

# **1 Building the GwBlock Graphical Programming Development Environment**

We creatively provide users with Arduino graphical programming tools-GwBlock. Using graphical program instruction blocks to achieve control of Arduino with the Web page. Compared with the traditional pure character interface code programming platform, graphical programming is more conducive to learners who have not mastered C/C++. If you have studied Scratch, then you will be able to easily master the graphical programming of Arduino. Next we will teach you how to Building the GwBlock Graphical Programming Development Environment.

## **1. Downloading and installing Python**

1.1 Log in to the official website by browser:

<https://www.python.org/downloads/>



1.2 Click the "Download Python 3.8.3" button to download and wait for the

download to complete. (The version will be updated, you can download the latest version)



1.3 Open the downloaded file, double-click to open it to install:



1.4 Select the "Add Python 3.8 to PATH" option:



1.5 Then click "Install Now" to install.



1.6 Wait for the Python installation to complete and click "Close" to close.

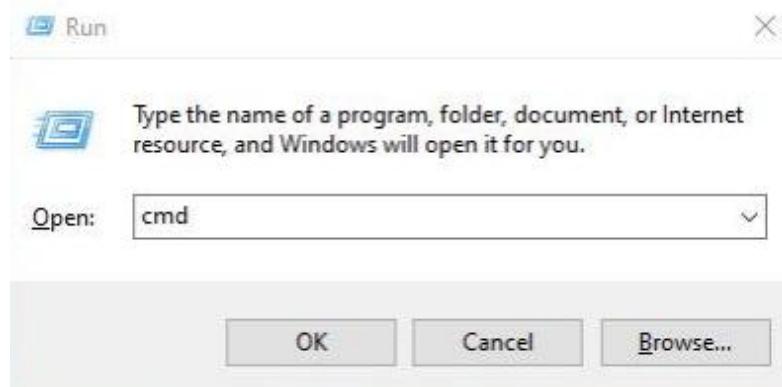


## 2. Installing pySerial and connecting GwBlock graphical editor

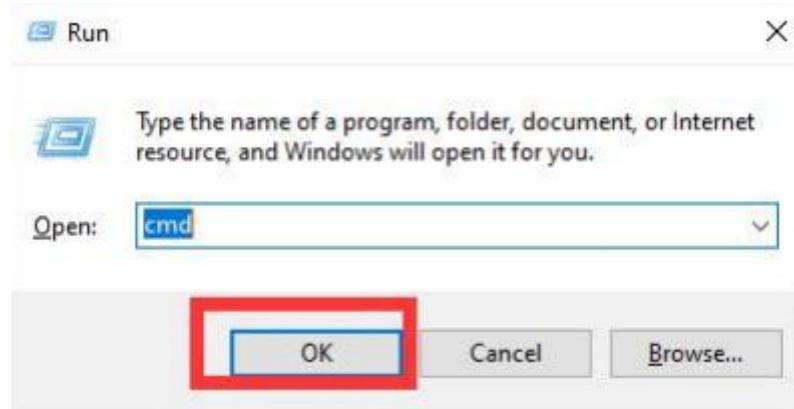
pySerial encapsulates the serial communication module, supporting Linux, Windows, BSD (may support all operating systems that support POSIX), Jython (Java) and IconPython (.NET and Mono). The pyserial module encapsulates access to the serial port. The port number starts from 0 by default. There is no need to know the port name in the program. APIs like file reading and writing, read and write (readline, etc. are also supported), support binary transmission, no null elimination, no cr-lf conversion. All programs are completed by Python In addition to the standard library, it does not depend on other packages, except pywin32 (windows), JavaComm (Jython). POSIX (Linux, BSD) only depends on the Python standard library. APIs like file read and write, read, write (readline, etc. are also supported), support binary transmission, no null elimination, no cr-lf conversion, all programs are all done by Python, and do not depend on other packages except the standard library, except pywin32 (windows), JavaComm (Jython). POSIX (Linux, BSD) only depends on the Python standard library.

Before downloading and installing, you need to connect the Adeept Arm Drive Board development board to your computer.

