Performing actions

1.1 Experimental purpose

In this course, we will learn how to control DOGZILLA 12-joint steering gear.

1.2 Experimental preparation

Dogzilla robot dog has four legs in total, and each leg is composed of three servos, a total of twelve joint servos, which can be controlled simultaneously or separately.

The functions of dogzilla Python library involved in this course are:

motor(motor_id, data)： Control the rotation angle of the steering gear.

motor_id： Corresponding to the ID number of each steering gear (ID number is marked on the steering gear of the fuselage). The meaning of ID number: the first digit represents the leg where the steering gear is located, and the second digit represents the position on the leg. From the bottom to the top, it is 1, 2 and 3.

data： Indicates the angle to be moved by the steering gear, in °.

motor_speed(speed)： Adjust the rotation speed of the steering gear, which is suitable for controlling the steering gear separately.

speed： The value range is [0, 255], 0 is the lowest speed, and 255 is the highest speed.

unload_motor(leg_id)： Unloading of single leg steering gear unloads the three steering gear on one leg without outputting torque. After that, it can be rotated by hand. It is generally used to write actions.

leg_id： The value range is 1, 2, 3 and 4, which respectively represent left front leg, right front leg, right rear leg and left rear leg.

unload_allmotor()： Unload all the steering gear, so that all the steering gear can be unloaded without output torque, and can be rotated by hand at will.***load_motor(leg_id)***： The single leg steering gear is loaded to keep the three steering gears on one leg loaded at the current position and output torque. After that, it can not be rotated by hand. It is generally used to write actions.

leg_id： The value range is 1, 2, 3 and 4, which respectively represent left front leg, right front leg, right rear leg and left rear leg.

load_allmotor()： Load all the steering gear, keep all the steering gear loaded at the current position, output torque, and then do not rotate by hand.

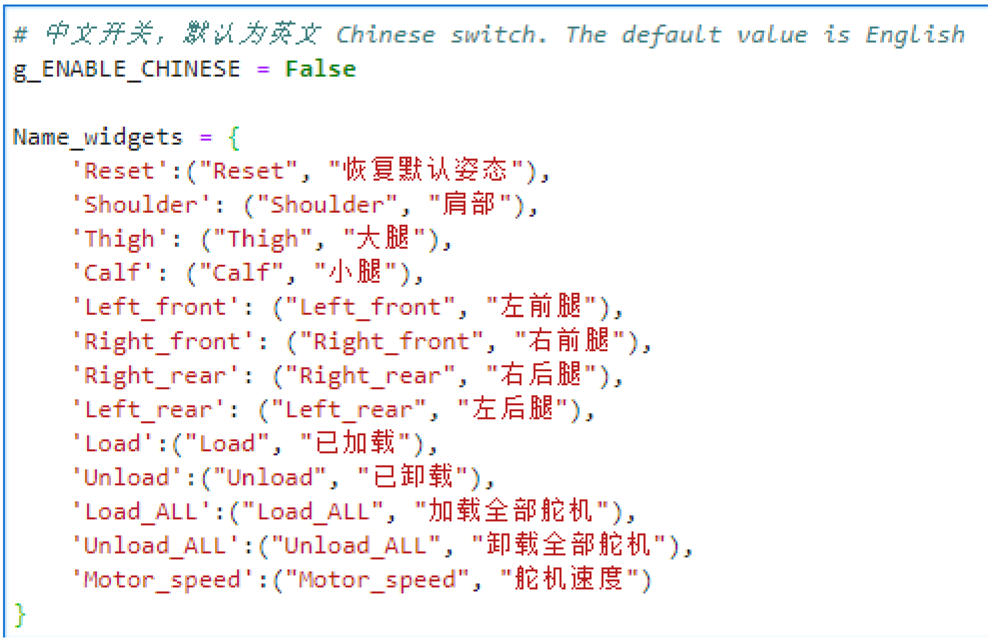
reset()*： Stop all movements and all States are restored to the initial state.

