ARDUINO

PRESENTATION BY

ARJUN SHASTRY – 1DA21CS025


DIGITAL (PWM~)
UNO
SMDEDITION
ARDUINO
ON
TX
RX
WWW.ARDUINO.CC
MADE IN ITALY

WHAT IS ARDUINO

- Arduino is a tool for making computers that can sense and control more of the physical world than your desktop computer.
- It's an open-source physical computing platform based on a simple microcontroller board, and a development environment for writing software for the board

HISTORY OF ARDUINO

- Founded in 2005 by Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, and David Mellis.
- Originated as a tool for students at the Interaction Design Institute Ivrea, Italy.
- Goals:
  - Simplify electronics for non-engineers.
  - Make prototyping accessible and affordable.
- Name derived from a bar named “Bar di Re Arduino” in Ivrea.
- Evolution of Arduino:
  - Early boards (Arduino Serial) to modern variants (Uno, Mega, etc.).
  - Open-source model contributed to rapid adoption worldwide.

THE ARDUINO ECOSYSTEM

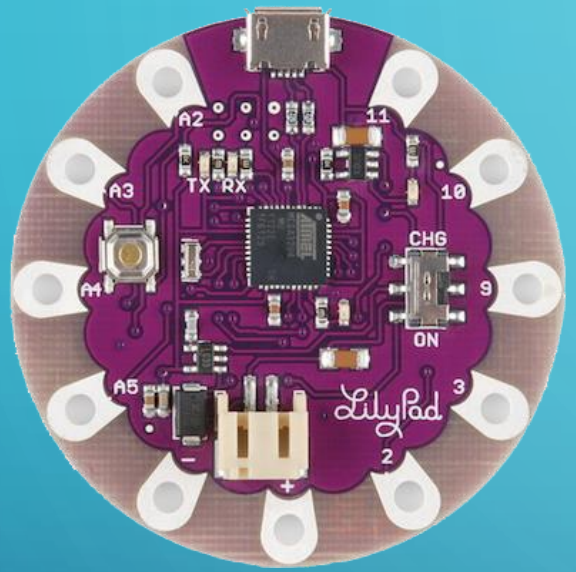
Hardware

Software

ARDUINO
OPEN-SOURCE
COMMUNITY

Community

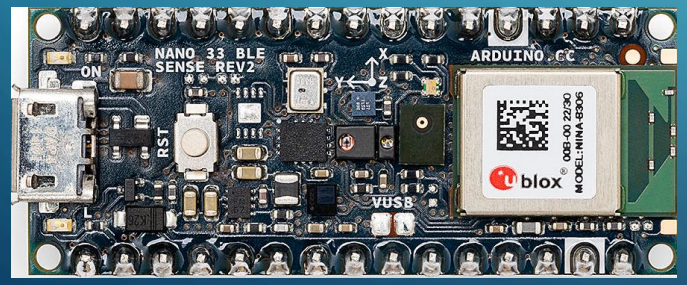
TYPES OF ARDUINOS



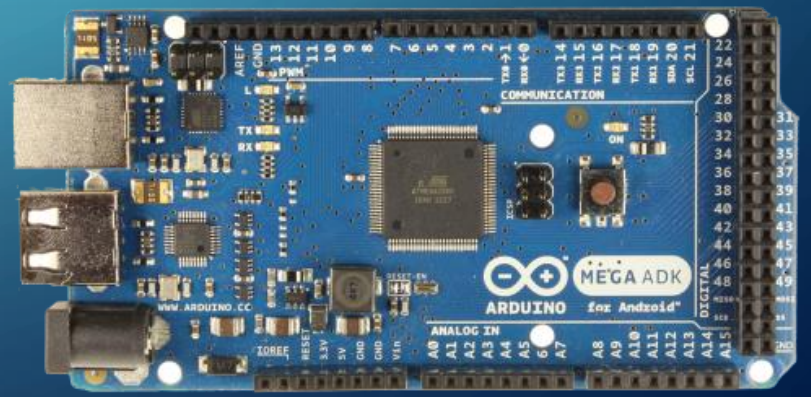
Arduino LILYPAD

Arduino UNO R2

Arduino NANO

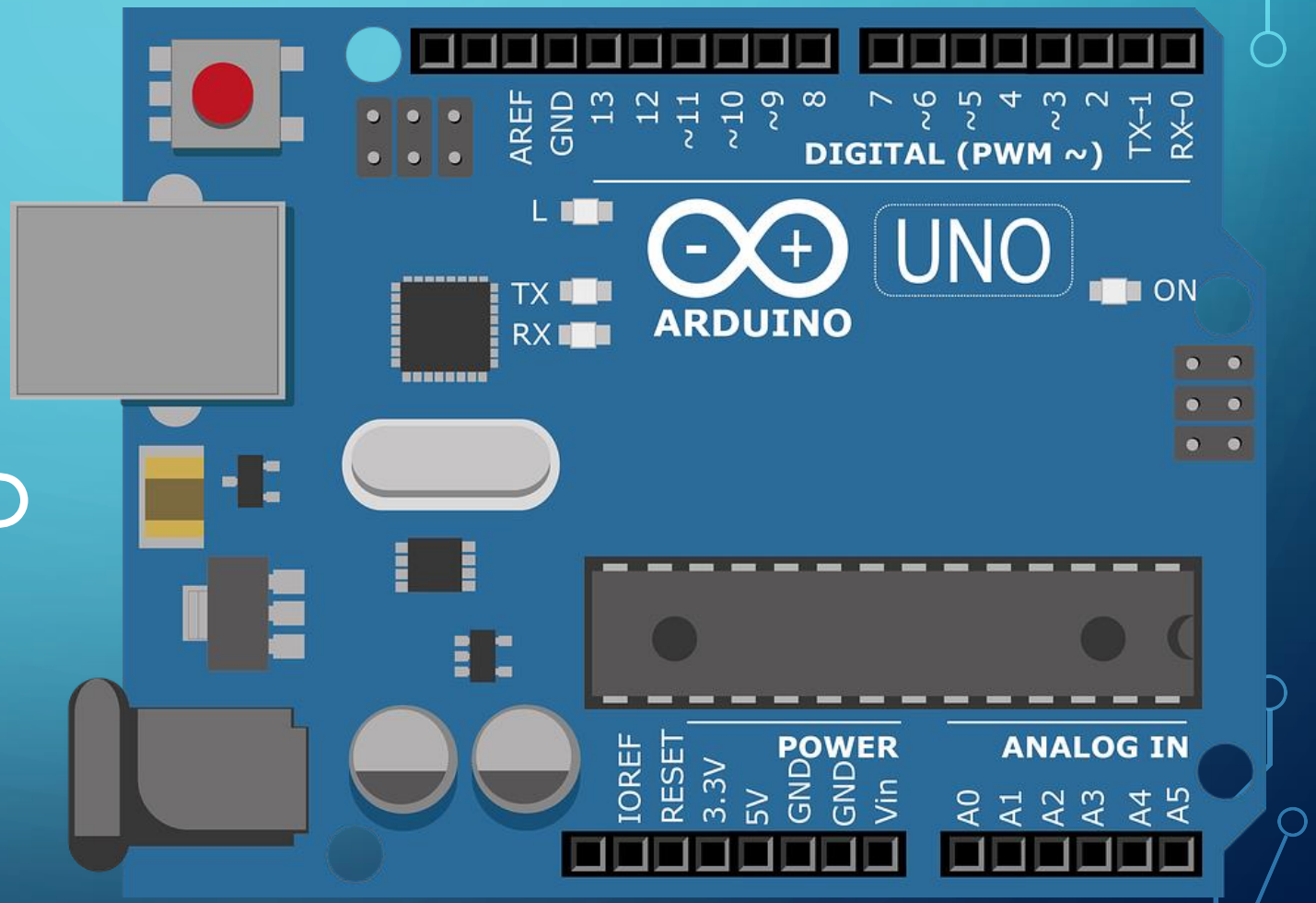


Arduino NANO 33
BLE Sense Rev2



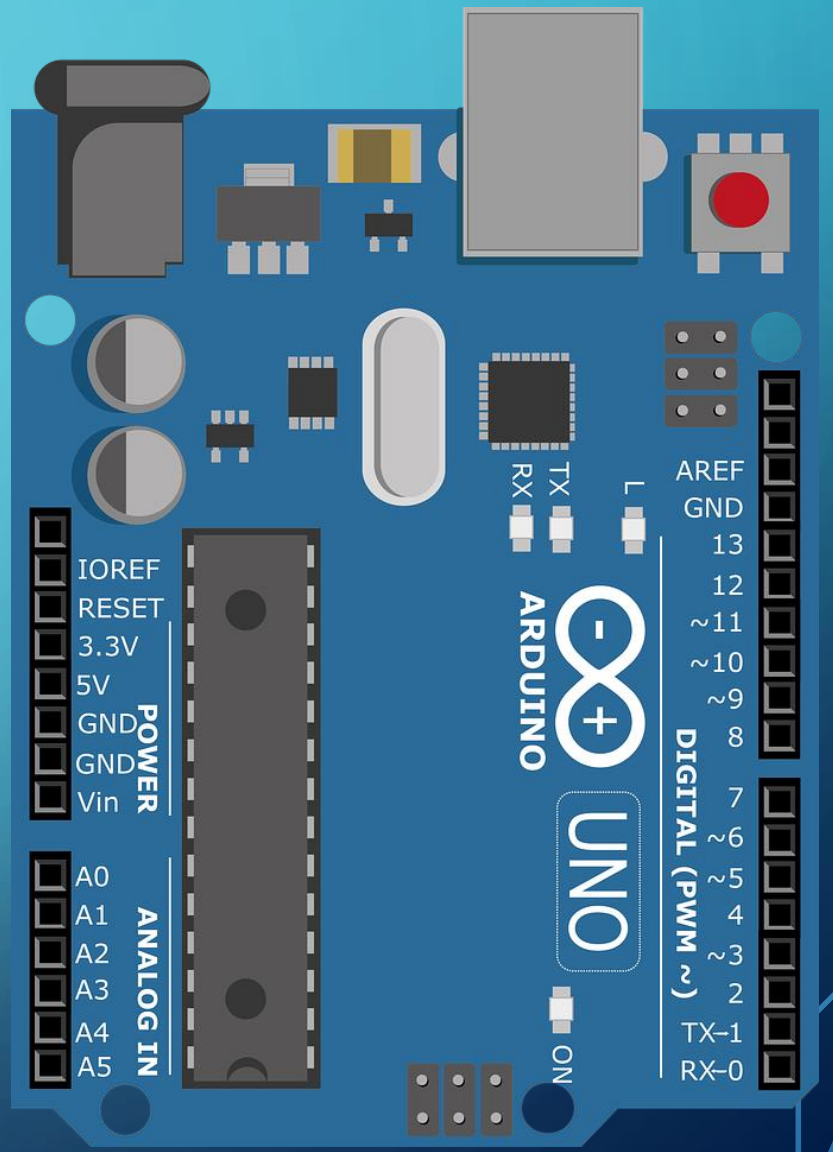
Arduino MEGA

THE ARDUINO BOARD

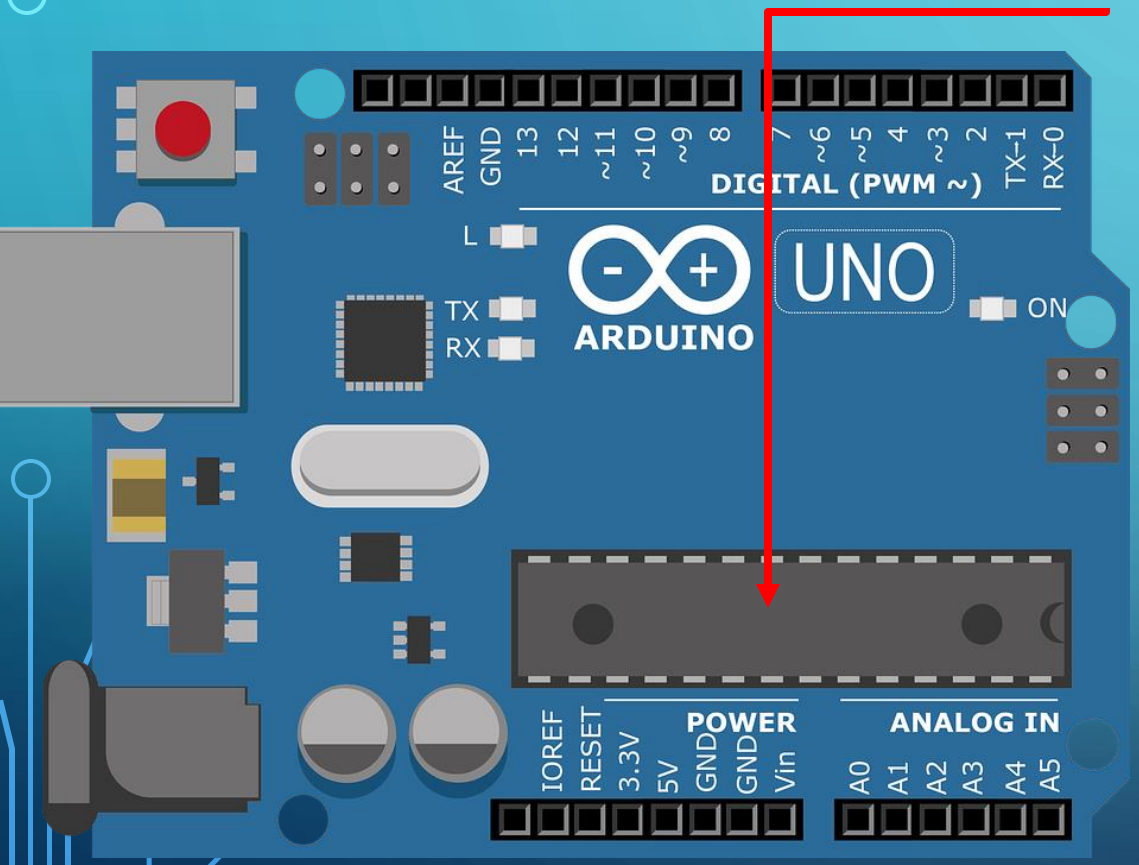


THE ARDUINO BOARD

- 14 Digital In/Out pins (6 can be used as PWM)
- 6 Analog Inputs
- A USB Connection
- A Power Jack
