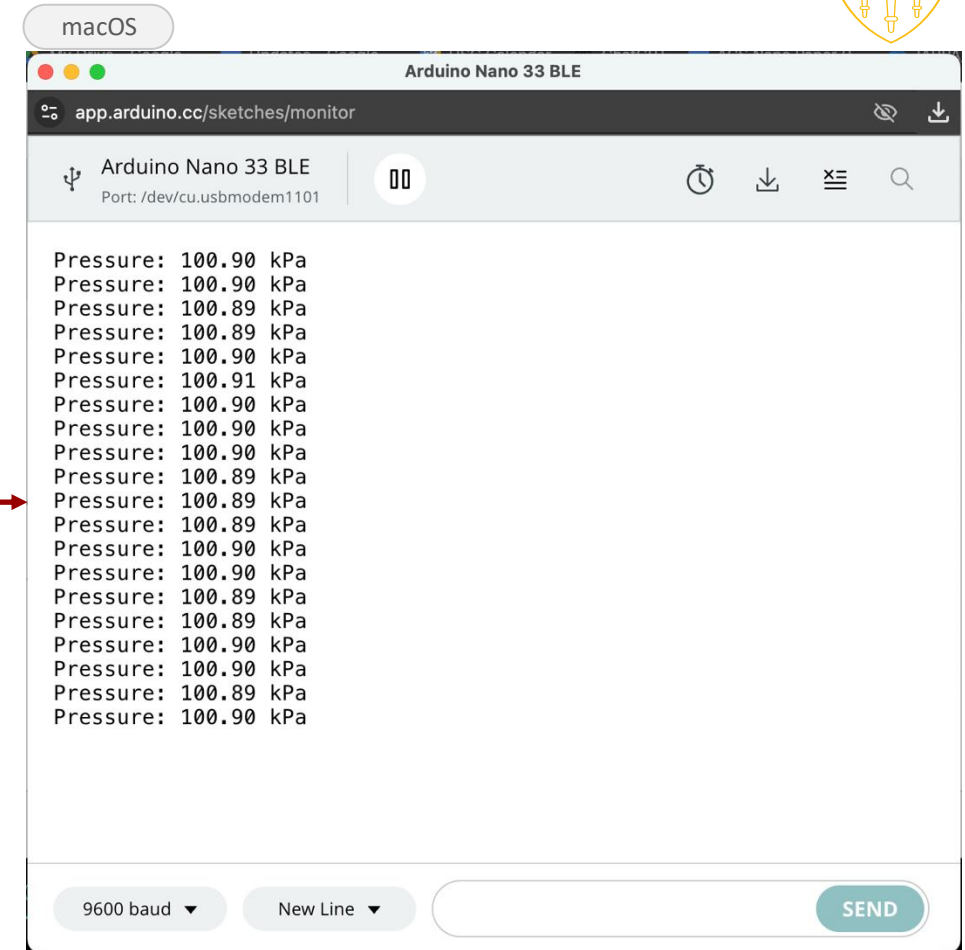
Using the Serial Monitor



- Testing pressure sensor: Open the sketch in 'pressure_mesurement'

[app.arduino.cc](https://app.arduino.cc/sketches/monitor) → [pressure-sensor test sketch](#)

