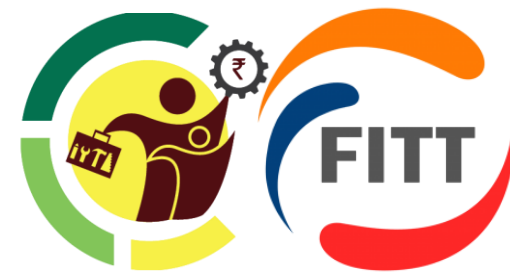


Lab Work and Q&A on IoT Devices and its connectivity

Setting Up IoT Devices with Arduino



Agenda

- Setting Up IoT Devices with Arduino
- IDE installation and basic code compilation and upload
- Online Simulator tools hands on practicals
- Weekly Recap and Q&A
- Preparation for Weekend Assignment

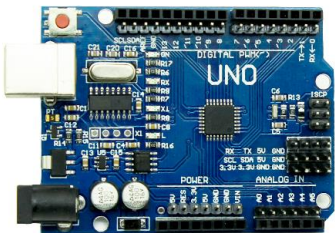
Basic Requirements for Experiment:



- IoT development Boards
- Electronic Components and Power supply
- IDE for Programming
- Basic Understanding of any programming language (Python preferred)
- Online tools for simulation purpose

Hardware Requirements for Practical:

- Boards - Most widely used boards



Arduino - UNO

<https://www.arduino.cc/>



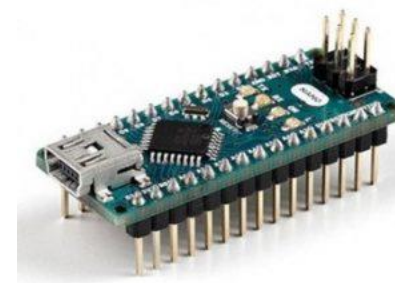
Raspberry-Pi

<https://www.raspberrypi.org/>



ESP-32

<https://www.espressif.com/en/products/socs/esp32>



Arduino-Nano

<https://www.arduino.cc/>
























Bharat Pi
IoT Prototyping Platform

Made in India








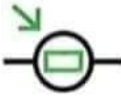














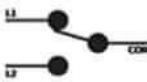








<https://bharatpi.net/>

Other Available Development Boards-

	Arduino	▼		Raspberry Pi	▼		Particle Photon	▼
	Arduino Nano 33 IoT	▼		ESP8266	▼		Microcontroller boards	▼
	Raspberry Pi 4 Model B	▼		Adafruit	▼		BeagleBone Black	▼
	Jetson Nano	▼		Single-board computer	▼		Tessel	▼
	Intel Edison	▼		Omega 2	▼		Banana Pi	▼
	ESP32	▼		Raspberry Pi Zero W	▼		Giant Board	▼
	Particle Boron	▼		System On Chipboards	▼		UDOO BOLT V8	▼

Lab Components



ACTIVE			PASSIVE		
Transistor			Resistor		
Diode			LDR		
LED			Thermistor		
Photodiode			Capacitor		
Integrated Circuit		-	Inductor		
Operational Amplifier			Switch		
Seven Segment Display			Variable Resistor		
Battery			Transformer		

source:
<https://www.hackatronic.com/electronics-components-type-and-explanation/#>

Lab Components Explained

list of essential components -

1. Breadboard and Jumper Wires:

- a. Breadboard for prototyping
- b. Jumper wires for connecting components on the breadboard

2. Prototyping Accessories:

- a. Resistors, capacitors, LEDs, and other basic electronic components for circuit building

3. Display:

- a. Optional: OLED or LCD display for visual output

4. Programming Cable:

- a. USB cables for programming and debugging Arduino

5. Software:

- a. Arduino IDE for programming Arduino boards
- b. required library for Arduino

6. Documentation and Learning Resources:

- a. Relevant datasheets, documentation, and learning materials for Arduino and other components

Lab Components Explained

list of essential components -

7. Arduino Board: Arduino Uno

8. Sensors:

8.1. Various sensors based on your application (e.g., temperature, humidity, motion, light)

8.2 Sensor modules compatible with Arduino (e.g., DHT series, PIR motion sensor)

9. Power Supply:

9.1. Batteries or power supply for Arduino

9.2. Consider low-power options for extended operation in battery-powered scenarios

• Electronic Components Kit

Transistor

USB Power Cable

DHT sensor

Indicator LEDs

ESP32 Board

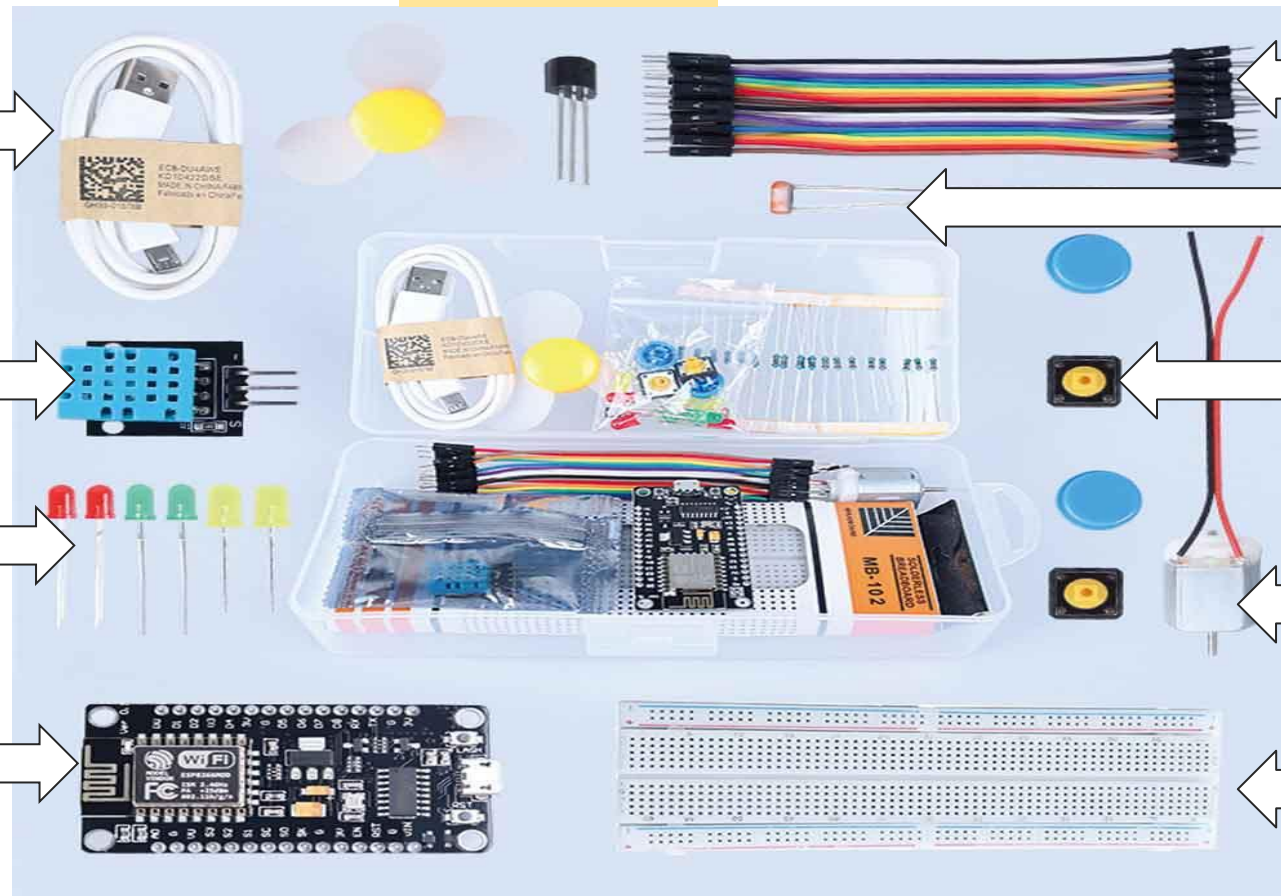
Jumper Wires

Temperature sensor

Switch

Motor

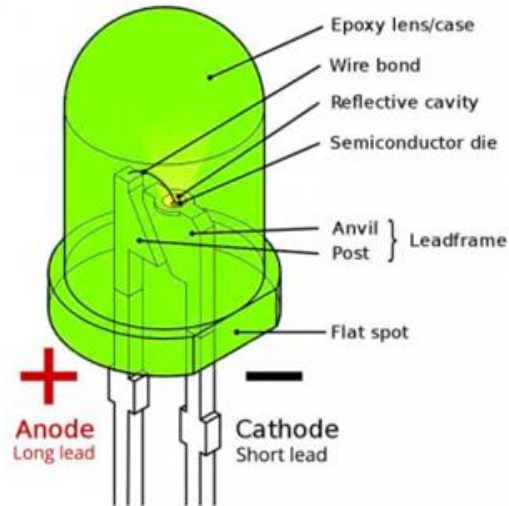
Bread Board



• LED and Resistors



LED POLARITY

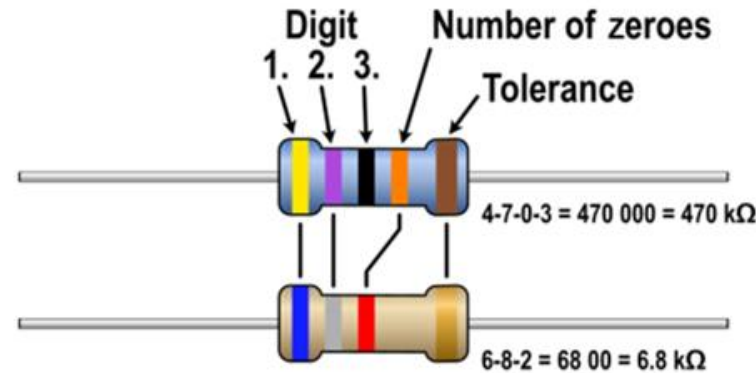


Note
If there is no flat side marked on the pcb, use a multimeter to find the ground. Usually ground is the square pad = cathode.

SCHEMATIC SYMBOL



SMD LED



Digit	0	1	2	3	4	5	6	7	8	9
Tolerance	Silver ±10 %	Gold ±5 %	±1 %	±0.5 %	±0.1 %					

Source:
https://en.m.wikipedia.org/wiki/File:Resistor_Color_Code.svg

	1 st Digit	2 nd Digit	Multiplier	Tolerance
Black	0	0	x 1	
Brown	1	1	x10	±1%
Red	2	2	x10 ²	±2%
Orange	3	3	x10 ³	±3%
Yellow	4	4	x10 ⁴	±4%
Green	5	5	x10 ⁵	±0.5%
Blue	6	6	x10 ⁶	±0.25%
Violet	7	7	x10 ⁷	±0.1%
Grey	8	8	x10 ⁸	±0.05%
White	9	9	x10 ⁹	
Gold			x10 ⁻¹	±5%
Silver			x10 ⁻²	±10%

Source:
<https://www.te.com/usa/en/products/passivecomponents/resistors/intersection/resistor-color-codes.html>

Source:
<https://www.tips.modularparts.net/led-polarity-orientation-pcb/>

• Arduino IDE

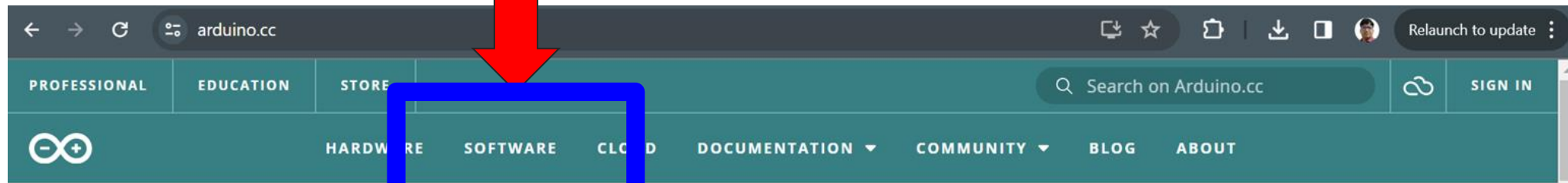
Arduino IDE (Integrated Development Environment) is a software platform used for programming Arduino microcontrollers. Here's a step-by-step guide on how to download it and an overview of its components:

How to Download Arduino IDE:

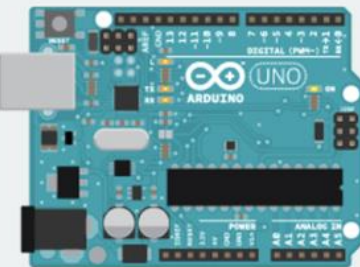
- Visit the official Arduino website (<https://www.arduino.cc/>).
- Navigate to the "Software" tab.
- Choose your operating system (Windows, macOS, or Linux) and follow the instructions to download the installer.
- Install the IDE by running the downloaded installer.


- # Arduino IDE for Programming


Get it from- <https://www.arduino.cc/>



WHAT IS ARDUINO?



