Laboratory Manual
for
CE/CZ1003
Introduction to Computational Thinking

Practical Exercise #2:
Network Access of Raspberry Pi
and Cloud9 IDE

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING
NANYANG TECHNOLOGICAL UNIVERSITY

Ex. #2 – Network Access of RPi

Learning Objectives

This practical exercise is to let the students learn how the Raspberry Pi (RPi) can be accessed by a desktop computer (or Notebook) through its Ethernet network connection, and the various software tools that can be used to code programs for the RPi through the remote desktop and execute them on the Raspberry Pi.

Intended Learning Outcomes

At the end of this exercise, you should

- Know the setup required to enable remote computer connection to the Raspberry Pi through its Ethernet interface.
- Be able to access the Raspberry Pi board using software tools such as **PuTTY** and **VNC Viewer**.
- Know how to develop Python programs using the **Cloud9 online IDE**.

Equipment and accessories required

i) Raspberry Pi 3 Model B (RPi3) board with Sense HAT add-on display module/board.

ii) A USB power source to power the RPi3 board (E.g. Power Bank, Adaptor or ***USB port of a desktop computer***).

iii) A computer (desktop PC or notebook) with Ethernet port and cable for remote access of RPi3. Software (open source) to be installed on the computer – PuTTY, VNC Viewer and WinSCP

1. Introduction

While the RPi can be used as a standalone computer by connecting it with keyboard/mouse/monitor as in Practical Exercise #1, this setup is rather cumbersome due to the connections required. An alternative way to use the RPi is to operate it in a 'headless' setup, i.e. without connecting it with keyboard/mouse/monitor. Instead, it is connected via one of its network interfaces to a computer/notebook. In this course, the setup is to have the RPi connects to the desktop computer in the laboratory using its wired Ethernet interface as shown in Figure 1 (and powered by one of the USB port on the computer).

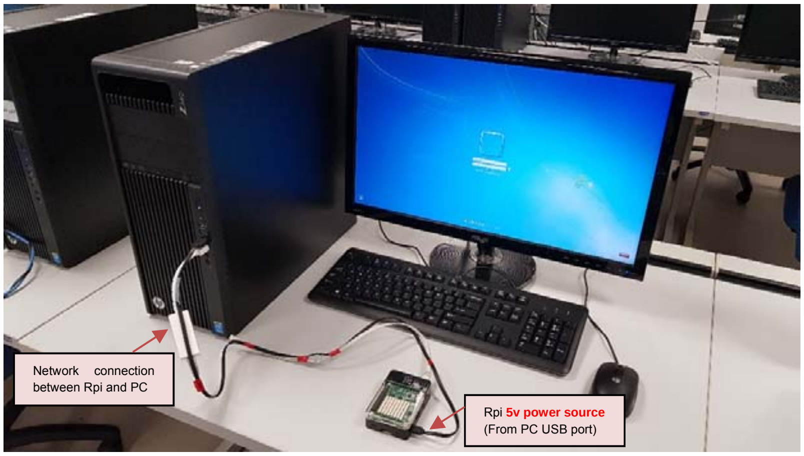


Fig.1 - Raspberry Pi in Headless mode

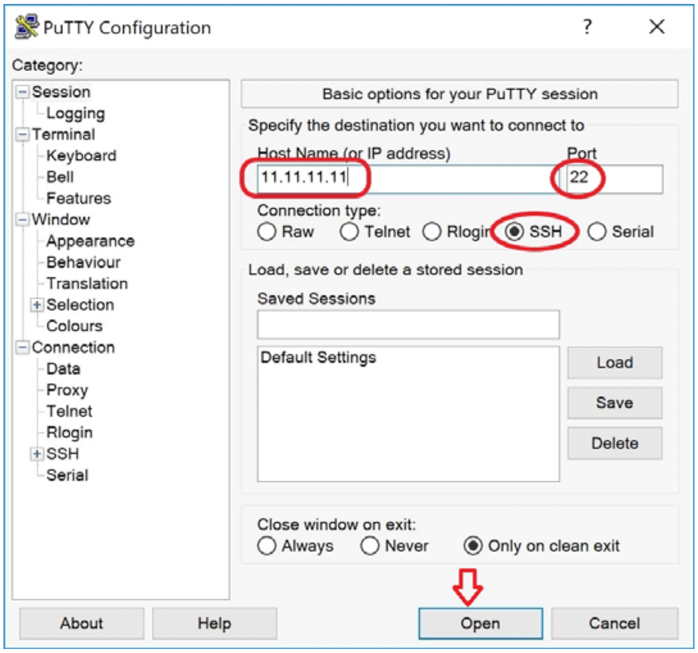
(In practice, it is much more convenient to have the RPi wirelessly connects to a notebook through its Wifi interface.)

2 Network Access of RPi - PuTTY

Several programs will be used in this course to access the RPi through the network. To start with, you will use the **PuTTY** program running the Secure Shell (SSH) network protocol.