Your serial monitor output should look like this →



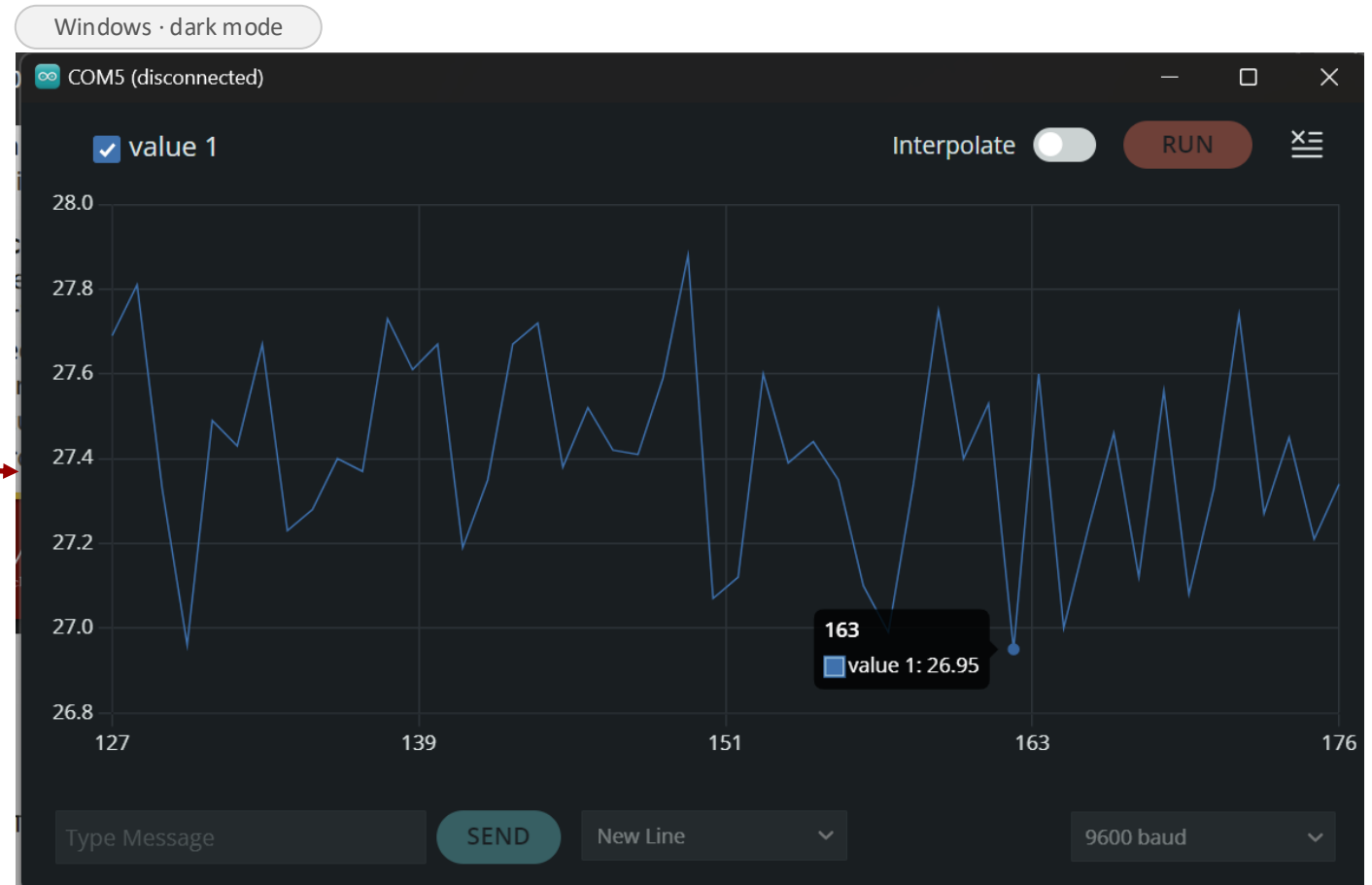
Scan for the sketch

Using the Serial Plotter



- Just like Serial monitor
- Plots values with time in x axis
- Use `Serial.println()` instead of `Serial.print()`

Your serial plotter output should look like this





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.

What the Sensors Can Do — Light & Gesture



Color Detection

The APDS9960 measures the red, green, and blue light reflected from an object — the web app's Colorimeter uses it to identify colors.

Proximity Sensing

Detects objects from a few centimeters up to about 10 cm away — the basis of touchless interaction with devices.

Gesture Detection

Recognizes up, down, left, and right swipes — control media or navigate menus without touching anything.

What the Sensors Can Do — Motion & Sound



Accelerometer

Measures acceleration along X, Y, and Z — motion, tilt, and free-fall detection in phones and wearables.

Gyroscope

Measures rotation rate around X, Y, and Z — stabilization in drones, gesture recognition, gaming controllers.

Microphone + FFT

The PDM mic converts sound into electrical signals; an FFT reveals the frequencies — plotted over time, that's a spectrogram.