BUY AN ARDUINO 

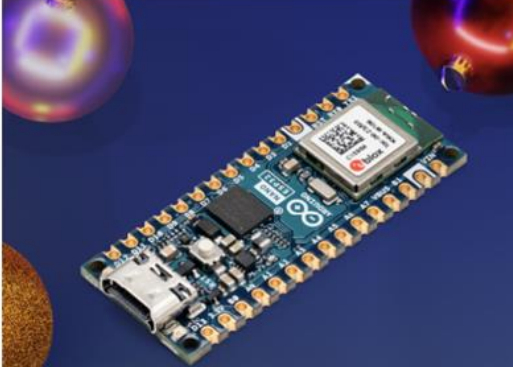
LEARN ARDUINO 



UNO R4

Surprise your loved ones with the **future of making**

[Discover more](#)

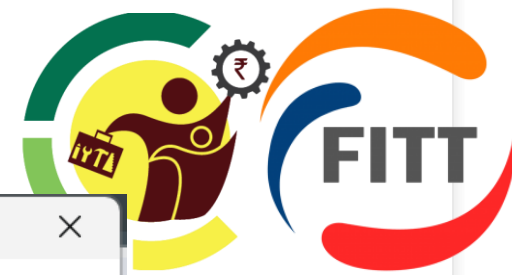


Nano ESP32

The compact board for **projects** of all sizes

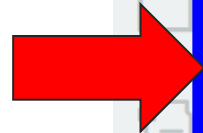
[Discover more](#)

• Components of Arduino IDE:



• Sketch Area:

This is the main working area where you write your Arduino code. It's called a sketch in Arduino terminology.

A screenshot of the Arduino IDE 2.2.1 interface. The window title is 'sketch_jan18a | Arduino IDE 2.2.1'. The menu bar includes 'File', 'Edit', 'Sketch', 'Tools', and 'Help'. Below the menu bar is a toolbar with icons for checking, running, and uploading code, along with a dropdown menu showing 'Arduino Uno'. The main area is the sketch editor, which contains the code for 'sketch_jan18a.ino'. The code is as follows:

```
1 void setup() {  
2   // put your setup code here, to run once:  
3  
4 }  
5  
6 void loop() {  
7   // put your main code here, to run repeatedly:  
8  
9 }  
10
```

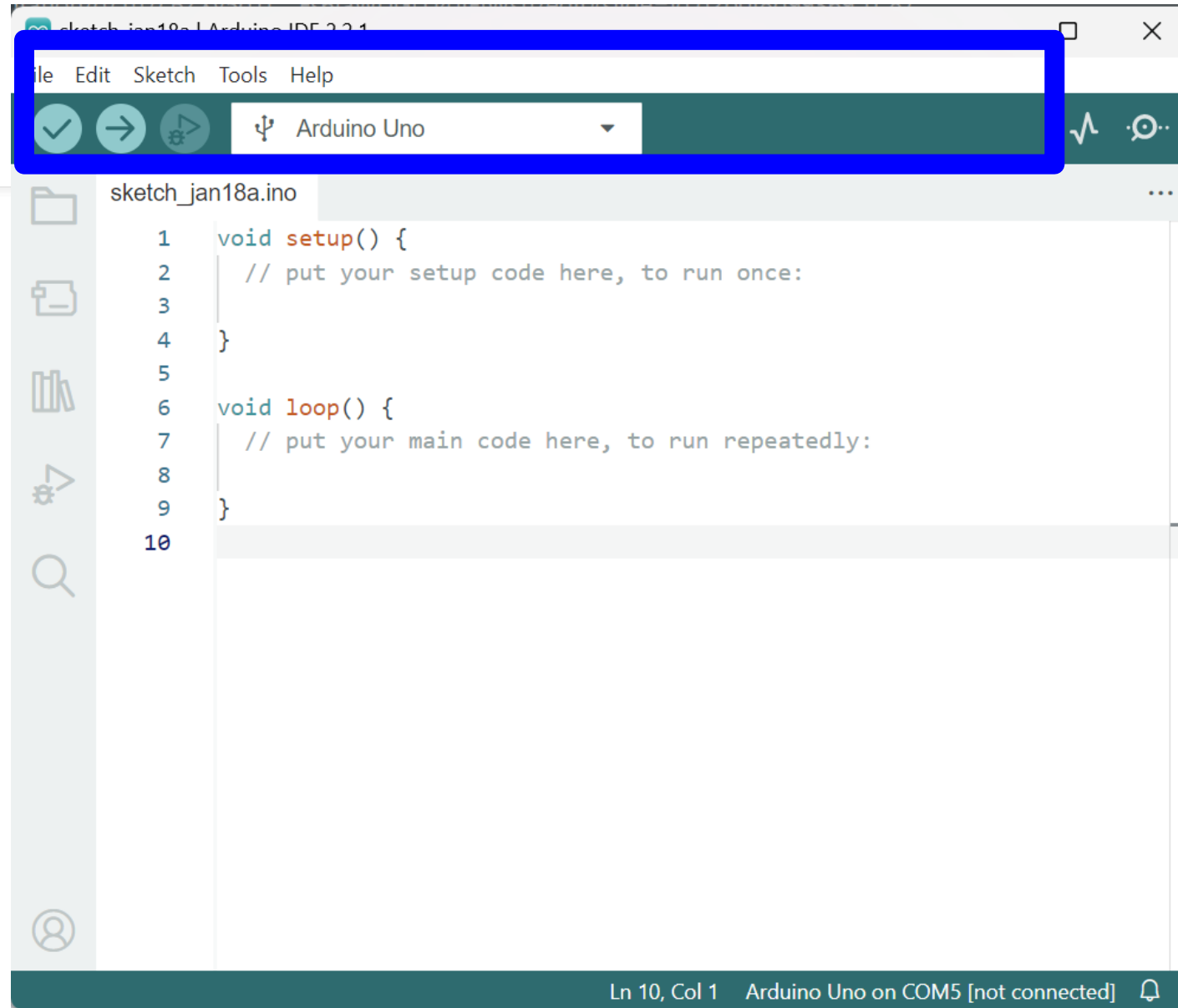
The sketch area is highlighted with a blue border. The status bar at the bottom shows 'Ln 10, Col 1' and 'Arduino Uno on COM5 [not connected]'.

```
sketch_jan18a.ino  
  
1 void setup() {  
2   // put your setup code here, to run once:  
3  
4 }  
5  
6 void loop() {  
7   // put your main code here, to run repeatedly:  
8  
9 }  
10
```

• Components of Arduino IDE:

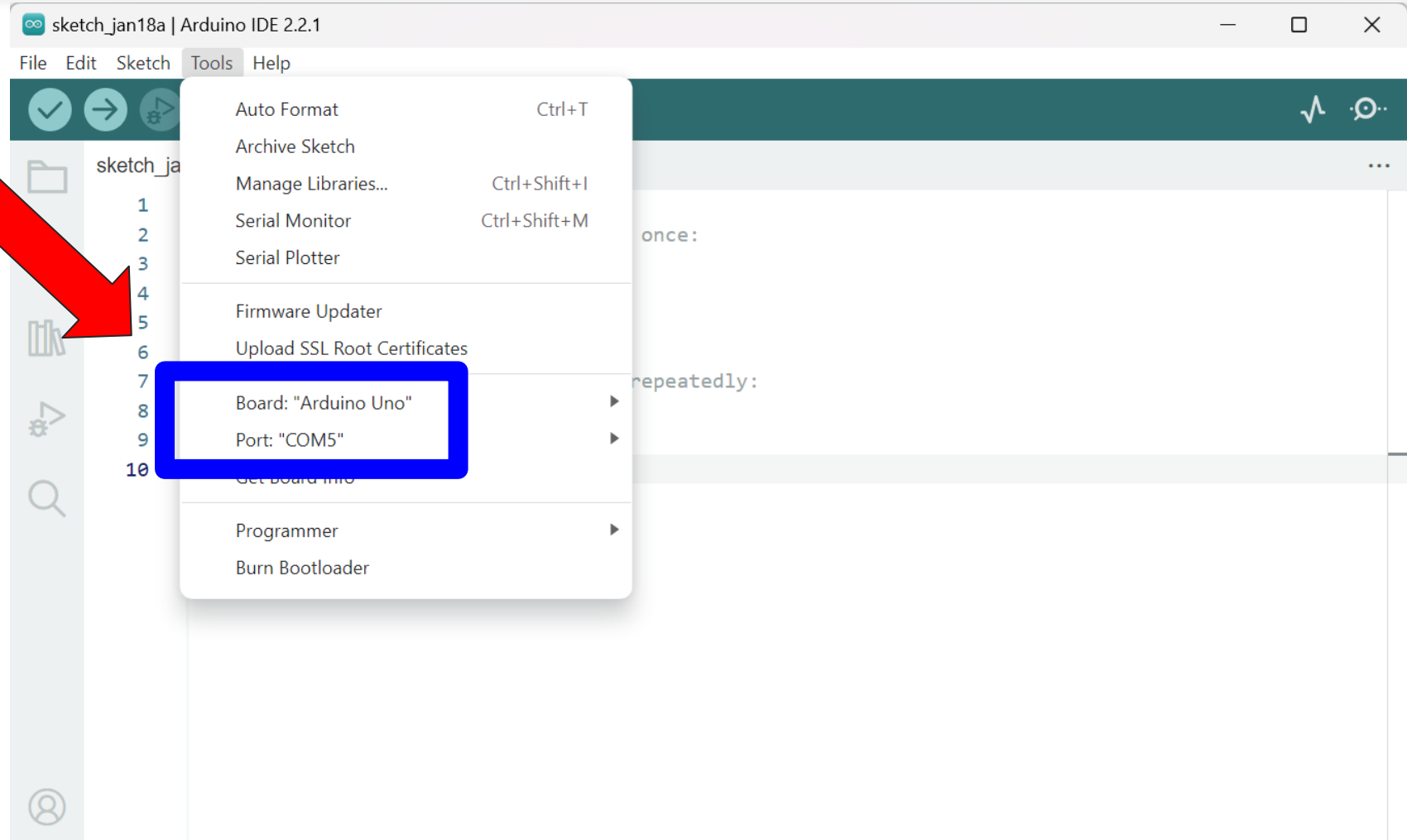


- **Toolbar:** Contains buttons for common actions like uploading code to the Arduino board, opening or saving sketches, and verifying code.



• Components of Arduino IDE:

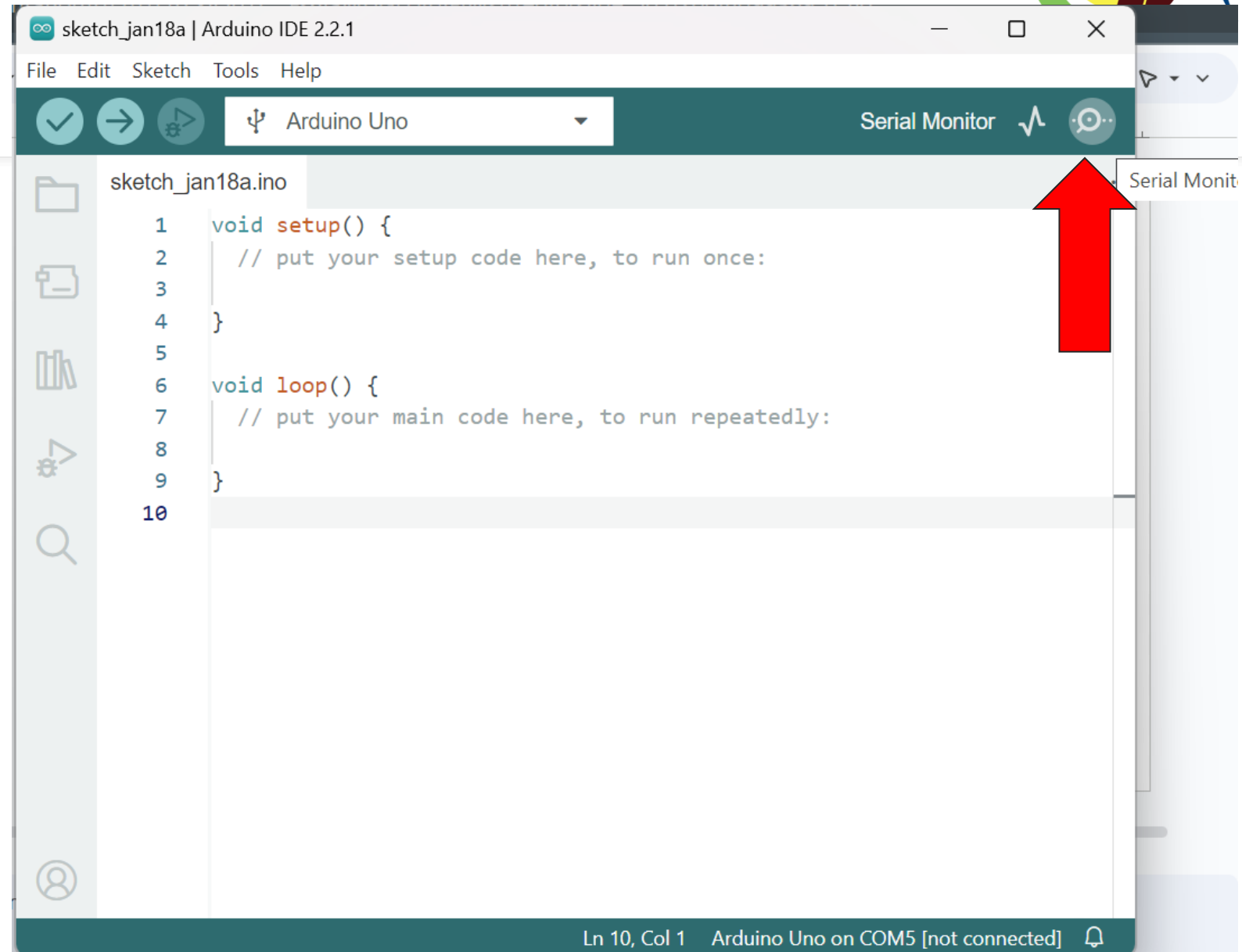
- **Board and Port Selection:** You need to select the specific Arduino board model and the communication port to which your Arduino is connected.