- Reset Button
- On-board LED
- SCL/SDA pins (Serial Clock/ Serial Data pins)



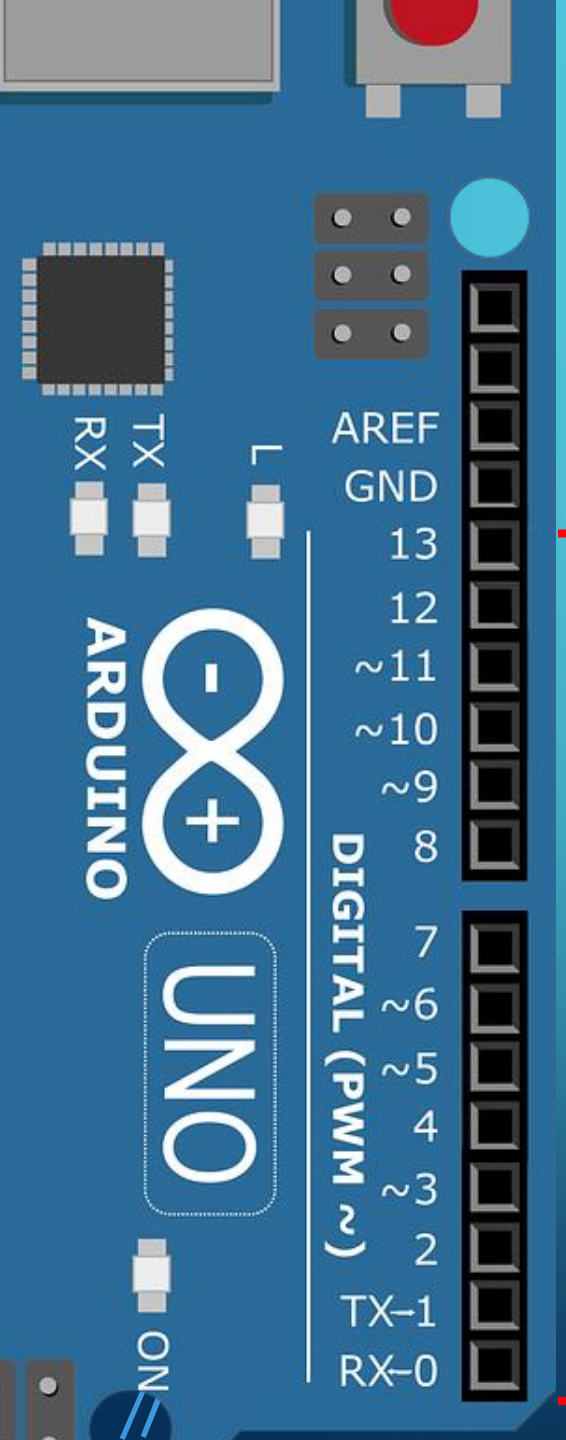
THE ARDUINO BOARD



The ATmega328P

- An 8-bit microcontroller that acts as the "brain" of the Arduino Uno.
- 32KB of flash memory
- 2KB of SRAM
- 1KB of EEPROM
- Integrated peripherals include timers, analog-to-digital converters (ADCs), and communication interfaces like UART, SPI, and I2C.

THE ARDUINO BOARD



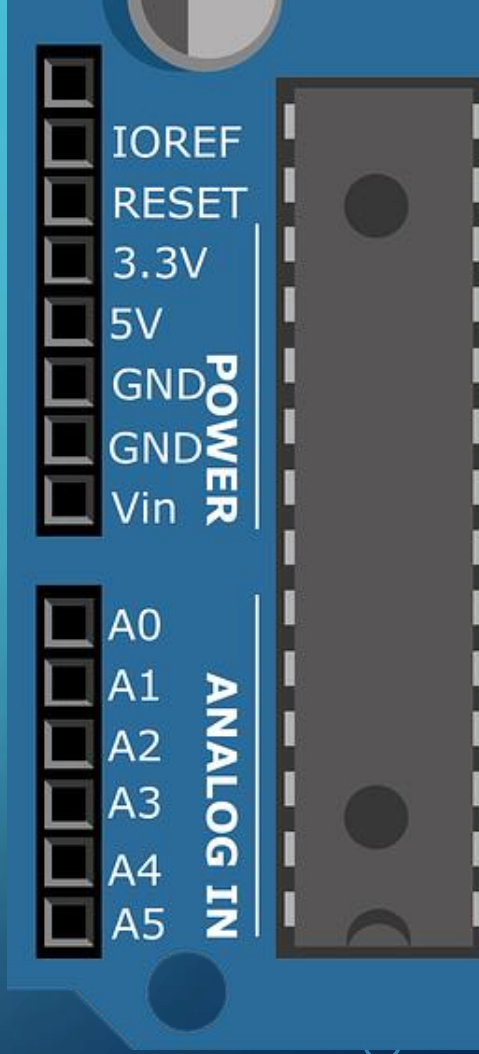
Digital I/O Pins

- The Arduino Uno has 14 digital pins that can be configured as inputs or outputs.
- Six of these pins support Pulse Width Modulation (PWM), enabling finer control for tasks like motor speed or LED brightness.
- Each pin operates at a 5V logic level with a maximum current of 40mA.