1.1 Press Win+R shortcut key to open CMD under Windows.



1.2 Click "OK":



1.3 Enter the command in the window:

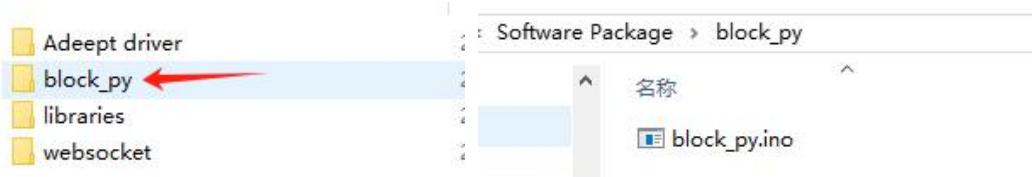
**pip install pyserial**

Press the Enter and wait for the installation to complete.

```
C:\Windows\system32\cmd.exe
Microsoft Windows [版本 10.0.18362.836]
(c) 2019 Microsoft Corporation. 保留所有权利。
C:\Users\ASUS>pip install pyserial
Collecting pyserial
  Downloading https://files.pythonhosted.org/packages/0d/e4/2a744dd9e3be04a0c0907414e2a01a7c88bb3915cba3c8cc06e209f59c30/pyserial-3.4-py2.py3-none-any.whl (193kB)
|██████████| 193kB 3.2kB/s
Installing collected packages: pyserial
Successfully installed pyserial-3.4
WARNING: You are using pip version 19.2.3, however version 20.1.1 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.
C:\Users\ASUS>
```

1.4 Open the folder "ADA031-Adeept\_Robotic\_Arm\_Kit\_for\_Arduino-V4.0"

provided by Adeept to the user → “Software Package”→"block\_py" and find this file: "[block\\_py.ino](#)".



```

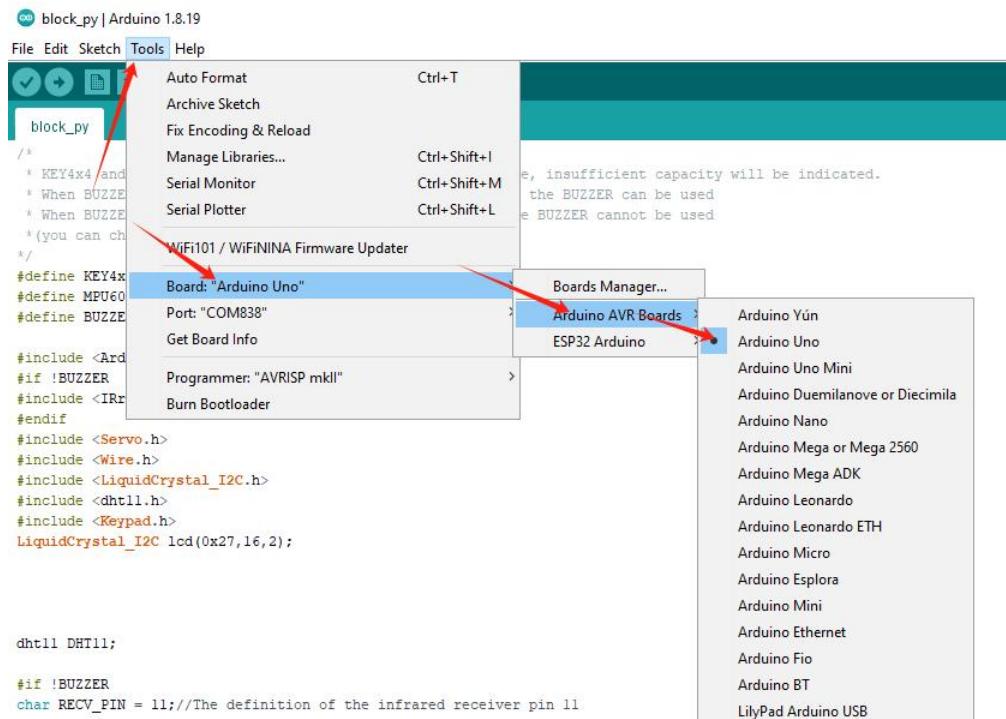
block_py | Arduino 1.8.19
文件 编辑 项目 工具 帮助
block_py $ 1
/*
 * KEY4x4 and MPU6050 cannot be 1 at the same time. Otherwise, insufficient capacity will be indicated.
 * When BUZZER is 1, the infrared sensor cannot be used, but the BUZZER can be used
 * When BUZZER is 0, the infrared sensor can be used, but the BUZZER cannot be used
 * (you can change these values as you apply them)
 */
#define KEY4x4 0
#define MPU6050 0
#define BUZZER 0

#include <ArduinoJson.h>
#if !BUZZER
#include <IRremote.h>
#endif
#include <Servo.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <dht11.h>
#include <Keypad.h>
LiquidCrystal_I2C lcd(0x27,16,2);