• Components of Arduino IDE:



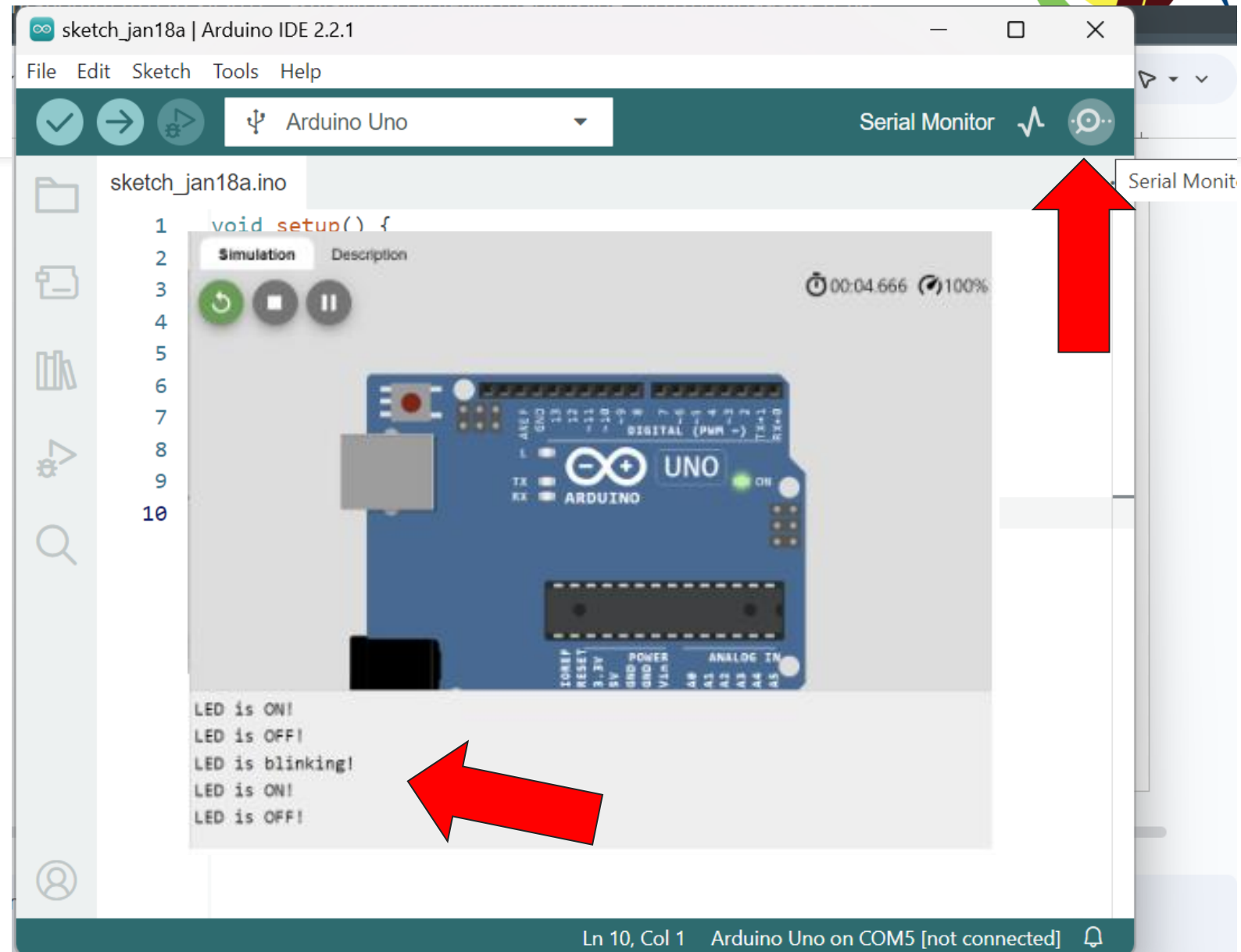
- **Serial Monitor:** This tool allows you to communicate with your Arduino board and monitor the data being sent or received via the serial port.



• Components of Arduino IDE:



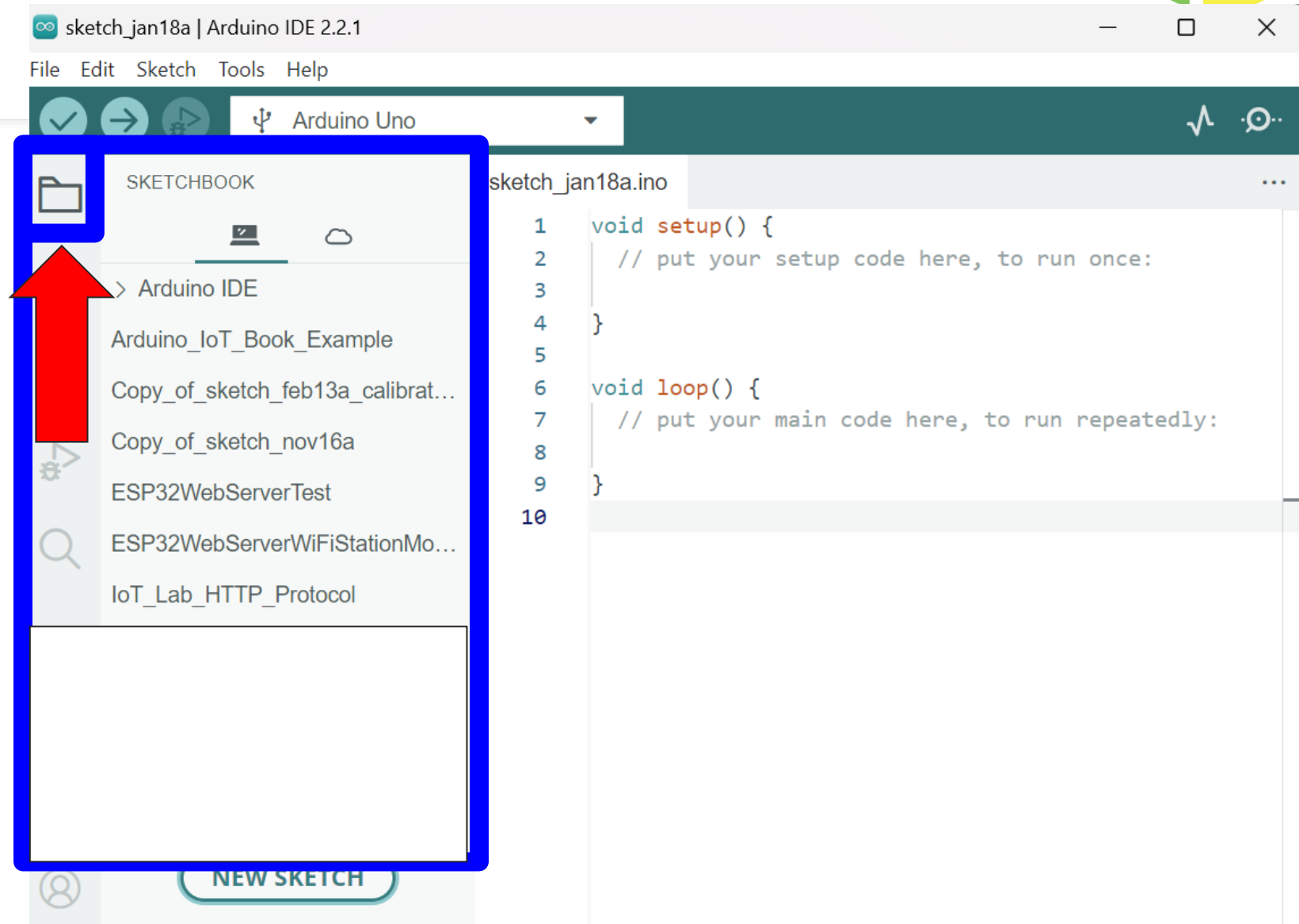
- **Serial Monitor:** This tool allows you to communicate with your Arduino board and monitor the data being sent or received via the serial port.
- Also, **Very helpful in Debugging the code.**



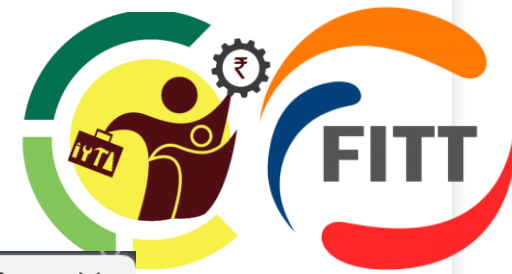
• Components of Arduino IDE:



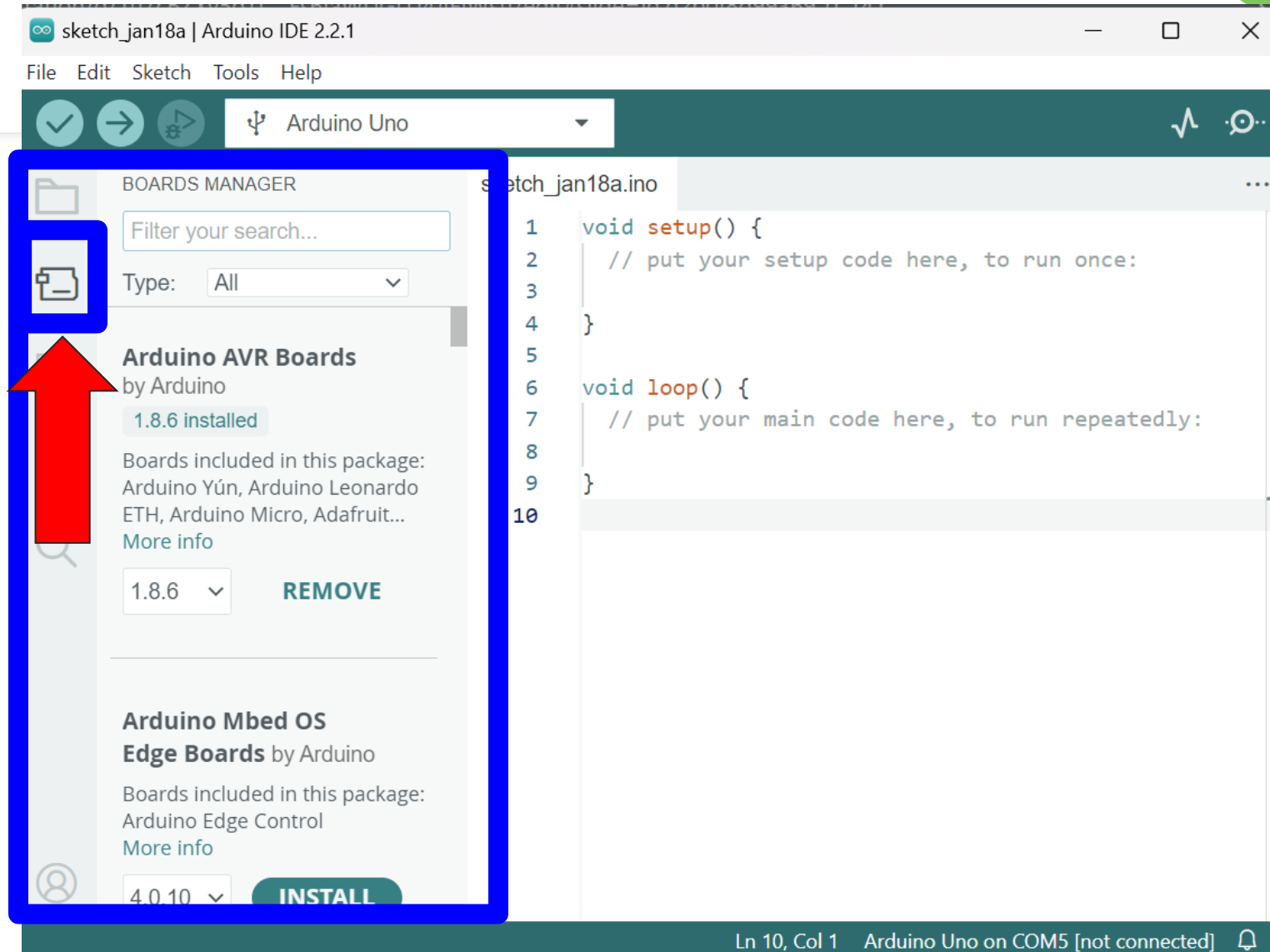
"The Arduino sketchbook is a designated folder where Arduino IDE stores and organizes your sketches (code files), facilitating easy access and management of your projects."



