

VG710 Python APP Development Guide

1. Introduction

This document briefly explains how to develop and deploy Python APPs based on VG710 **Python SDK** and **Visual Studio Code** IDE.

Python at its core is an interpreted, object-oriented, high-level programming language with dynamic semantics. So it's simple and easy to learn.

Python also comes with extensive standard libraries, numerous third-party modules(PyPI), which made its reputation for productive development in lots of areas.

By reusing proven standard libraries and third-party modules, you can accomplish more complicated tasks with less code and less development and debugging time.

VG710 has thus embedded up to date Python 3.7.x support for users to extend its ability and achieve their goals easily.

Programming an embedded system was once a tedious low level task: setup cross-compiling development environment, coding with C/C++, cross-compiling and building firmware, erasing and reprogramming flash memory...

Users need a decent experience to program and deploy an embedded system. After all, customizable is necessary, and made it easy is even more important.

The awesome **Visual Studio Code** IDE has made it easy for you to program your VG710 device.

By using Visual Studio Code IDE and its powerful remote development plug-ins, you import, edit, debug and package your APPs all within it. It just makes life much more easier than ever before.

Since Visual Studio Code IDE is cross-platform and runs everywhere, you can download and install it on your Windows PC, MAC, or even Ubuntu desktop.

With Python & SFTP plug-ins, you could easily code in python, and establish connection to VG710 to debug and package your APPs.

After you have packaged your APPs, you could redistribute and deploy them via VG710 WebUI or Device Manager.

At this deployment stage, all you need is the packaged APPs and your favorite web browser to access VG710 WebUI or our **Device Manager**.

During your development process, we recommend you first deploy via VG710 WebUI to make sure your APPs work as expected.

Once you have the confidence that your APPs are OK for mass deployment, you could use our Device Manager to import and deploy them on lots of VG710 devices.

2. Overview

The remaining document is divided into several topics, you can easily read your interested topics.

Install Python SDK

Our Python SDK provides a Python environment and necessary Python libraries to run your Python APPs.

This SDK should be installed on VG710 so you can debug, package and run your APPs.

You can refer to [3.2 Install Python SDK](#) for more details.

Install IDE

We use **Visual Studio Code** with some necessary plug-ins as our IDE. You should install it on your PC.

Using it, you can easily import, edit, debug and package your APPs.

You can refer to [3.4 Install IDE](#) and [3.5 Install IDE plug-ins](#) for more details.

APP Development

We provide a sample APP to guide you to understand the steps needed to develop your APPs.

By using VS Code IDE, you can:

- Import your APP, please refer to [4.1 Import HelloWorld](#)
- Configure your APP, please refer to [4.2 Configure HelloWorld](#)
- Turn on IDE debugging, please refer to [3.3 Enable APP Manager & IDE debug](#)
- Debug your APP, please refer to [4.3 Debug HelloWorld](#) (This step requires connection to VG710)

APP Packaging

After you have developed your APP and completed its debugging, you need to package it for further deployment.

By using VS Code IDE, you can:

- Package your APP, please refer to [4.4 Package HelloWorld](#) (This step requires connection to VG710)

APP Deployment

After you have packaged your APP, you can deploy it on your VG710 gateways.

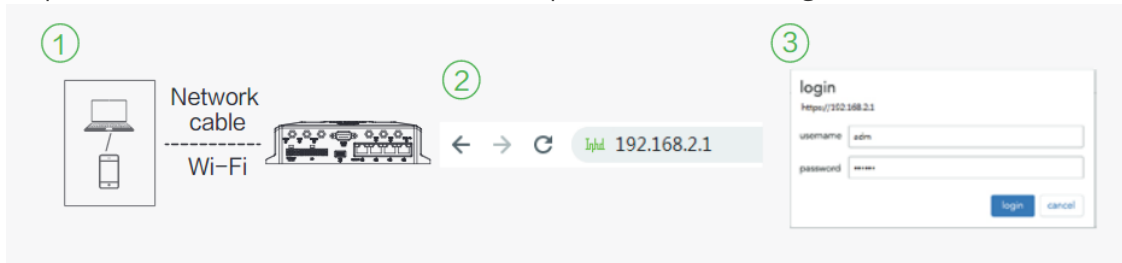
From VG710 WebUI or our Device Manager, you can install your packaged APPs on your VG710 gateways, run them and check their logs and status.

You can refer to [5. APP management](#) for more details.

3. Setup the development environment

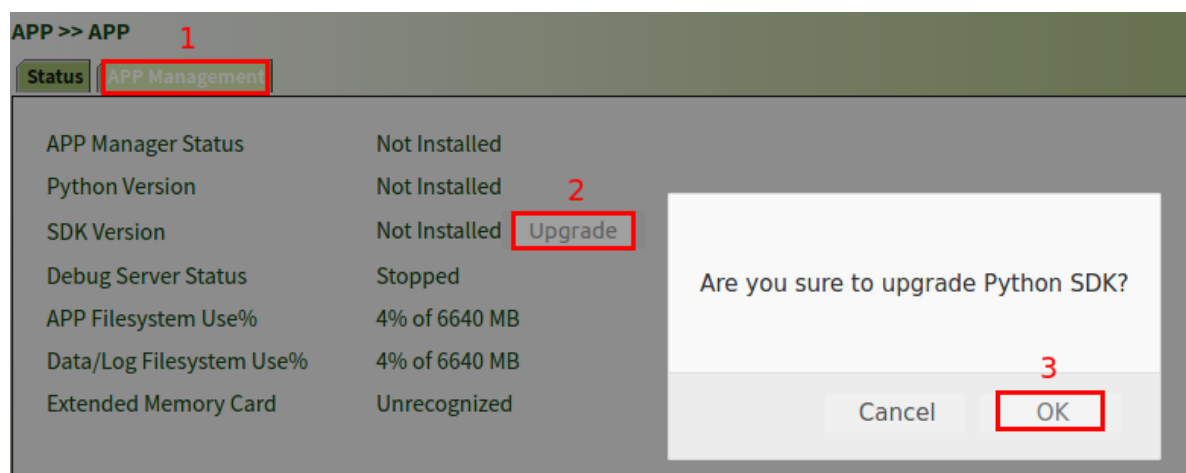
3.1 Access VG710 WebUI

- Step 1: Connect to VG710 through Ethernet cable or Wi-Fi (see the SSID and key on the nameplate).
- Step 2: Enter the default device IP address 192.168.2.1 in the address bar of the web browser to open the login page.
- Step 3: Enter the default username adm and password 123456 to go to the web interface



