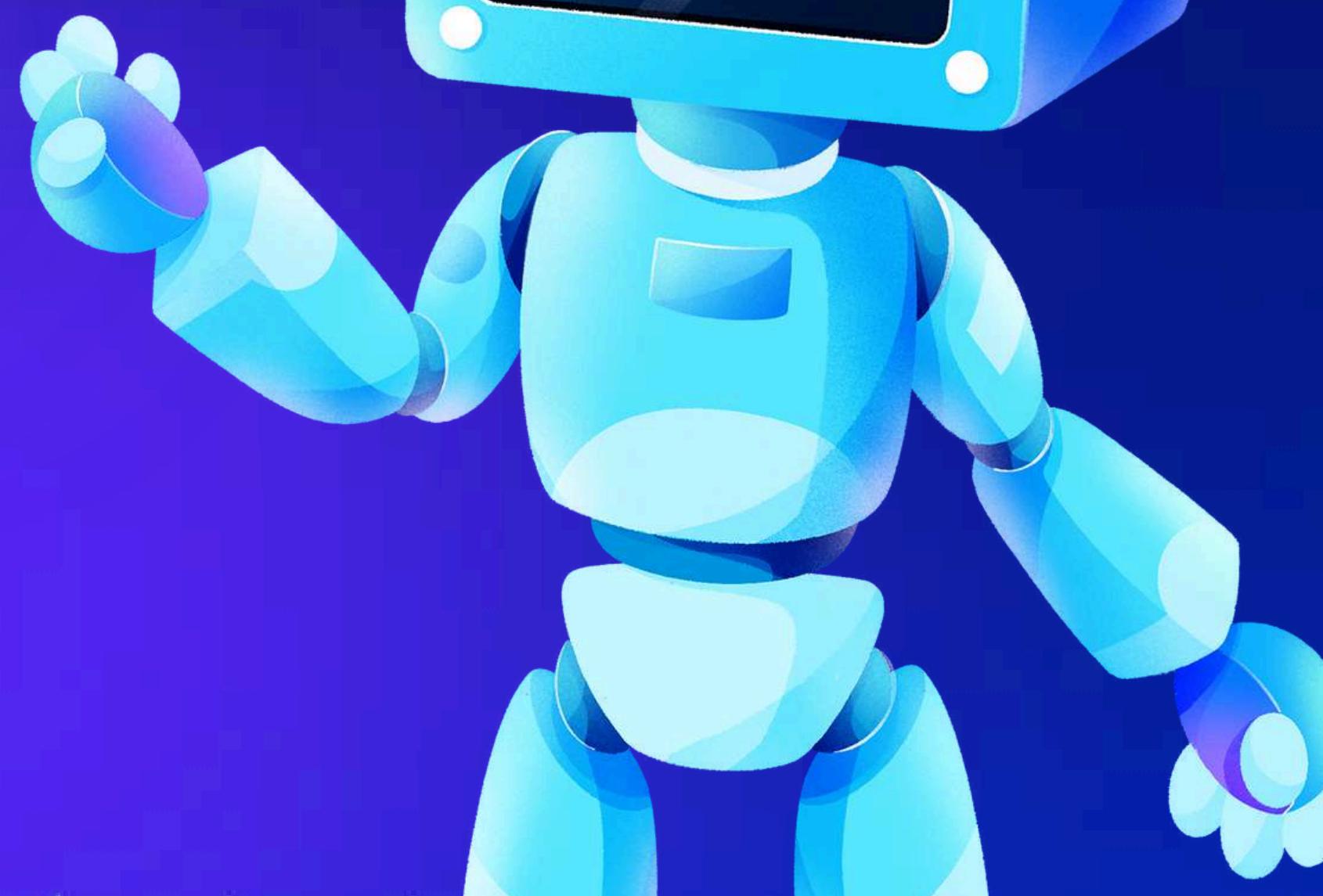




ROBOTICS WORKSHOP

ARDUINO



By Washio Ferdous (Rubai)



INTRODUCTION TO ARDUINO

01

- Why use Arduino?
 - Arduino is an affordable, beginner-friendly platform for versatile electronics projects, supported by open-source tools and a strong community.
-

02

- What is Arduino IDE?
 - A platform to write, compile, and upload code to microcontrollers. Simple interface with extensive community support.

