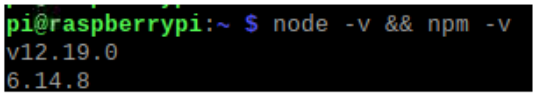
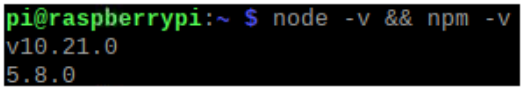
4. Jupyter widgets configuration tutorial

1. Install ipywidgets

1. Refer to `1. Jupyter lab environment setup tutorial` to install Jupyter lab.
2. Check whether `node` and `nmp` are installed. If the version number is displayed, it means that they are installed. You can continue to operate (the second part **Install Node.js** can be skipped directly). Otherwise, you should jump to **Second, Install Node.js** first, install Node.js and then operate below. (As shown in the figure, both versions can be used directly)

```
node -v && npm -v
```





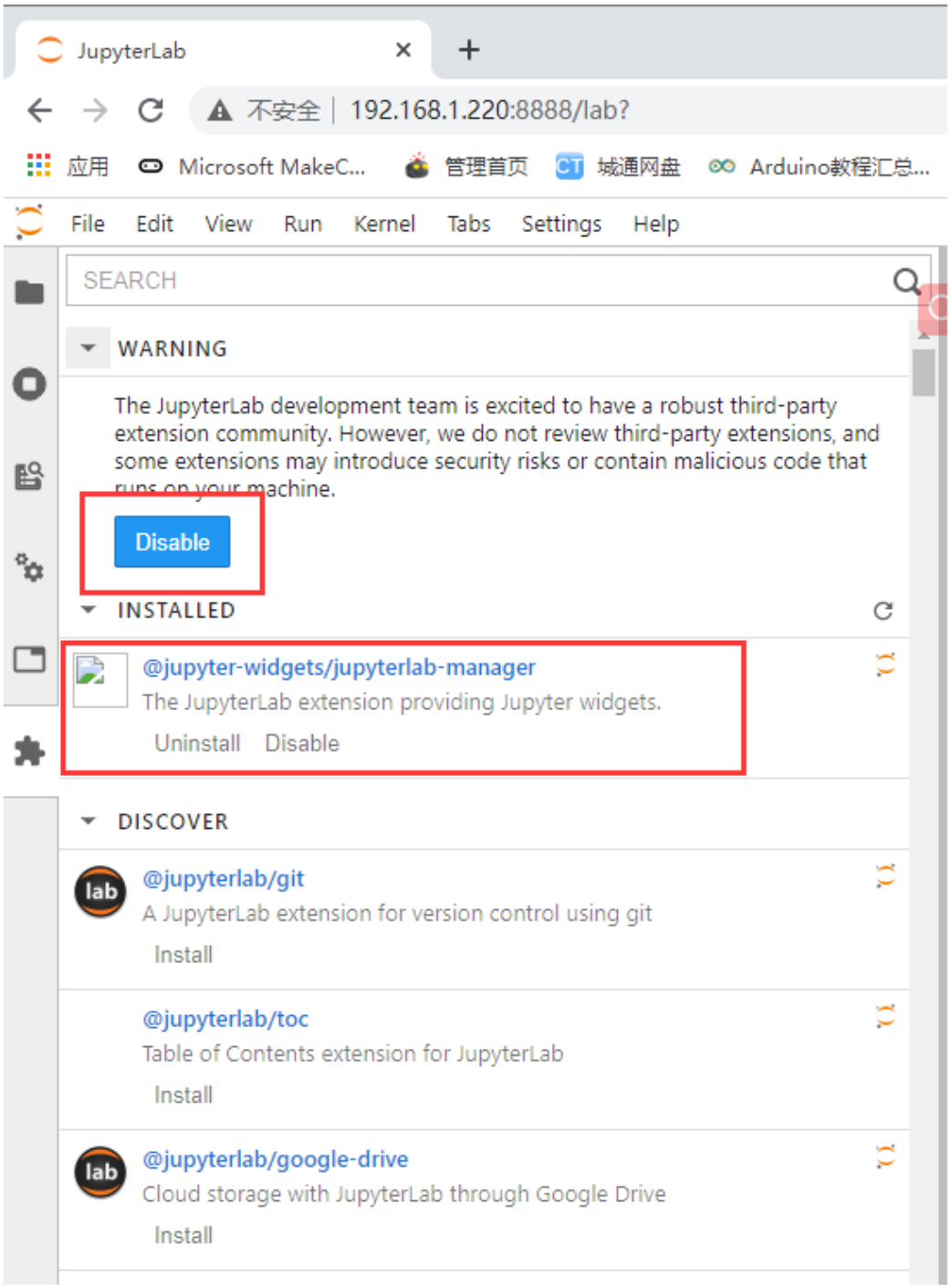
3. Install @jupyter-widgets/jupyterlab-manager (this step needs to be done in jupyter lab)

First, enable the third-party extension community

Then install (install) @jupyter-widgets/jupyterlab-manager management plug-in

After the installation is complete, the plug-in will appear in the **INSTALLED** (installed) area, as shown below.

Generally, after the installation is complete, you will be prompted to re-**Build JupyterLab**. This process will be relatively long, and there will be no prompt after success. It is recommended to wait for about 2-3 minutes before re-entering jupyterlab. If there is no prompt **Build JupyterLab**, it means that the build is successful.



3. Install `ipywidgets`

```
pip3 install ipywidgets
```

4. Enable `widgetsnbextension`

```
jupyter nbextension enable --py widgetsnbextension
```

5. Delete temporary and static directories

```
jupyter lab clean
jupyter lab path
```

6. Restart

```
sudo reboot
```

2. Install Node.js

1. Check the system architecture, as shown below.

```
uname -a
```

```
pi@raspberrypi:~ $ uname -a
Linux raspberrypi 5.4.51-v7l+ #1333 SMP Mon Aug 10 16:51:40 BST 2020 armv7l GNU/Linux
```

2. Enter the Nood.js official website [download page] and find the corresponding version to download

3. Unzip the downloaded compressed file (the current Nood.js official website provides version 12.19.0. If the subsequent version changes, please follow the actual situation)

```
xz -d node-v12.19.0-linux-armv7l.tar.xz
tar -xavf node-v12.19.0-linux-armv7l.tar
```

4. Delete the existing `/usr/bin.node` in the system

```
sudo rm -rf /usr/bin/node
```

5. Move the binary package to `/usr/local/node`

```
sudo mv ./node-v12.19.0-linux-armv7l /usr/local/node
```

6. Create a soft link for `node` and `npm`

```
sudo ln -s /usr/local/node/bin/node /usr/bin/node
sudo ln -s /usr/local/node/bin/npm /usr/bin/npm
```

3. Verification

1. Open and enter jupyter lab

```
jupyter lab
```

2. Enter Notebook

3. Copy and paste the sample code

```
from __future__ import print_function
from ipywidgets import interact, interactive, fixed, interact_manual
import ipywidgets as widgets
def f(x):
return x
interact(f, x=10);
```

4. Use the shortcut key to start running. If the result is as follows, the Jupyter Widgets configuration is complete

Enter+Shift

[1]:
```
from __future__ import print_function
from ipywidgets import interact, interactive, fixed, interact_manual
import ipywidgets as widgets
def f(x):
    return x
interact(f, x=10);
```