How can we monitor streamed data?



- Open sketch in '**ble_dashboard**':
[Google Drive → EE105 dashboard sketch \(.ino\)](#)



sketch

- Install the 5 required libraries

or

- Install the libraries from the '**libraries**' folder:



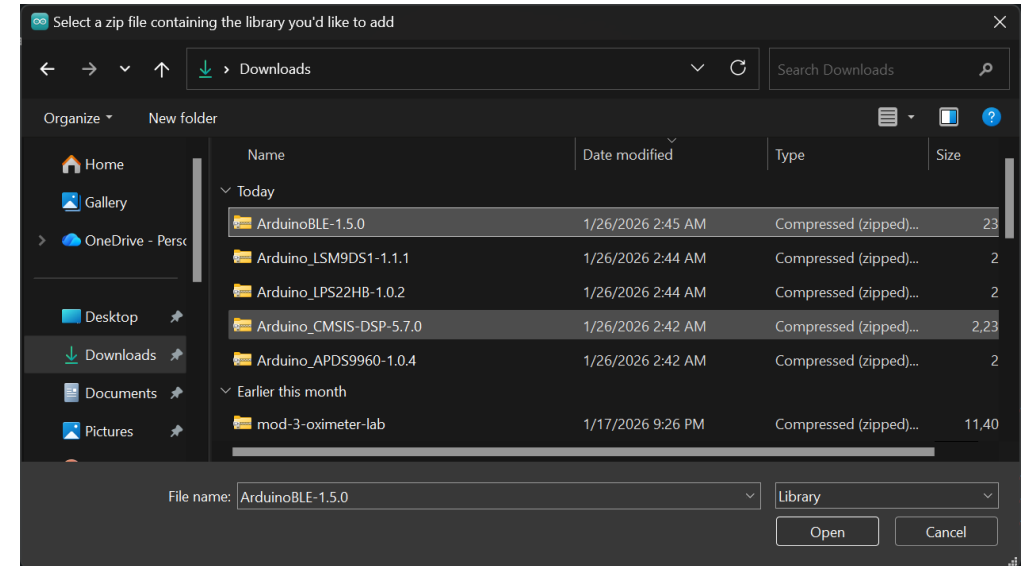
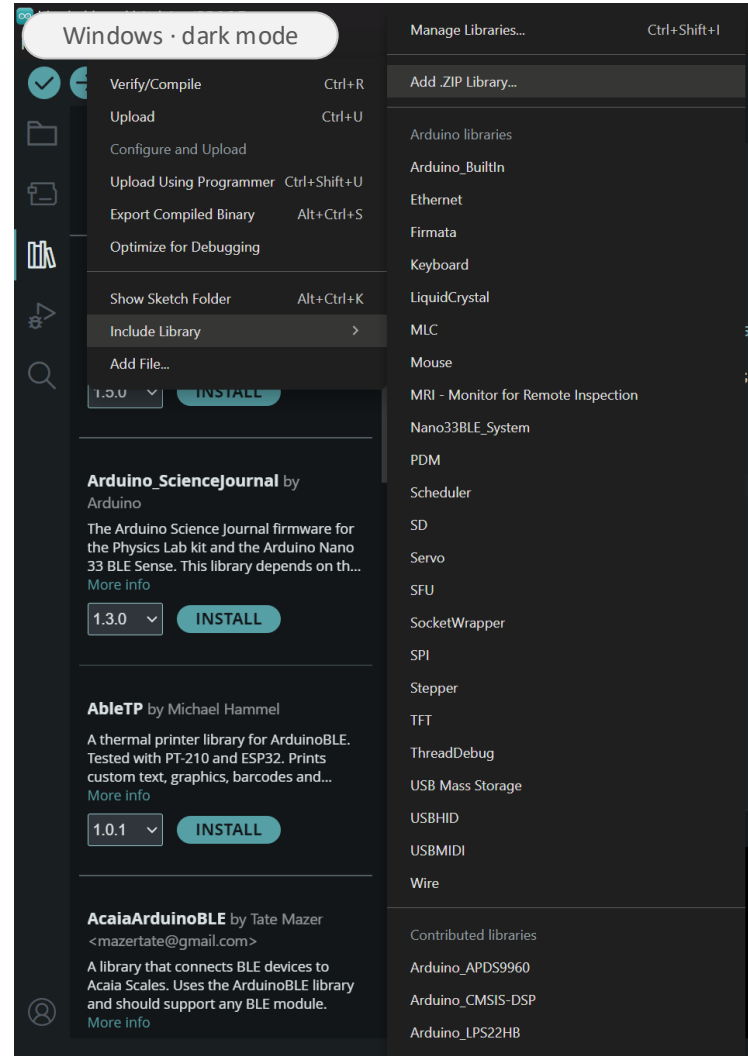
libraries