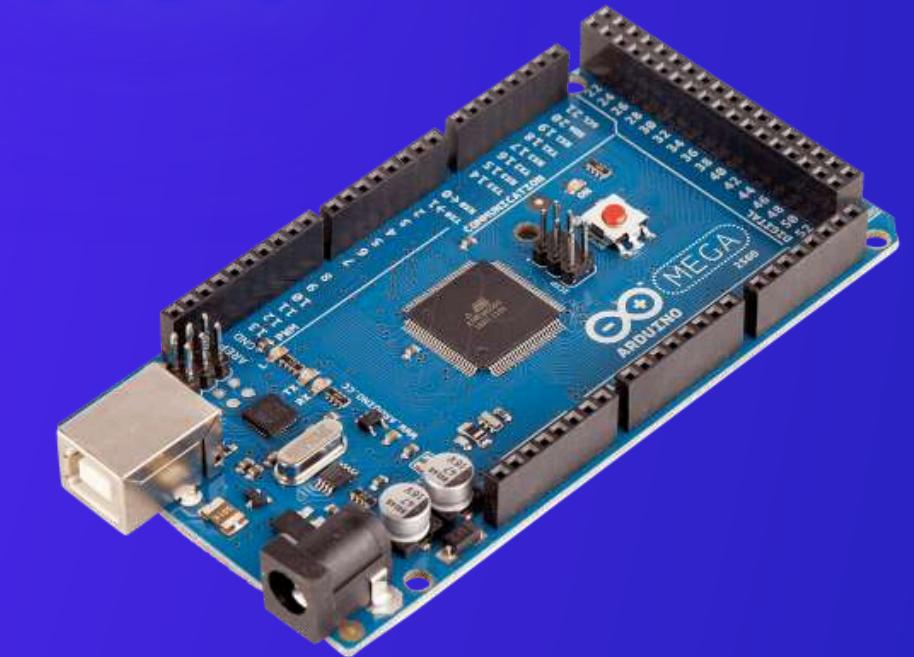
DIFFERENT TYPES OF MCU



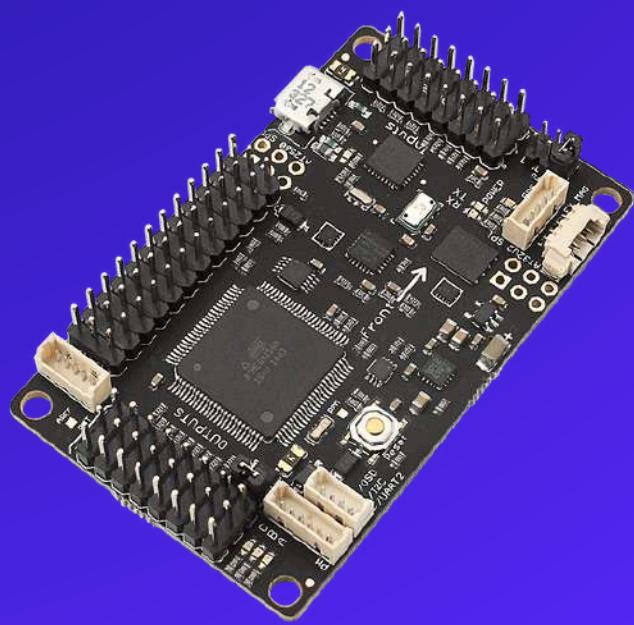
Arduino Uno



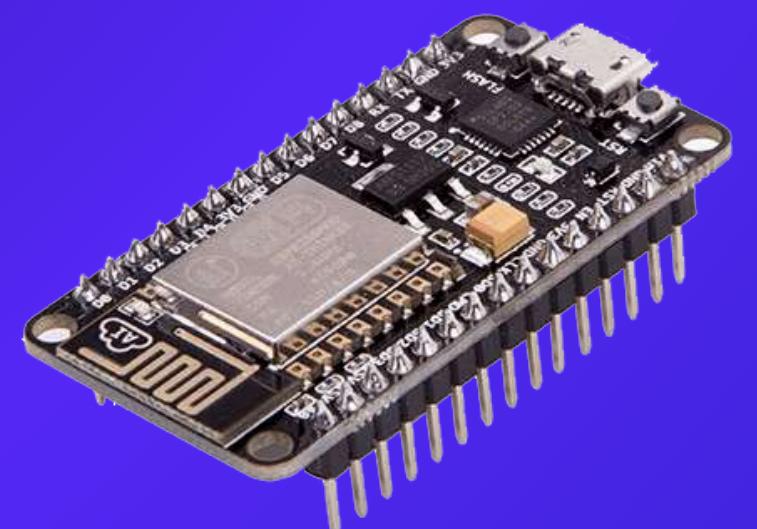
Arduino Nano



Arduino Mega



ESP32

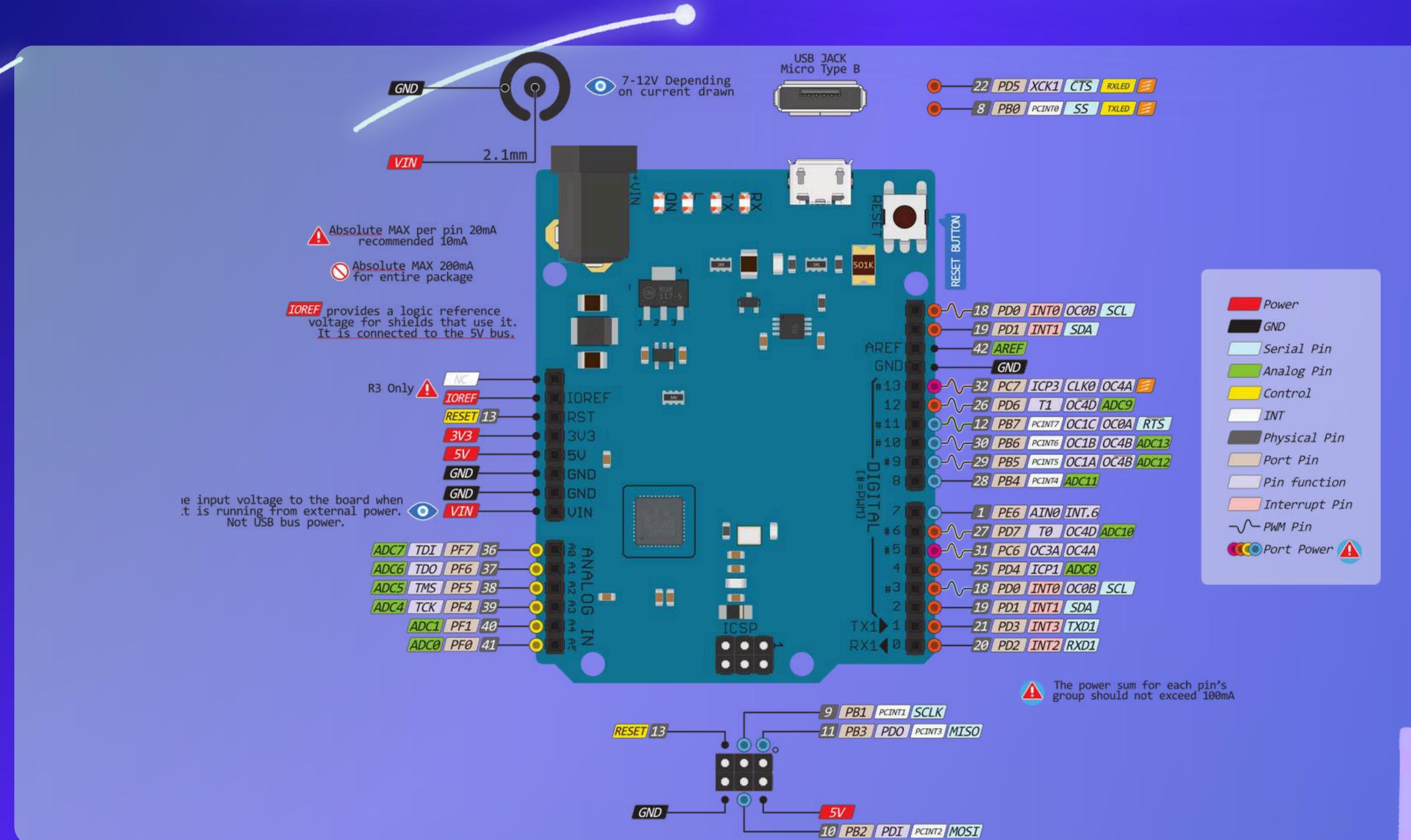


ESP8266



Raspberry Pi

ARDUINO UNO PIN DIAGRAM

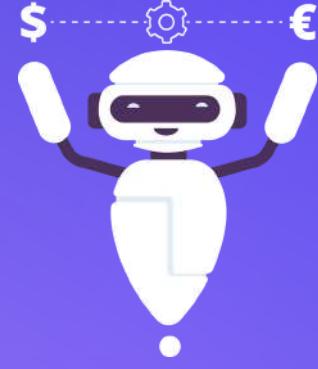


HOW TO GET STARTED



STEP 01

Download and
install the Arduino
IDE



STEP 02

Write the code to
operate the
Arduino UNO as
you desire



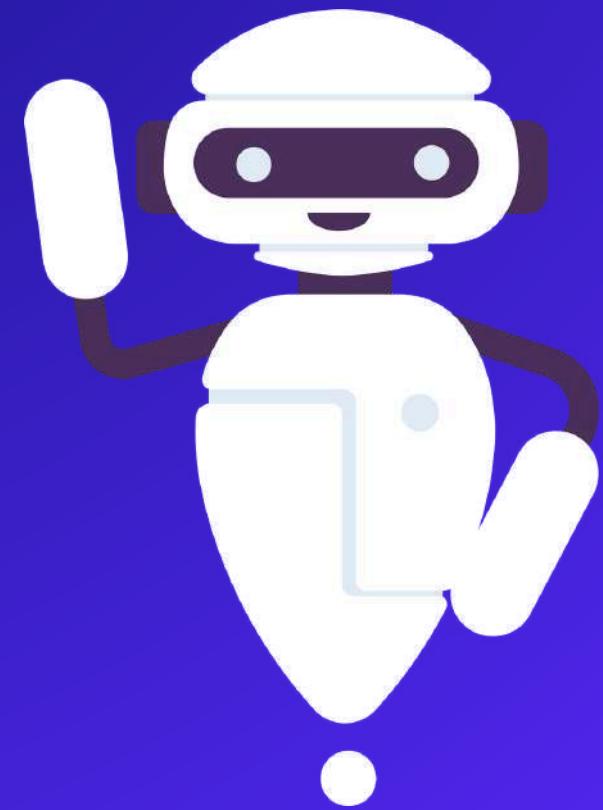
STEP 03

Select the board
and port to verify
and upload the
code to your
Arduino UNO

STEP 01



DOWNLOAD AND INSTALL



Download

- Visit arduino.cc and download for your OS (Windows/Mac/Linux).

Install

- Follow the installation wizard.

And you are ready to go

STEP 02



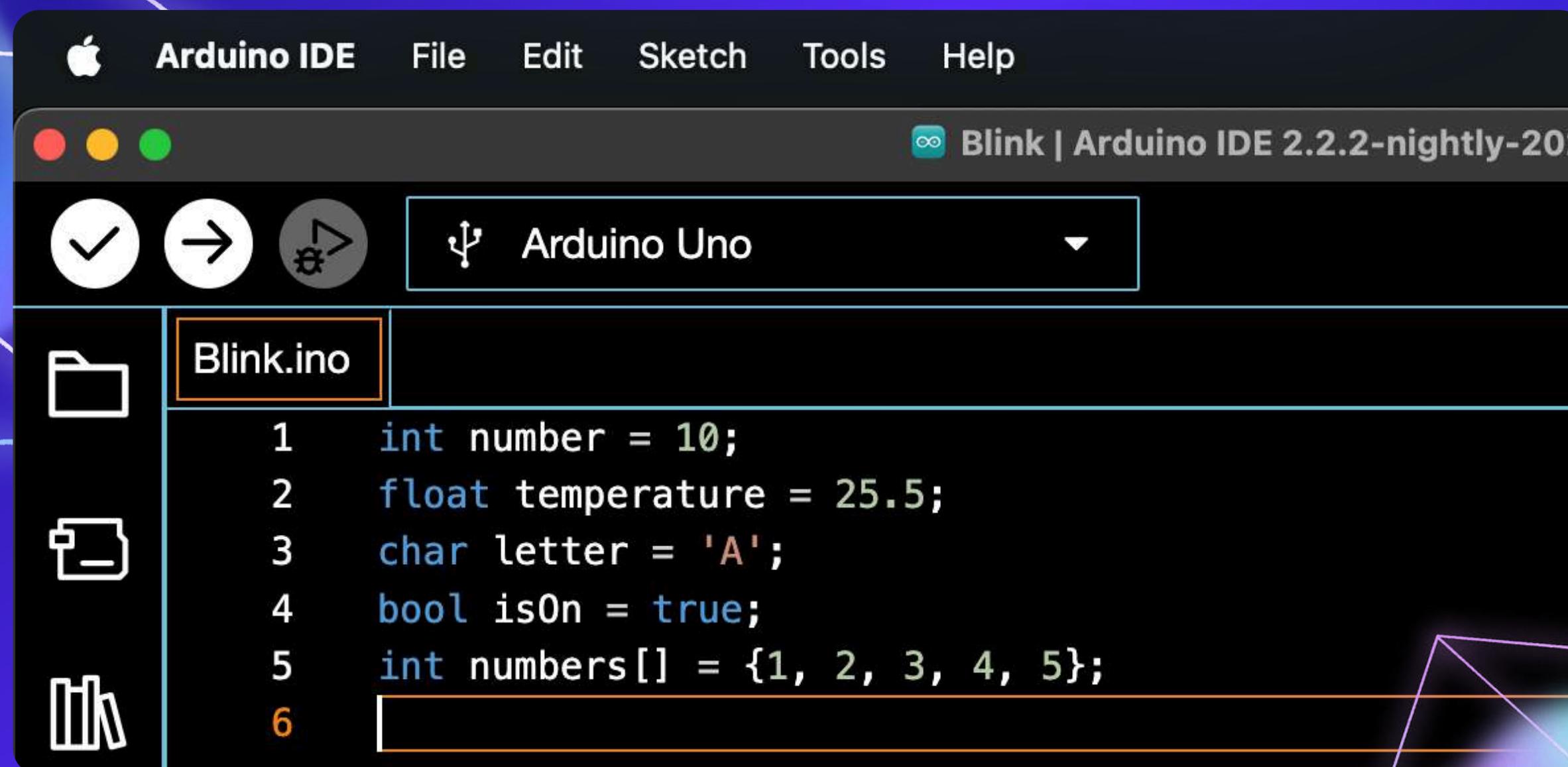


LET'S START CODING

- Data types
- Array
- Operators
- Conditional function
- Loops
- Structure of Arduino code
- Pin mode setups
- Input output methods
- Function



DATA TYPES AND ARRAY

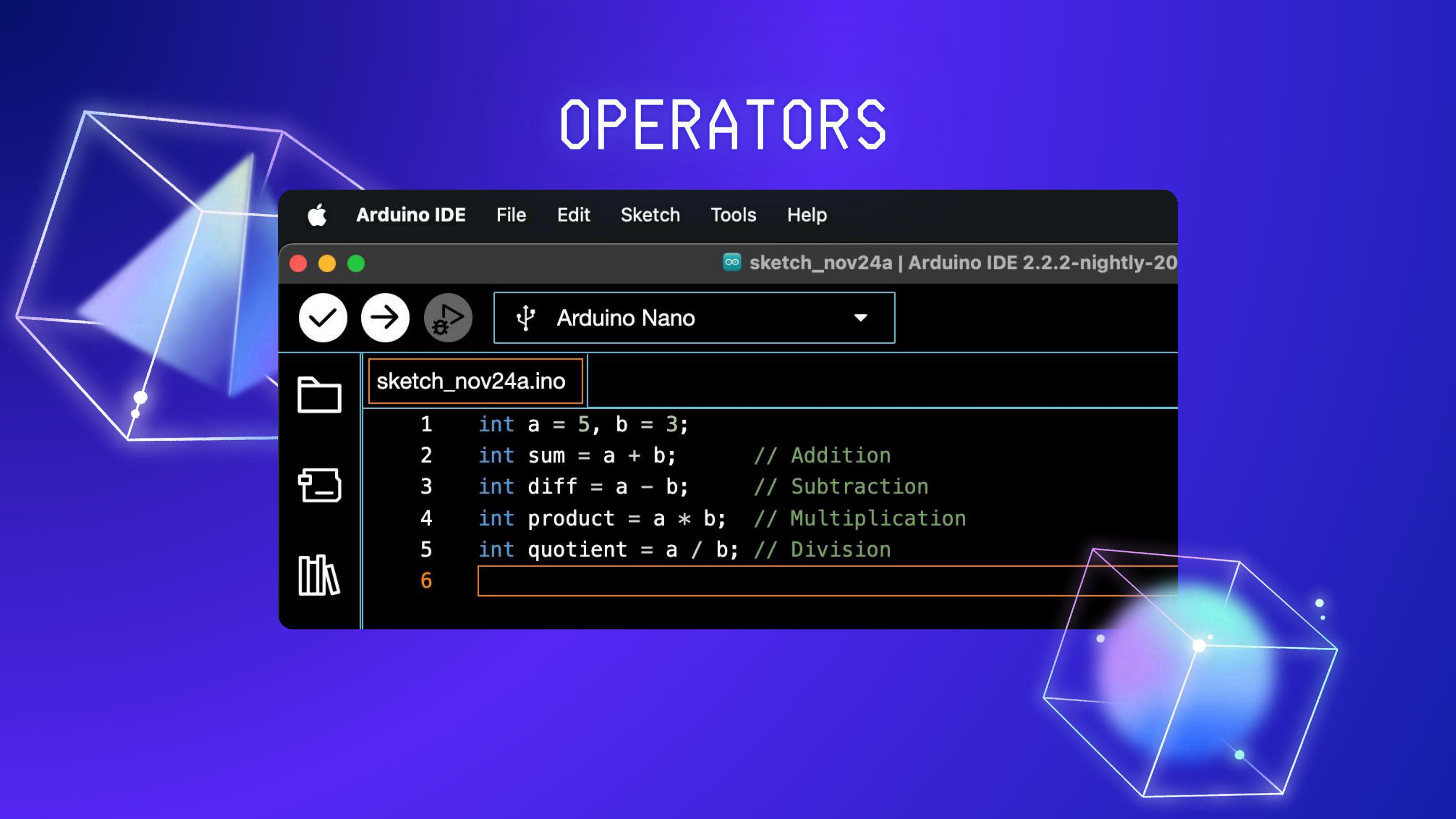


The screenshot shows the Arduino IDE interface with the following details:

- Title Bar:** Arduino IDE, File, Edit, Sketch, Tools, Help.
- Sketch Name:** Blink | Arduino IDE 2.2.2-nightly-202
- Board Selection:** Arduino Uno
- Code Area:** Displays the following C++ code:

```
1 int number = 10;
2 float temperature = 25.5;
3 char letter = 'A';
4 bool isOn = true;
5 int numbers[] = {1, 2, 3, 4, 5};
6 
```
- Sidebar:** Shows icons for File (Folder), Sketch (File), and Preferences (Books).