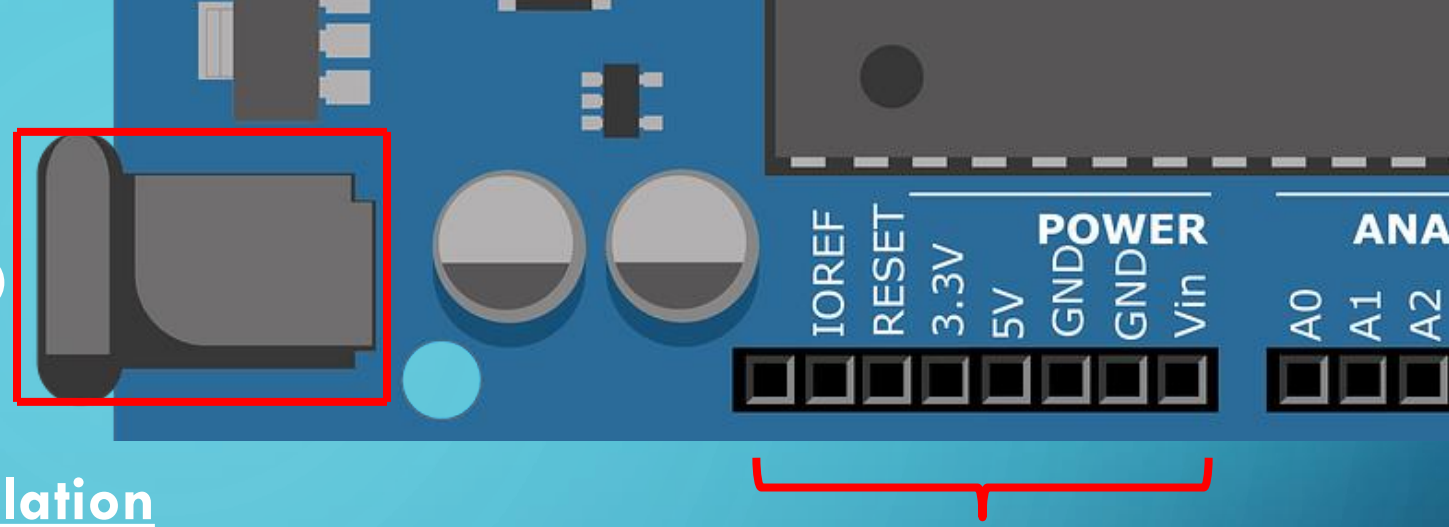
THE ARDUINO BOARD

Analog I/O Pins

- There are six analog input pins (A0–A5) for reading analog signals, such as sensor outputs.
- They use the ATmega328P's ADC to convert analog signals (0–5V) into a digital value (0–1023).
- This allows the Arduino to interface with variable inputs like temperature, light, or potentiometer adjustments. The pins can also be configured as digital I/O if needed.