```

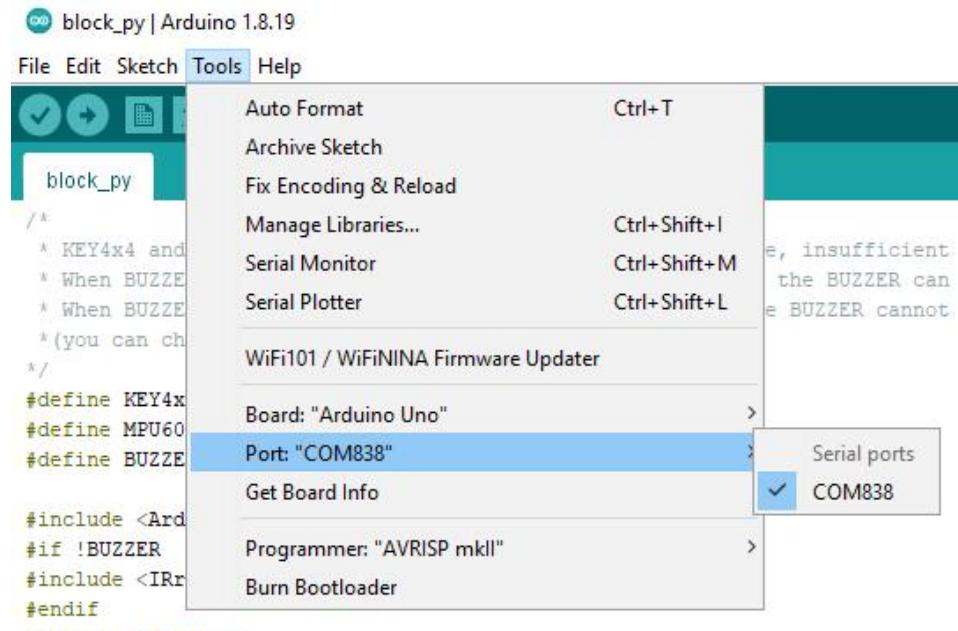
Arduino Uno at COM10

## 1.5 First select the Arduino development board as UNO version with Tools.



## 1.6 Then continue to select the "Port" of the Adeept Arm Drive Board connected

to the computer with Tools.