• Components of Arduino IDE:



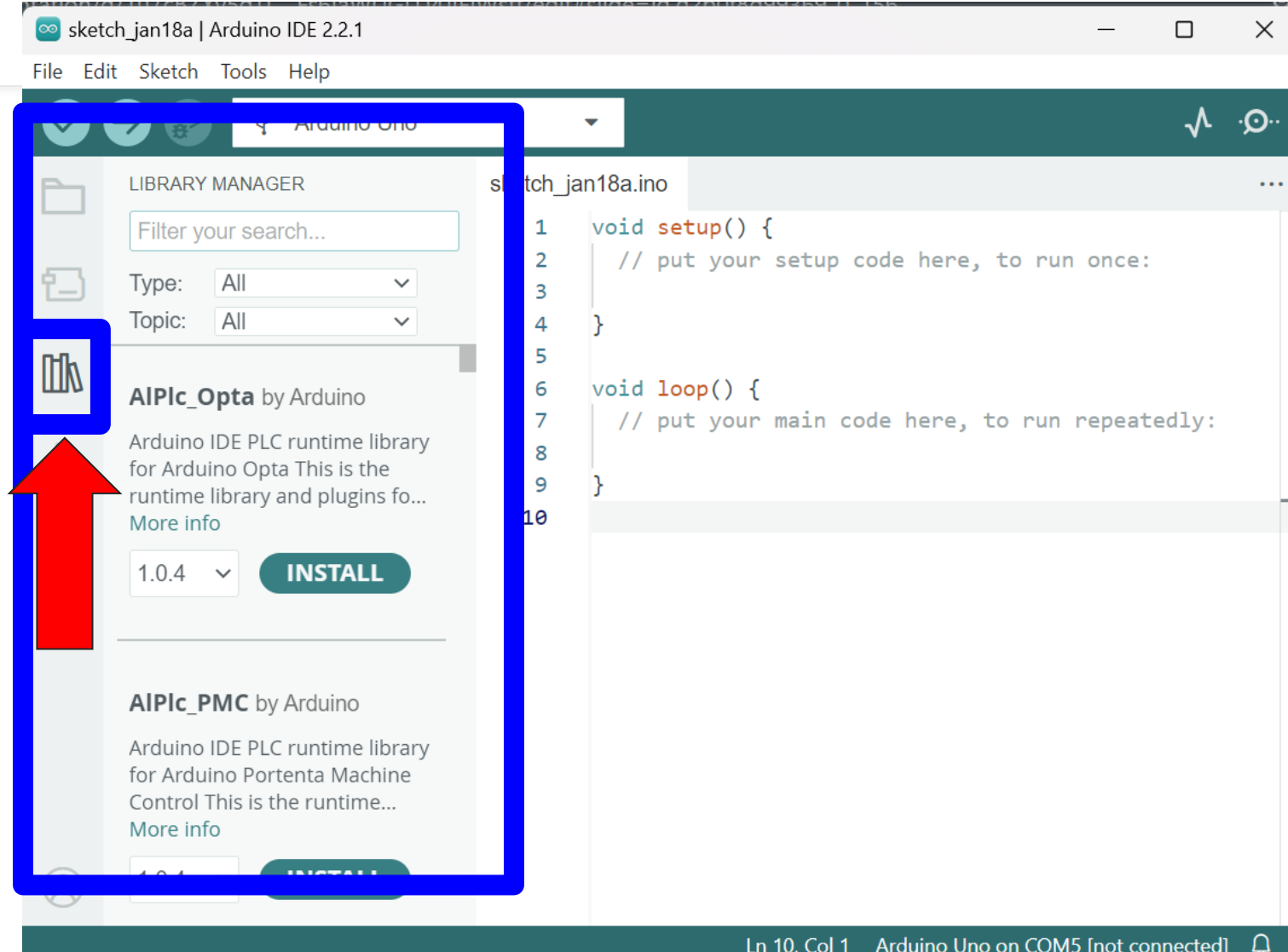
The **Arduino Boards Manager** is a tool within the Arduino IDE that allows users to easily install, update, and manage additional board support packages for various microcontroller platforms "



• Components of Arduino IDE:



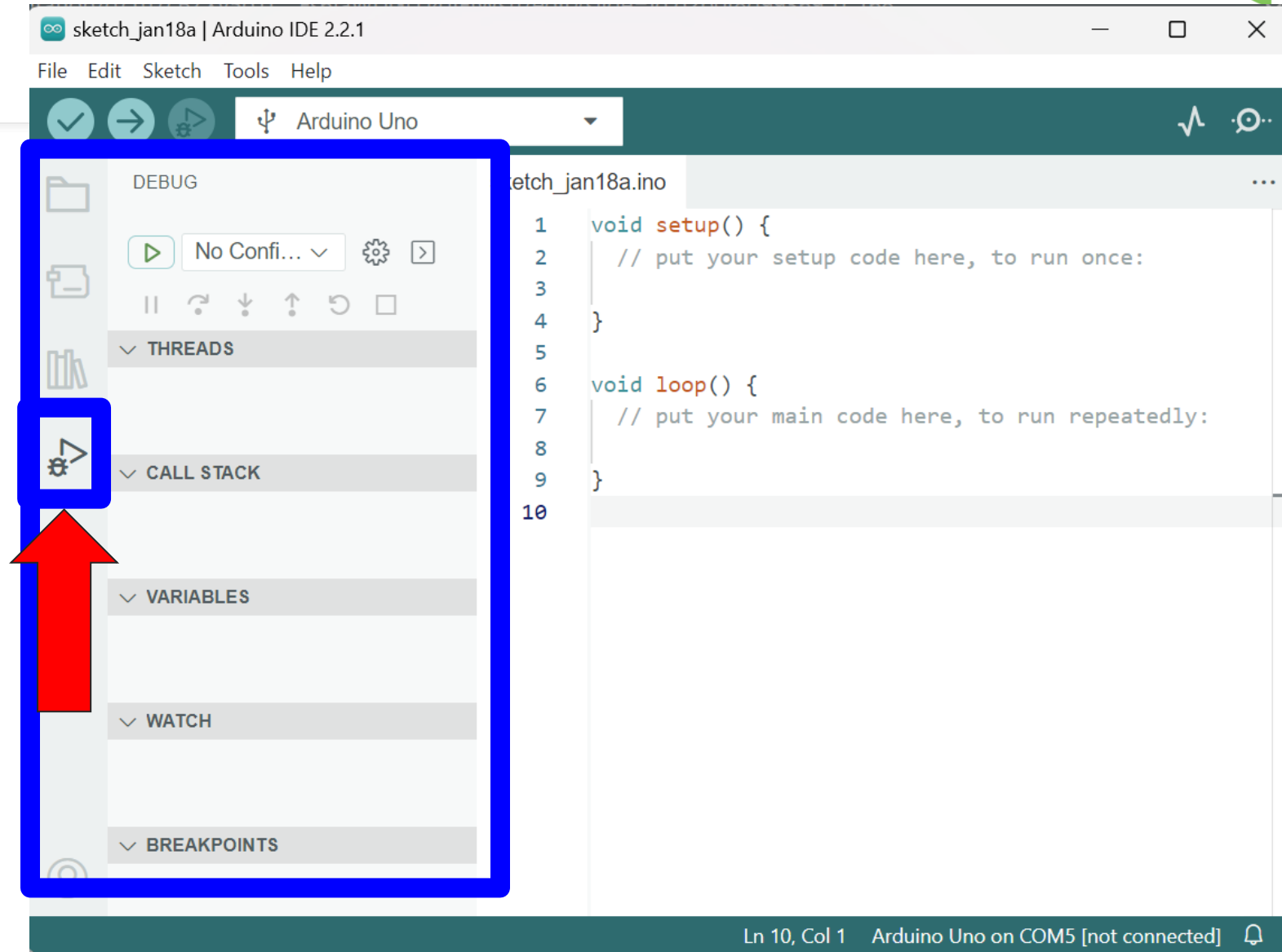
"The **Arduino Library Manager** is a feature in the Arduino IDE enabling users to effortlessly discover, install, and manage libraries, extending the functionality of their projects with pre-written code modules."



• Components of Arduino IDE:



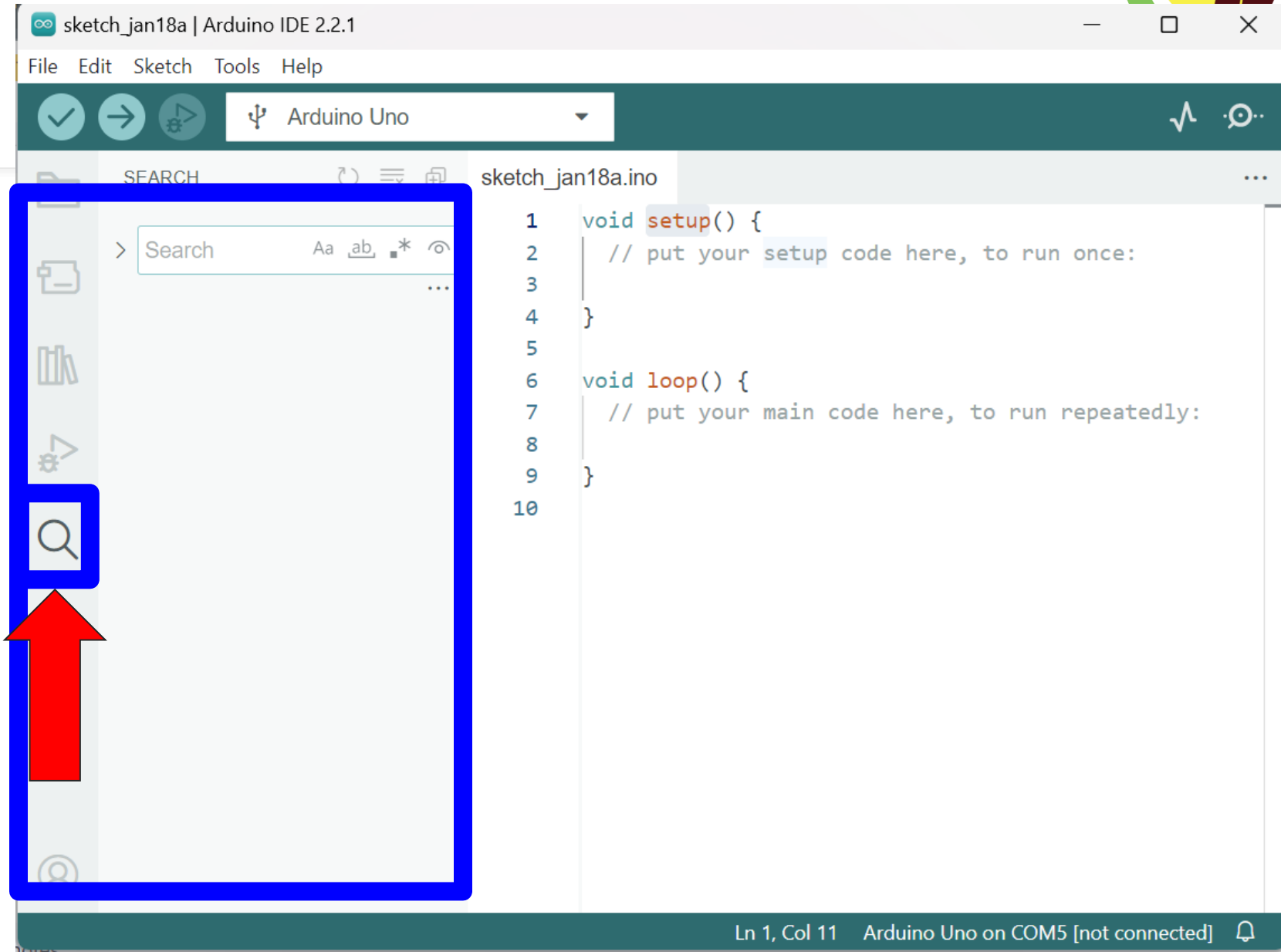
"The Arduino debug option is a feature that facilitates code debugging by allowing users to set breakpoints, inspect variable values, and step through their code for troubleshooting and optimization."



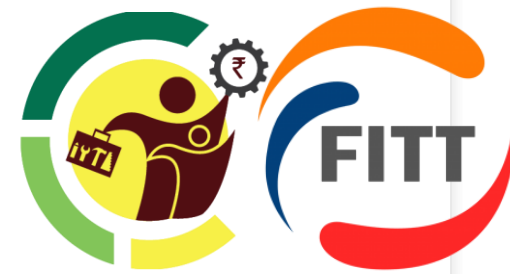
• Components of Arduino IDE:



The Arduino search option is a tool in the IDE that enables users to quickly find functions, variables, and code snippets within their sketches, streamlining the development process."



• Components of Arduino IDE:



Things to Keep in Mind Before Writing the First Code:

- **Select the Correct Board:** Before uploading your code, ensure you've selected the correct Arduino board model in the "Tools" menu.
- **Choose the Correct Port:** In the "Tools" menu, select the appropriate port to which your Arduino is connected.
- **Understand the Basics:** Familiarize yourself with basic Arduino syntax, functions, and the structure of a sketch. Arduino uses a simplified version of C++, so some programming knowledge is beneficial.
- **Power Supply:** If you are using external components (sensors, LEDs, etc.), make sure to connect a power supply to your Arduino board if it requires more power than the USB connection provides.
- **Upload Baud Rate:** Ensure that the baud rate in the code matches the baud rate in the Serial Monitor if you plan to communicate via the serial port.

By keeping these considerations in mind, you'll be well-prepared to write and upload your first Arduino code using the Arduino IDE.