THE ARDUINO BOARD



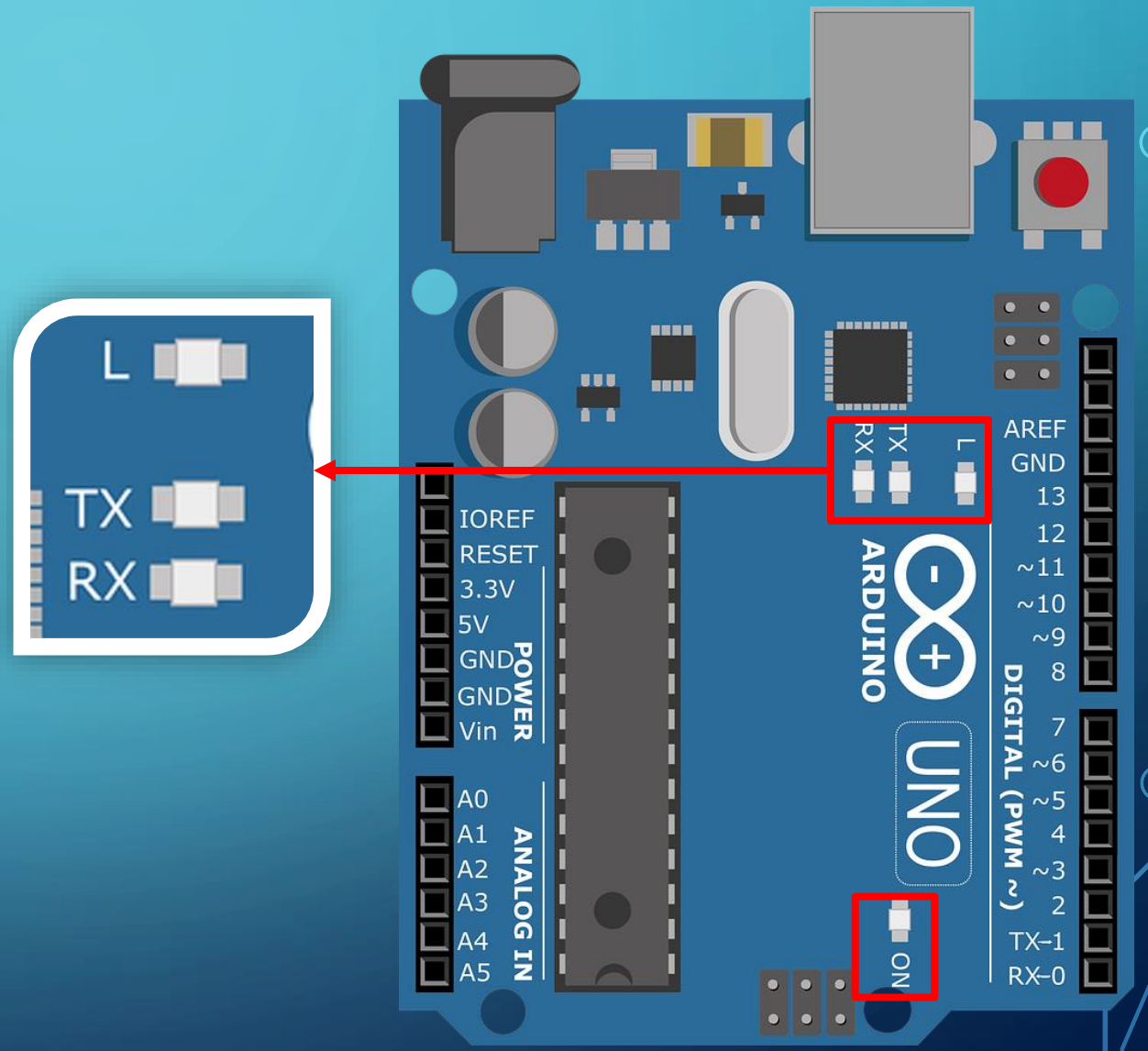
Power Supply and Voltage Regulation

- The Arduino Uno can be powered through a USB cable (5V) or an external DC power jack (7–12V).
- It includes a voltage regulator to ensure a stable 5V and 3.3V supply for components.
- The board has multiple power pins: 3.3V, 5V, GND (ground), and VIN for external power. These pins provide power to external sensors, modules, or other electronics.

THE ARDUINO BOARD

LED Indicators

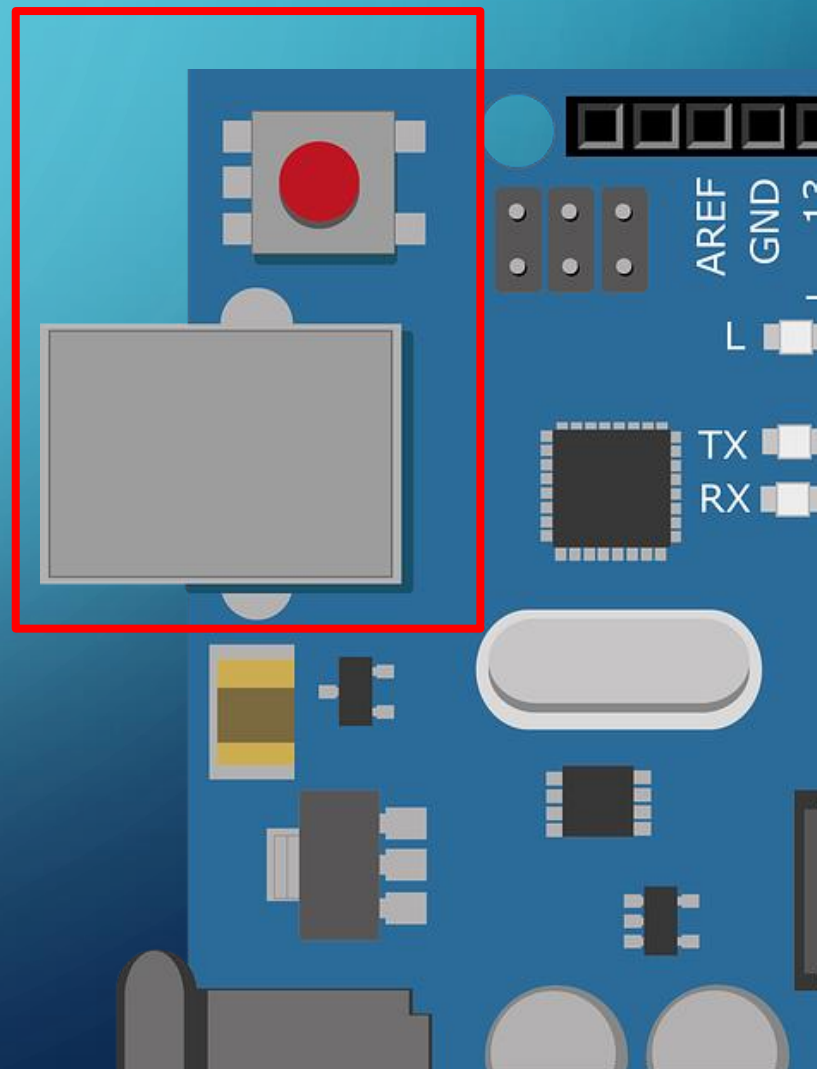
- Power LED: Indicates the board is receiving power.
- TX/RX LEDs: Blink during data transmission or reception via USB.
- Pin 13 LED: A programmable LED connected to digital pin 13 for quick testing or debugging. These LEDs help monitor board activity and functionality.



THE ARDUINO BOARD

Other Components

- **USB Interface:** The ATmega16U2 chip acts as a USB-to-serial converter, allowing the Arduino Uno to communicate with a computer.
- **Reset Button:** The reset button restarts the Arduino Uno, halting the current program and reinitializing the microcontroller.