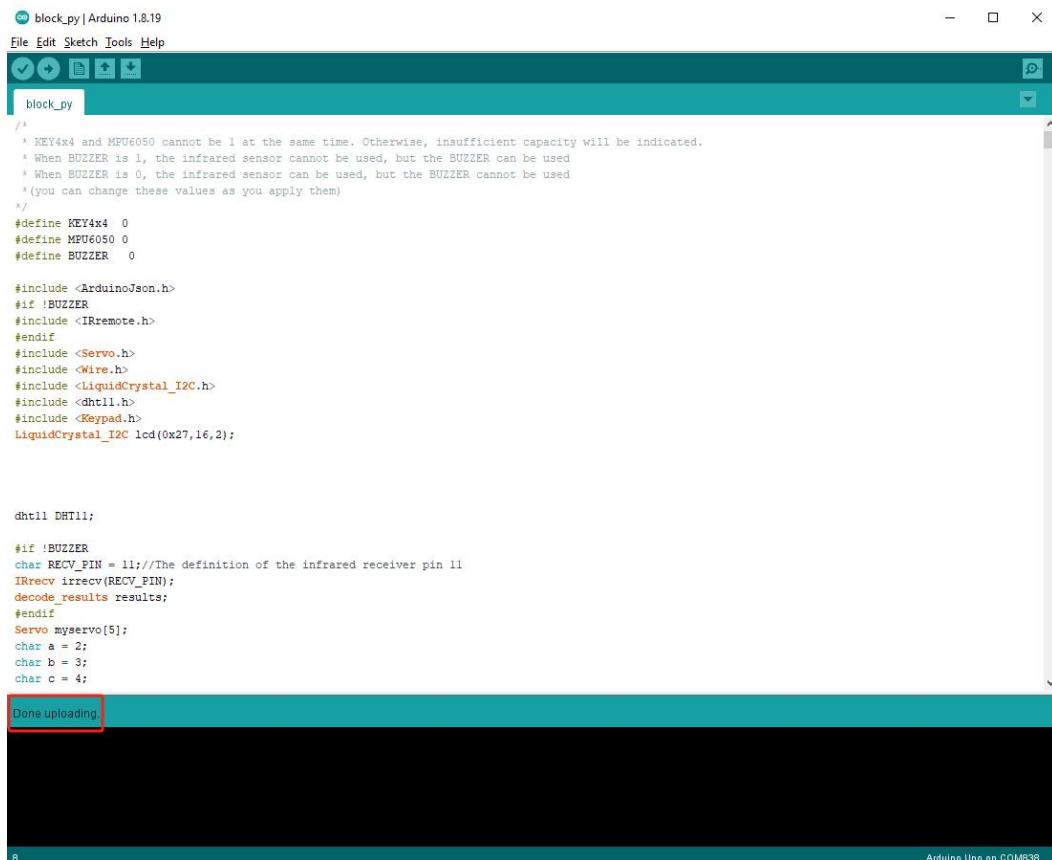


1.7 Click the Upload button to download the code program to the Arduino development board. Note: Please install the 18650 battery first and turn on the switch.

The screenshot shows the Arduino IDE with the code editor open. The code is for a project named 'block\_py'. The upload button, represented by a circular arrow icon, is highlighted with a red circle. The code itself includes definitions for KEY4x4, MPU6050, and BUZZER, along with various library includes and pin declarations. At the bottom of the code, there is a section for the DHT11 sensor and a loop that checks for infrared receiver input and moves a servo based on the received command.

```
/*  
 * KEY4x4 and MPU6050 cannot be 1 at the same time. Otherwise, insufficient capacity will be indicated.  
 * When BUZZER is 1, the infrared sensor cannot be used, but the BUZZER can be used  
 * When BUZZER is 0, the infrared sensor can be used, but the BUZZER cannot be used  
 * (you can change these values as you apply them)  
 */  
  
#define KEY4x4 0  
#define MPU6050 0  
#define BUZZER 0  
  
#include <ArduinoJson.h>  
#if !BUZZER  
#include <IRremote.h>  
#endif  
#include <Servo.h>  
#include <Wire.h>  
#include <LiquidCrystal_I2C.h>  
#include <DHT.h>  
#include <Keypad.h>  
LiquidCrystal_I2C lcd(0x27,16,2);  
  
  
dht11 DHT11;  
  
#if !BUZZER  
char RECV_PIN = 11;//The definition of the infrared receiver pin 11  
I2Crecv irrecv(RECV_PIN);  
decode_results results;  
#endif  
Servo myservo[5];  
char a = 2;  
char b = 3;  
char c = 4;
```

1.8 We re-use the Arduino software to open the "block\_py.ino" file, and then click the Upload button again to download the code program to the Adeept Arm Drive Board development board. After the download is successful, the following picture is shown.