• Arduino IDE for Programming

- The Blink LED Example

Blink | Arduino IDE 2.2.1

File Edit Sketch Tools Help

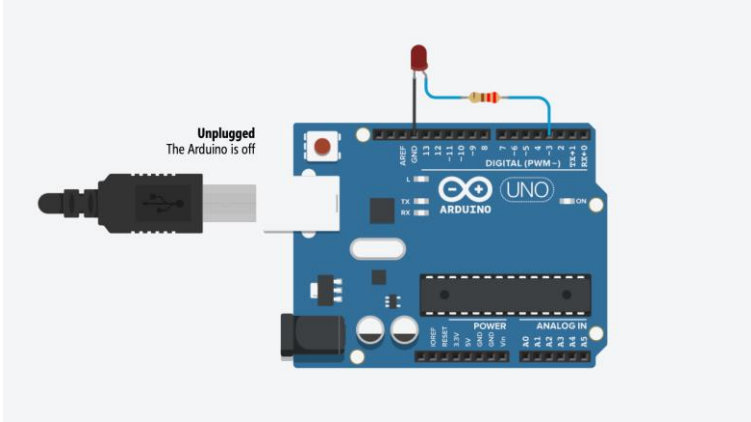
✓ → ⚙ Arduino Uno

Blink.ino

```

11  https://www.arduino.cc/en/Main/Products
12
13  modified 8 May 2014
14  by Scott Fitzgerald
15  modified 2 Sep 2016
16  by Arturo Guadalupi
17  modified 8 Sep 2016
18  by Colby Newman
19
20  This example code is in the public domain.
21
22  https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blink
23  */
24
25  // the setup function runs once when you press reset or power the board
26  void setup() {
27    // initialize digital pin LED_BUILTIN as an output.
28    pinMode(LED_BUILTIN, OUTPUT);
29  }
30
31  // the loop function runs over and over again forever
32  void loop() {
33    digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
34    delay(1000); // wait for a second
35    digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
36    delay(1000); // wait for a second
37  }
38

```

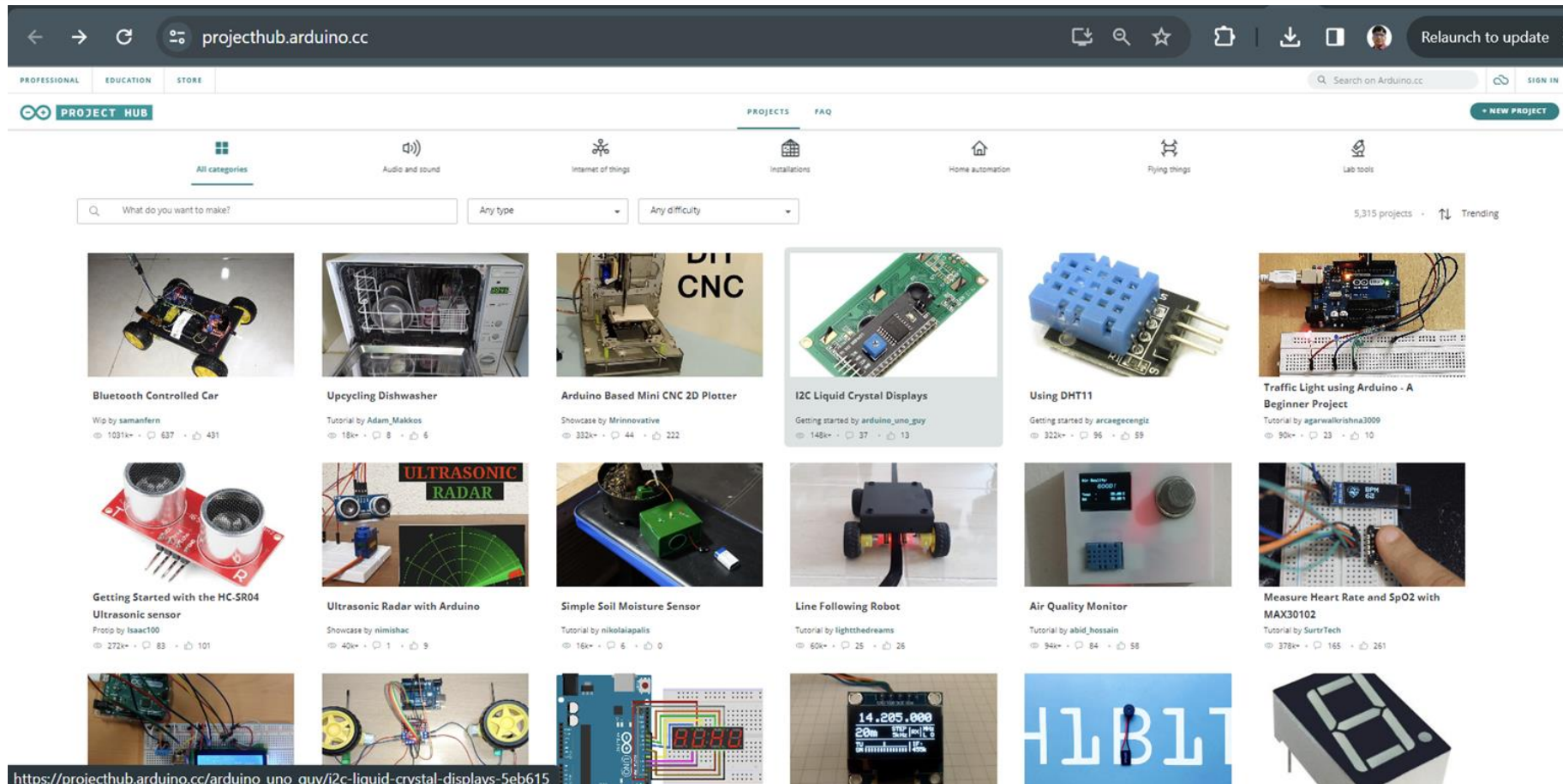


Unplugged
The Arduino is off

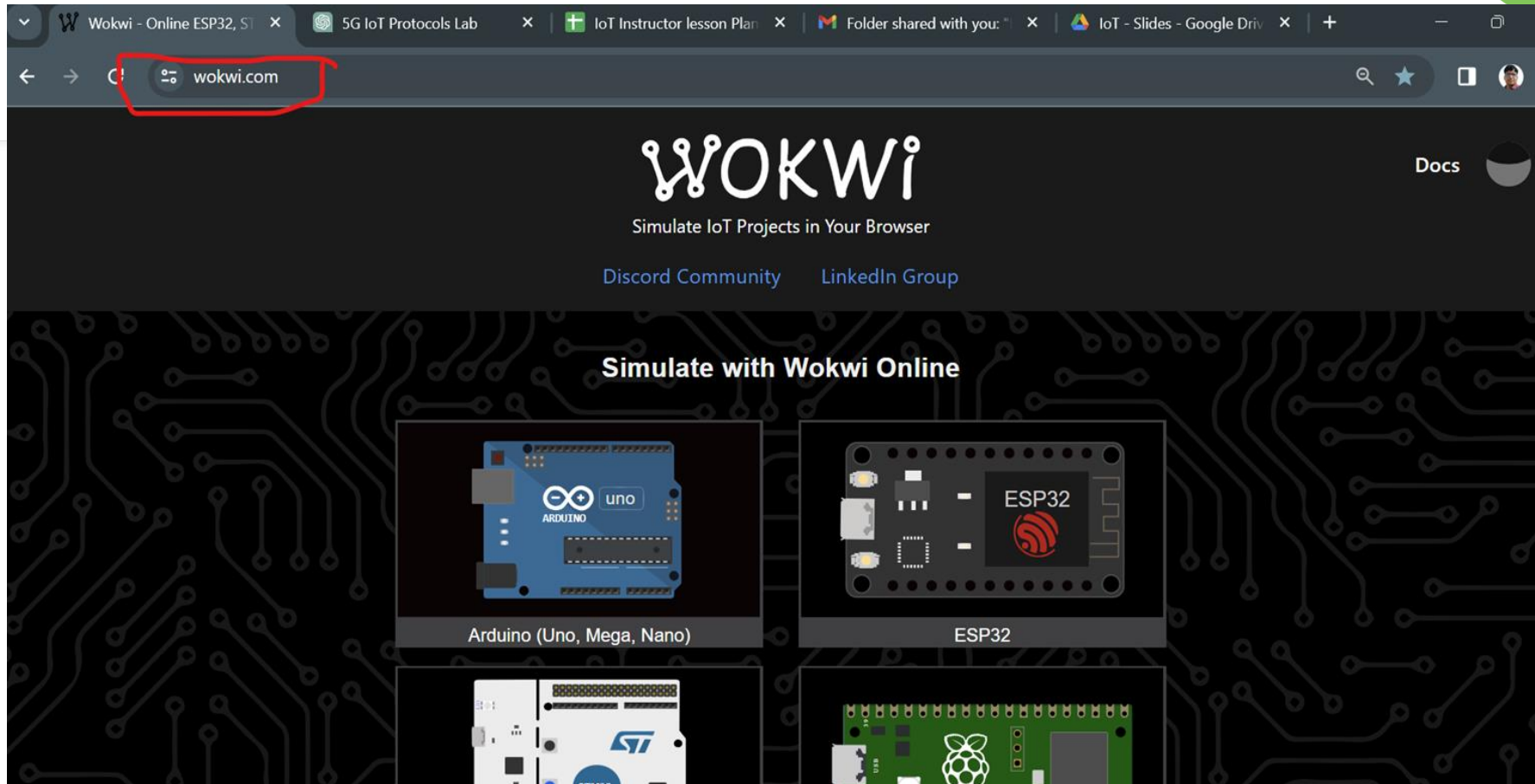
Ln 1, Col 1 Arduino Uno on COM5 [not connected]

- # Arduino IDE for Programming

Find Interesting Projects at-<https://projecthub.arduino.cc/>



Online tools (Simulator) - Wokwi.com



Let's have a look at the browser screen to interact with the online simulator.

Online tools (Simulator) - Wokwi.com

Let's Try Blink-LED with Wokwi

WOKWI

SAVE

SHARE

ArduinoUnoBlink by naimish

Docs

sketch.ino

diagram.json

Library Manager

```

1  /*
2   Blink
3
4   Turns an LED on for one second, then off for one second, repeatedly.
5
6   Most Arduinos have an on-board LED you can control. On the UNO, MEGA and ZERO
7   it is attached to digital pin 13, on MKR1000 on pin 6. LED_BUILTIN is set to
8   the correct LED pin independent of which board is used.
9   If you want to know what pin the on-board LED is connected to on your Arduino
10  model, check the Technical Specs of your board at:
11  https://www.arduino.cc/en/Main/Products
12
13  modified 8 May 2014
14  by Scott Fitzgerald
15  modified 2 Sep 2016
16  by Arturo Guadalupi
17  modified 8 Sep 2016
18  by Colby Newman
19
20  This example code is in the public domain.
21
22  https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blink
23  */
24
25  // the setup function runs once when you press reset or power the board
26  void setup() {
27    // initialize digital pin LED_BUILTIN as an output.
28    pinMode(LED_BUILTIN, OUTPUT);
29  }
30

```

Simulation

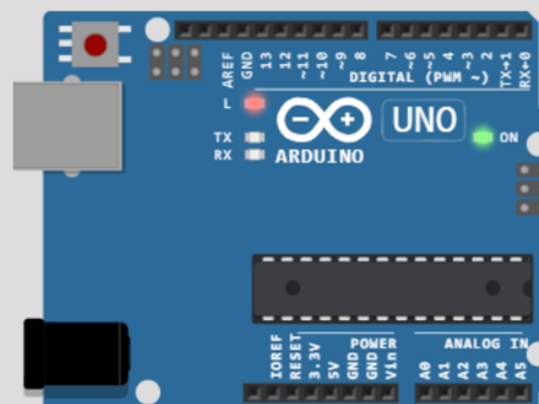
00:17.483

99%

↺

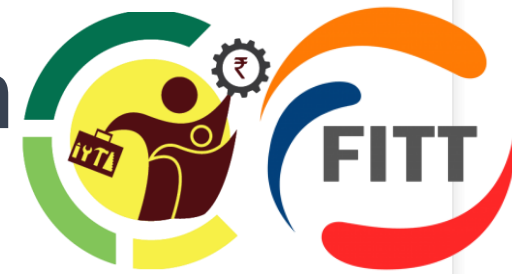
⏏

⏸



Online tools - Example Projects simulation

Interesting Featured Projects



wokwi.com/arduino

Featured projects

Simon Says Game

Simon Says Game
Copyright (C) 2017, UNIST
Released under the MIT License.
#include "pitches.h"
// Constants - define pin numbers for buttons, speaker, and LED
#define BUTTONS_PIN 2
#define SPEAKER_PIN 4
#define LED_PIN 13

32 Servos Dancing

// ServoDance.ino
//
// Example for multiple servos in a circle
//
// Version 1, 28 July 2015
// Version 2, 15 Aug 2015
// Changed timing, added
// diagram, pin has been changed (is visually better)
// Added fourth sequence
//
// Public Domain

32x32 LED Matrix Tunnel

// This version uses bit-banged SPI.
// If you need hardware-based SPI, see the other version.
//
// Define WGM (segments) to (segments * 4) * 4 (SEGMENTS)
#define WGM (SEGMENTS * 4 * 4)

Mini Piano

Mini piano for Arduino.
You can control the piano by pressing the keys on the board.
After starting the simulation, press the keys to play a melody.
Then press any key between 1 and 10 to play a note (1 is the lowest note, 10 is the highest).
Copyright (C) 2021, UNIST
Released under the MIT License.
#include "pitches.h"

Alarm Clock with RTC

Arduino Display
Copyright (C) 2017, UNIST
Released under the MIT License.
#include <Servo.h>
#include <Buttons.h>
#include <AlarmClock.h>
#include <RTC.h>

Electronic Safe

Arduino Electronics
Copyright (C) 2017, UNIST
Released under the MIT License.
#include <LiquidCrystal.h>
#include <Keypad.h>
#include <Servo.h>
#include <SafeState.h>
#include <com.h>

Nano Pong

A simple Pong game.
https://medium.com/@maverickwong
Based on Arduino Pong by
https://github.com/maverickwong
#include <SPI.h>
#include <Adafruit_GFX.h>
#include <Adafruit_GFX.h>

Touch LCD Breakout Game

#include <SD.h>
#include <Adafruit_1947_Oled.h>
#include <ourOSD.h>
ourOSD myOSD;
void setup() {
 Serial.begin(57600);
 if (!myScreen) {
 Serial.println("No screen found. Try a different screen or cable. Tell us:");
 Serial.println("https://github.com/maverickwong");
 }
}