How can we monitor streamed data?



Installing Libraries:

- Sketch
- Include Library
- Add ZIP Library
- Repeat for each downloaded Library



How can we monitor streamed data?



- Verify and upload
- Open the following link:
[arduino.github.io → BLESense test dashboard](https://arduino.github.io/ble-sense/test-dashboard/)



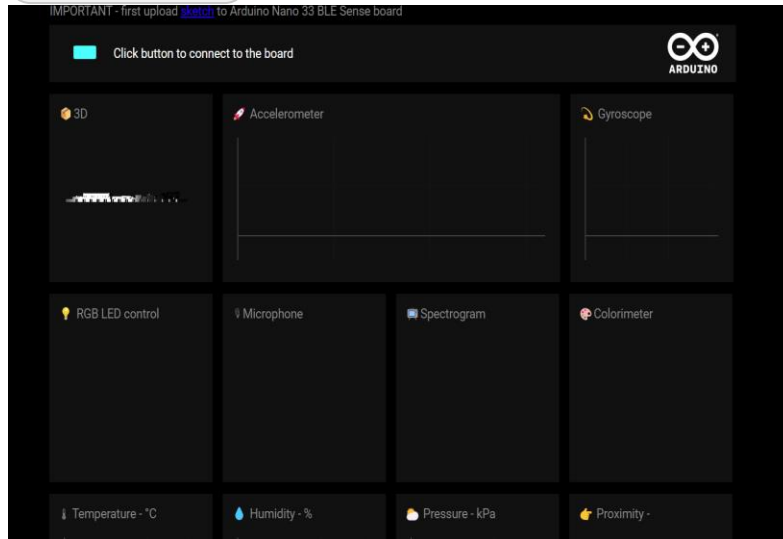
Open the dashboard



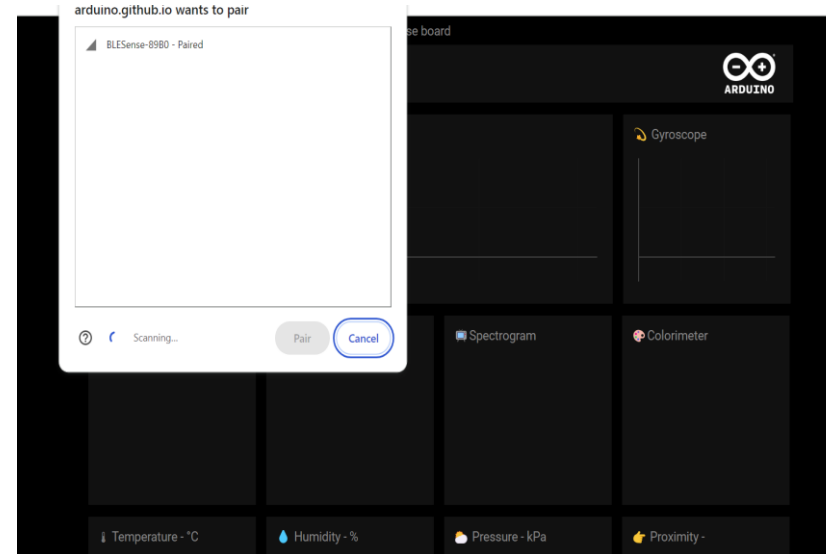
Pairing With the Dashboard

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

