

OTTO Robot Operating Manual V1.0

1. Power On and Startup

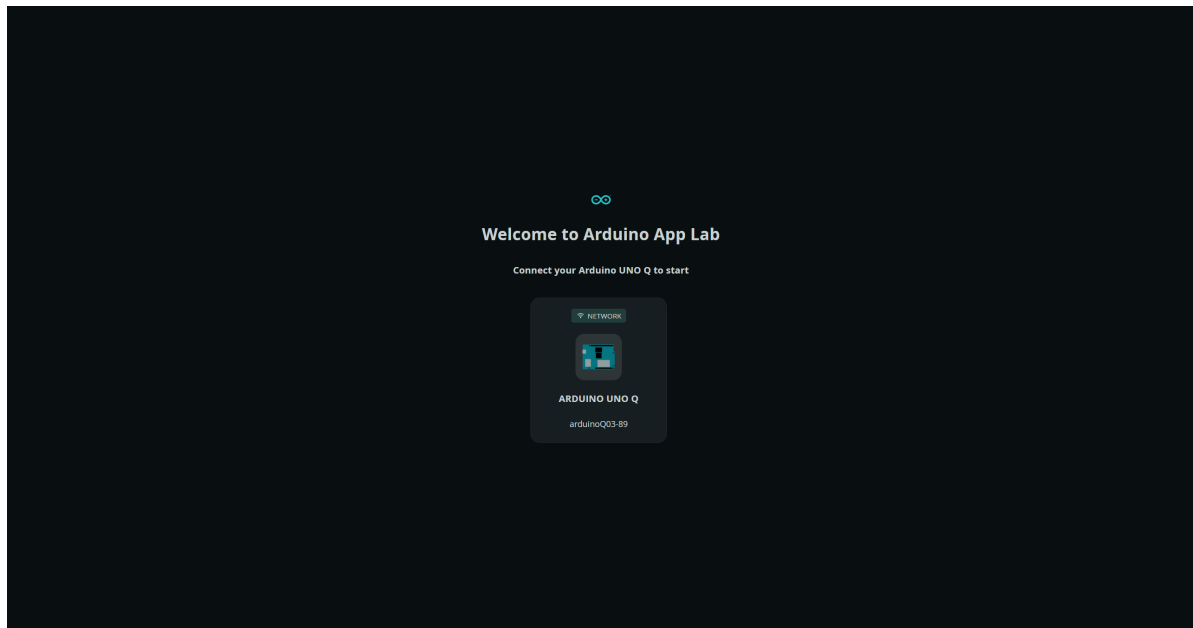
Turn on the switch and wait for the robot to light up. After normal startup, the lights can be seen on the side.