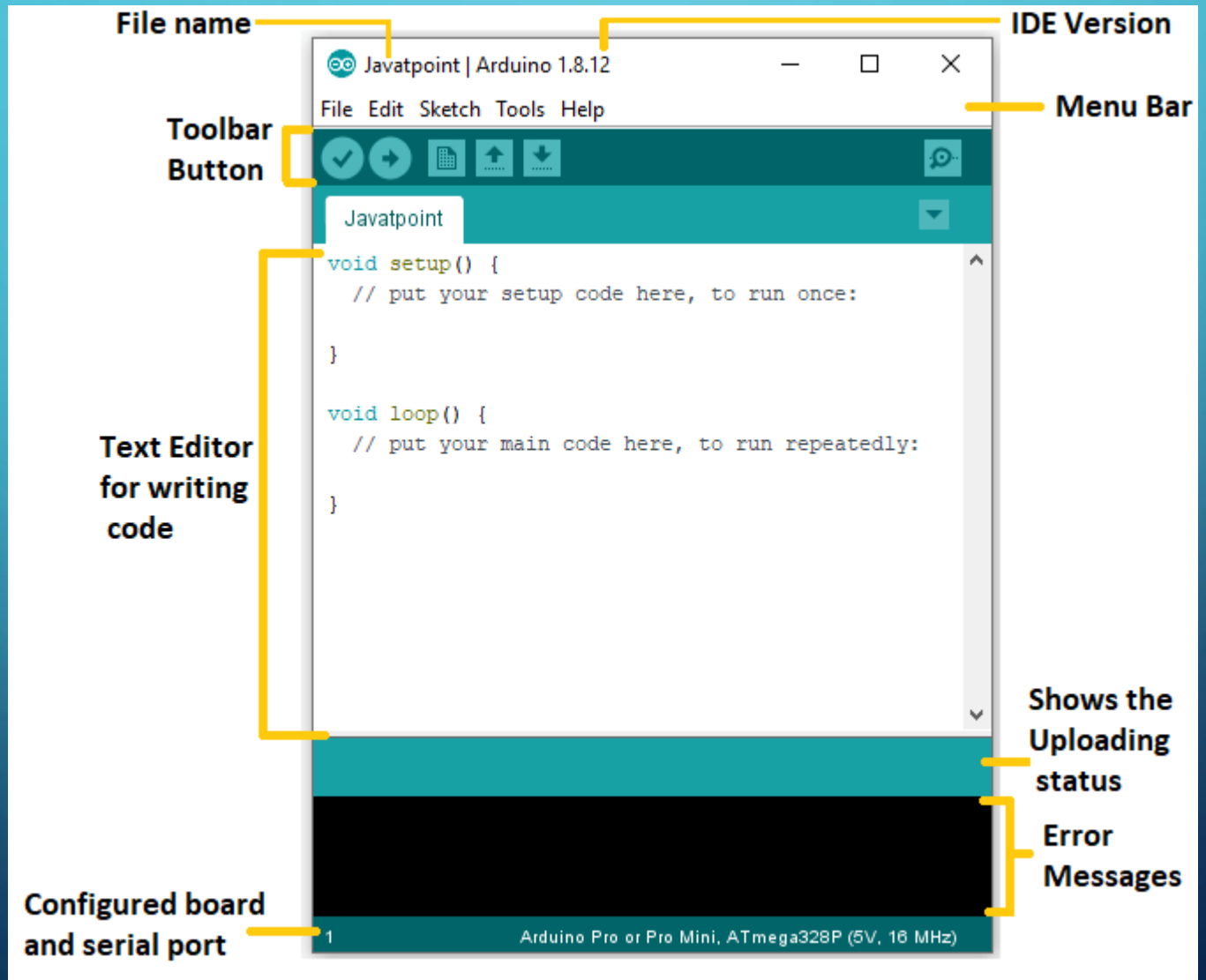
PROGRAMMING IN ARDUINO

- An Arduino board is programmed using the Arduino IDE. It can be downloaded here: https://www.arduino.cc/en/software
- The IDE includes a text editor, a message area, a text console, a toolbar, and menus. It also has features like autocompletion, code navigation, and a live debugger.

PROGRAMMING IN ARDUINO

- Programs written using the Arduino IDE are called sketches. Sketches are written in the text editor and are saved with the file extension .ino.
- Programming language is called **Wiring**, which is based on C++, but it's designed to be easier to learn.