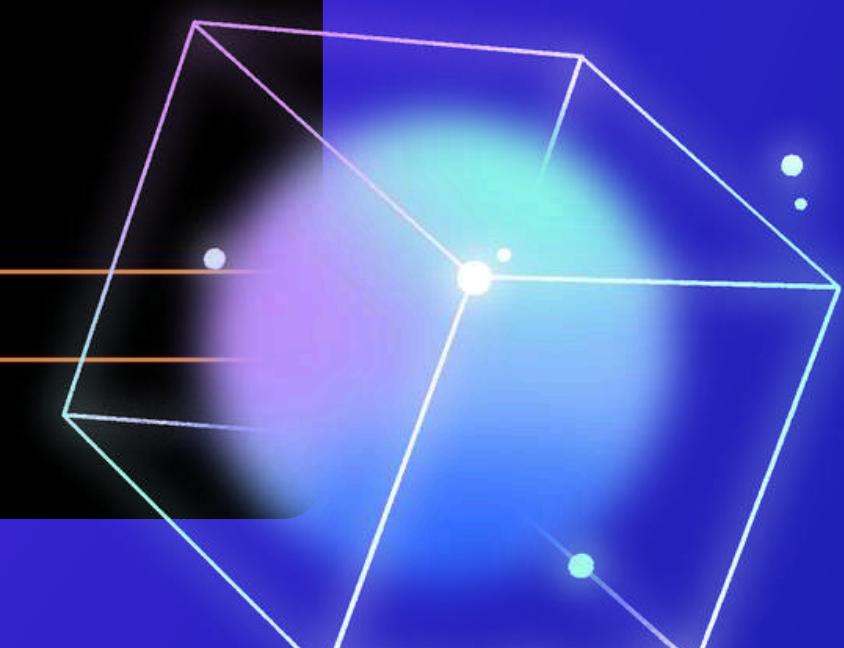
OPERATORS



The image shows the Arduino IDE interface running on a Mac OS X system. The title bar reads "Arduino IDE" and "sketch_nov24a | Arduino IDE 2.2.2-nightly-20". The main area displays a sketch named "sketch_nov24a.ino" containing the following code:

```
1 int a = 5, b = 3;
2 int sum = a + b;      // Addition
3 int diff = a - b;    // Subtraction
4 int product = a * b; // Multiplication
5 int quotient = a / b; // Division
6 
```

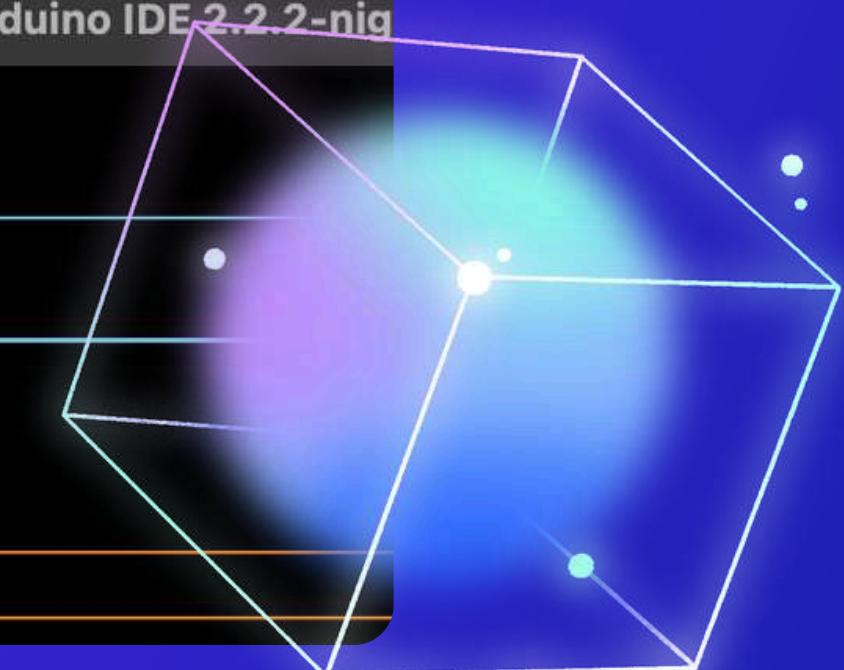
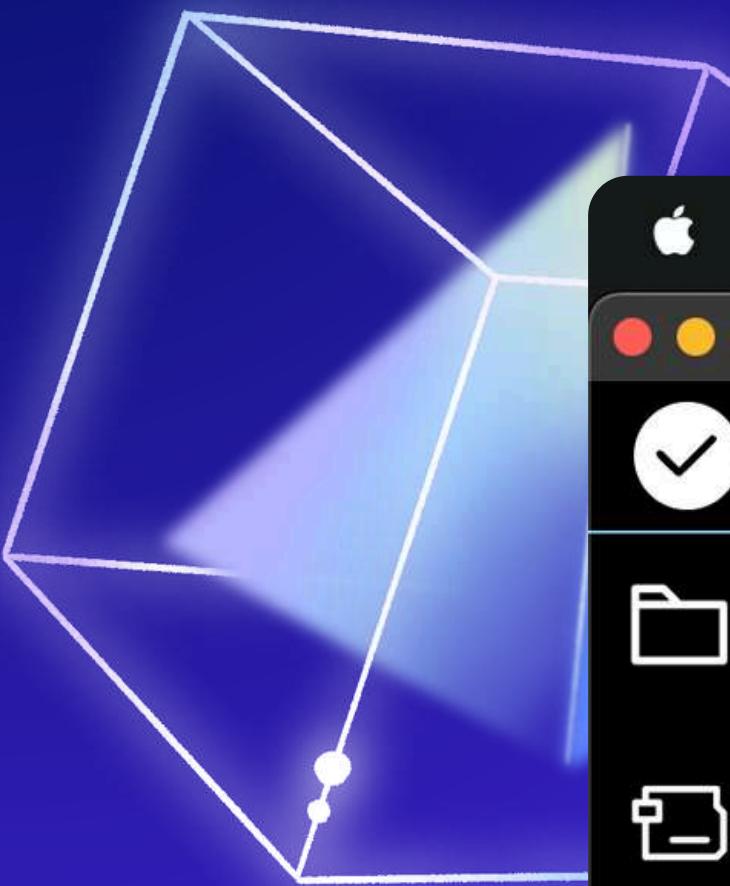
CONDITIONAL FUNCTION



An Arduino IDE window showing a sketch named "sketch_nov24a.ino". The code contains an if-else conditional function:

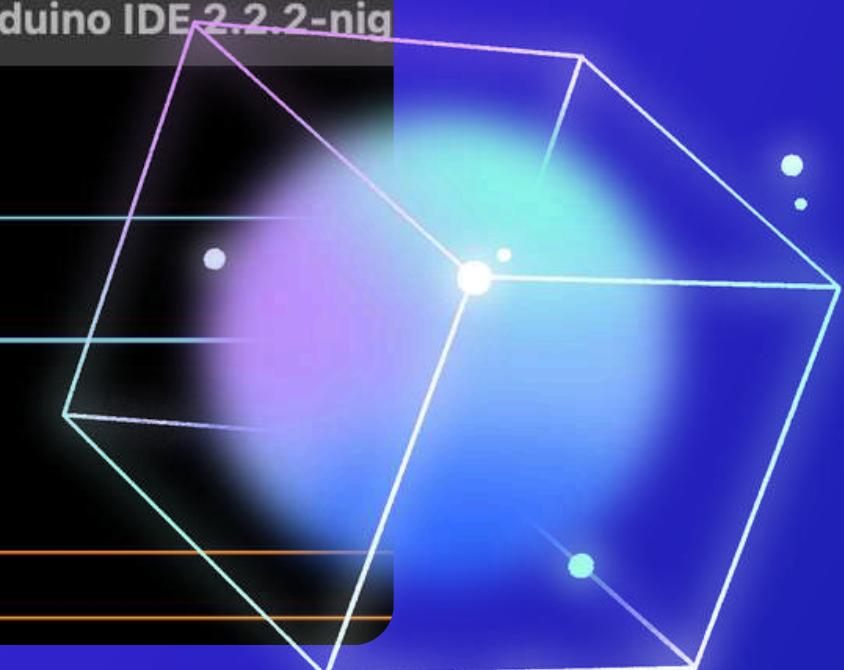
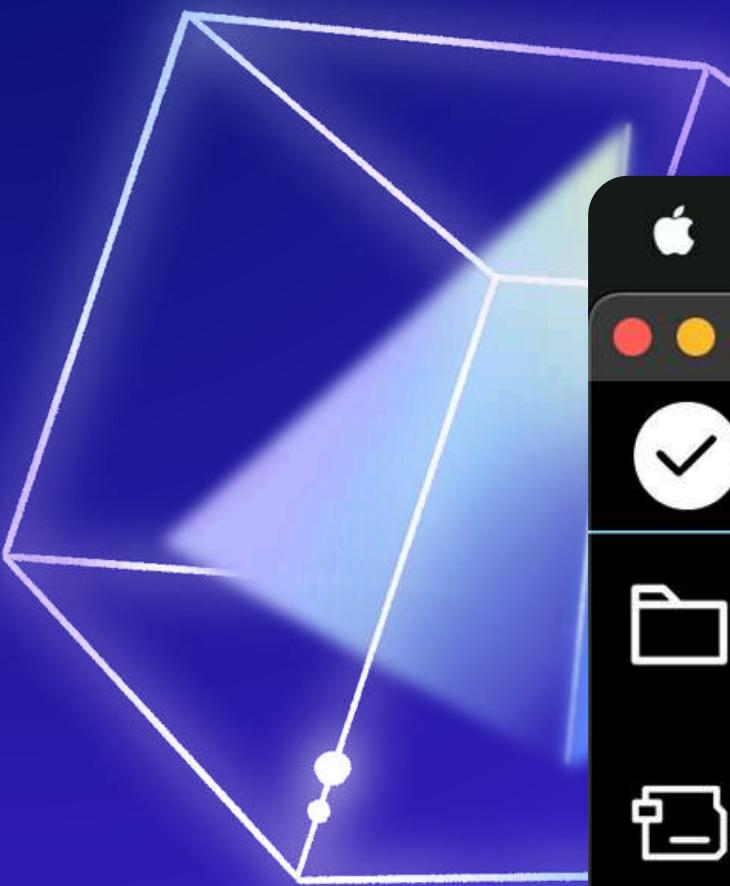
```
1  if (a > b) {  
2      // Do something  
3  } else if (a == b) {  
4      // Do something else  
5  } else {  
6      // Default action  
7  }  
8  
```

FOR AND WHILE LOOP



An Arduino IDE window showing a sketch named "sketch_nov24a.ino". The code uses a while loop:

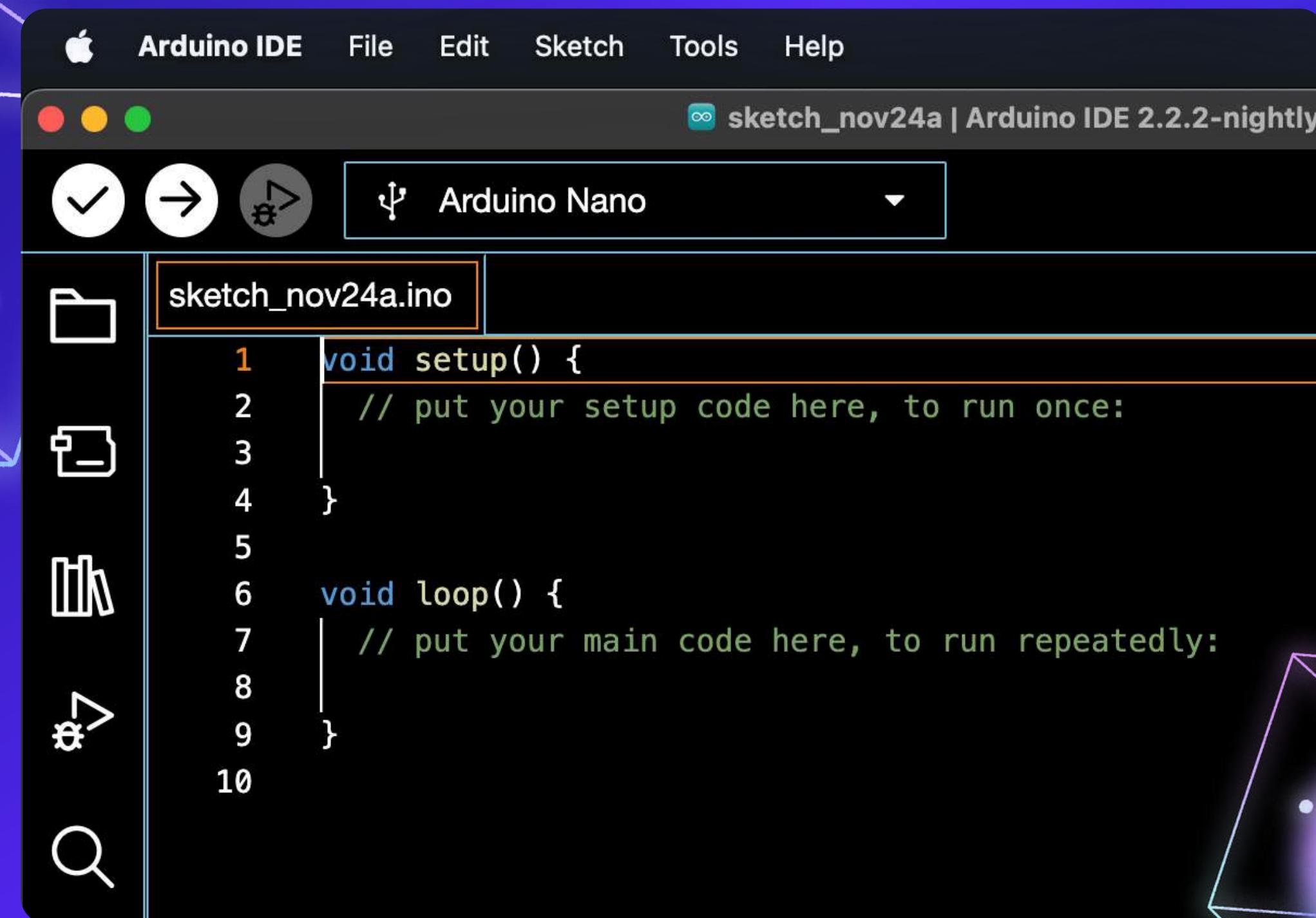
```
1 int count = 0;
2 while (count < 5) {
3     // loop action
4 }
```



An Arduino IDE window showing a sketch named "sketch_nov24a.ino". The code uses a for loop:

```
1 for (int i = 0; i < 5; i++) {
2     // loop action
3 }
4 
```

STRUCTURE OF ARDUINO CODE

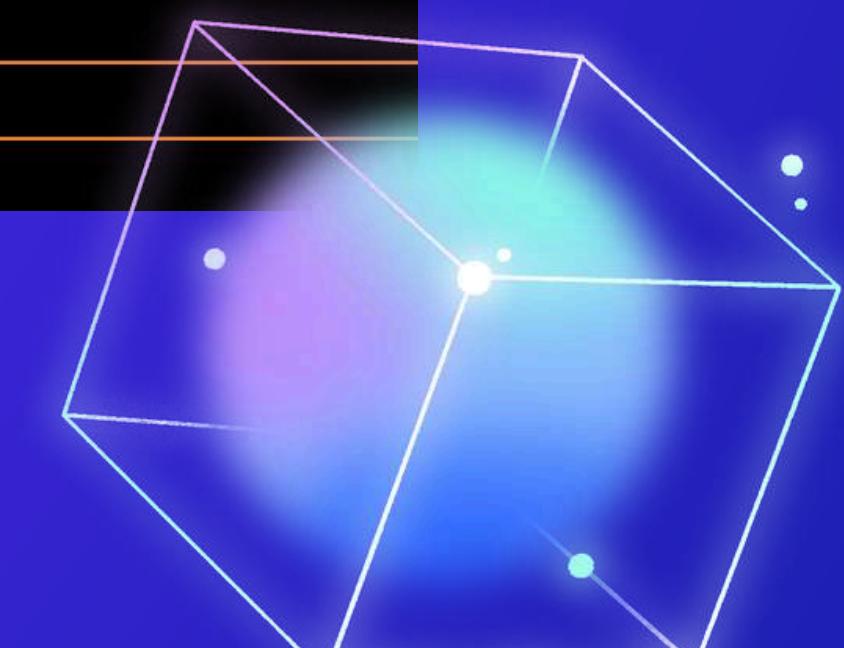
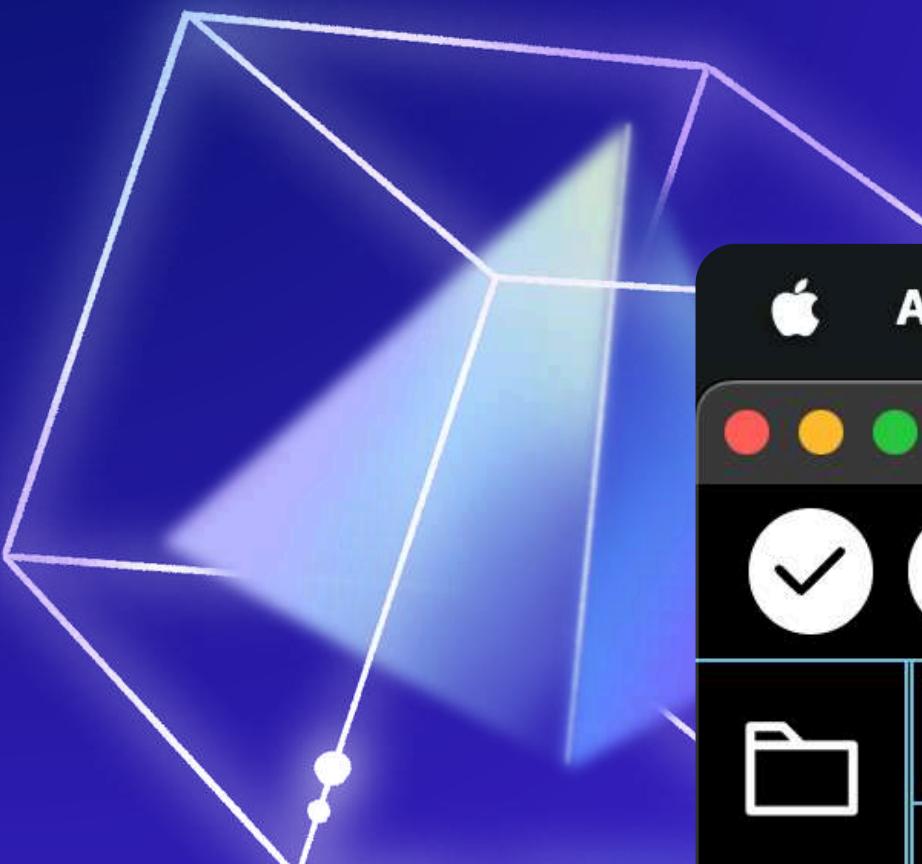


The image shows a screenshot of the Arduino IDE interface. The title bar reads "Arduino IDE" and "sketch_nov24a | Arduino IDE 2.2.2-nightly". The main window displays the code for "sketch_nov24a.ino". The code is structured as follows:

```
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 code editor has a dark background with light-colored text. The file browser on the left shows a single folder icon labeled "sketch_nov24a.ino". The toolbar at the top includes icons for upload, verify, and other functions.

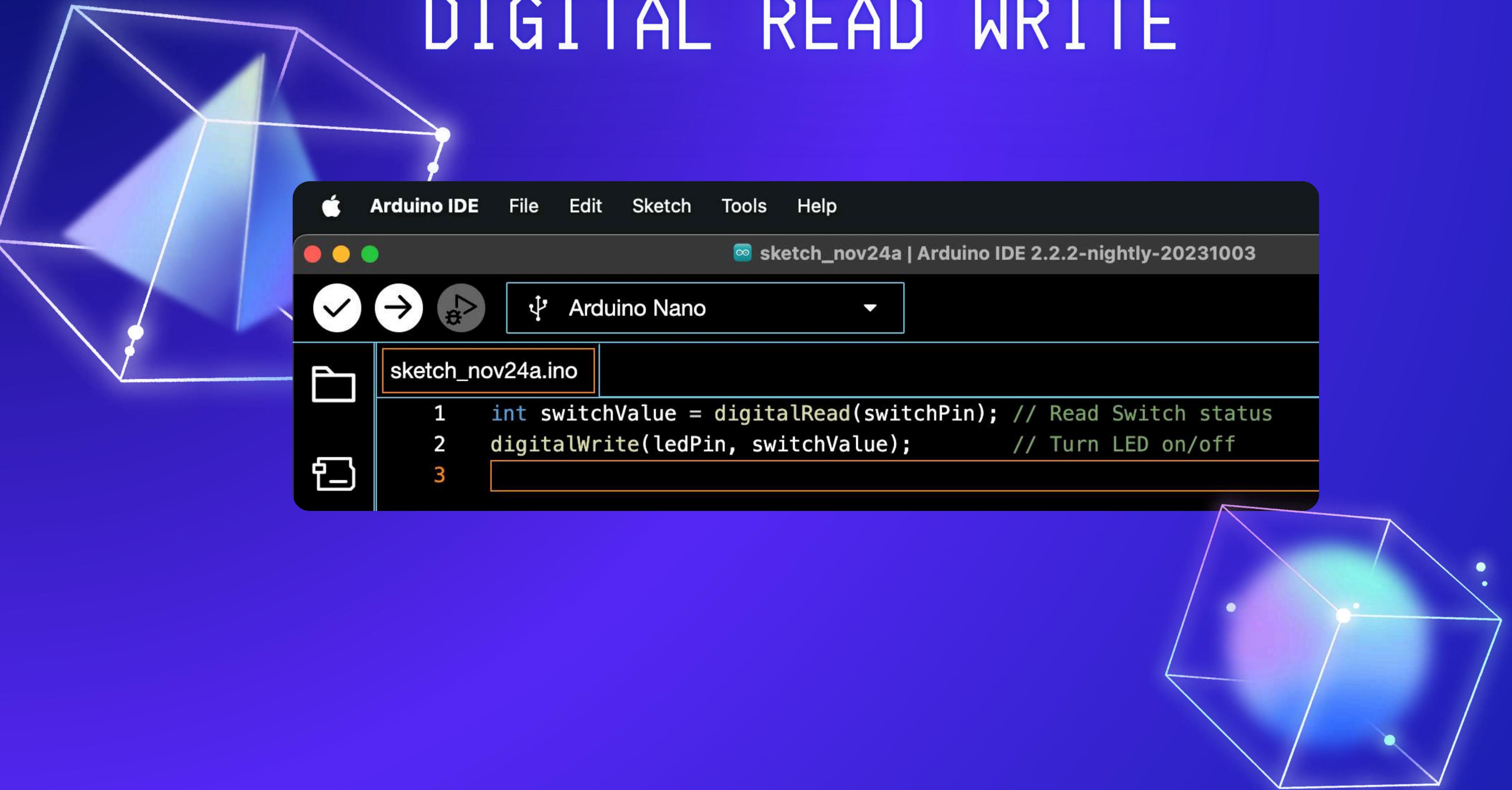
PIN MODE SETUP



An image of the Arduino IDE interface on a Mac OS X system. The window title is "sketch_nov24a | Arduino IDE 2.2.2-nightly-2023". The toolbar includes icons for power, upload, and serial monitor, with "Arduino Nano" selected as the port. The left sidebar shows a folder icon and the sketch name "sketch_nov24a.ino". The main code editor contains the following code:

```
1 const int ledPin = 13; // Pin 13 for LED
2
3 void setup() {
4     pinMode(ledPin, OUTPUT); // Set pin as output
5 }
6
```

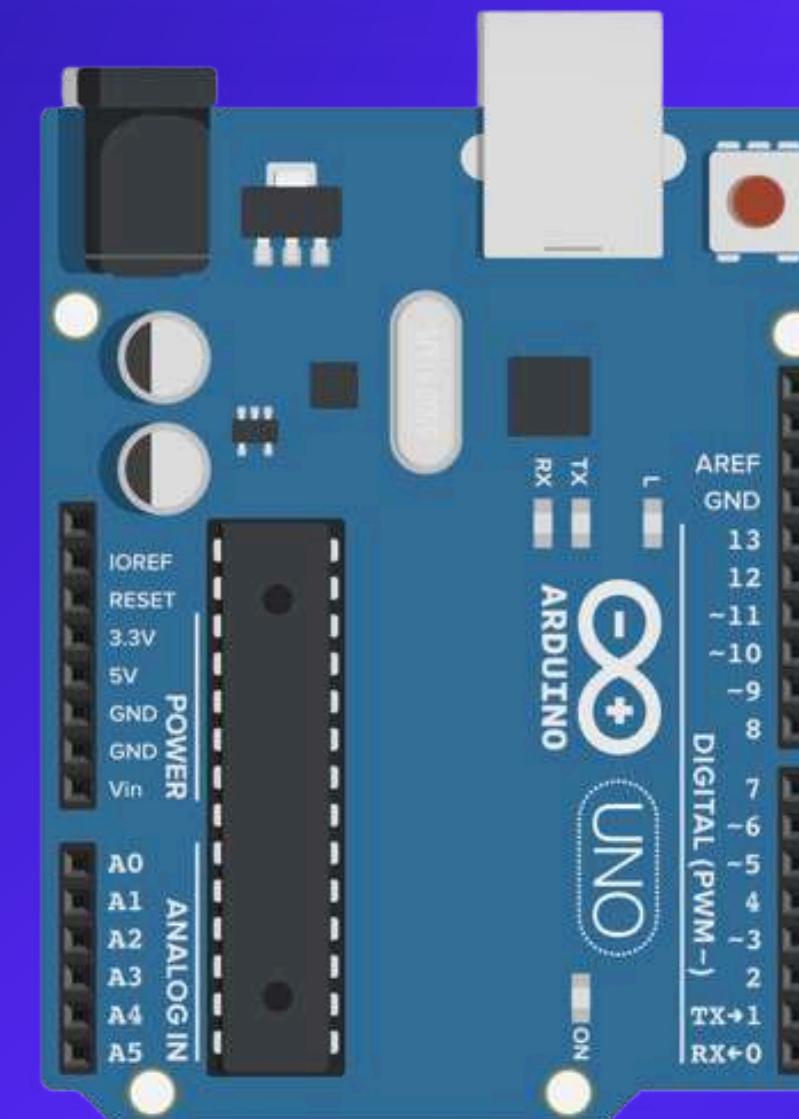
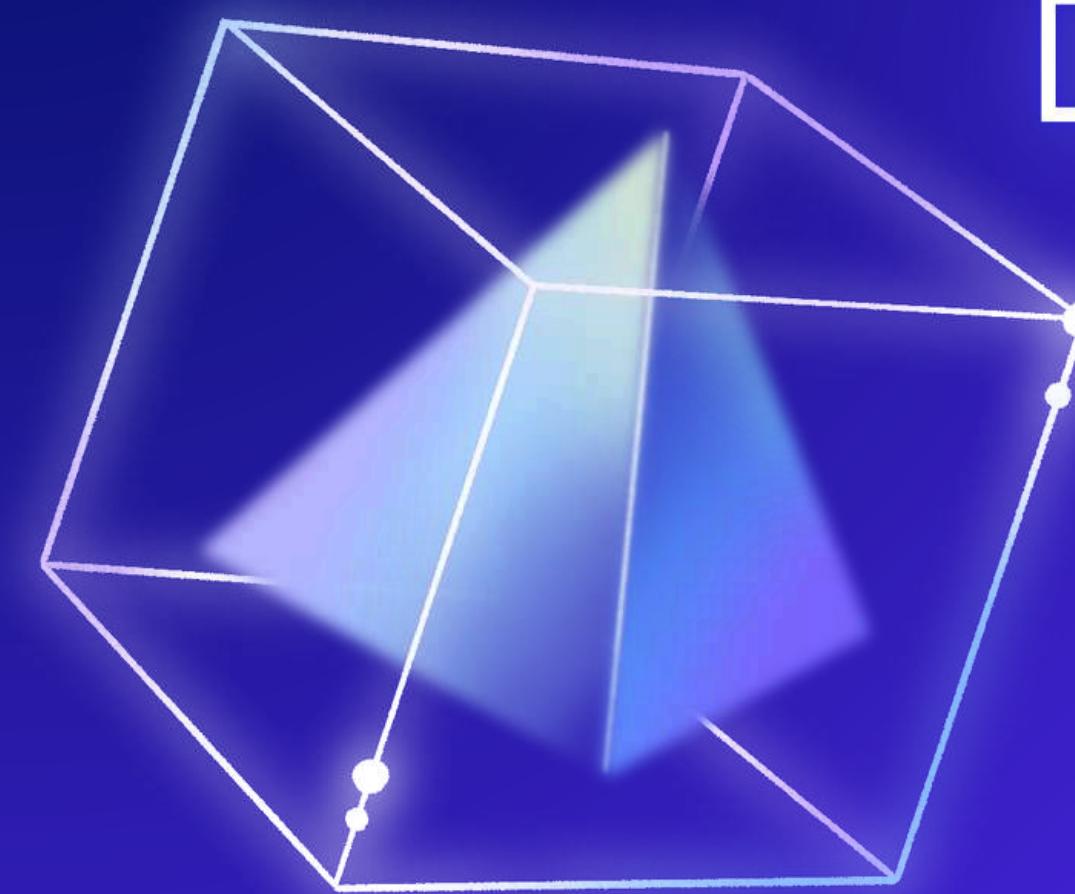
DIGITAL READ WRITE



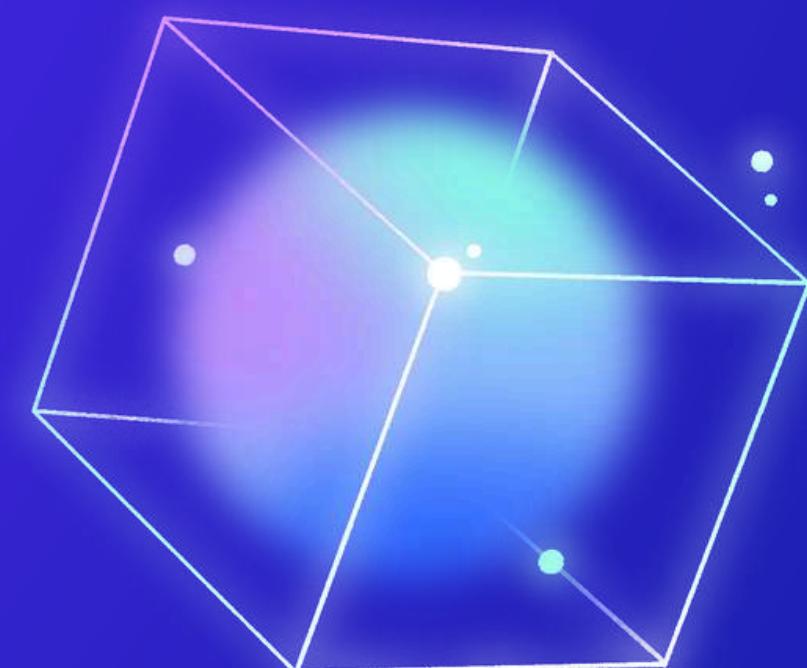
The image shows the Arduino IDE interface on a Mac OS X system. The title bar reads "Arduino IDE" and the active tab is "sketch_nov24a | Arduino IDE 2.2.2-nightly-20231003". The central code editor window displays the following sketch:

```
1 int switchValue = digitalRead(switchPin); // Read Switch status
2 digitalWrite(ledPin, switchValue);           // Turn LED on/off
3 
```

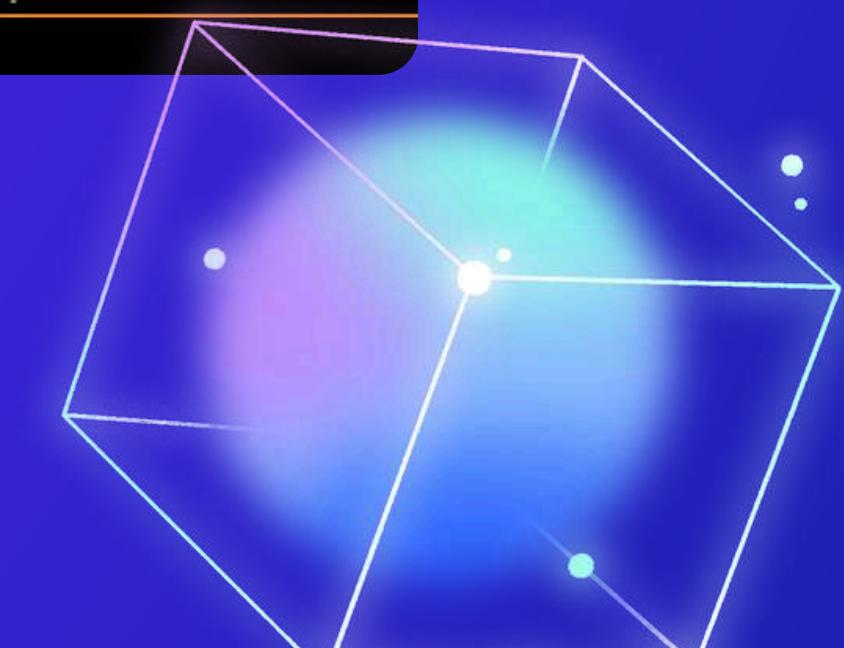
DIGITAL READ WRITE



Only digital pins can be used
to input output digital values



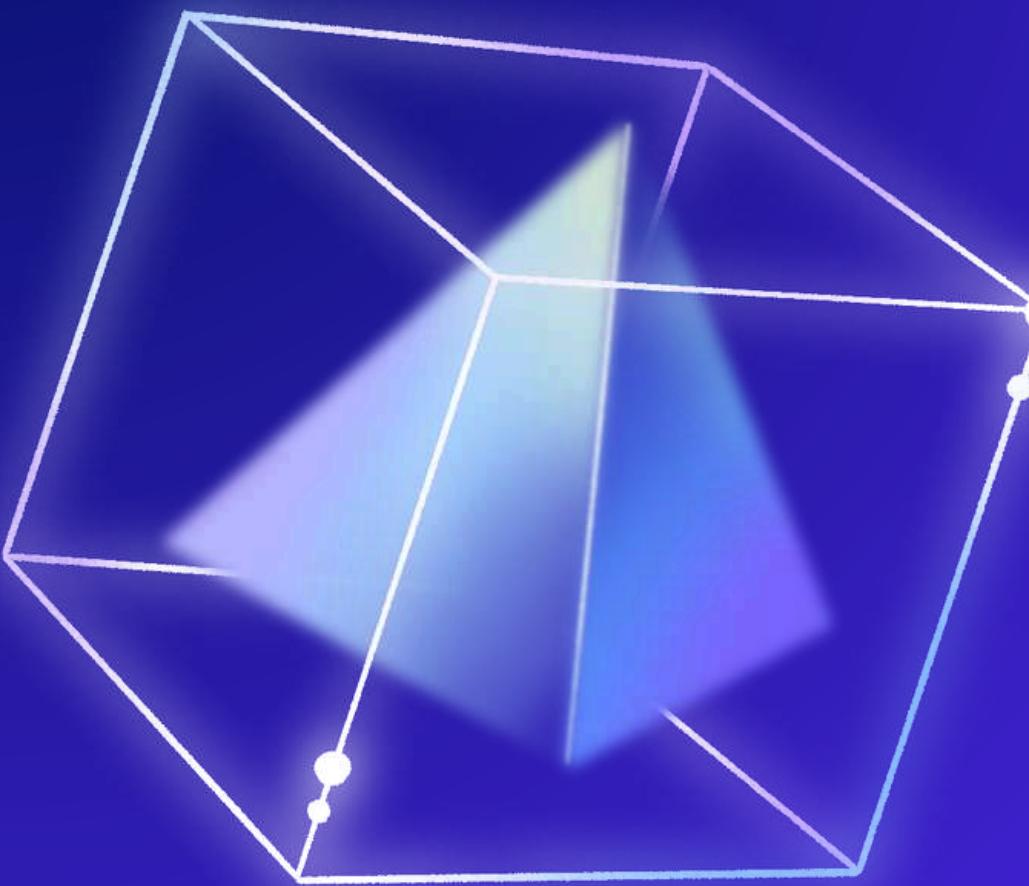
ANALOG READ



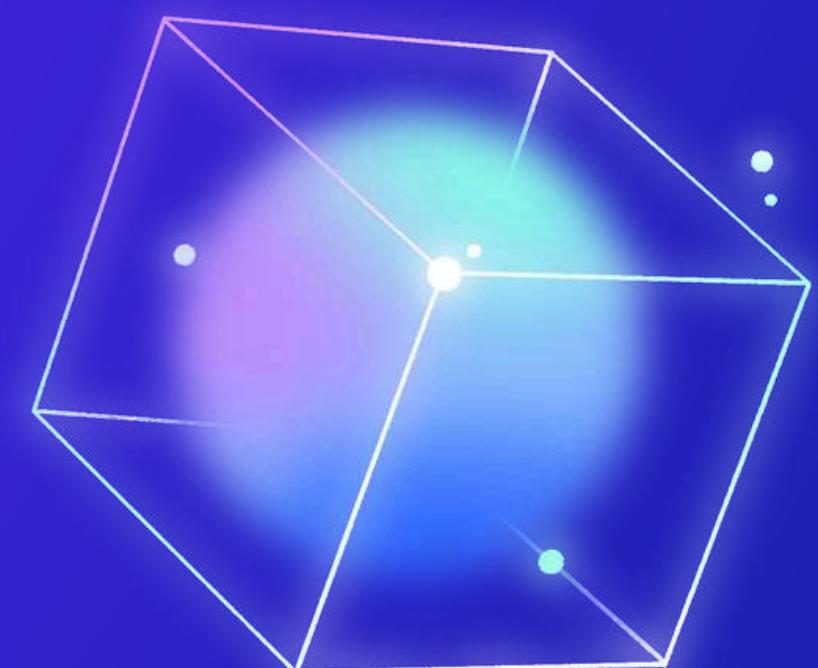
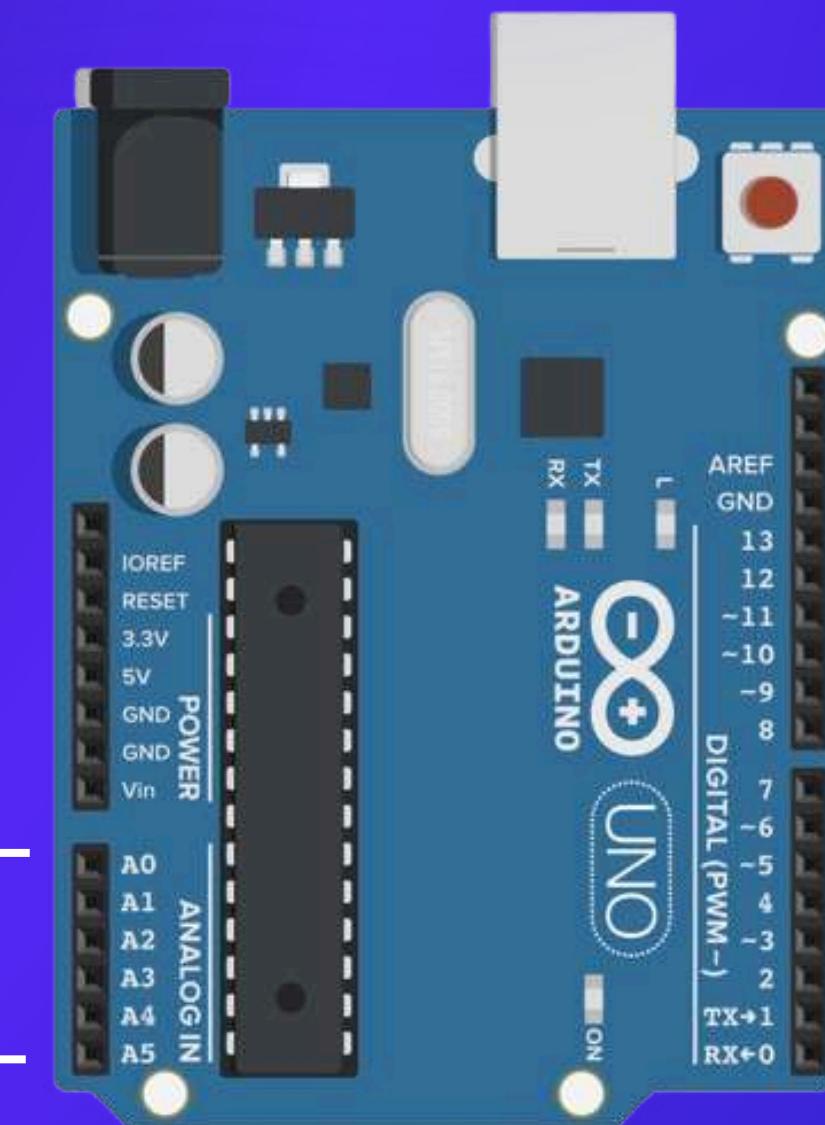
An Arduino IDE window showing the sketch `sketch_nov24a.ino`. The code reads an analog value from pin A0.