Dino Game

{
 "version": 1,
 "author": "Ami Sung",
 "editor": "wokwi",
 "parts": [
 {
 "type": "wokwi:arduino:uno",
 "x": 100, "y": 100, "label": "Arduino Uno",
 "pin": "GND",
 "pin": "5V",
 "pin": "D0",
 "pin": "D1",
 "pin": "D2",
 "pin": "D3",
 "pin": "D4",
 "pin": "D5",
 "pin": "D6",
 "pin": "D7",
 "pin": "D8",
 "pin": "D9",
 "pin": "D10",
 "pin": "D11",
 "pin": "D12",
 "pin": "D13",
 "pin": "A0",
 "pin": "A1",
 "pin": "A2",
 "pin": "A3",
 "pin": "A4",
 "pin": "A5",
 "pin": "A6",
 "pin": "A7",
 "pin": "A8",
 "pin": "A9",
 "pin": "A10",
 "pin": "A11",
 "pin": "A12",
 "pin": "A13",
 "pin": "A14",
 "pin": "A15",
 "pin": "A16",
 "pin": "A17",
 "pin": "A18",
 "pin": "A19",
 "pin": "A20",
 "pin": "A21",
 "pin": "A22",
 "pin": "A23",
 "pin": "A24",
 "pin": "A25",
 "pin": "A26",
 "pin": "A27",
 "pin": "A28",
 "pin": "A29",
 "pin": "A30",
 "pin": "A31",
 "pin": "A32",
 "pin": "A33",
 "pin": "A34",
 "pin": "A35",
 "pin": "A36",
 "pin": "A37",
 "pin": "A38",
 "pin": "A39",
 "pin": "A40",
 "pin": "A41",
 "pin": "A42",
 "pin": "A43",
 "pin": "A44",
 "pin": "A45",
 "pin": "A46",
 "pin": "A47",
 "pin": "A48",
 "pin": "A49",
 "pin": "A50",
 "pin": "A51",
 "pin": "A52",
 "pin": "A53",
 "pin": "A54",
 "pin": "A55",
 "pin": "A56",
 "pin": "A57",
 "pin": "A58",
 "pin": "A59",
 "pin": "A60",
 "pin": "A61",
 "pin": "A62",
 "pin": "A63",
 "pin": "A64",
 "pin": "A65",
 "pin": "A66",
 "pin": "A67",
 "pin": "A68",
 "pin": "A69",
 "pin": "A70",
 "pin": "A71",
 "pin": "A72",
 "pin": "A73",
 "pin": "A74",
 "pin": "A75",
 "pin": "A76",
 "pin": "A77",
 "pin": "A78",
 "pin": "A79",
 "pin": "A80",
 "pin": "A81",
 "pin": "A82",
 "pin": "A83",
 "pin": "A84",
 "pin": "A85",
 "pin": "A86",
 "pin": "A87",
 "pin": "A88",
 "pin": "A89",
 "pin": "A90",
 "pin": "A91",
 "pin": "A92",
 "pin": "A93",
 "pin": "A94",
 "pin": "A95",
 "pin": "A96",
 "pin": "A97",
 "pin": "A98",
 "pin": "A99",
 "pin": "A100",
 "pin": "A101",
 "pin": "A102",
 "pin": "A103",
 "pin": "A104",
 "pin": "A105",
 "pin": "A106",
 "pin": "A107",
 "pin": "A108",
 "pin": "A109",
 "pin": "A110",
 "pin": "A111",
 "pin": "A112",
 "pin": "A113",
 "pin": "A114",
 "pin": "A115",
 "pin": "A116",
 "pin": "A117",
 "pin": "A118",
 "pin": "A119",
 "pin": "A120",
 "pin": "A121",
 "pin": "A122",
 "pin": "A123",
 "pin": "A124",
 "pin": "A125",
 "pin": "A126",
 "pin": "A127",
 "pin": "A128",
 "pin": "A129",
 "pin": "A130",
 "pin": "A131",
 "pin": "A132",
 "pin": "A133",
 "pin": "A134",
 "pin": "A135",
 "pin": "A136",
 "pin": "A137",
 "pin": "A138",
 "pin": "A139",
 "pin": "A140",
 "pin": "A141",
 "pin": "A142",
 "pin": "A143",
 "pin": "A144",
 "pin": "A145",
 "pin": "A146",
 "pin": "A147",
 "pin": "A148",
 "pin": "A149",
 "pin": "A150",
 "pin": "A151",
 "pin": "A152",
 "pin": "A153",
 "pin": "A154",
 "pin": "A155",
 "pin": "A156",
 "pin": "A157",
 "pin": "A158",
 "pin": "A159",
 "pin": "A160",
 "pin": "A161",
 "pin": "A162",
 "pin": "A163",
 "pin": "A164",
 "pin": "A165",
 "pin": "A166",
 "pin": "A167",
 "pin": "A168",
 "pin": "A169",
 "pin": "A170",
 "pin": "A171",
 "pin": "A172",
 "pin": "A173",
 "pin": "A174",
 "pin": "A175",
 "pin": "A176",
 "pin": "A177",
 "pin": "A178",
 "pin": "A179",
 "pin": "A180",
 "pin": "A181",
 "pin": "A182",
 "pin": "A183",
 "pin": "A184",
 "pin": "A185",
 "pin": "A186",
 "pin": "A187",
 "pin": "A188",
 "pin": "A189",
 "pin": "A190",
 "pin": "A191",
 "pin": "A192",
 "pin": "A193",
 "pin": "A194",
 "pin": "A195",
 "pin": "A196",
 "pin": "A197",
 "pin": "A198",
 "pin": "A199",
 "pin": "A200",
 "pin": "A201",
 "pin": "A202",
 "pin": "A203",
 "pin": "A204",
 "pin": "A205",
 "pin": "A206",
 "pin": "A207",
 "pin": "A208",
 "pin": "A209",
 "pin": "A210",
 "pin": "A211",
 "pin": "A212",
 "pin": "A213",
 "pin": "A214",
 "pin": "A215",
 "pin": "A216",
 "pin": "A217",
 "pin": "A218",
 "pin": "A219",
 "pin": "A220",
 "pin": "A221",
 "pin": "A222",
 "pin": "A223",
 "pin": "A224",
 "pin": "A225",
 "pin": "A226",
 "pin": "A227",
 "pin": "A228",
 "pin": "A229",
 "pin": "A230",
 "pin": "A231",
 "pin": "A232",
 "pin": "A233",
 "pin": "A234",
 "pin": "A235",
 "pin": "A236",
 "pin": "A237",
 "pin": "A238",
 "pin": "A239",
 "pin": "A240",
 "pin": "A241",
 "pin": "A242",
 "pin": "A243",
 "pin": "A244",
 "pin": "A245",
 "pin": "A246",
 "pin": "A247",
 "pin": "A248",
 "pin": "A249",
 "pin": "A250",
 "pin": "A251",
 "pin": "A252",
 "pin": "A253",
 "pin": "A254",
 "pin": "A255",
 "pin": "A256",
 "pin": "A257",
 "pin": "A258",
 "pin": "A259",
 "pin": "A260",
 "pin": "A261",
 "pin": "A262",
 "pin": "A263",
 "pin": "A264",
 "pin": "A265",
 "pin": "A266",
 "pin": "A267",
 "pin": "A268",
 "pin": "A269",
 "pin": "A270",
 "pin": "A271",
 "pin": "A272",
 "pin": "A273",
 "pin": "A274",
 "pin": "A275",
 "pin": "A276",
 "pin": "A277",
 "pin": "A278",
 "pin": "A279",
 "pin": "A280",
 "pin": "A281",
 "pin": "A282",
 "pin": "A283",
 "pin": "A284",
 "pin": "A285",
 "pin": "A286",
 "pin": "A287",
 "pin": "A288",
 "pin": "A289",
 "pin": "A290",
 "pin": "A291",
 "pin": "A292",
 "pin": "A293",
 "pin": "A294",
 "pin": "A295",
 "pin": "A296",
 "pin": "A297",
 "pin": "A298",
 "pin": "A299",
 "pin": "A300",
 "pin": "A301",
 "pin": "A302",
 "pin": "A303",
 "pin": "A304",
 "pin": "A305",
 "pin": "A306",
 "pin": "A307",
 "pin": "A308",
 "pin": "A309",
 "pin": "A310",
 "pin": "A311",
 "pin": "A312",
 "pin": "A313",
 "pin": "A314",
 "pin": "A315",
 "pin": "A316",
 "pin": "A317",
 "pin": "A318",
 "pin": "A319",
 "pin": "A320",
 "pin": "A321",
 "pin": "A322",
 "pin": "A323",
 "pin": "A324",
 "pin": "A325",
 "pin": "A326",
 "pin": "A327",
 "pin": "A328",
 "pin": "A329",
 "pin": "A330",
 "pin": "A331",
 "pin": "A332",
 "pin": "A333",
 "pin": "A334",
 "pin": "A335",
 "pin": "A336",
 "pin": "A337",
 "pin": "A338",
 "pin": "A339",
 "pin": "A340",
 "pin": "A341",
 "pin": "A342",
 "pin": "A343",
 "pin": "A344",
 "pin": "A345",
 "pin": "A346",
 "pin": "A347",
 "pin": "A348",
 "pin": "A349",
 "pin": "A350",
 "pin": "A351",
 "pin": "A352",
 "pin": "A353",
 "pin": "A354",
 "pin": "A355",
 "pin": "A356",
 "pin": "A357",
 "pin": "A358",
 "pin": "A359",
 "pin": "A360",
 "pin": "A361",
 "pin": "A362",
 "pin": "A363",
 "pin": "A364",
 "pin": "A365",
 "pin": "A366",
 "pin": "A367",
 "pin": "A368",
 "pin": "A369",
 "pin": "A370",
 "pin": "A371",
 "pin": "A372",
 "pin": "A373",
 "pin": "A374",
 "pin": "A375",
 "pin": "A376",
 "pin": "A377",
 "pin": "A378",
 "pin": "A379",
 "pin": "A380",
 "pin": "A381",
 "pin": "A382",
 "pin": "A383",
 "pin": "A384",
 "pin": "A385",
 "pin": "A386",
 "pin": "A387",
 "pin": "A388",
 "pin": "A389",
 "pin": "A390",
 "pin": "A391",
 "pin": "A392",
 "pin": "A393",
 "pin": "A394",
 "pin": "A395",
 "pin": "A396",
 "pin": "A397",
 "pin": "A398",
 "pin": "A399",
 "pin": "A400",
 "pin": "A401",
 "pin": "A402",
 "pin": "A403",
 "pin": "A404",
 "pin": "A405",
 "pin": "A406",
 "pin": "A407",
 "pin": "A408",
 "pin": "A409",
 "pin": "A410",
 "pin": "A411",
 "pin": "A412",
 "pin": "A413",
 "pin": "A414",
 "pin": "A415",
 "pin": "A416",
 "pin": "A417",
 "pin": "A418",
 "pin": "A419",
 "pin": "A420",
 "pin": "A421",
 "pin": "A422",
 "pin": "A423",
 "pin": "A424",
 "pin": "A425",
 "pin": "A426",
 "pin": "A427",
 "pin": "A428",
 "pin": "A429",
 "pin": "A430",
 "pin": "A431",
 "pin": "A432",
 "pin": "A433",
 "pin": "A434",
 "pin": "A435",
 "pin": "A436",
 "pin": "A437",
 "pin": "A438",
 "pin": "A439",
 "pin": "A440",
 "pin": "A441",
 "pin": "A442",
 "pin": "A443",
 "pin": "A444",
 "pin": "A445",
 "pin": "A446",
 "pin": "A447",
 "pin": "A448",
 "pin": "A449",
 "pin": "A450",
 "pin": "A451",
 "pin": "A452",
 "pin": "A453",
 "pin": "A454",
 "pin": "A455",
 "pin": "A456",
 "pin": "A457",
 "pin": "A458",
 "pin": "A459",
 "pin": "A460",
 "pin": "A461",
 "pin": "A462",
 "pin": "A463",
 "pin": "A464",
 "pin": "A465",
 "pin": "A466",
 "pin": "A467",
 "pin": "A468",
 "pin": "A469",
 "pin": "A470",
 "pin": "A471",
 "pin": "A472",
 "pin": "A473",
 "pin": "A474",
 "pin": "A475",
 "pin": "A476",
 "pin": "A477",
 "pin": "A478",
 "pin": "A479",
 "pin": "A480",
 "pin": "A481",
 "pin": "A482",
 "pin": "A483",
 "pin": "A484",
 "pin": "A485",
 "pin": "A486",
 "pin": "A487",
 "pin": "A488",
 "pin": "A489",
 "pin": "A490",
 "pin": "A491",
 "pin": "A492",
 "pin": "A493",
 "pin": "A494",
 "pin": "A495",
 "pin": "A496",
 "pin": "A497",
 "pin": "A498",
 "pin": "A499",
 "pin": "A500",
 "pin": "A501",
 "pin": "A502",
 "pin": "A503",
 "pin": "A504",
 "pin": "A505",
 "pin": "A506",
 "pin": "A507",
 "pin": "A508",
 "pin": "A509",
 "pin": "A510",
 "pin": "A511",
 "pin": "A512",
 "pin": "A513",
 "pin": "A514",
 "pin": "A515",
 "pin": "A516",
 "pin": "A517",
 "pin": "A518",
 "pin": "A519",
 "pin": "A520",
 "pin": "A521",
 "pin": "A522",
 "pin": "A523",
 "pin": "A524",
 "pin": "A525",
 "pin": "A526",
 "pin": "A527",
 "pin": "A528",
 "pin": "A529",
 "pin": "A530",
 "pin": "A531",
 "pin": "A532",
 "pin": "A533",
 "pin": "A534",
 "pin": "A535",
 "pin": "A536",
 "pin": "A537",
 "pin": "A538",
 "pin": "A539",
 "pin": "A540",
 "pin": "A541",
 "pin": "A542",
 "pin": "A543",
 "pin": "A544",
 "pin": "A545",
 "pin": "A546",
 "pin": "A547",
 "pin": "A548",
 "pin": "A549",
 "pin": "A550",
 "pin": "A551",
 "pin": "A552",
 "pin": "A553",
 "pin": "A554",
 "pin": "A555",
 "pin": "A556",
 "pin": "A557",
 "pin": "A558",
 "pin": "A559",
 "pin": "A560",
 "pin": "A561",
 "pin": "A562",
 "pin": "A563",
 "pin": "A564",
 "pin": "A565",
 "pin": "A566",
 "pin": "A567",
 "pin": "A568",
 "pin": "A569",
 "pin": "A570",
 "pin": "A571",
 "pin": "A572",
 "pin": "A573",
 "pin": "A574",
 "pin": "A575",
 "pin": "A576",
 "pin": "A577",
 "pin": "A578",
 "pin": "A579",
 "pin": "A580",
 "pin": "A581",
 "pin": "A582",
 "pin": "A583",
 "pin": "A584",
 "pin": "A585",
 "pin": "A586",
 "pin": "A587",
 "pin": "A588",
 "pin": "A589",
 "pin": "A590",
 "pin": "A591",
 "pin": "A592",
 "pin": "A593",
 "pin": "A594",
 "pin": "A595",
 "pin": "A596",
 "pin": "A597",
 "pin": "A598",
 "pin": "A599",
 "pin": "A600",
 "pin": "A601",
 "pin": "A602",
 "pin": "A603",
 "pin": "A604",
 "pin": "A605",
 "pin": "A606",
 "pin": "A607",
 "pin": "A608",
 "pin": "A609",
 "pin": "A610",
 "pin": "A611",
 "pin": "A612",
 "pin": "A613",
 "pin": "A614",
 "pin": "A615",
 "pin": "A616",
 "pin": "A617",
 "pin": "A618",
 "pin": "A619",
 "pin": "A620",
 "pin": "A621",
 "pin": "A622",
 "pin": "A623",
 "pin": "A624",
 "pin": "A625",
 "pin": "A626",
 "pin": "A627",
 "pin": "A628",
 "pin": "A629",
 "pin": "A630",
 "pin": "A631",
 "pin": "A632",
 "pin": "A633",
 "pin": "A634",
 "pin": "A635",
 "pin": "A636",
 "pin": "A637",
 "pin": "A638",
 "pin": "A639",
 "pin": "A640",
 "pin": "A641",
 "pin": "A642",
 "pin": "A643",
 "pin": "A644",
 "pin": "A645",
 "pin": "A646",
 "pin": "A647",
 "pin": "A648",
 "pin": "A649",
 "pin": "A650",
 "pin": "A651",
 "pin": "A652",
 "pin": "A653",
 "pin": "A654",
 "pin": "A655",
 "pin": "A656",
 "pin": "A657",
 "pin": "A658",
 "pin": "A659",
 "pin": "A660",
 "pin": "A661",
 "pin": "A662",
 "pin": "A663",
 "pin": "A664",
 "pin": "A665",
 "pin": "A666",
 "pin": "A667",
 "pin": "A668",
 "pin": "A669",
 "pin": "A670",
 "pin": "A671",
 "pin": "A672",
 "pin": "A673",
 "pin": "A674",
 "pin": "A675",
 "pin": "A676",
 "pin": "A677",
 "pin": "A678",
 "pin": "A679",
 "pin": "A680",
 "pin": "A681",
 "pin": "A682",
 "pin": "A683",
 "pin": "A684",
 "pin": "A685",
 "pin": "A686",
 "pin": "A687",
 "pin": "A688",
 "pin": "A689",
 "pin": "A690",
 "pin": "A691",
 "pin": "A692",
 "pin": "A693",
 "pin": "A694",
 "pin": "A695",
 "pin": "A696",
 "pin": "A697",
 "pin": "A698",
 "pin": "A699",
 "pin": "A700",
 "pin": "A701",
 "pin": "A702",
 "pin": "A703",
 "pin": "A704",
 "pin": "A705",
 "pin": "A706",
 "pin": "A707",
 "pin": "A708",
 "pin": "A709",
 "pin": "A710",
 "pin": "A711",
 "pin": "A712",
 "pin": "A713",
 "pin": "A714",
 "pin": "A715",
 "pin": "A716",
 "pin": "A717",
 "pin": "A718",
 "pin": "A719",
 "pin": "A720",
 "pin": "A721",
 "pin": "A722",
 "pin": "A723",
 "pin": "A724",
 "pin": "A725",
 "pin": "A726",
 "pin": "A727",
 "pin": "A728",
 "pin": "A729",
 "pin": "A730",
 "pin": "A731",
 "pin": "A732",
 "pin": "A733",
 "pin": "A734",
 "pin": "A735",
 "pin": "A736",
 "pin": "A737",
 "pin": "A738",
 "pin": "A739",
 "pin": "A740",
 "pin": "A741",
 "pin": "A742",
 "pin": "A743",
 "pin": "A744",
 "pin": "A745",
 "pin": "A746",
 "pin": "A747",
 "pin": "A748",
 "pin": "A749",
 "pin": "A750",
 "pin": "A751",
 "pin": "A752",
 "pin": "A753",
 "pin": "A754",
 "pin": "A755",
 "pin": "A756",
 "pin": "A757",
 "pin": "A758",
 "pin": "A759",
 "pin": "A760",
 "pin": "A761",
 "pin": "A762",
 "pin": "A763",
 "pin": "A764",
 "pin": "A765",
 "pin": "A766",
 "pin": "A767",
 "pin": "A768",
 "pin": "A769",
 "pin": "A770",
 "pin": "A771",
 "pin": "A772",
 "pin": "A773",
 "pin": "A774",
 "pin": "A775",
 "pin": "A776",
 "pin": "A777",
 "pin": "A778",
 "pin": "A779",
 "pin": "A780",
 "pin": "A781",
 "pin": "A782",
 "pin": "A783",
 "pin": "A784",
 "pin": "A785",
 "pin": "A786",
 "pin": "A787",
 "pin": "A788",
 "pin": "A789",
 "pin": "A790",
 "pin": "A791",
 "pin": "A792",
 "pin": "A793",
 "pin": "A794",
 "pin": "A795",
 "pin": "A796",
 "pin": "A797",
 "pin": "A798",
 "pin": "A799",
 "pin": "A800",
 "pin": "A801",
 "pin": "A802",
 "pin": "A803",
 "pin": "A804",
 "pin": "A805",
 "pin": "A806",
 "pin": "A807",
 "pin": "A808",
 "pin": "A809",
 "pin": "A810",
 "pin": "A811",
 "pin": "A812",
 "pin": "A813",
 "pin": "A814",
 "pin": "A815",
 "pin": "A816",
 "pin": "A817",
 "pin": "A818",
 "pin": "A819",
 "pin": "A820",
 "pin": "A821",
 "pin": "A822",
 "pin": "A823",
 "pin": "A824",
 "pin": "A825",
 "pin": "A826",
 "pin": "A827",
 "pin": "A828",
 "pin": "A829",
 "pin": "A830",
 "pin": "A831",
 "pin": "A832",
 "pin": "A833",
 "pin": "A834",
 "pin": "A835",
 "pin": "A836",
 "pin": "A837",
 "pin": "A838",
 "pin": "A839",
 "pin": "A840",
 "pin": "A841",
 "pin": "A842",
 "pin": "A843",
 "pin": "A844",
 "pin": "A845",
 "pin": "A846",
 "pin": "A847",
 "pin": "A848",
 "pin": "A849",
 "pin": "A850",
 "pin": "A851",
 "pin": "A852",
 "pin": "A853",
 "pin": "A854",
 "pin": "A855",
 "pin": "A856",
 "pin": "A857",
 "pin": "A858",
 "pin": "A859",
 "pin": "A860",
 "pin": "A861",
 "pin