3.2 Install Python SDK

- Step 1: Switch to APP Status page
- Step 2: Click **Upgrade** button, choose Python SDK package you downloaded.
- Step 3: Click **OK** to confirm the operation, then wait for the upgrading process to complete



3.3 Enable APP Manager and IDE debug

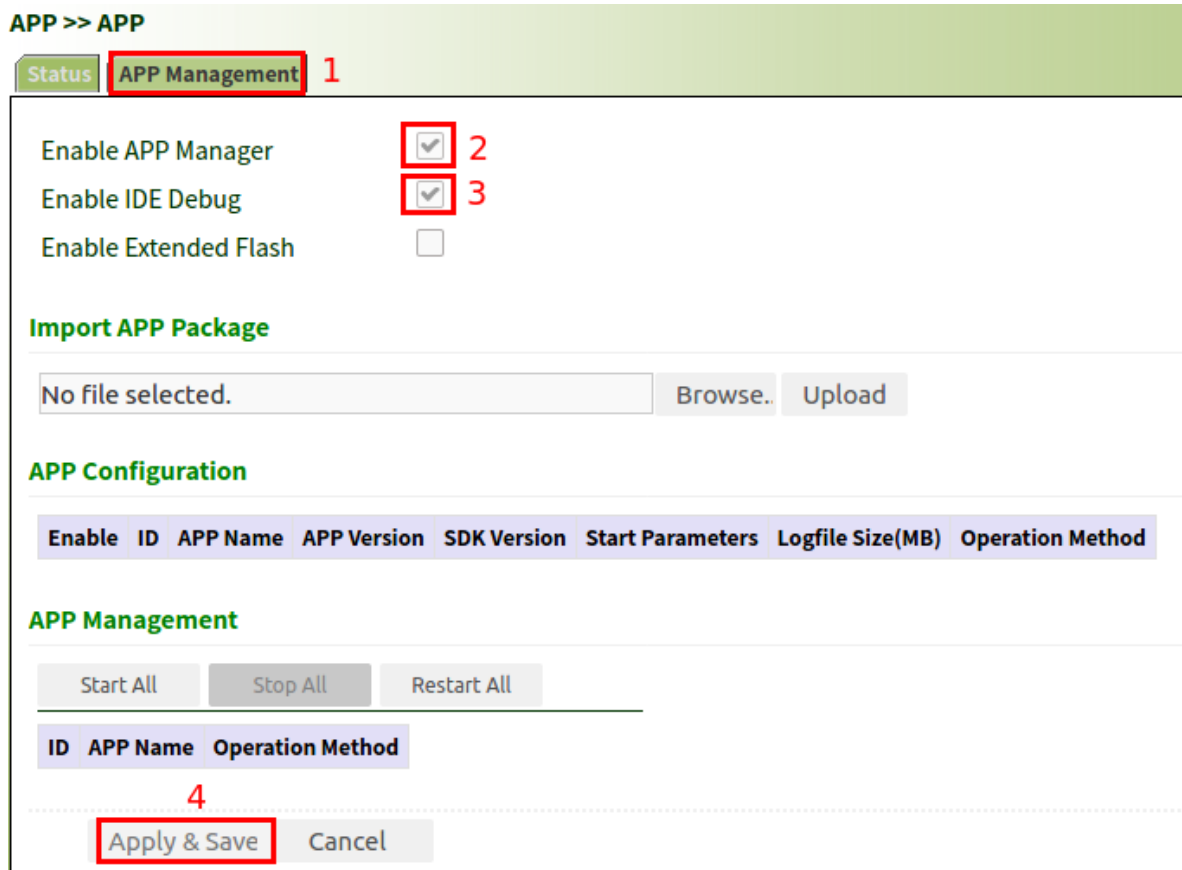
APP manager is used to run and manage your APPs.

IDE debug is needed at development stage to synchronize files with VG710 to debug and package your APPs.

Steps are as follows:

- Step 1: switch to APP Management page
- Step 2: check the check-box of **Enable IDE Debug**
- Step 3: check the check-box of **Enable APP Manager**
- Step 4: click **Apply&Save** and wait for the operation to complete

Note: IDE Debug will be disabled after each reboot, so you need to re-enable it for your development.