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

- Title Bar:** block\_py | Arduino 1.8.19
- Menu Bar:** File Edit Sketch Tools Help
- Toolbar:** Standard icons for Open, Save, Print, etc.
- Sketch Name:** block\_py
- Code Area:**

```

/*
 * KEY4x4 and MFU6050 cannot be 1 at the same time. Otherwise, insufficient capacity will be indicated.
 * When BUZZER is 1, the infrared sensor cannot be used, but the BUZZER can be used
 * When BUZZER is 0, the infrared sensor can be used, but the BUZZER cannot be used
 * (you can change these values as you apply them)
 */
#define KEY4x4 0
#define MFU6050 0
#define BUZZER 0

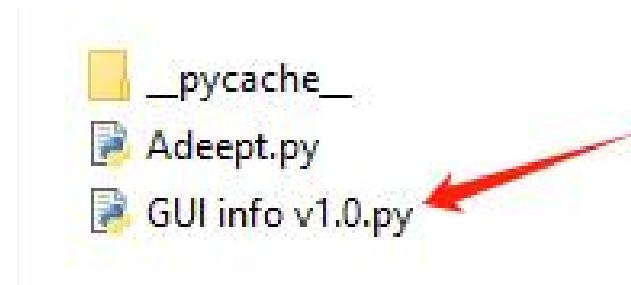
#include <ArduinoJson.h>
#if !BUZZER
#include <IRremote.h>
#endif
#include <Servo.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <DHT.h>
#include <Keypad.h>
LiquidCrystal_I2C lcd(0x27,16,2);

dht11 DHT11;

#if !BUZZER
char RECV_PIN = 11;//The definition of the infrared receiver pin 11
IRrecv irrecv(RECV_PIN);
decode_results results;
#endif
Servo myservo[5];
char a = 2;
char b = 3;
char c = 4;

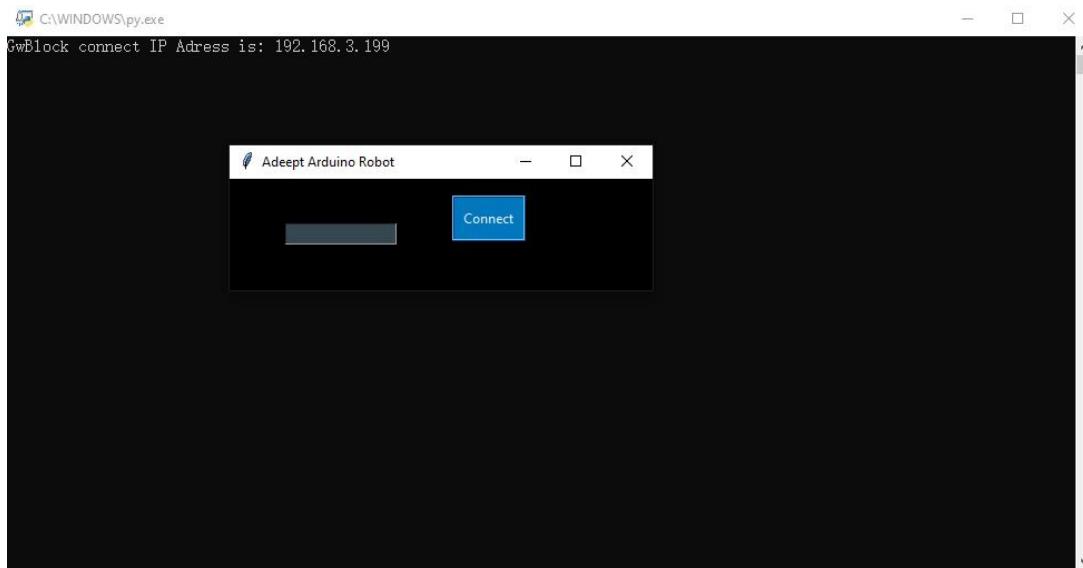
```
- Status Bar:** Done uploading.
- Bottom Right:** Arduino Uno on COM38

1.9 Find the folder "Adeept\_Robotic\_Arm\_Kit\_for\_Arduino-V4.0"→"Software Package"→"websocket", find this file: "GUI info v1.0.py".