Online tools - ServoMotor Example

WOKWI

SAVE

SHARE

Docs

ServoOverdone.ino

diagram.json

libraries.txt

Library Manager

```

1 // ServoOverdone.ino
2 //
3 // Example for multiple Servo objects in a array.
4 //
5 // Version 1, 28 July 2021, by Koepel.
6 // Version 2, 15 August 2021, by Koepel.
7 // changed timing, a little slower
8 // diagram.json has servos in reverse order (I think it is visually better)
9 // Added fourth sequence: "compass"
10 //
11 // Public Domain
12 //
13
14 #include <Servo.h>
15
16 #define NUM_SERVOS 32
17 Servo myServo[NUM_SERVOS];
18
19 void setup()
20 {
21   // Attach pins from the Arduino Mega board to the Servo objects.
22   // Starting from pin 22, there happen to be exactly 32 pins on the double row pi
23   for( int i=0; i<NUM_SERVOS; i++)
24   {
25     myServo[i].attach( i + 22);    // pin 22 up to 53 is 32 pins
26   }
27 }
28
29 void loop()
30 {

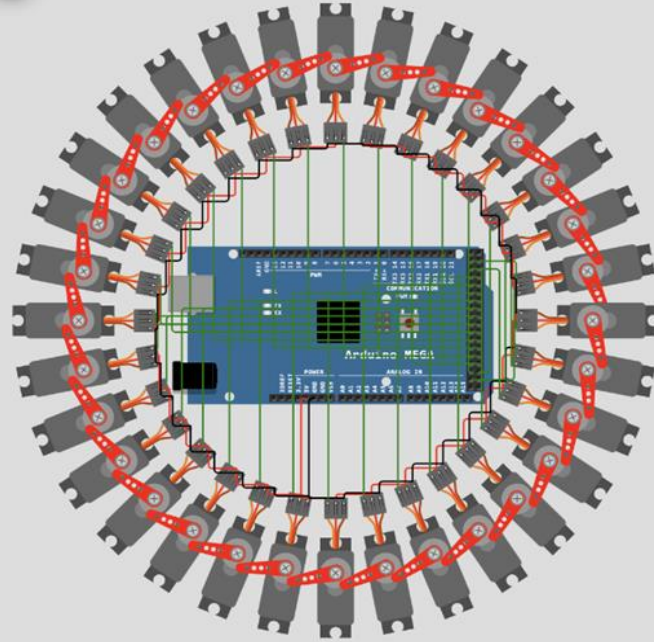
```

Simulation

▶

+

⋮



Online tools - Mini-Piano example

WOKWI SAVE SHARE mini-piano.ino by urish Docs

mini-piano.ino pitches.h diagram.json Library Manager

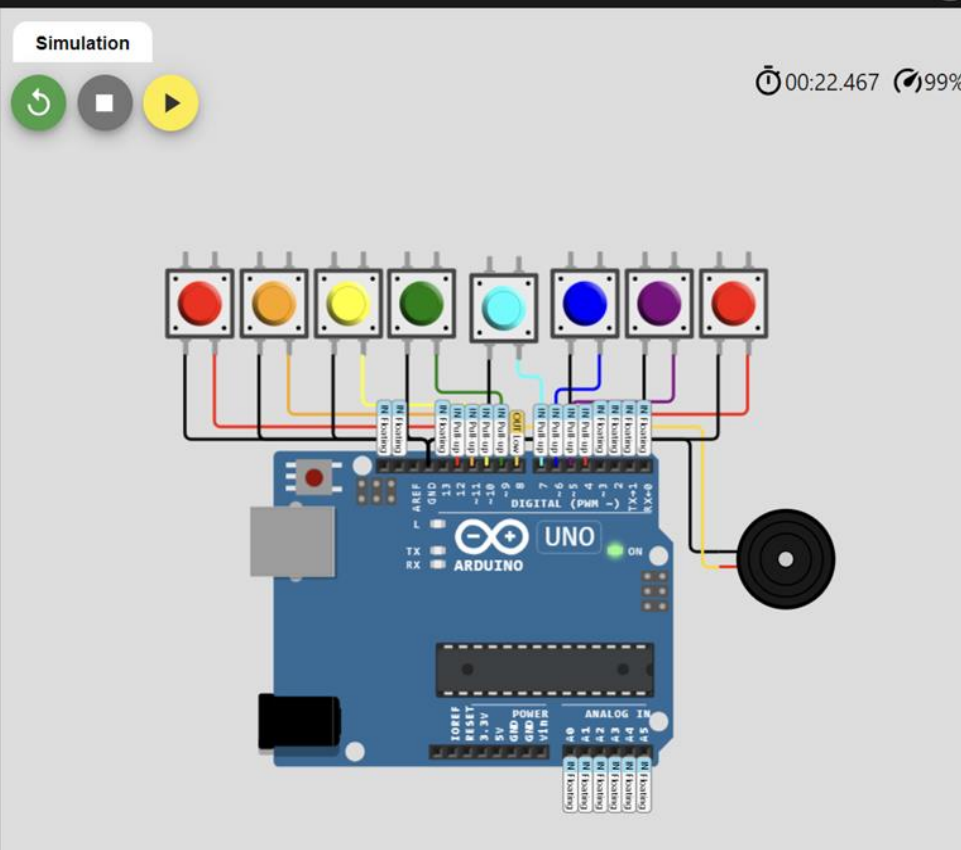
```

1  /**
2   * Mini piano for Arduino.
3   *
4   * You can control the colorful buttons with your keyboard:
5   * After starting the simulation, click anywhere in the diagram to focus it.
6   * Then press any key between 1 and 8 to play the piano (1 is the lowest note,
7   * 8 is the highest).
8   *
9   * Copyright (C) 2021, Uri Shaked. Released under the MIT License.
10  */
11
12  #include "pitches.h"
13
14  #define SPEAKER_PIN 8
15
16  const uint8_t buttonPins[] = { 12, 11, 10, 9, 7, 6, 5, 4 };
17  const int buttonTones[] = {
18    NOTE_C4, NOTE_D4, NOTE_E4, NOTE_F4,
19    NOTE_G4, NOTE_A4, NOTE_B4, NOTE_C5
20  };
21  const int numTones = sizeof(buttonPins) / sizeof(buttonPins[0]);
22
23  void setup() {
24    for (uint8_t i = 0; i < numTones; i++) {
25      pinMode(buttonPins[i], INPUT_PULLUP);
26    }
27    pinMode(SPEAKER_PIN, OUTPUT);
28  }
29
30  void loop() {