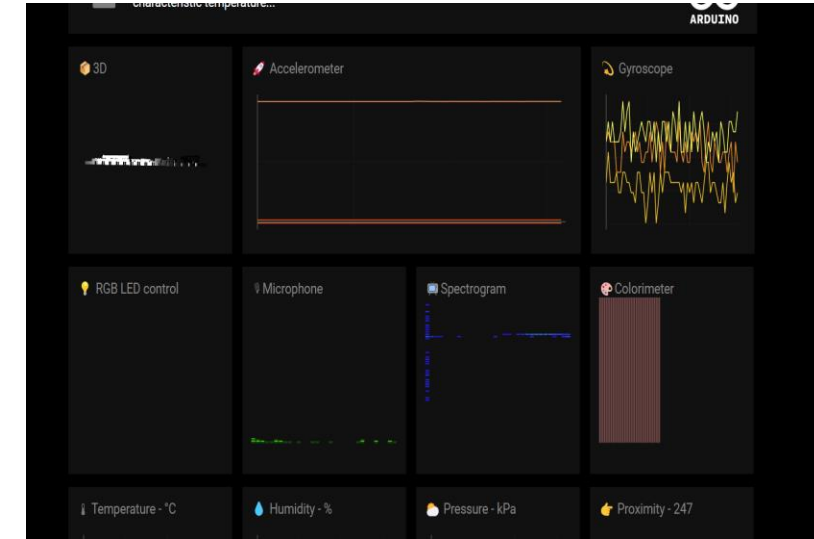
Google Chrome



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



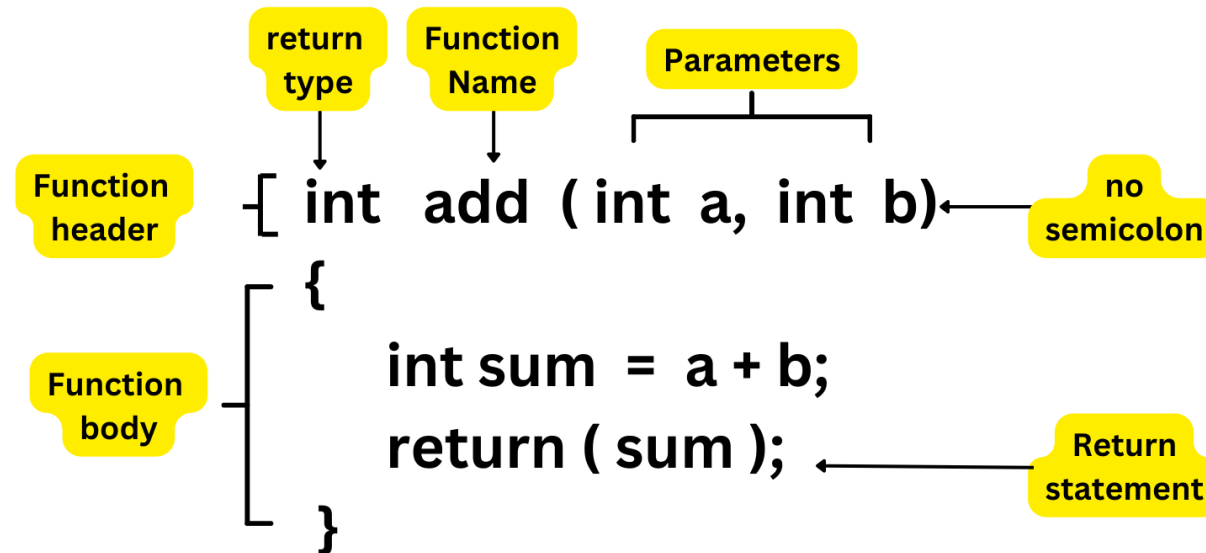
3. The sensor data is streaming to the web app



Classwork 1



- Solve all the tasks in the lab completion form.
- Hint 1: This is how to define a function in the C programming language.



- Hint 2: In C / Arduino, you cannot use `^` for exponentiation. Instead, you must use the `pow()` function from `<math.h>`: `pow(base, exponent)`