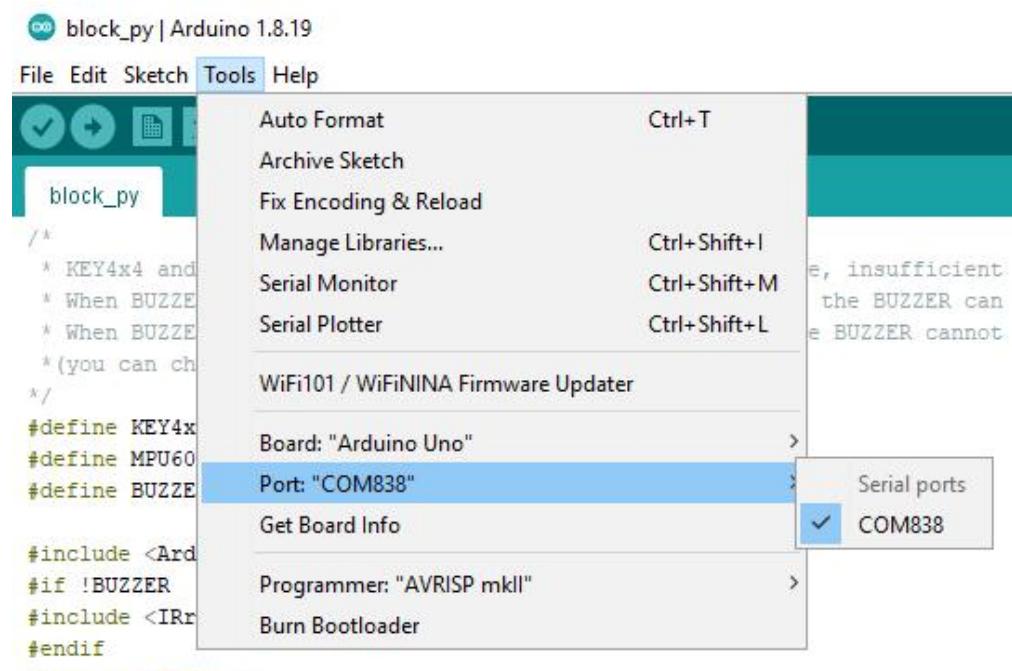


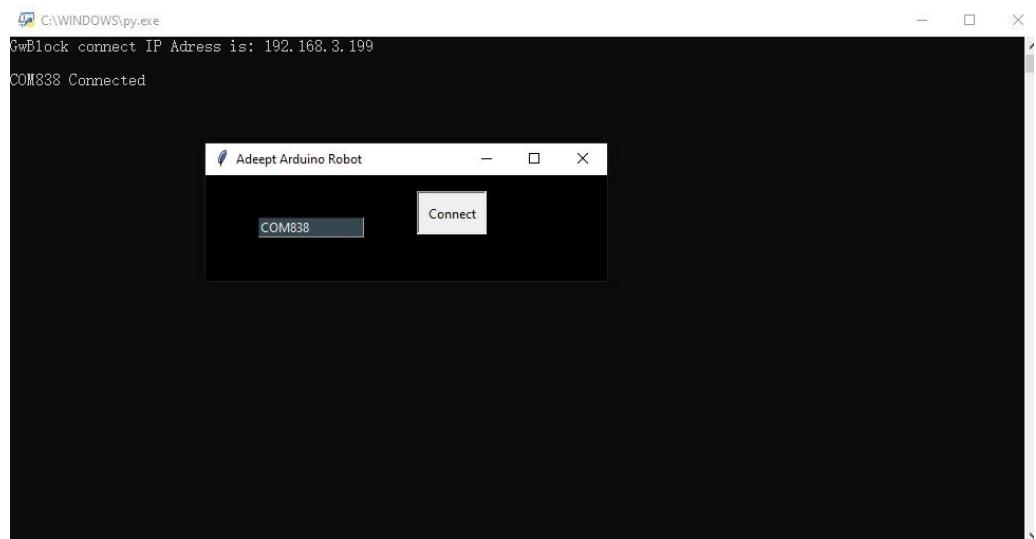
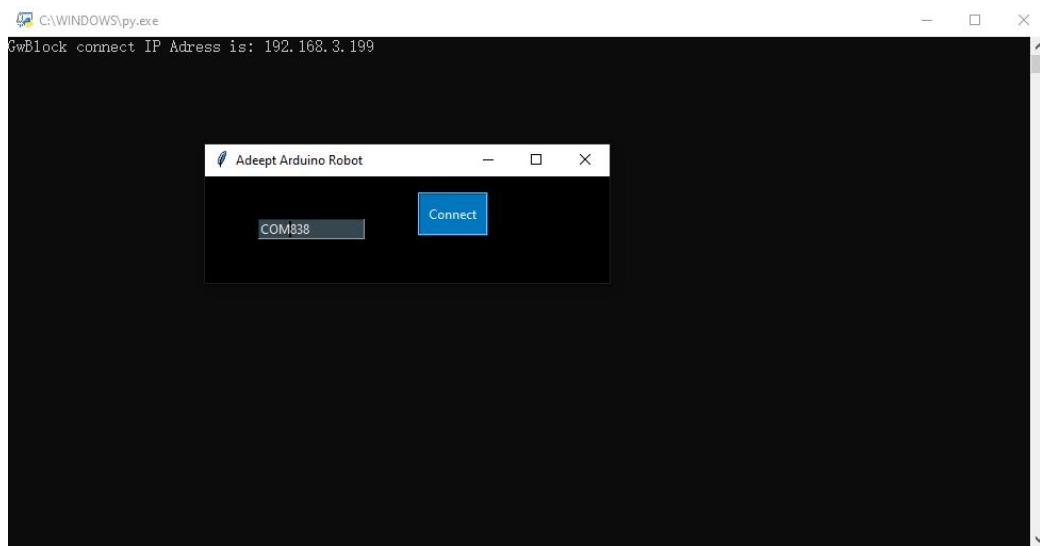
1.10 Double-click to open this file: GUI info v1.0.py