1.3 Experimental process

Open the jupyterlab client and find the code path:

```
DOGZILLA/Samples/2_Control/6.control_motor.ipynb
```

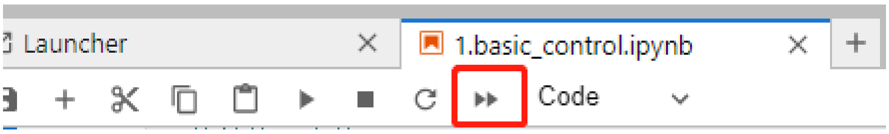
By default g_ENABLE_CHINESE=False, if you need to display Chinese, please set g_ENABLE_CHINESE=True.

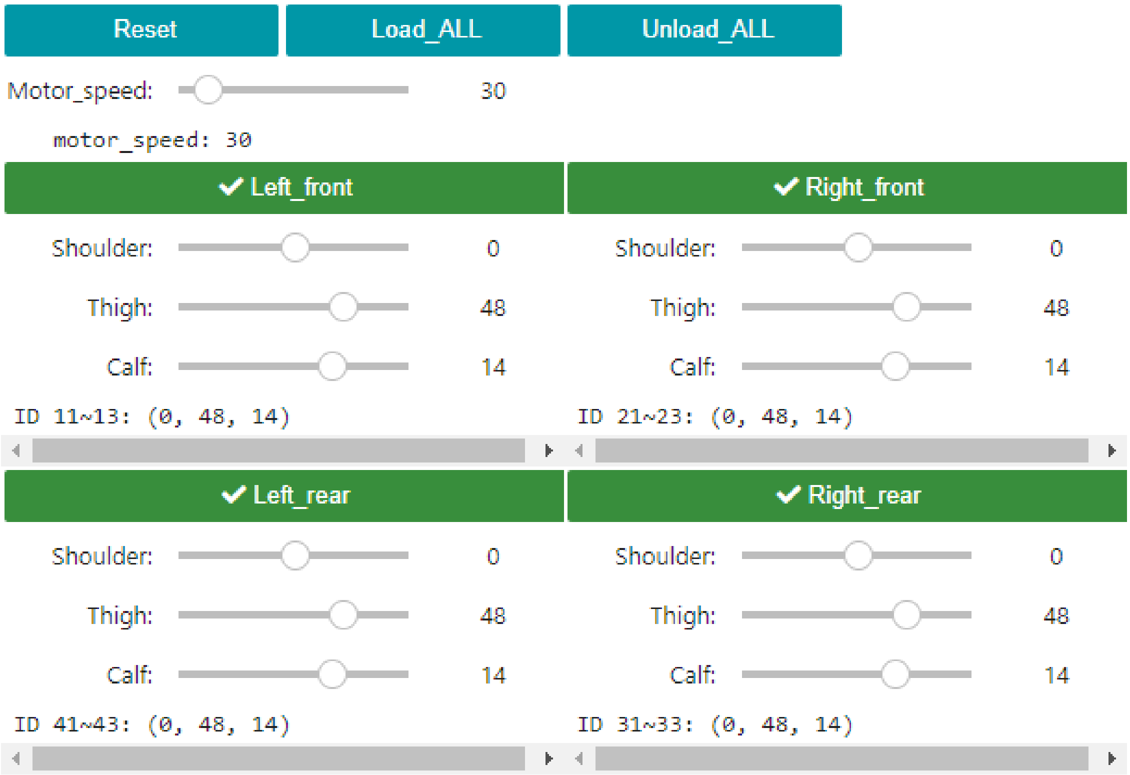


```
# 中文开关，默认为英文 Chinese switch. The default value is English
g_ENABLE_CHINESE = False

Name_widgets = {
    'Reset':("Reset", "恢复默认姿态"),
    'Shoulder': ("Shoulder", "肩部"),
    'Thigh': ("Thigh", "大腿"),
    'Calf': ("Calf", "小腿"),
    'Left_front': ("Left_front", "左前腿"),
    'Right_front': ("Right_front", "右前腿"),
    'Right_rear': ("Right_rear", "右后腿"),
    'Left_rear': ("Left_rear", "左后腿"),
    'Load':("Load", "已加载"),
    'Unload':("Unload", "已卸载"),
    'Load_ALL':("Load_ALL", "加载全部舵机"),
    'Unload_ALL':("Unload_ALL", "卸载全部舵机"),
    'Motor_speed':("Motor_speed", "舵机速度")
}
```