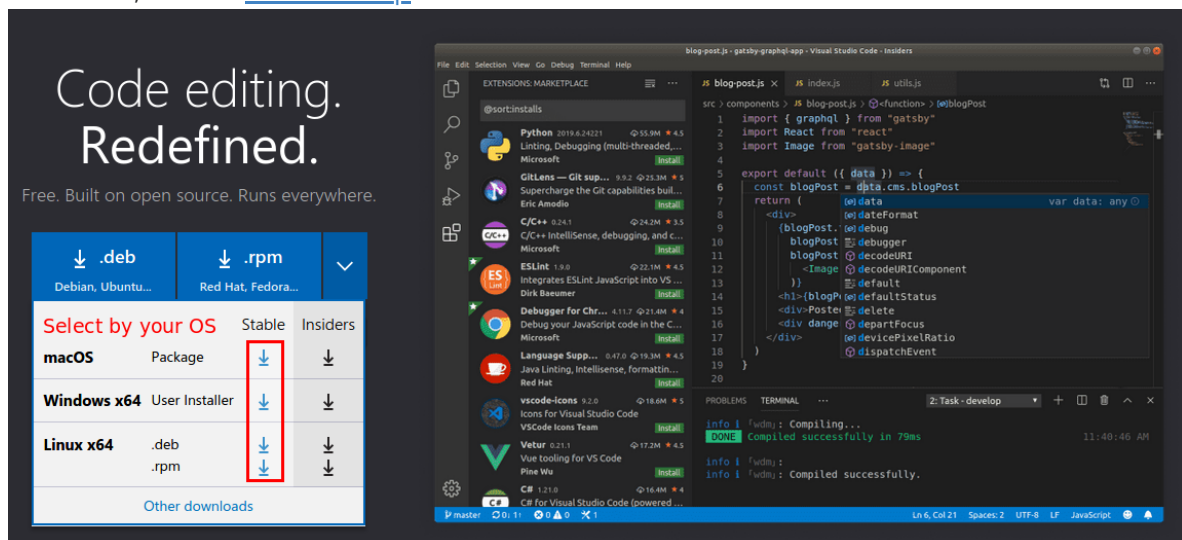


3.4 Install IDE

We use **Visual Studio Code** editor as IDE to develop python APP.

Users need to access code.visualstudio.com to download and install this editor.

For details, refer to [VScode Setup](#).



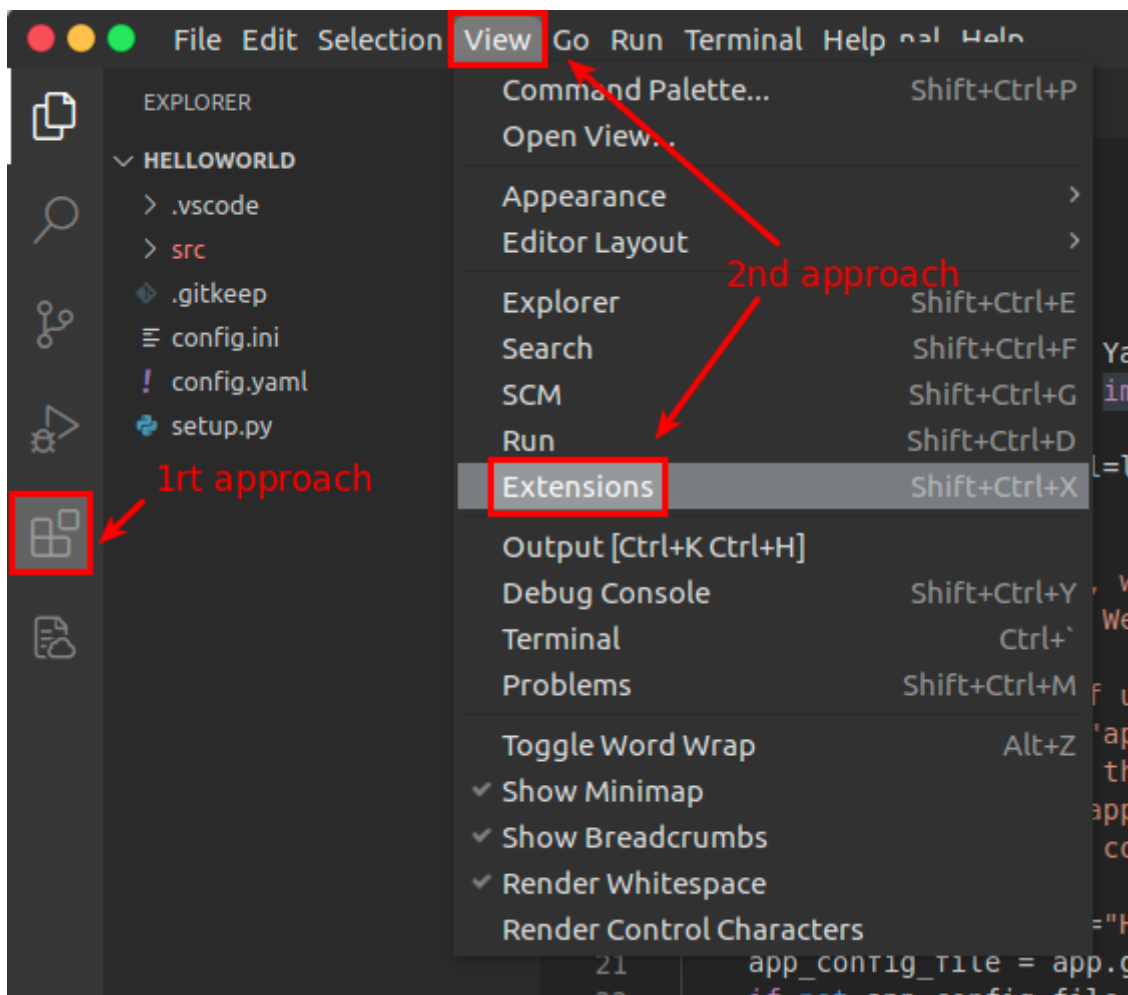
3.5 Install IDE plug-ins

For better and efficient APP development, you need to install **python** and **sftp** plug-ins. Steps are as follows:

- Browse for extensions

You can browse and install extensions from within VS Code.