```

Simulation

00:22.467 99%



Online tools - External LED blink with Arduino UNO



WOKWI SAVE SHARE External LED blink on Arduino UNO_Naimish Docs

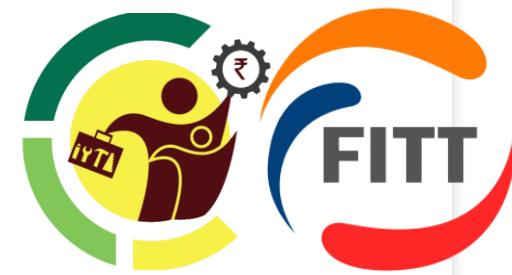
blink.ino • README.md diagram.json • Library Manager

```
21
22 http://www.arduino.cc/en/Tutorial/Blink
23 */
24 const int ledPin = 12;    // LED pin
25 // the setup function runs once when you press reset or power the board
26 void setup() {
27   // initialize digital pin LED_BUILTIN as an output.
28   pinMode(ledPin, OUTPUT);
29 }
30
31 // the loop function runs over and over again forever
32 void loop() {
33   digitalWrite(ledPin, HIGH);  // turn the LED on (HIGH is the voltage level)
34   delay(1000);                 // wait for a second
35   digitalWrite(ledPin, LOW);   // turn the LED off by making the voltage LOW
36   delay(1000);                 // wait for a second
37 }
38
```

Simulation Description

00:22.817 100%

Online tools - Serial monitor and LED blink with Arduino UNO



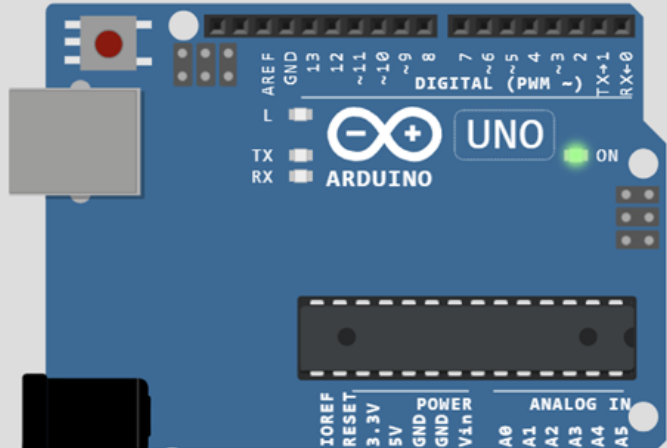
WOKWI Arduino serial monitor use by naimish

sketch.ino README.md diagram.json Library Manager

```
1 const int ledPin = 13; // Define the LED pin
2
3 void setup() {
4   pinMode(ledPin, OUTPUT); // Set the LED pin as an output
5   Serial.begin(9600);      // Initialize serial communication at 9600 bps
6 }
7
8 void loop() {
9   digitalWrite(ledPin, HIGH); // Turn the LED on
10  Serial.println("LED is ON!");
11  delay(1000);                // Wait for 1 second
12  digitalWrite(ledPin, LOW);  // Turn the LED off
13  Serial.println("LED is OFF!");
14  delay(1000);                // Wait for 1 second
15
16  // Send blink status to the Serial Monitor
17  Serial.println("LED is blinking!");
18  delay(1000);
19 }
```

Simulation Description

00:04.666 100%



LED is ON!
LED is OFF!
LED is blinking!
LED is ON!
LED is OFF!

The image shows a screenshot of the Wokwi online Arduino IDE. The left pane displays the 'sketch.ino' file with a C++ program for an LED blink. The right pane shows a simulation of an Arduino UNO board. Below the board, a serial monitor window displays the output of the program: 'LED is ON!', 'LED is OFF!', 'LED is blinking!', 'LED is ON!', and 'LED is OFF!'. The board itself has a green LED that is currently lit, and the serial monitor shows the status of the LED at each step of the loop.

Online tools - DTH22 (digital Temp and Humidity sensor) with Arduino



WOKWI SAVE SHARE ♥ DHT with Arduino _Naimish by naimish Docs

sketch.ino README.md diagram.json libraries.txt Library Manager

```
17 // Variables to store temperature and humidity values
18 float temp = 0;
19 float Humidity = 0;
20 float c = 0;
21 float f = 0;
22
23
24 void setup() {
25     // Initialize the LCD display
26     lcd.begin(20,4);
27     lcd.backlight();
28
29     // Display a welcome message on the LCD
30     lcd.setCursor(4,1);
31     lcd.print("Weather Station");
32     delay(3000);
33
34     // Clear the LCD and initialize the DHT sensor
35     lcd.clear();
36     Sensor.begin();
37
38     // Set the DHT22 pin as INPUT
39     pinMode(dhtpin,INPUT);
40
41 }
42
43 void loop() {
44     // Read humidity and temperature values from the DHT22 sensor
45     Humidity = Sensor.readHumidity();
```

Simulation Description

00:05.266 100%

TEMP (F): 80.6
Temp(C): 27.0 C
Humidity: 18.0 %

Setting Up ESP32 for Soil Moisture Sensor



Circuit Creation on simulator for Soil Moisture Sensor

WOKWI SAVE SHARE Soil Sensor with ESP32_Naimish Docs

soil_sensor.ino diagram.json soil_sensor.chip.c soil_sensor.chip.json libraries.txt

Library Manager

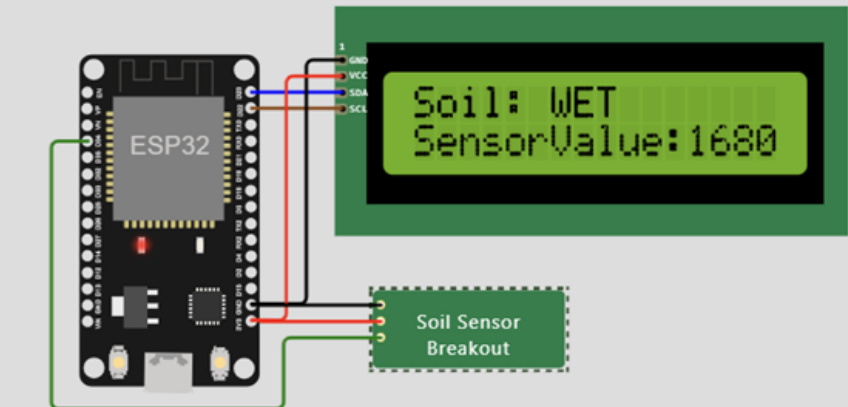
```
1  #include <Wire.h>
2  #include <LiquidCrystal_I2C.h>
3  LiquidCrystal_I2C lcd(0x27,16,2);
4
5  void setup()
6  {
7      Wire.begin(23, 22);
8      Serial.begin(9600);
9      lcd.init();
10     lcd.backlight();
11 }
12
13 void loop()
14 {
15     int16_t i = analogRead(34);
16     String msg = i < 2165 ? "WET" : i > 3135 ? "DRY" : "OK";
17     lcd.clear();
18     lcd.print("Soil: ");
19     lcd.print(msg);
20
21     lcd.setCursor(0, 1); // Set cursor to the first column and second row
22     lcd.print("SensorValue:");
23     lcd.print(i);
24
25     delay(500);
26 }
27
28
```

Simulation

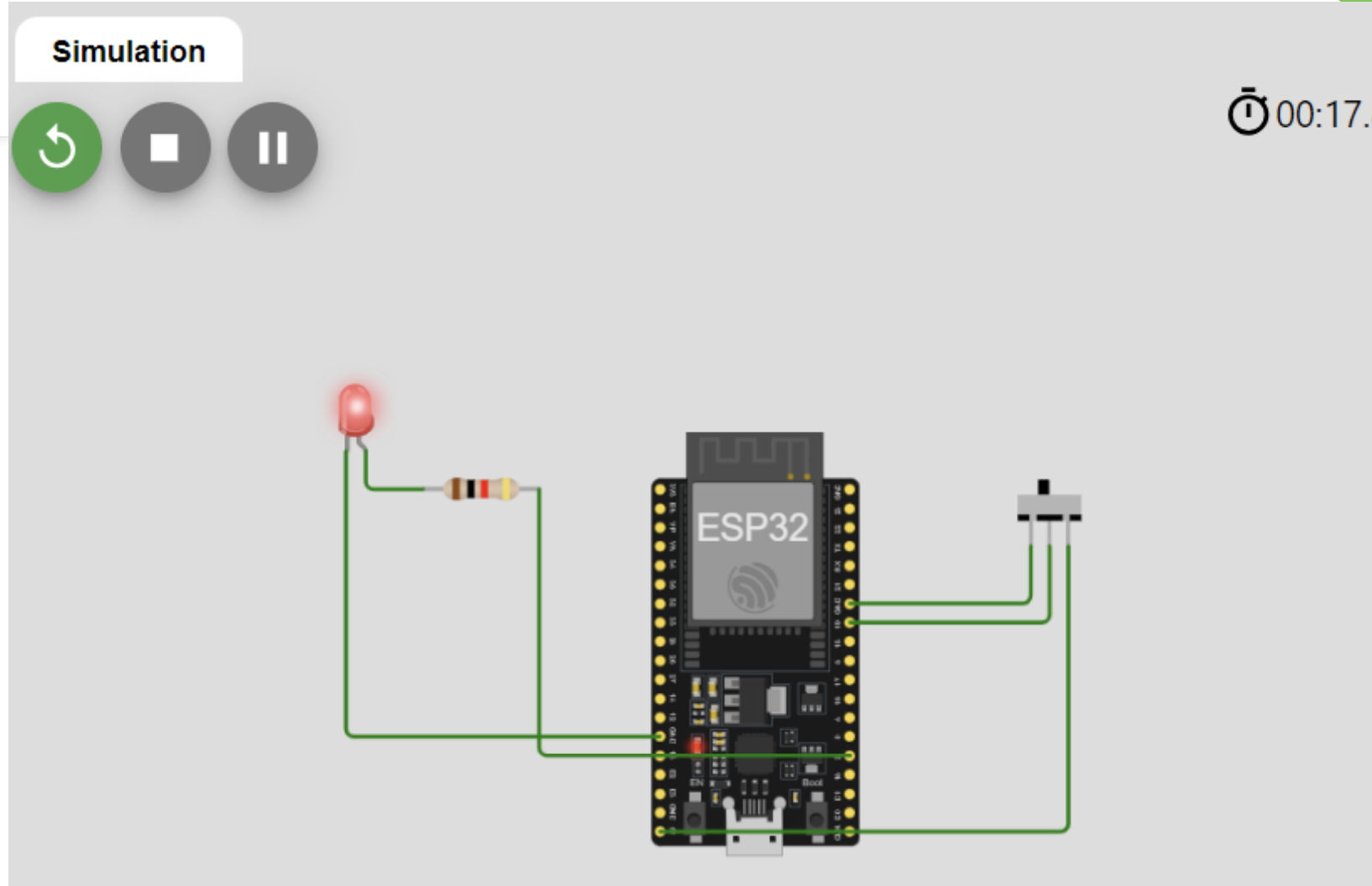
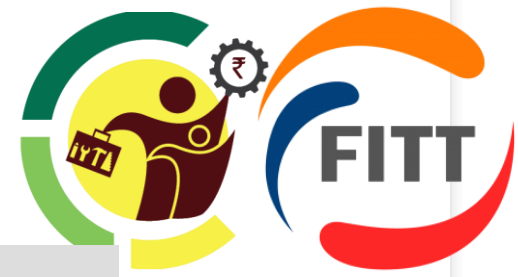
01:15.689 90%

Soil Sensor

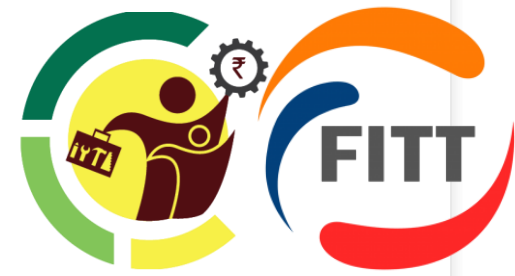
Soil Moisture



Setting Up Arduino for Button operated LED Project



Thank you



Assignment -

1. Install Arduino IDE
2. Visit wokwi.com
3. Run the Blink-LED code on wokwi.com
4. Explore more featured projects on wokwi.com
5. Make a circuit with one button, one LED and one resistor and write a code to operate the LED with the button.