Click the following icon to run all cells, and then pull to the bottom to see the generated controls.





Motor_speed: 30

```
motor_speed: 30
```

Left_front — Shoulder: 0, Thigh: 48, Calf: 14

```
ID 11~13: (0, 48, 14)
```

Right_front — Shoulder: 0, Thigh: 48, Calf: 14

```
ID 21~23: (0, 48, 14)
```

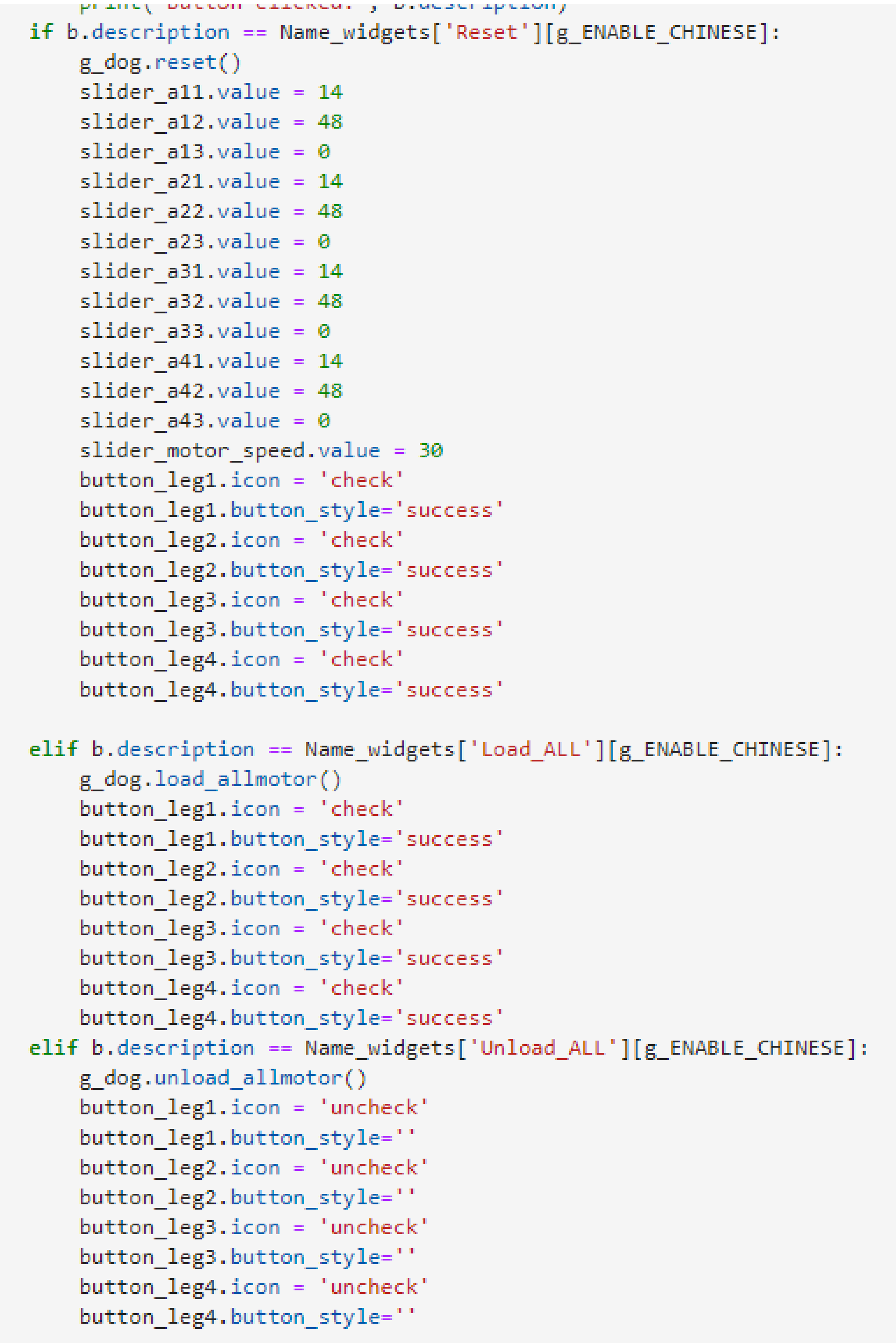
Left_rear — Shoulder: 0, Thigh: 48, Calf: 14

```
ID 41~43: (0, 48, 14)
```