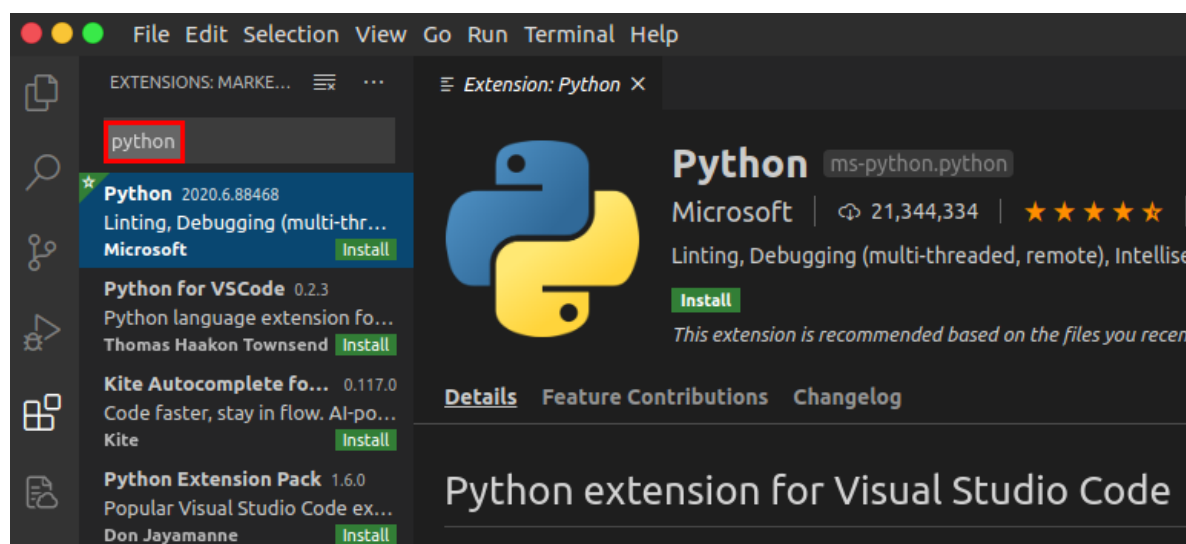
Bring up the Extensions view by clicking on the Extensions icon in the Activity Bar on the side of VS Code or the **View: Extensions** command ([Ctrl+Shift+X](#)).



- Search for an extension

You can clear the Search box at the top of the Extensions view and type in the name of the extension, tool, or programming language you're looking for.

For example, typing 'python' will bring up a list of Python language extensions:

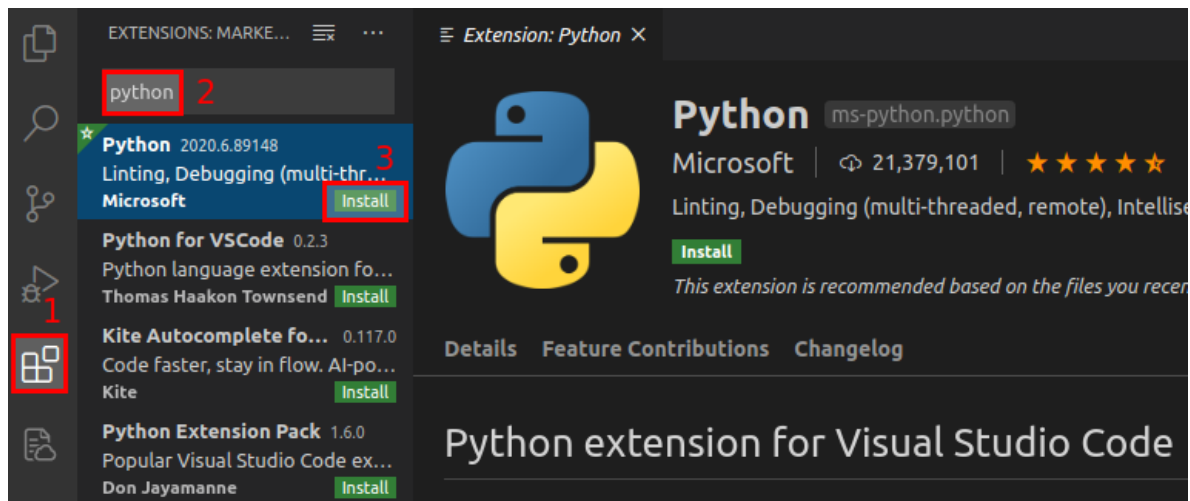


- Install an extension

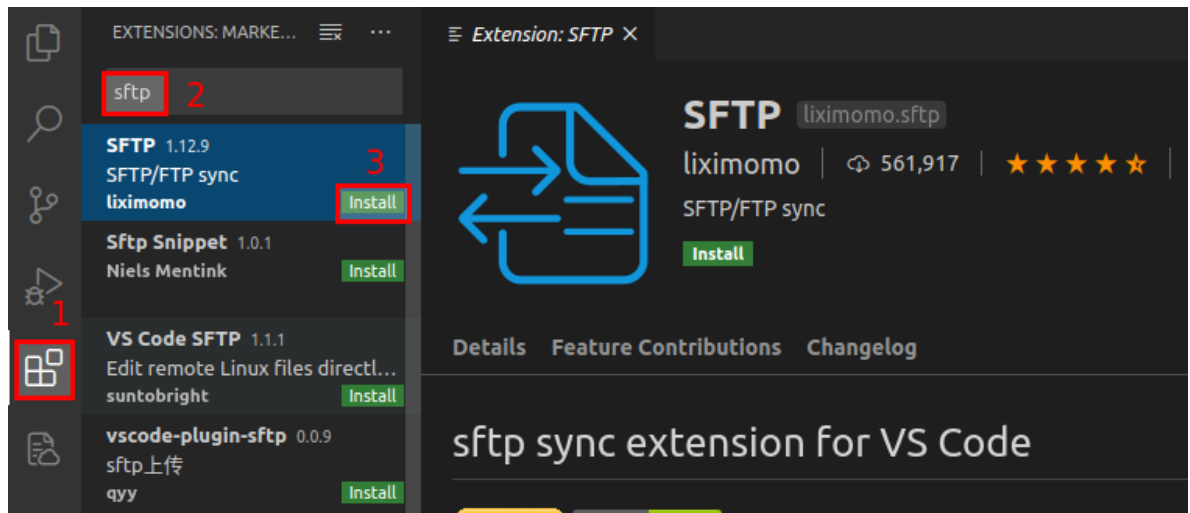
To install an extension, click the **Install** button. Once the installation is complete, the **Install** button will change to the **Manage gear** button.