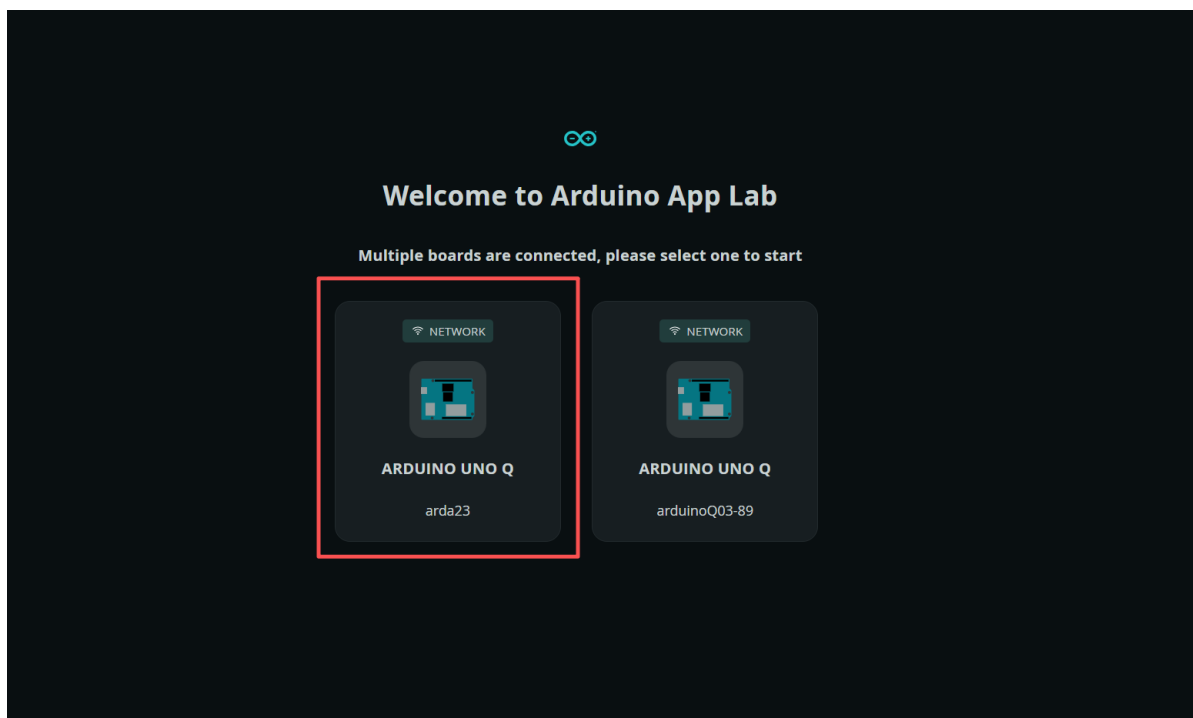


2. Arduino AppLab Debugging

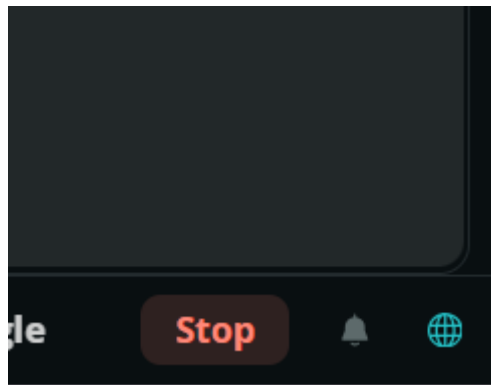
Open the Arduino AppLab software:



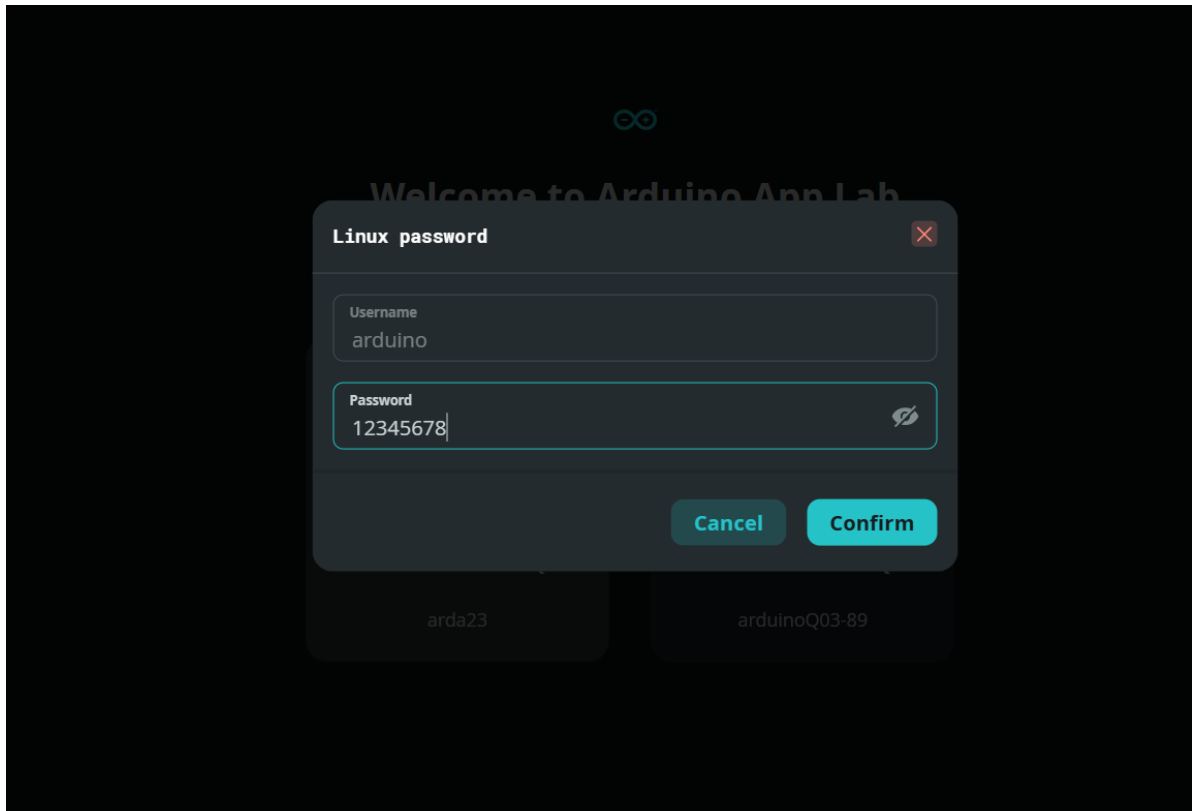
Wait for the corresponding UNO Q development board to appear on the page:



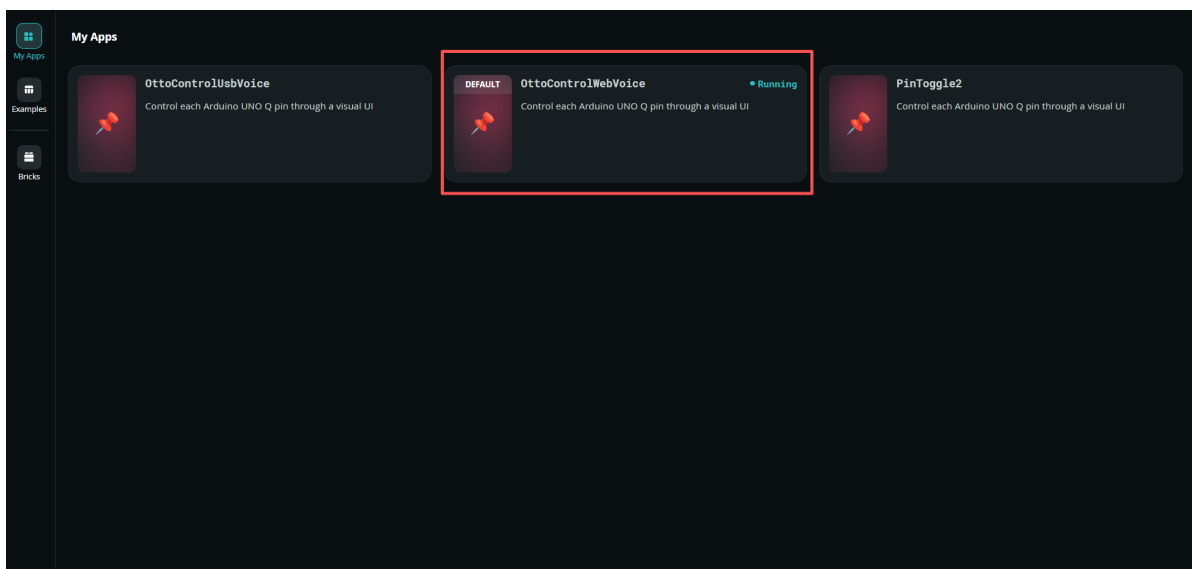
If this is the first time connecting to a new WiFi network, please use a USB cable to connect the TypeC port to your computer's USB port first. Then in the App Lab software, after selecting the board, a network connection option will appear, usually in the bottom right corner:



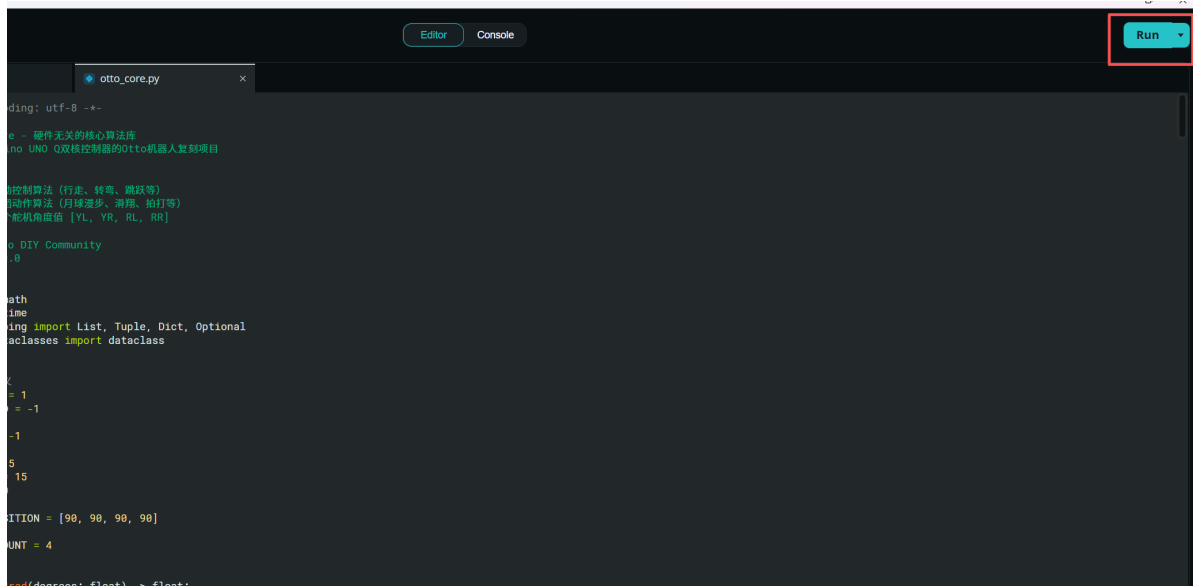
Enter the password 12345678



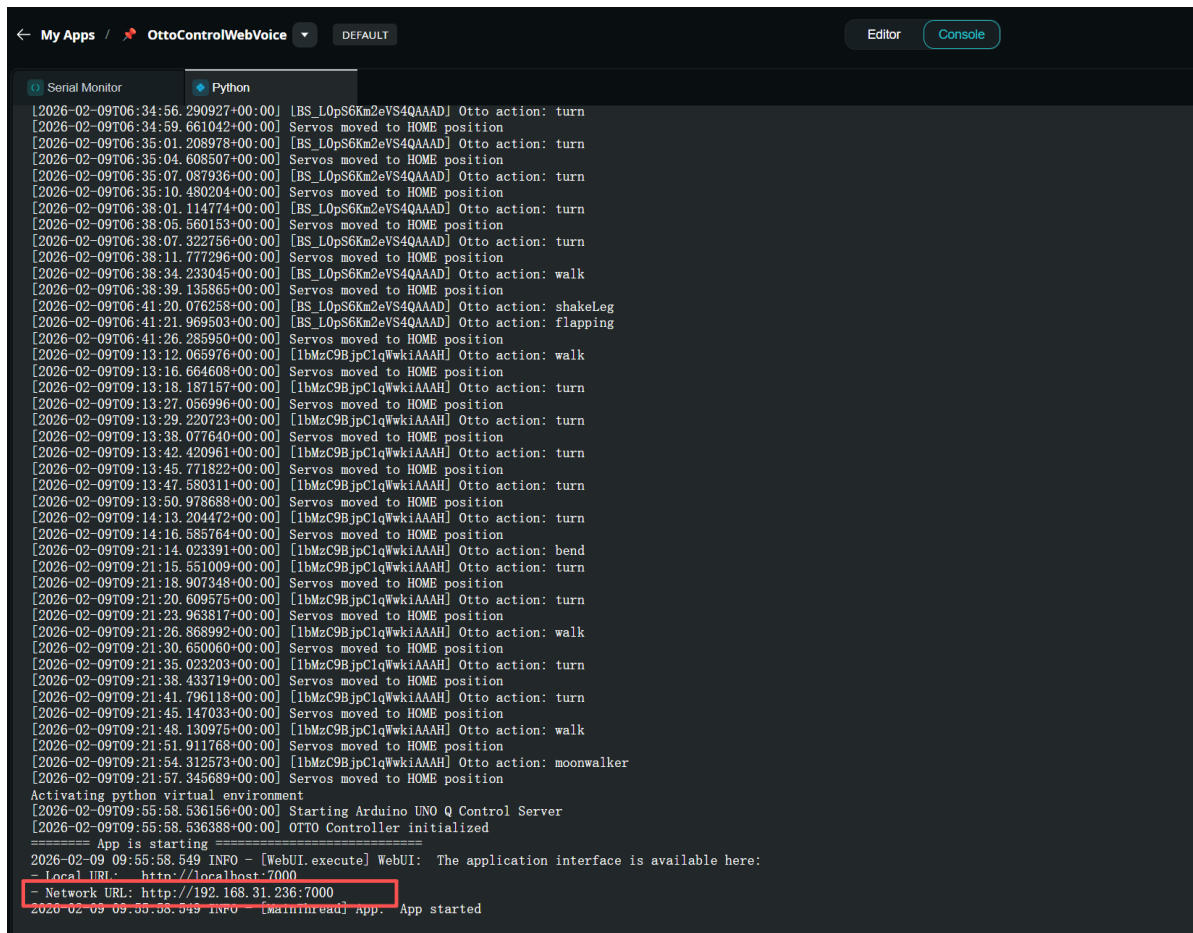
After successful connection, enter the My Apps interface to find the program. The program has two versions: one uses web-based voice output [OttoControlWebVoice], and the other uses speaker-based voice output [OttoControlUsbVoice]. Due to library and driver limitations on the development board, the speaker-based voice output quality is not ideal. It is recommended to prioritize using the web-based voice version.



Click the Run button to start the program. The development board uses OttoControlWebVoice code as the default startup program:



After successful startup, the Web access address will be printed in the log output area:

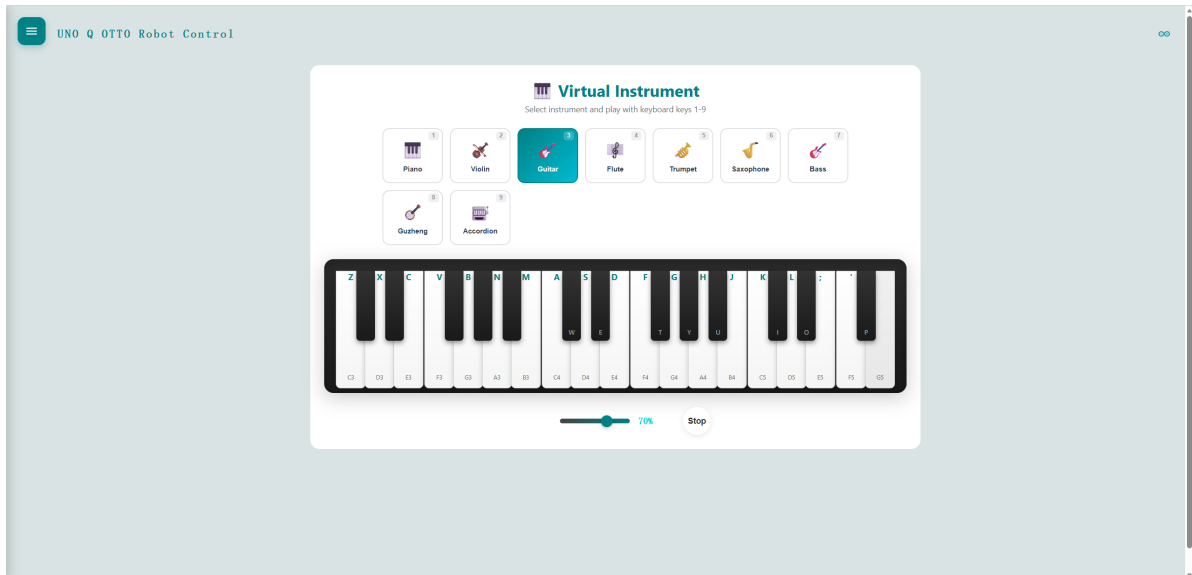


3. Web Page Operations

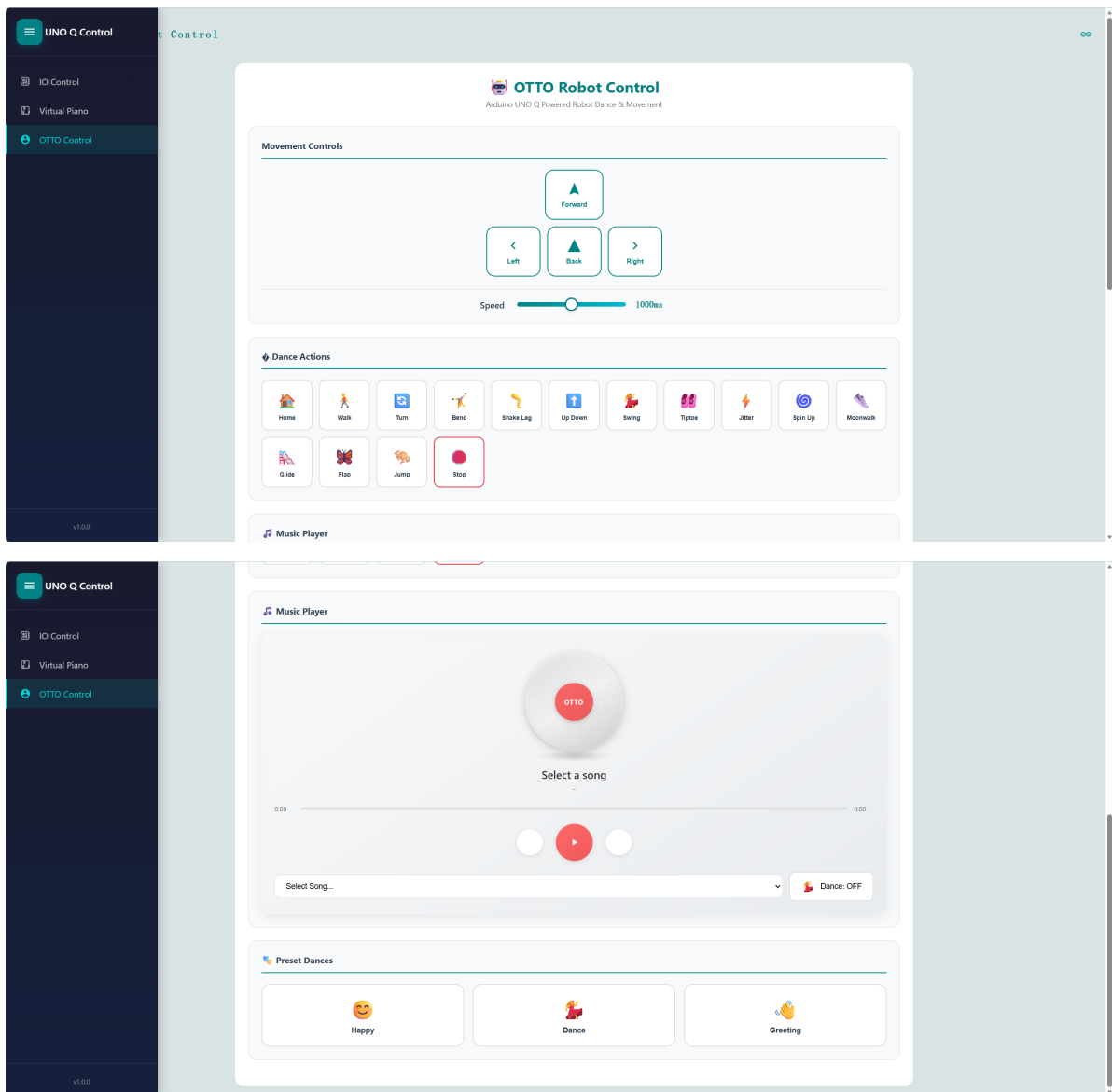
After opening the page, the first page can test whether each servo is working normally:



The second page is a virtual piano function that can test synchronized sound and motion:

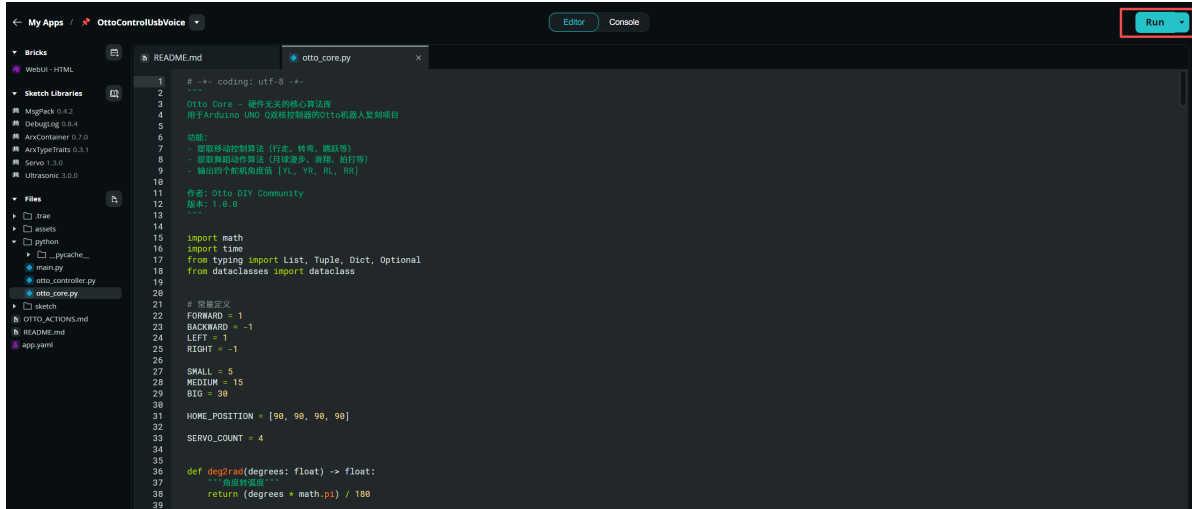


The third page contains various motion operations and song playback:



4. Speaker Version Code Description

When using the OttoControlUsbVoice program, please disconnect the C to C data cable before powering on. Due to UNO Q driver limitations, if the USB speaker cannot be recognized during power-on, after the robot has successfully powered on and can enter the Arduino App Lab software, connect the C to C data cable properly. Then you can run the speaker version code. The operation is the same as above:



```
1  # -*- coding: utf-8 -*-
2
3  Otto Core - 硬件无关的核心库
4  用于Arduino UNO C双核控制器的Otto机器人控制项目
5
6  功能:
7  - 获取移动控制算法 (行走、转弯、跳跃等)
8  - 获取舞蹈动作算法 (月球漫步、滑翔、拍打等)
9  - 输出四个舵机角度值 [FL, FR, RL, RR]
10
11  作者: Otto DIY Community
12  版本: 1.0.0
13
14
15  import math
16  import time
17  from typing import List, Tuple, Dict, Optional
18  from dataclasses import dataclass
19
20
21  # 常量定义
22  FORWARD = 1
23  BACKWARD = -1
24  LEFT = 1
25  RIGHT = -1
26
27  SMALL = 5
28  MEDIUM = 15
29  BIG = 30
30
31  HOME_POSITION = [90, 90, 90, 90]
32
33  SERVO_COUNT = 4
34
35
36  def deg2rad(degrees: float) -> float:
37  """角度转弧度"""
38  return (degrees + math.pi) / 180
39
```