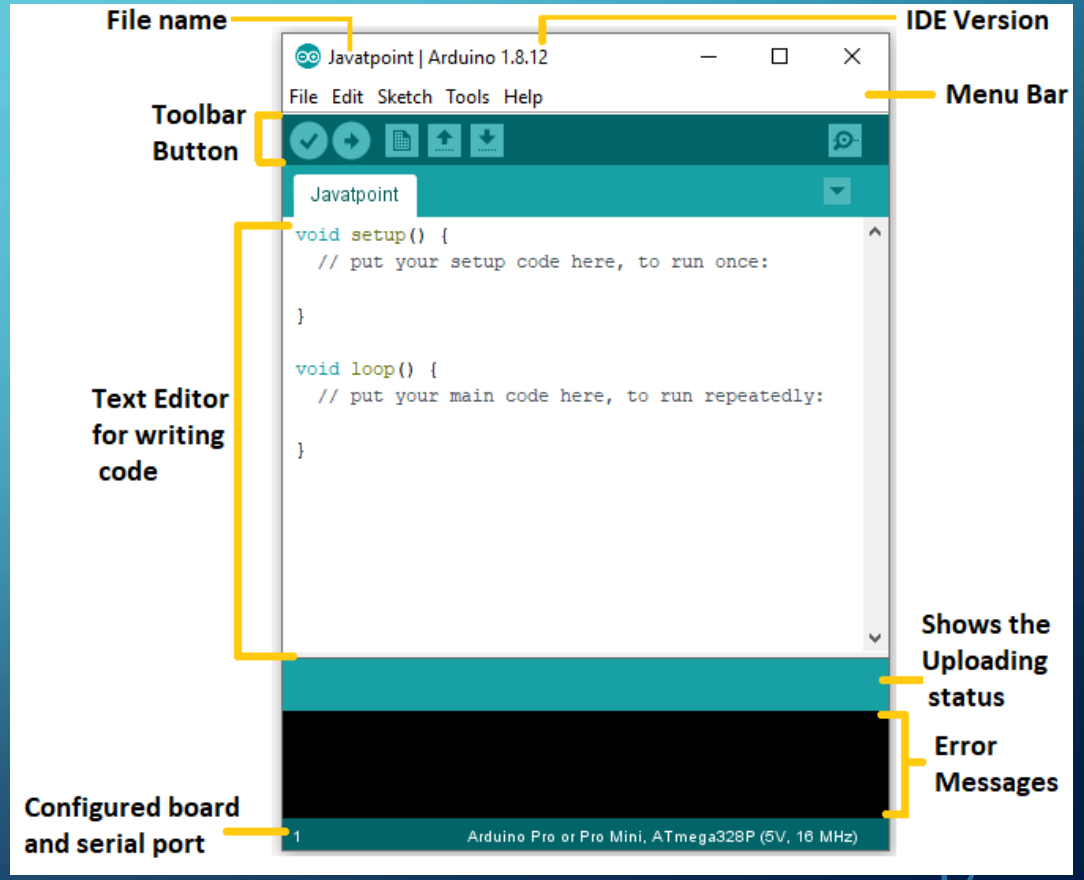
PROGRAMMING IN ARDUINO



PROGRAMMING IN ARDUINO

Setup() Function

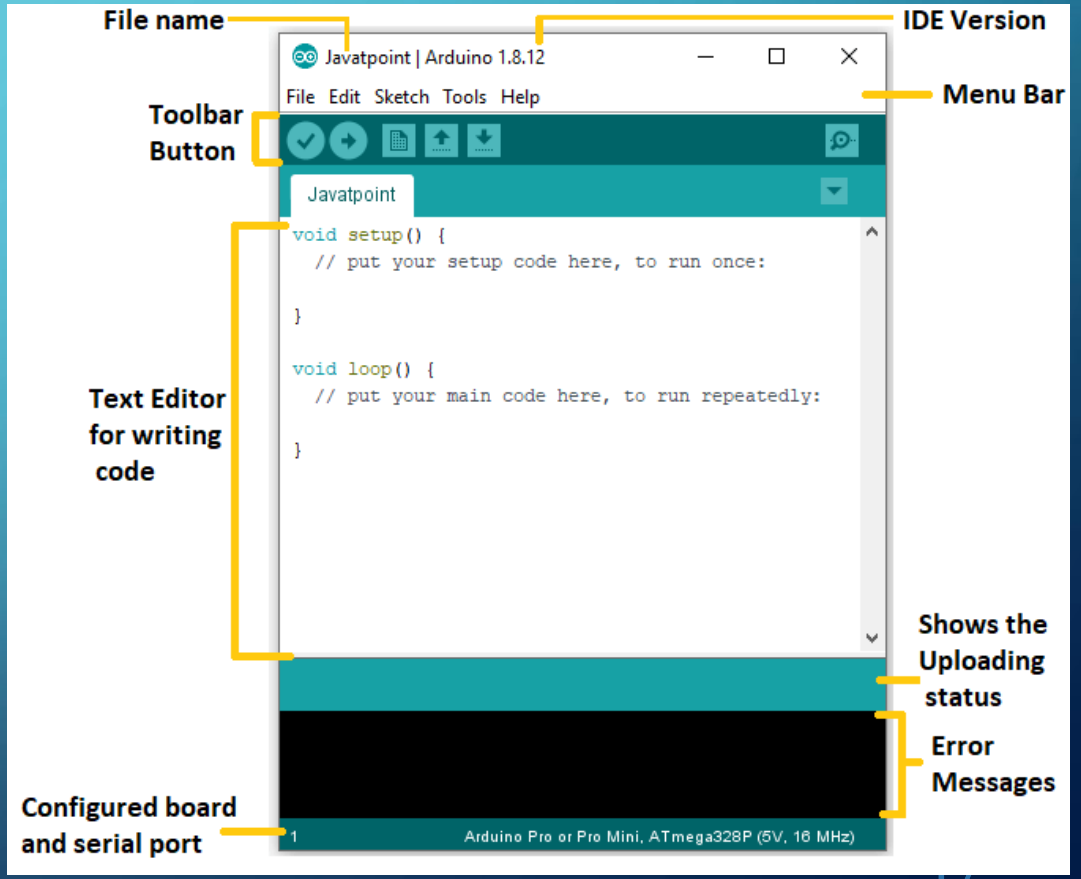
- Runs only once after the board starts.
- Common tasks include setting pin modes (e.g., INPUT or OUTPUT) and initializing libraries like Serial.begin().
- Used to prepare the Arduino environment for the main program logic.



PROGRAMMING IN ARDUINO

Setup() Function

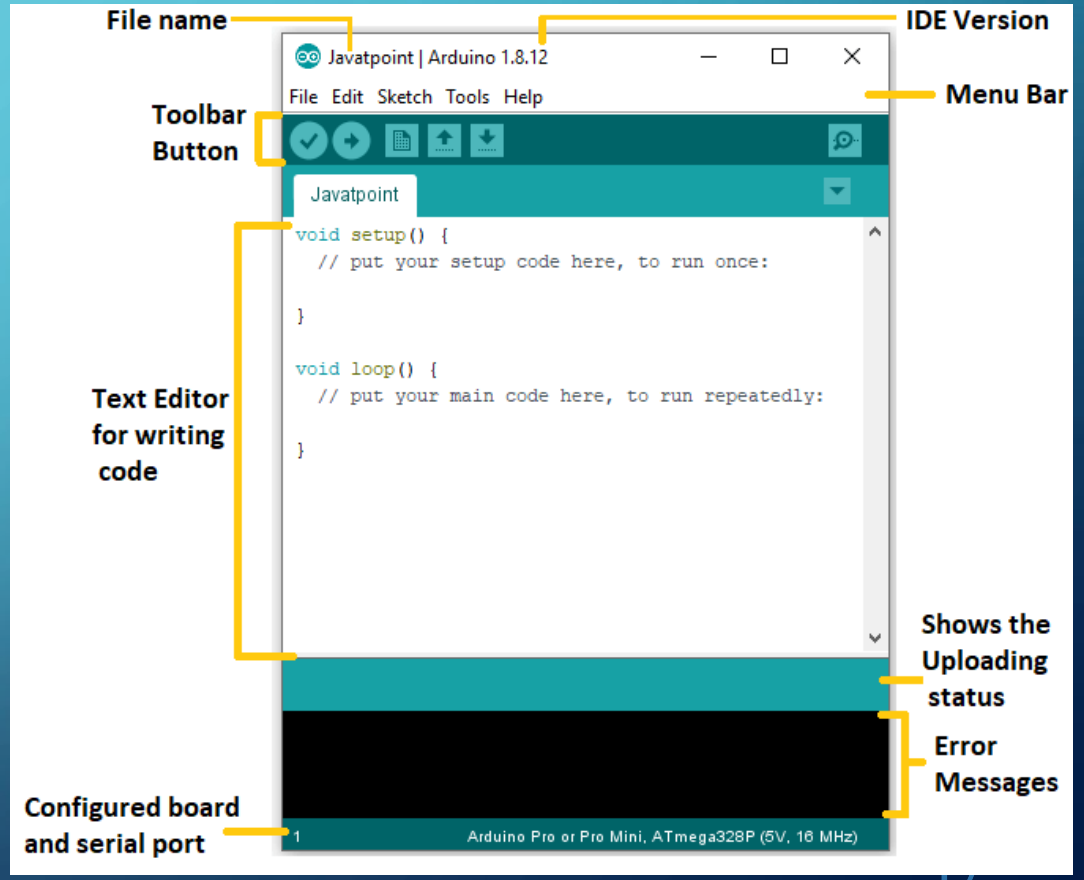
```
void setup() {
    pinMode(13, OUTPUT);    // Set digital pin 13 as an output
    Serial.begin(9600);     // Initialize serial communication at 9600 baud
    Serial.println("Setup Complete!"); // Print a message to the serial monitor
}
```