Examples of plug-in installation required for APP development are as follows:

Install **python** plug-in



Install **sftp** plug-in



4. HelloWorld example

We provide a template project **HelloWorld** to demonstrate how to develop an APP based on our Python SDK.

4.1 Import HelloWorld

- Open **File: Open Folder**(`Ctrl+K Ctrl+O`) command to import **HelloWorld** project.

This project includes a complete development framework. its directory structure is as follows:

```
dengzt@Vostro3470: /home/dengzt/workspace/APP_source/HelloWorld$ tree -a
.
├── config.yaml
├── setup.py                APP info: name, version, author, SDK version, etc
├── src                    APP source code
│   ├── main.py
│   └── parse_config.py
├── .vscode
│   └── sftp.json          STFP configuration to access VG710
└── 2 directories, 5 files
```

4.2 Configure HelloWorld

- In HelloWorld/setup.py, you need to set APP's name, version, author and so on. These information are used to package APP.

```
1  ...
2  from setuptools import setup, find_packages
3  setup(name='HelloWorld', #APP name
4        sdk_version='0.2.0', #SDK version
5        version='1.0.0', #APP version
6        author='Inhand', #APP author
7  ...
```

- In HelloWorld/src/main.py, you also need to set APP's name, which must be the same as name in setup.py.

```
1  ...
2  app = APPConfig(name="HelloWorld") #APP name
3  app_config_file = app.get_app_cfg_file()
4  ...
```

- In HelloWorld/.vscode/sftp.json, you need to configure host, username, password and remotePath to synchronize with VG710.

```
1  {
2      "name": "Debug Server",
3      "host": "10.5.16.197", # VG710 IP address
4      "protocol": "sftp",
5      "port": 222, # SFTP port
6      "username": "pyuser", # SFTP username, the default is pyuser, please
7      "password": "VG7101234568888", # SFTP password, it is the serial number
8      "remotePath": "/var/app/HelloWorld", # APP install path in VG710
```

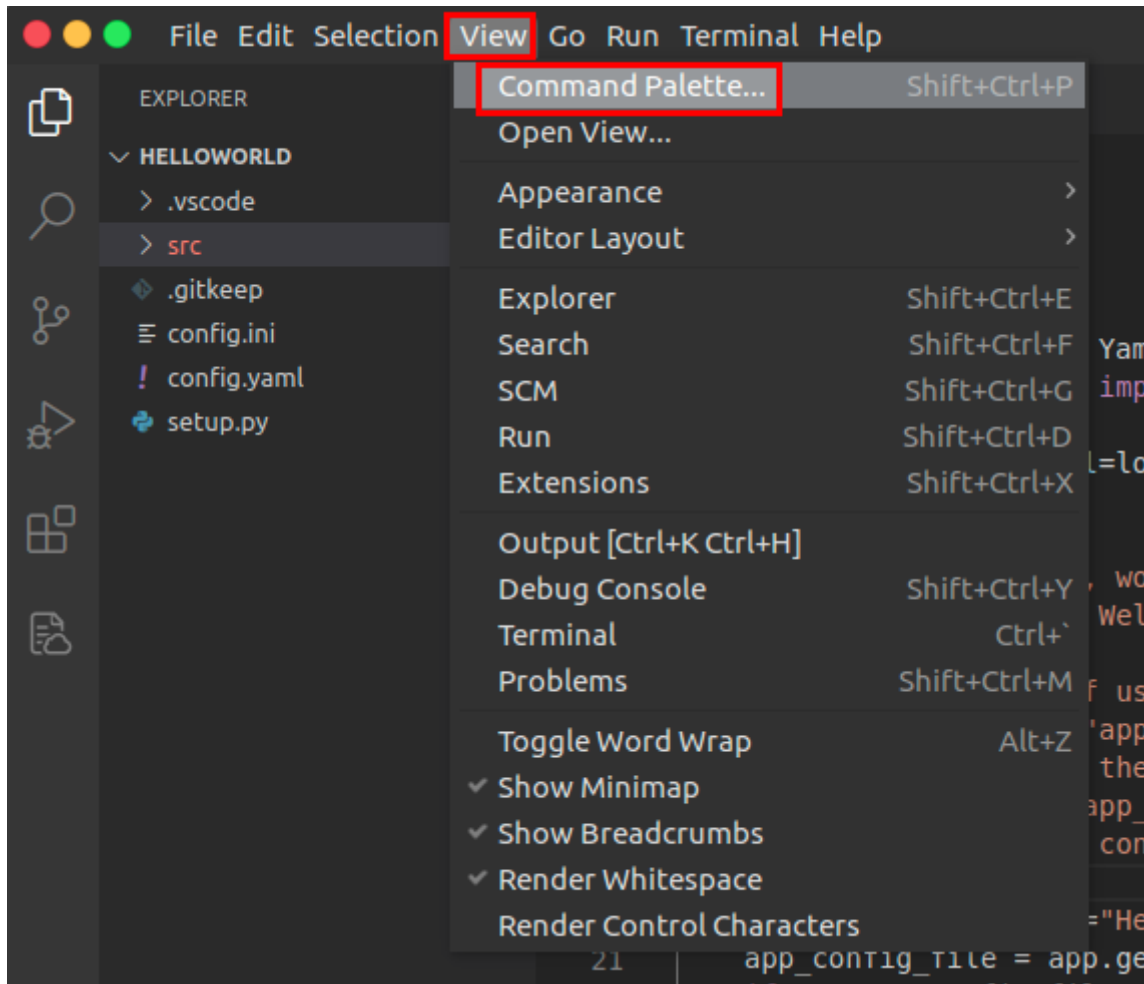
```

9      "uploadOnSave": true,
10     "ignore": [
11         ".vscode",
12         ".git",
13         ".DS_Store"
14     ]
15 }

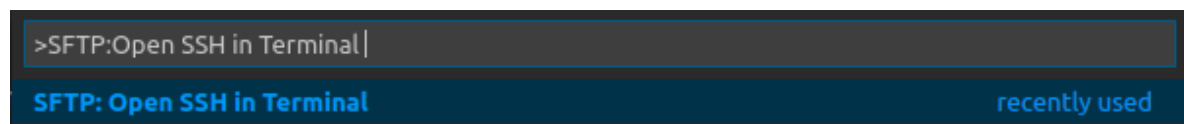
```