```
1 int sensorValue = analogRead(A0); // Read from pin A0
```

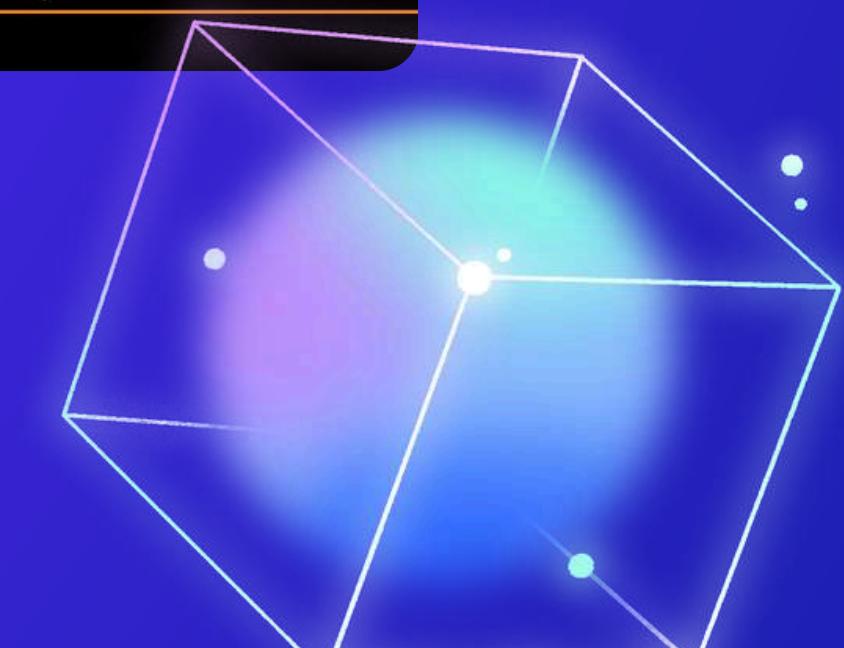
ANALOG READ



Only analog pins can be used
to input analog readings



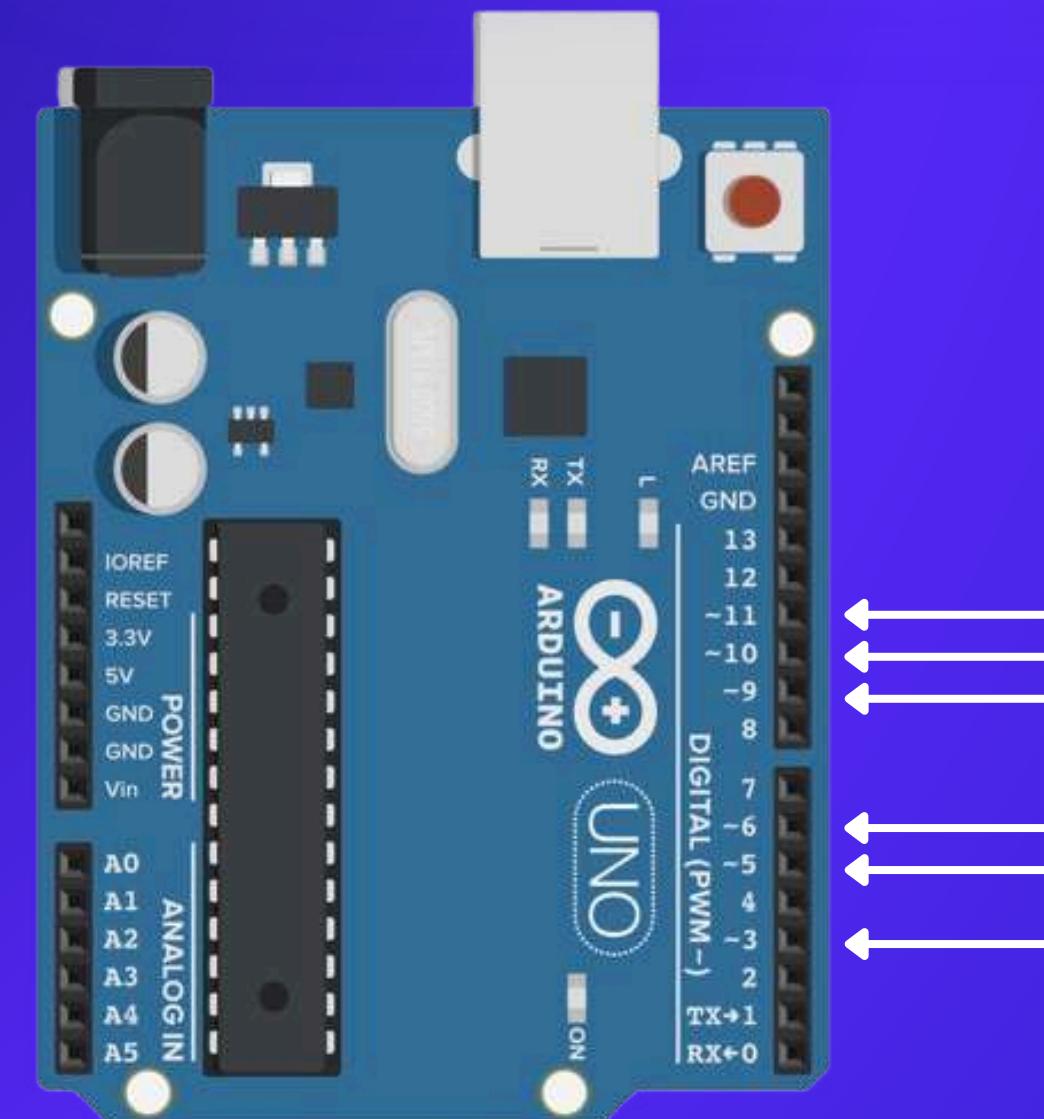
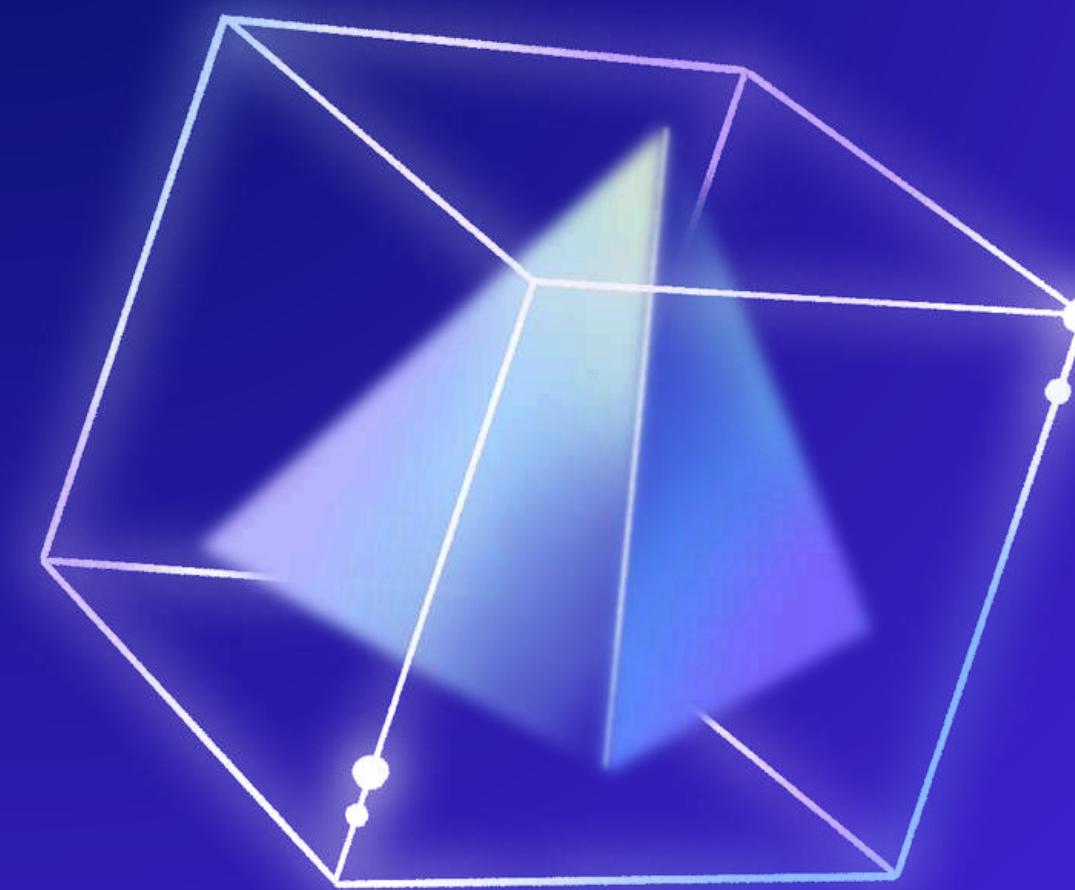
ANALOG WRITE



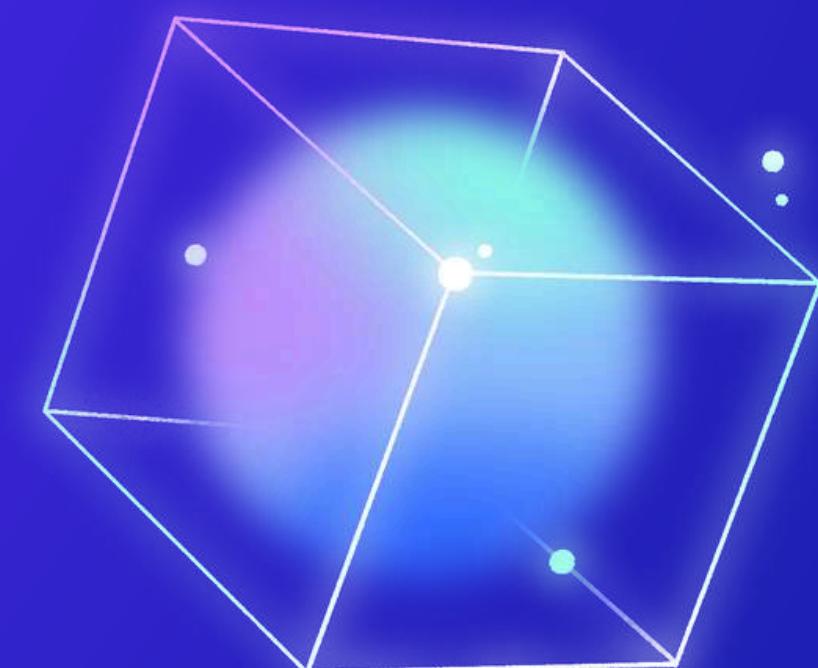
An Arduino IDE window showing the sketch `sketch_nov24a.ino`. The code contains the following lines:

```
1  analogWrite(ledPin, 128); // Set pin to 50% brightness
2 
```

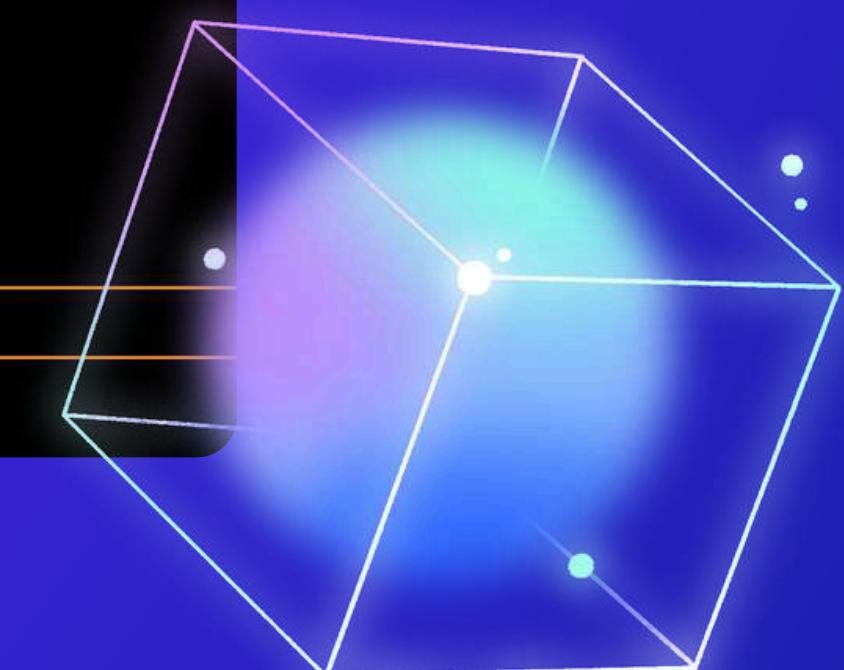
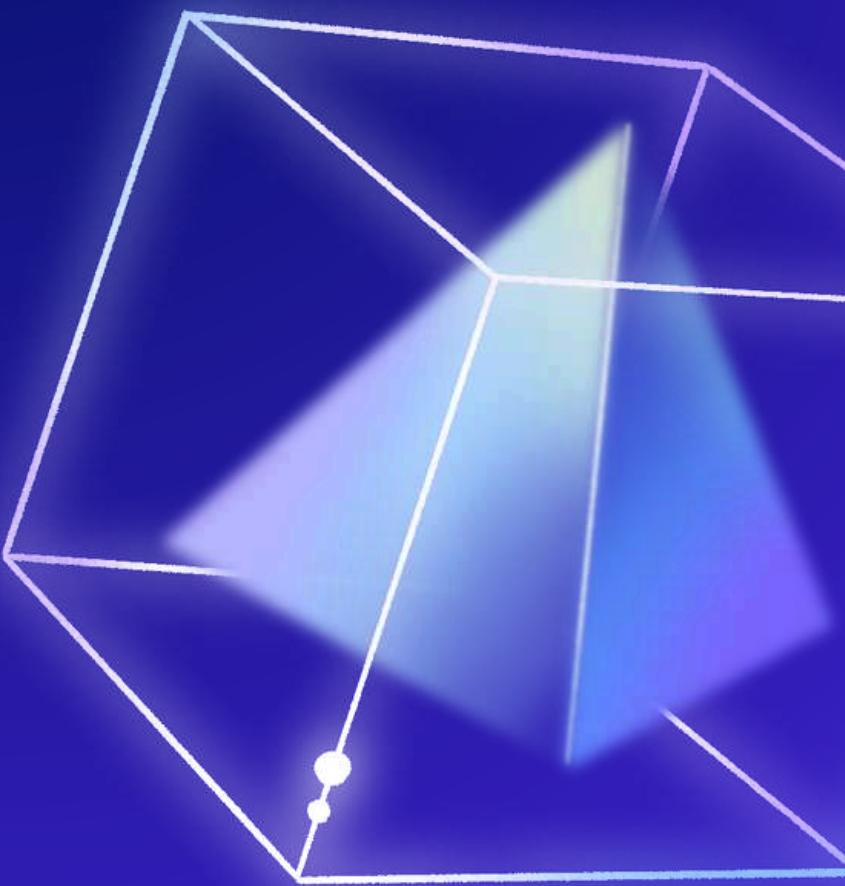
ANALOG WRITE



Only PWM pins can be used
to output analog values



CUSTOM FUNCTION



An Arduino IDE window showing a sketch named "sketch_nov24a.ino". The code defines a custom function `blinkLED()` which alternates the state of a digital pin (ledPin) between HIGH and LOW every second. This function is then called within the `loop()` function.

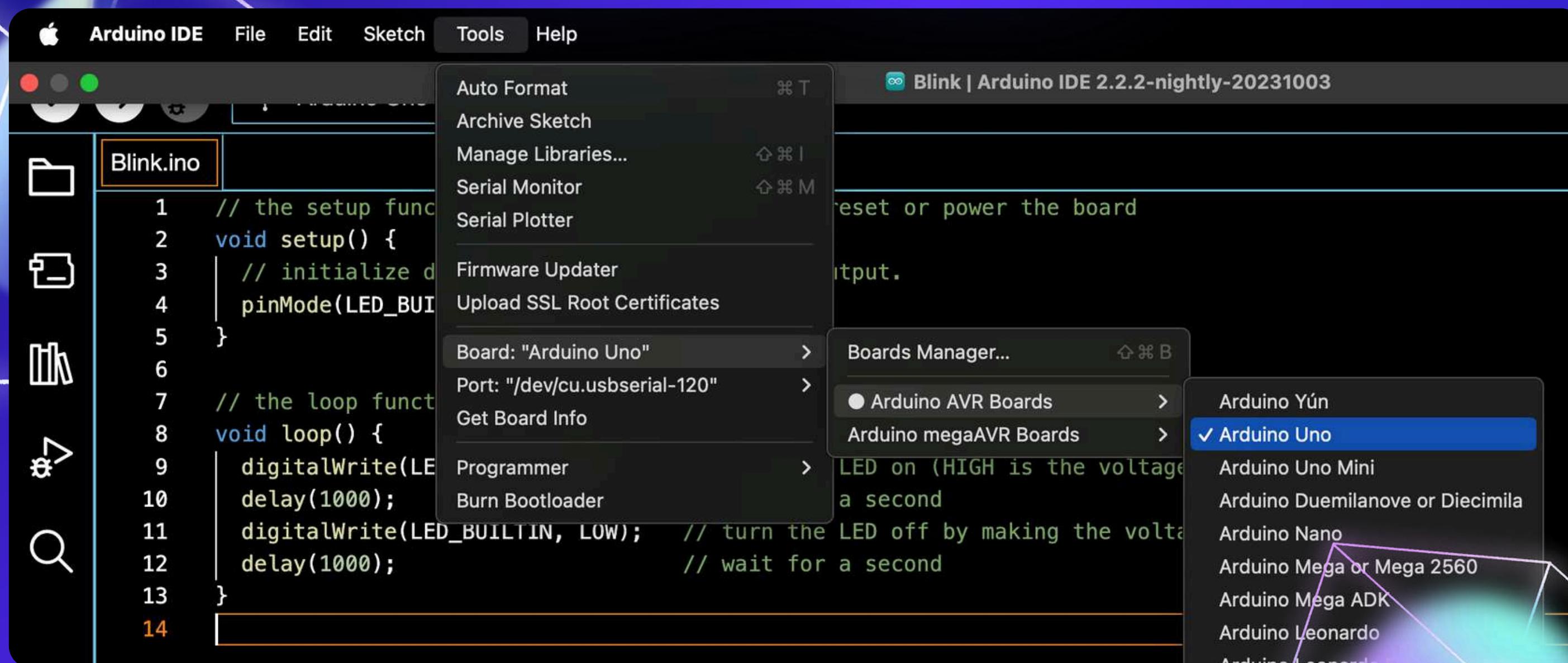
```
void blinkLED() {
    digitalWrite(ledPin, HIGH);
    delay(1000); // Wait for 1 second
    digitalWrite(ledPin, LOW);
    delay(1000); // Wait for 1 second
}