PROGRAMMING IN ARDUINO

Loop() Function

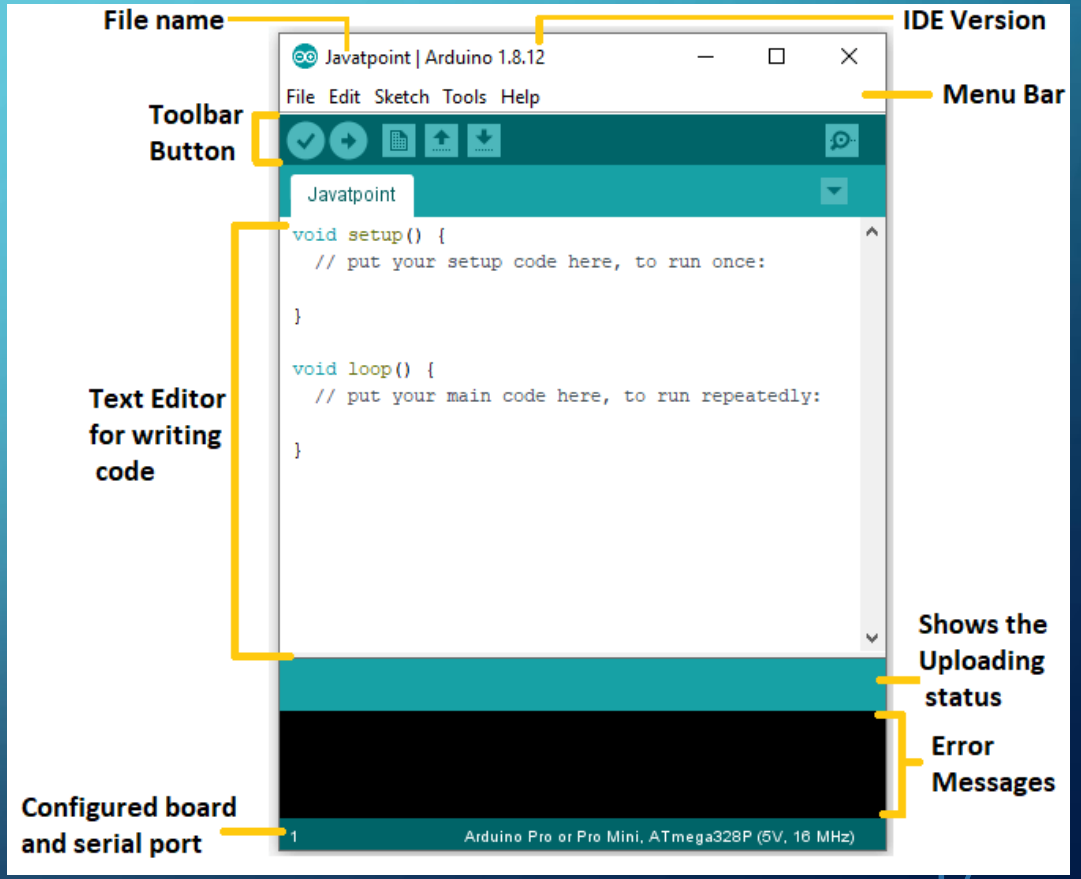
- Executes continuously after setup() finishes.
- Useful for tasks that require constant checking or repetitive actions, such as blinking an LED, reading sensor data, or communicating over serial.
- Can include delays (delay()) or logic to pace the execution.



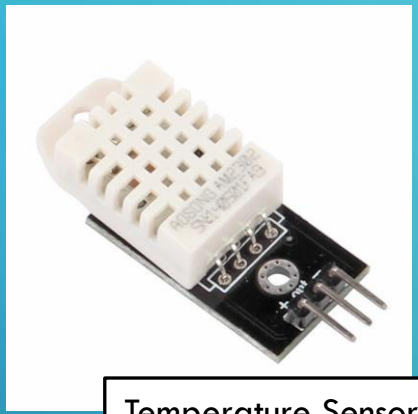
PROGRAMMING IN ARDUINO

Loop() Function

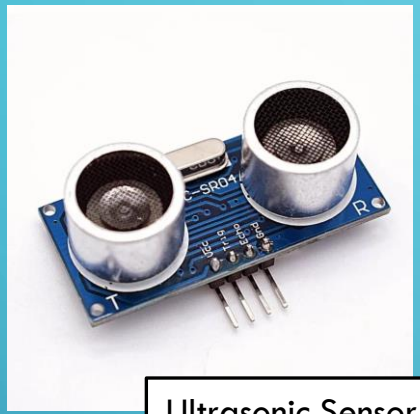
```
void loop() {
    digitalWrite(13, HIGH); // Turn LED on
    delay(1000);            // Wait 1 second
    digitalWrite(13, LOW);  // Turn LED off
    delay(1000);            // Wait 1 second
}
```



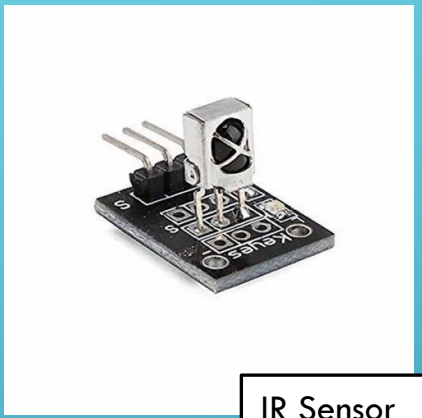
ARDUINO COMPATIBLE ELECTRONICS



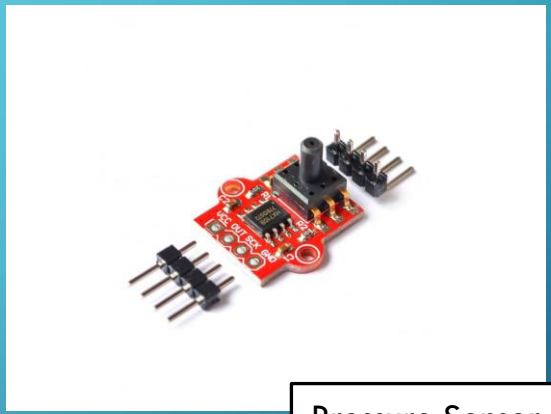
Temperature Sensor



Ultrasonic Sensor



IR Sensor



Pressure Sensor

Servo Motor

LCD Display

LED

WHY ARDUINO?

- Easy to learn and use, even for beginners.
- Cost-effective and widely available.
- Open-source hardware and software, enabling customization.
- Versatile platform suitable for various projects, from simple to complex.
- Cross-platform IDE supports Windows, macOS, and Linux.
- Compatible with a wide range of sensors, modules, and actuators.
- Widely used in STEM education for hands-on learning.
- Low power consumption, ideal for portable projects.

THANK YOU