4.3 Debug HelloWorld

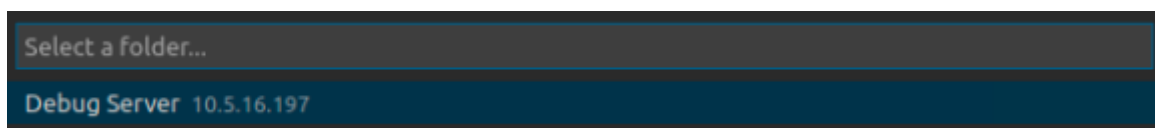
In the VS Code menu bar, open **View: Command Palette** (Shift+Ctrl+P) command.



Then type **SFTP:Open SSH in Terminal** in the pop-up box.



Press **enter**, then the drop-down list will prompt the following information:



After confirming that it is correct, press **enter**, and then follow the steps below on the VS Code **Terminal** window:

- Step 1: When connecting for the first time, the "Terminal" will prompt you whether you want to continue to connect, enter "yes" and press **enter** to continue;

- Step 2: Then the "Terminal" window will prompt you to enter the password, the password is the serial number of your VG710 device;
- Step 3: After entering the password, press `enter`. Once the "Terminal" prompts the following information, it means that VS Code has successfully established an SFTP connection with your VG710.

```
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL
```

2: Debug Server + [] [X] ^ X

```
dengzt@Vostro3470:/home/dengzt/workspace/APP_source>HelloWorld$ ssh -t pyuser@10.5.16.197 -p 222  
The authenticity of host '[10.5.16.197]:222 ([10.5.16.197]:222)' can't be established.  
RSA key fingerprint is SHA256:QwGsxufuuu/8nAICwMMNo0sKx7XU2TQJdja5tkXSkuUs.  
Are you sure you want to continue connecting (yes/no)? yes  
Warning: Permanently added '[10.5.16.197]:222' (RSA) to the list of known hosts.  
pyuser@10.5.16.197's password:   
  
BusyBox v1.26.2 (2020-05-29 08:42:51 CST) built-in shell (ash)  
  
# # ##### # ###  
# # # # # # #  
# # # # # # #  
# # ##### # # # #  
# # # # # # #  
# # # # # # #  
# ##### # #####  
  
-----  
Vehicle Gateway from InHand Networks  
-----  
  
/tmp/app $
```

FAQ 1: Why do i encounter ssh connection refused?

```
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL 1: Debug Server + [ ] [X] ^ X
dengzt@Vostro3470:/home/dengzt/workspace/APP_source/HelloWorld$ ssh -t pyuser@10.5.16.197 -p 222
ssh: connect to host 10.5.16.197 port 222: Connection refused
dengzt@Vostro3470:/home/dengzt/workspace/APP_source/HelloWorld$
```

Please ensure you have enabled **Enable IDE Debug** as [3.3 Enable APP Manager & IDE debug](#).