void loop() {
    blinkLED(); // Call function
}
```

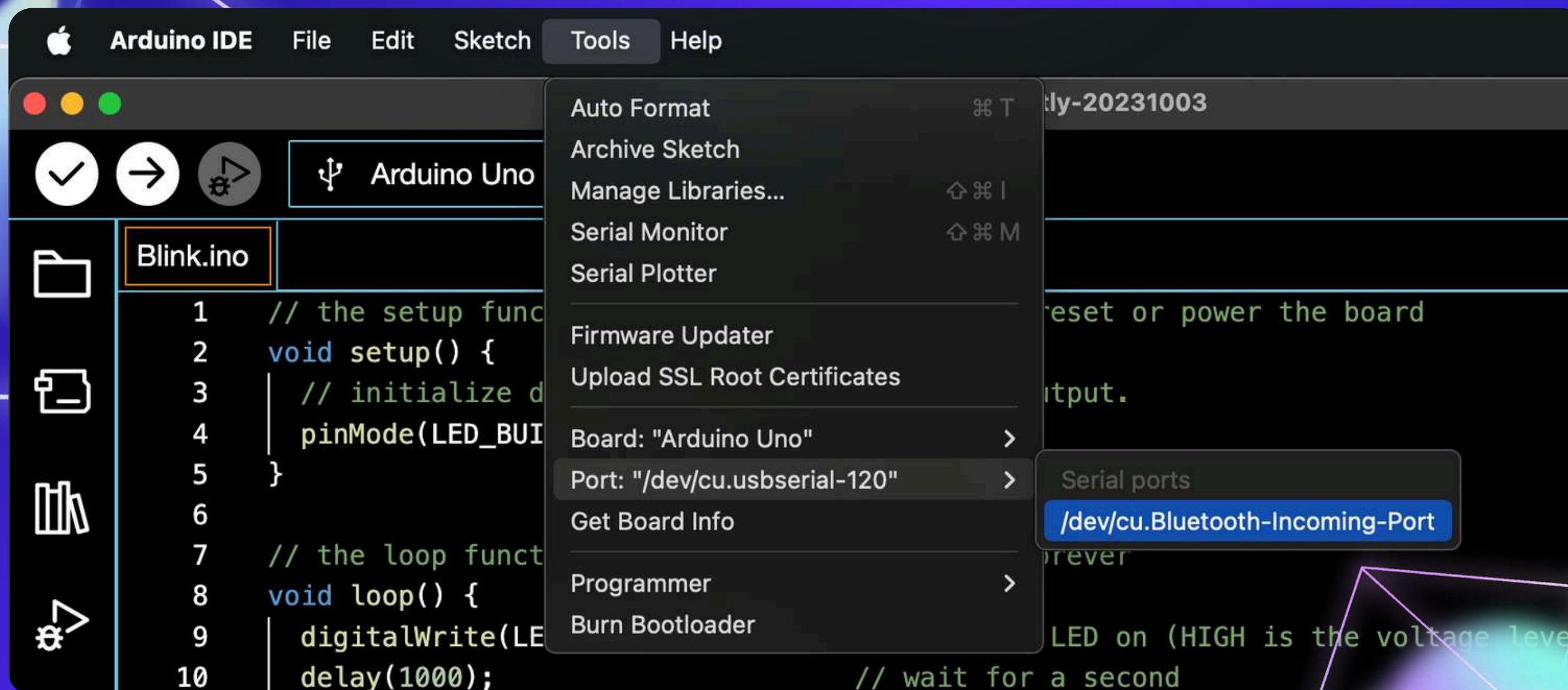
STEP 03



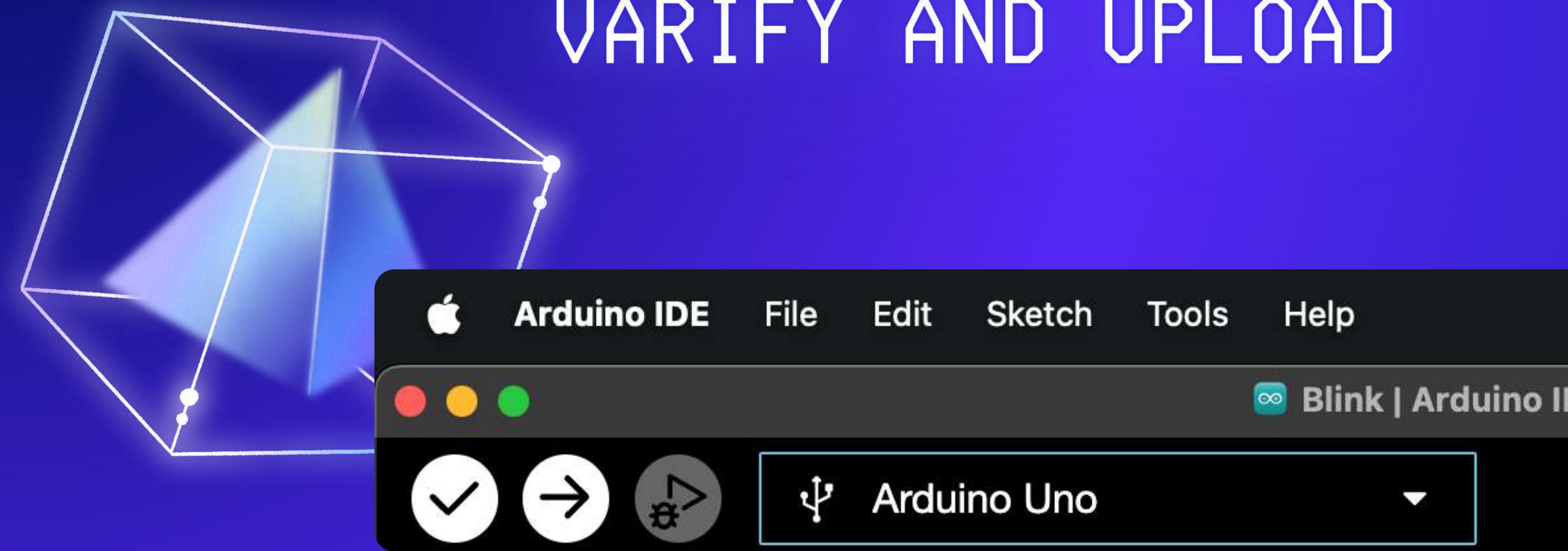
SELECT BOARD



SELECT PORT

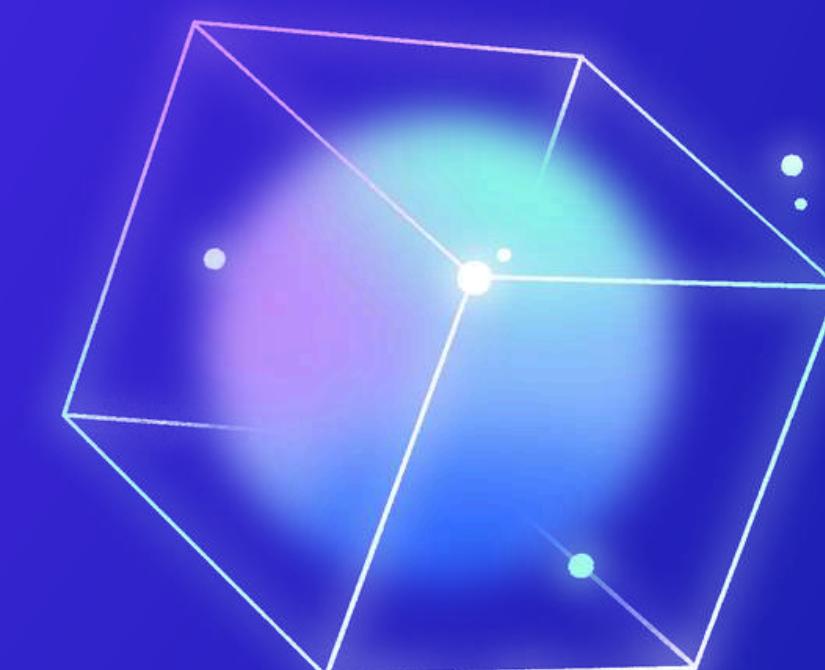


VERIFY AND UPLOAD



Verify Upload

Selected board appears here



NOW LET'S
START MAKING
ROBOTS



THANK YOU