- Run the PuTTY program on the desktop computer (which should be already install on the computer)

- A pop up window will be shown. Key in the IP address of the RPi, which is pre-set with the value **11.11.11.11**. Make sure that port 22 is selected, and SSH is selected as the connection type.



PuTTY Configuration

Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address): 11.11.11.11

Port: 22

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

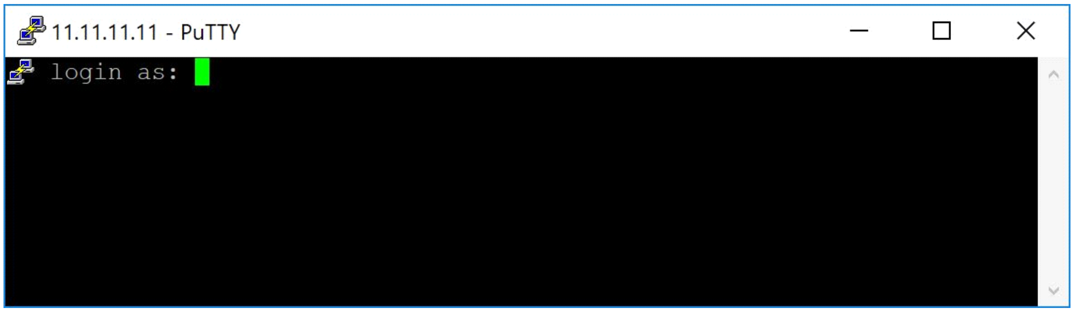
Load, save or delete a stored session

Saved Sessions

Default Settings

Close window on exit: ○ Always ○ Never ● Only on clean exit

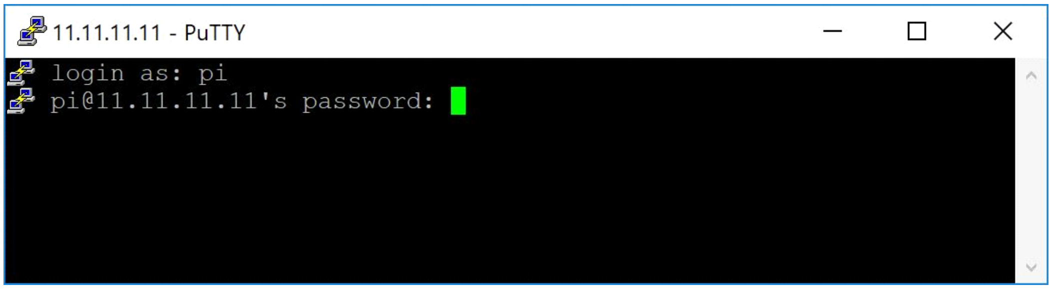
- Click 'Open', and a console interface will appear if the connection is successful. (Otherwise, check the IP address and other settings that you key in.)



11.11.11.11 - PuTTY

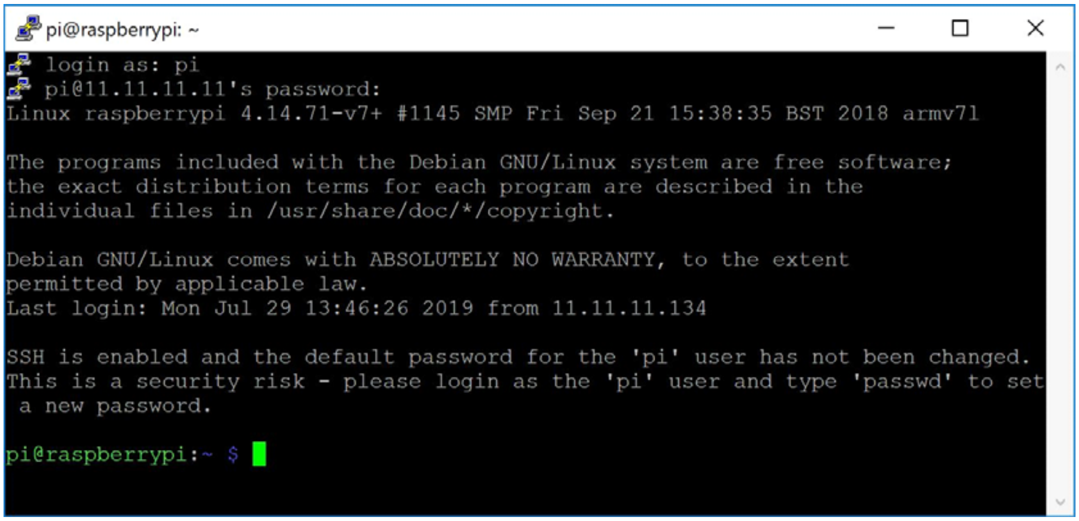
```
login as: 
```

- The default login ID is “**pi**” and the password is “**raspberry**”. (***Remarks : just type the password normally, you will not see the password been typed or shown***.)



```
login as: pi
pi@11.11.11.11's password:
```

- If the remote login is successful, the console interface display in the PuTTY program will be as shown below, which is essentially identical to the Terminal interface that you used during Practical Exercise #1.



```
login as: pi
pi@11.11.11.11's password:
Linux raspberrypi 4.14.71-v7+ #1145 SMP Fri Sep 21 15:38:35 BST 2018 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Jul 29 13:46:26 2019 from 11.11.11.134

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set
 a new password.

pi@raspberrypi:~ $
```

As such, you can now remotely operate the RPi as if you are running the Terminal program directly on the RPi. All the commands that you had used before can be similarly executed through the PuTTY console.