FAQ 2: Why do i encounter "WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED"?

```

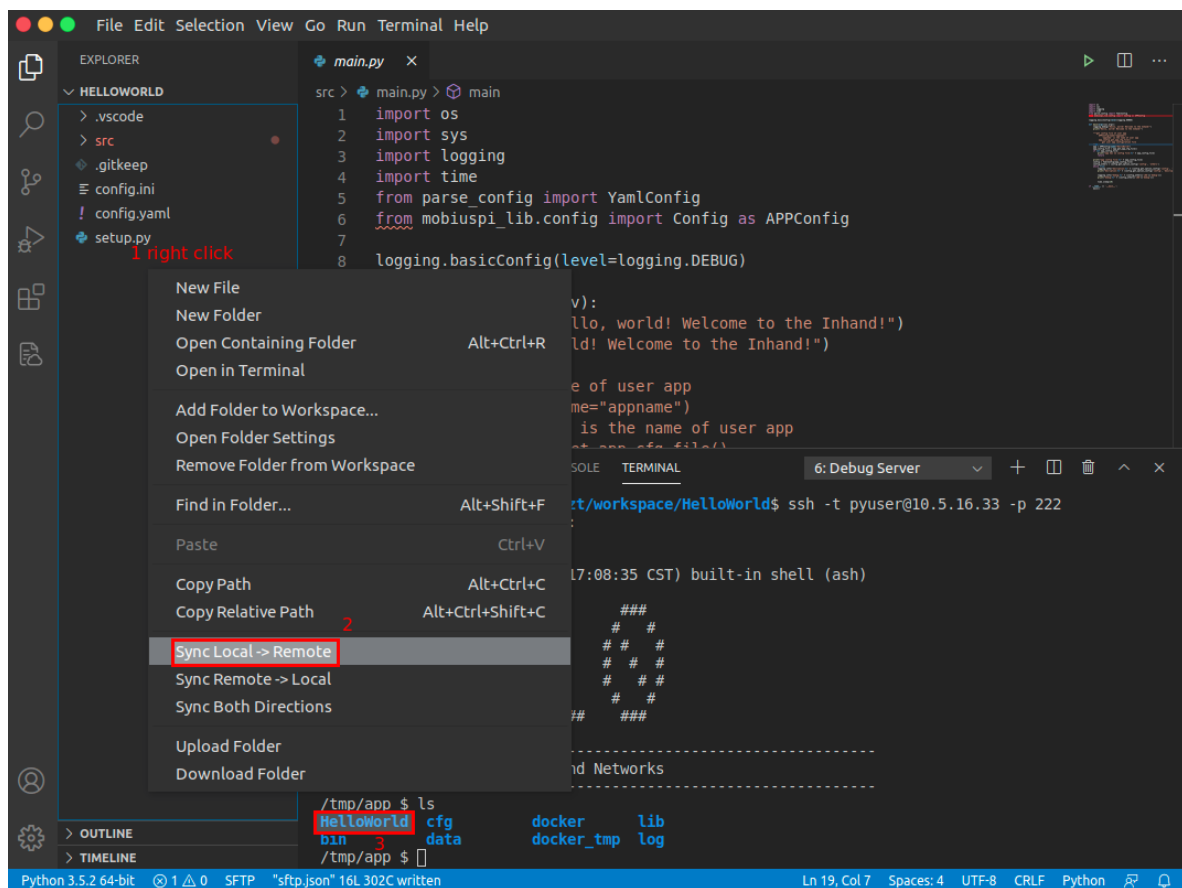
PROBLEMS 1 OUTPUT TERMINAL DEBUG CONSOLE

wucl@wucl-Vostro-3902:~/Desktop/pysdk/app_source/HelloWorld$ ssh -t pyuser@10.5.16.197 -p 222
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!    @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that a host key has just been changed.
The fingerprint for the RSA key sent by the remote host is
SHA256:+mAl1dDgpdKu+za/cf4EeJ05r1Q2J3AJYgtX0bYR4A0.
Please contact your system administrator.
Add correct host key in /home/wucl/.ssh/known_hosts to get rid of this message.
Offending RSA key in /home/wucl/.ssh/known_hosts:1
  remove with:
  ssh-keygen -f "/home/wucl/.ssh/known_hosts" -R "[10.5.16.197]:222"
RSA host key for [10.5.16.197]:222 has changed and you have requested strict checking.
Host key verification failed.
wucl@wucl-Vostro-3902:~/Desktop/pysdk/app_source/HelloWorld$ ssh-keygen -f "/home/wucl/.ssh/known_hosts" -R "[10.5.16.197]:222"
# Host [10.5.16.197]:222 found: line 1
/home/wucl/.ssh/known_hosts updated.
Original contents retained as /home/wucl/.ssh/known_hosts.old
wucl@wucl-Vostro-3902:~/Desktop/pysdk/app_source/HelloWorld$
wucl@wucl-Vostro-3902:~/Desktop/pysdk/app_source/HelloWorld$

```

The above warning means the fingerprint of VG710 has changed. Solution is to use the **ssh-keygen** utility to delete the fingerprint from your **known_hosts** file as below picture shows, then repeat steps 1-3 above to login again.

After completing code modification, right click the mouse in the blank area of "EXPLORER" on the left side of VS Code, select **Sync Local->Remote** to synchronize the code to VG710. As shown below:



- Step 4: change to APP source directory to run your APP.

```
1 tmp/app # cd HelloWorld/src/
2 tmp/app/HelloWorld/src # python3 main.py
```

Press `Ctrl+C` to terminate the debugger.