1.11 After opening the file, the following interface will appear. We need to record this IP address, which will be used later: 192.168.3.199



1.12 In the input box, enter the port we set in Arduino. Everyone's port is different. The port of my Adeebt Arm Drive Board development board is: COM838. After entering, click the Connect button.

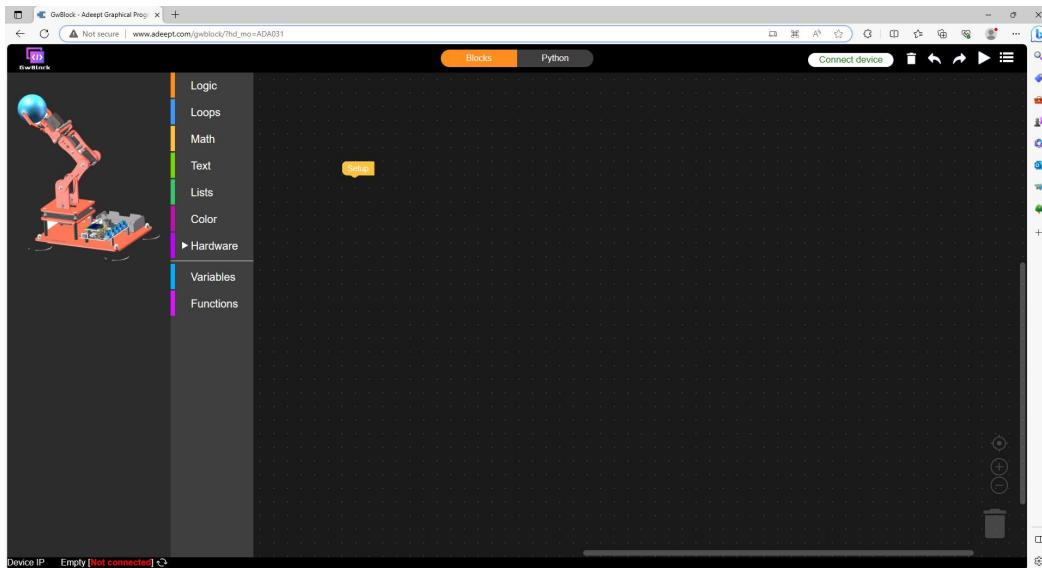




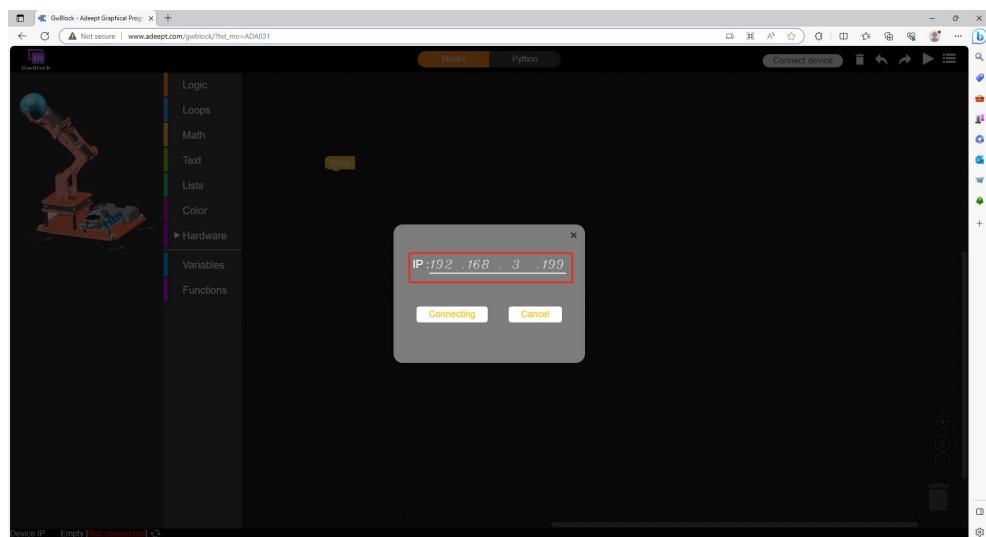
1.13 Enter the URL of the GwBlock graphical editor in the browser:

[http://www.adeept.com/gwblock/?hd\\_mo=ADA031](http://www.adeept.com/gwblock/?hd_mo=ADA031)

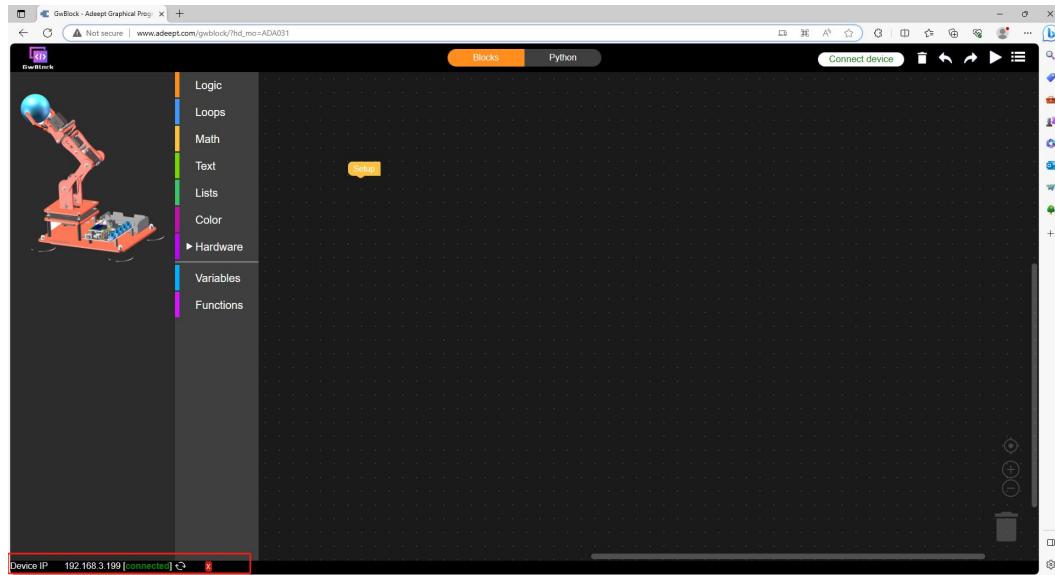
After successfully entering the website, the interface is as follows:



1.14 Then click the "Connecting device" button in the upper right corner. In the pop-up box, enter the IP address we recorded, as shown below, and then click the Connecting button. Note: If the connection fails, check whether the 18650 battery is installed and the switch is turned on. If not, please install the 18650 battery first and turn on the switch.



1.15 After the connection is successful, a green "connected" will appear in the lower left corner of the interface, indicating that the connection with the Adept Arm Drive Board is successful.



### 3. The Matters needing attention of reconnecting the GwBlock graphical editor

In the following cases, you need to reconnect to the GwBlock graphical editor:

- [1] When you close the GwBlock editor.
- [2] When you close the Adeept Arduino Robot window.
- [3] When you restart the computer.
- [4] When you log in to the GwBlock website again.
- [5] When you close.

You need to repeat the above operation to connect.