Right_rear — Shoulder: 0, Thigh: 48, Calf: 14

```
ID 31~33: (0, 48, 14)
```

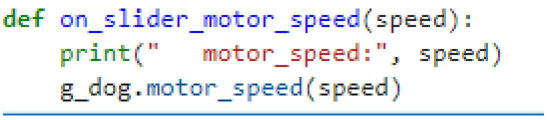
The control function of the steering gear is mainly divided into three parts.The first part is button control: the reset button is to reset the robot dog and all controls, and load_ All is to load all steering gear, unload_ All is to unload all steering gear.



```
    print("button clicked:", b.description)
if b.description == Name_widgets['Reset'][g_ENABLE_CHINESE]:
    g_dog.reset()
    slider_a11.value = 14
    slider_a12.value = 48
    slider_a13.value = 0
    slider_a21.value = 14
    slider_a22.value = 48
    slider_a23.value = 0
    slider_a31.value = 14
    slider_a32.value = 48
    slider_a33.value = 0
    slider_a41.value = 14
    slider_a42.value = 48
    slider_a43.value = 0
    slider_motor_speed.value = 30
    button_leg1.icon = 'check'
    button_leg1.button_style='success'
    button_leg2.icon = 'check'
    button_leg2.button_style='success'
    button_leg3.icon = 'check'
    button_leg3.button_style='success'
    button_leg4.icon = 'check'
    button_leg4.button_style='success'

elif b.description == Name_widgets['Load_ALL'][g_ENABLE_CHINESE]:
    g_dog.load_allmotor()
    button_leg1.icon = 'check'
    button_leg1.button_style='success'
    button_leg2.icon = 'check'
    button_leg2.button_style='success'
    button_leg3.icon = 'check'
    button_leg3.button_style='success'
    button_leg4.icon = 'check'
    button_leg4.button_style='success'
elif b.description == Name_widgets['Unload_ALL'][g_ENABLE_CHINESE]:
    g_dog.unload_allmotor()
    button_leg1.icon = 'uncheck'
    button_leg1.button_style=''
    button_leg2.icon = 'uncheck'
    button_leg2.button_style=''
    button_leg3.icon = 'uncheck'
    button_leg3.button_style=''
    button_leg4.icon = 'uncheck'
    button_leg4.button_style=''
```

The second part is to control the running speed of the steering gear: drag the motor_ speed slider controls the running speed of the steering gear. The higher the value, the faster the speed.



```
def on_slider_motor_speed(speed):
    print("   motor_speed:", speed)
    g_dog.motor_speed(speed)
```

The third part is the steering gear that controls the four legs. The button can unload and load the steering gear of which leg.

1.4 Summary

In this course, we use JupyterLab controls the steering gear of dogzilla.

The steering gear can be controlled individually or multiple at a time. After unloading the steering gear, you can turn the steering gear angle by hand. After the steering gear is loaded, the output torque can not be turned by hand.