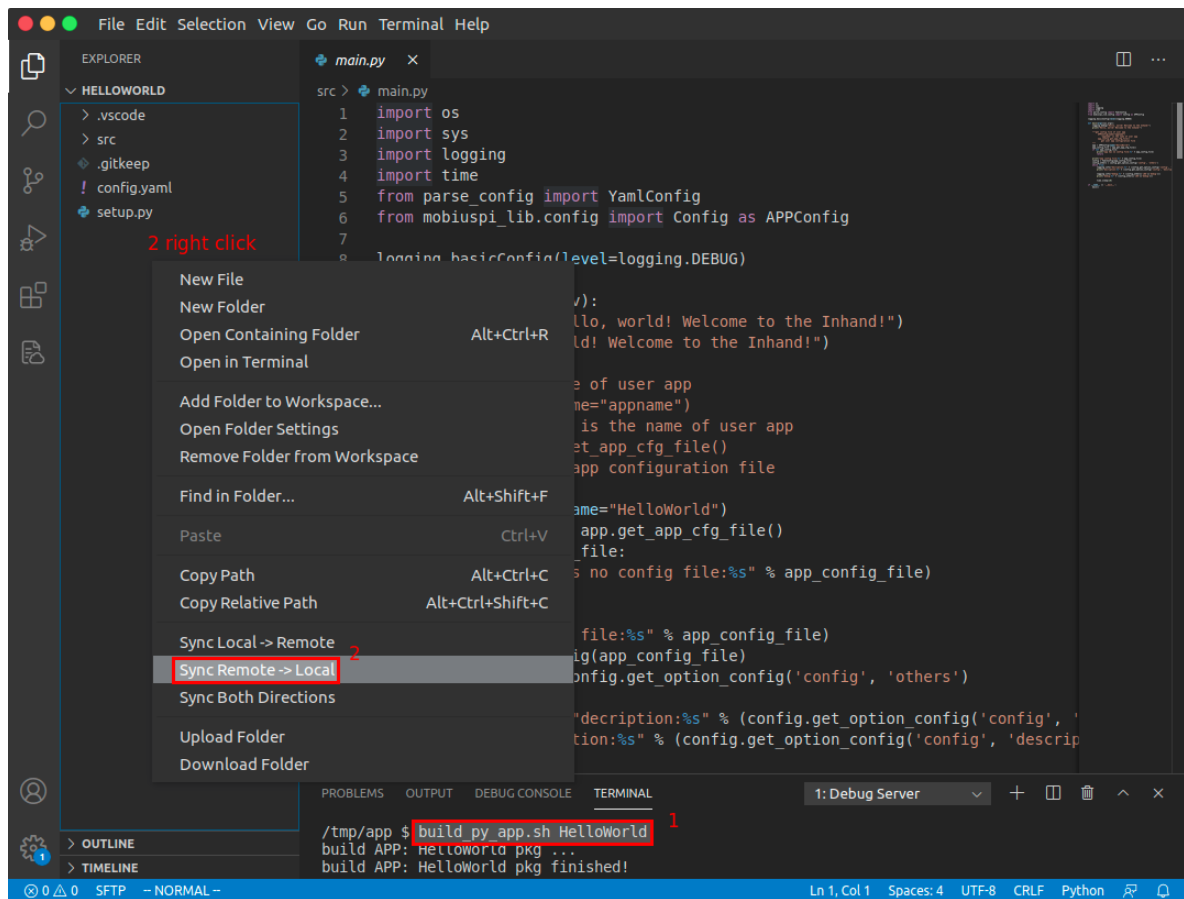
4.4 Package HelloWorld

After APP has been successfully developed, you can package it for deployment on other VG710 devices.

Since the entire packaging process need to be completed in VG710, please excute the following commands in the terminal:

```
1 /tmp/app/HelloWorld $ cd ..           # Return to APP top directory
2 /tmp/app $ build_py_app.sh HelloWorld  # Perform APP packaging
3 build APP:HelloWorld pkg!
4 build APP:HelloWorld pkg finished!    # Finish packaging
```

After successfully packaging, right click the mouse in the blank area of "EXPLORER" on the left side and select **Sync Remote->Local** to synchronize the code to local computer. After successfully transferred, a *.tar.gz file will be added to the build directory, steps as show below:



Project tree is as follows:

```

1 | .
2 | └─ build
3 |   └─ HelloWorld-V1.0.0.tar.gz
4 | └─ config.yaml
5 | └─ HelloWorld.info
6 | └─ setup.py
7 | └─ src
8 |   └─ main.py
9 |   └─ parse_config.py
10 |     └─ __pycache__
11 |       └─ parse_config.cpython-37.pyc
12 | └─ .vscode
13 |   └─ sftp.json
14 |
15 | 4 directories, 8 files
16 |

```

As above, the HelloWorld-V1.0.0.tar.gz package can be used to deploy on other VG710 devices.

5. APP management

After you have packaged your APP, you could deploy it via VG710 WebUI.

5.1 Import APP

Go to WebUI **APP Management** to import your APP package. Steps are as follows:

- Step 1: switch to APP Management
- Step 2: choose the python APP package(e.g. HelloWorld-V1.0.0.tar.gz)
- Step 3: click **Upload** to finish import

APP >> APP

Status **APP Management** 1

Enable APP Manager ☒

Enable IDE Debug ☒

Enable Extended Flash ☐

Import APP Package 2 3

No file selected. Browse. Upload

APP Configuration

Enable	ID	APP Name	APP Version	SDK Version	Start Parameters	Logfile Size(MB)	Operation Method
--------	----	----------	-------------	-------------	------------------	------------------	------------------

APP Management

Start All Stop All Restart All

ID	APP Name	Operation Method
----	----------	------------------

Apply & Save Cancel

The result of successful import is as follows:

APP >> APP

Status **APP Management**

Enable APP Manager ☒

Enable IDE Debug ☒

Enable Extended Flash ☐

Import APP Package

No file selected. Browse. Upload

APP Configuration

Enable	ID	APP Name	APP Version	SDK Version	Start Parameters	Logfile Size(MB)	Operation Method
<input checked="" type="checkbox"/>	1	HelloWorld	1.0.0	0.2.0		1	Import Conf Export Conf Export App Uninstall

APP Management

Start All Stop All Restart All

ID	APP Name	Operation Method
----	----------	------------------

Apply & Save Cancel

5.2 Run APP

Go to **APP Management** to run your imported APP.

- Step 1: check the **Enable** check-box
- Step 2: click **Apply&Save** , then wait for operation to complete

APP >> APP

Status **APP Management**

Enable APP Manager ☒
Enable IDE Debug ☒
Enable Extended Flash ☐

Import APP Package

No file selected. Browse.. Upload

APP Configuration

Enable	ID	APP Name	APP Version	SDK Version	Start Parameters	Logfile Size(MB)	Operation Method
1 <input checked="" type="checkbox"/>	1	HelloWorld	1.0.0	0.2.0		1	Import Conf Export Conf Export App Uninstall

APP Management

Start All Stop All Restart All

ID APP Name Operation Method

2 **Apply & Save** Cancel

5.3 Manage APP

In Operation area, you can do the following operation:

- import APP configuration
- export APP configuration
- export APP
- uninstall APP

In Management area, you can start, stop or restart your APP.

APP >> APP

Status **APP Management**

Enable APP Manager ☒
Enable IDE Debug ☒
Enable Extended Flash ☐

Import APP Package

No file selected. Browse.. Upload

APP Configuration

Enable	ID	APP Name	APP Version	SDK Version	Start Parameters	Logfile Size(MB)	Operation Method
<input checked="" type="checkbox"/>	1	HelloWorld	1.0.0	0.2.0		1	Import Conf Export Conf Export App Uninstall

APP Management

Start All Stop All Restart All

ID APP Name Operation Method

1 HelloWorld Start Stop Restart

Apply & Save Cancel

Operation

Management

5.4 Check APP status

In Status area, you can find APP status, process id and uptime.

In Action area, you can check APP's log.

APP >> APP

Status

APP Management

Extended Memory Card	Unrecognized
APP Manager Status	Running
Python Version	Python3
SDK Version	1.3.8 Upgrade
Debug Server Status	Running
APP Filesystem Use%	4% of 6640 MB
Data/Log Filesystem Use%	4% of 6640 MB
Extended Filesystem Use%	0%

APP Running Status

ID	APP Name	APP Version	SDK Version	State	Status	Uptime	Action	Action
1	HelloWorld	1.0.0	0.2.0	running	pid 3004, uptime 0:02:49		Clear Log	Show Log Download Lo