Things to try:

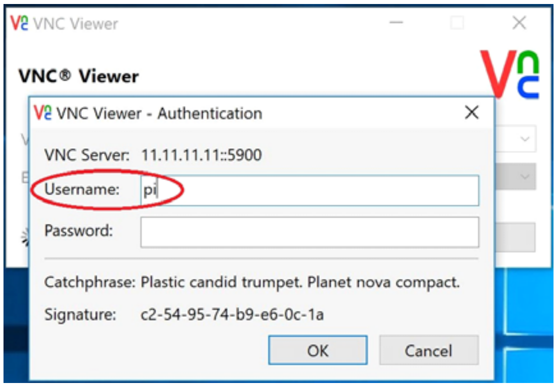
- Write the Python code to print the ‘Hello world’ message.
- Open another PuTTY console terminal and code the program to display a message on the SenseHat module (refers to manual of Lab Ex #1 for detail).

3 Network Access of RPi – VNC Viewer

There may be time that you will need to access the RPi through its GUI, such as to launch a web browser to access the internet. For such situation, you can also remotely access the RPI's GUI through the network, using the VNC Viewer program.

3.1 We will first launch the GUI through the VNC Viewer

- Run the VNC Viewer program on the computer (It should be already installed).
- Key in the IP address of the RPi (which is …). If this is the first time the RPi is accessed by the VNC Viewer program, an Authenticating pop up dialog will appear to request for the User's ID and password (which are ..)



VNC Viewer

VNC® Viewer

VNC Viewer - Authentication

VNC Server: 11.11.11.11::5900

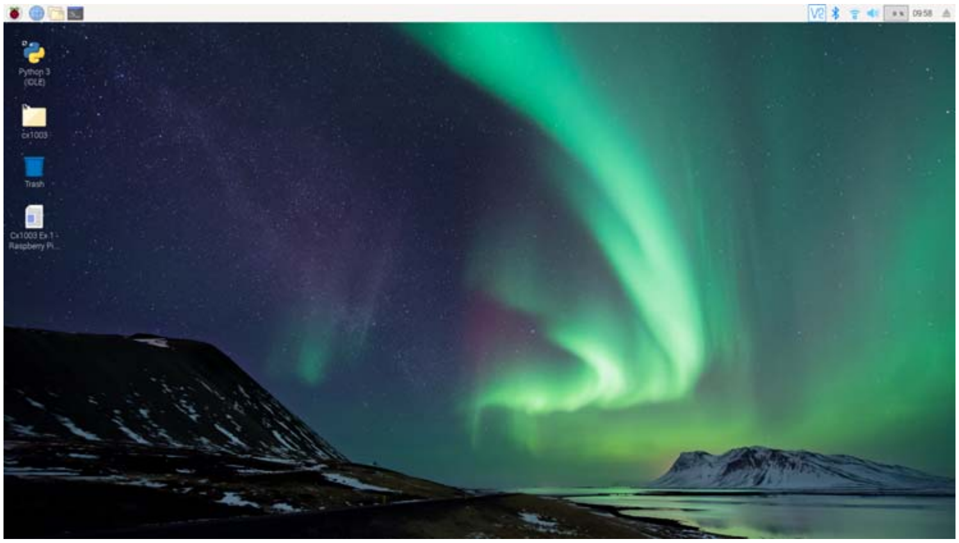
Username: pi

Password:

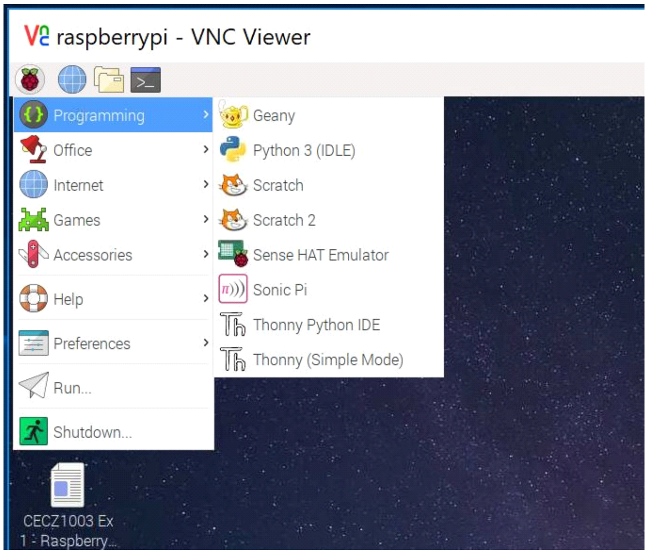
Catchphrase: Plastic candid trumpet. Planet nova compact.

Signature: c2-54-95-74-b9-e6-0c-1a

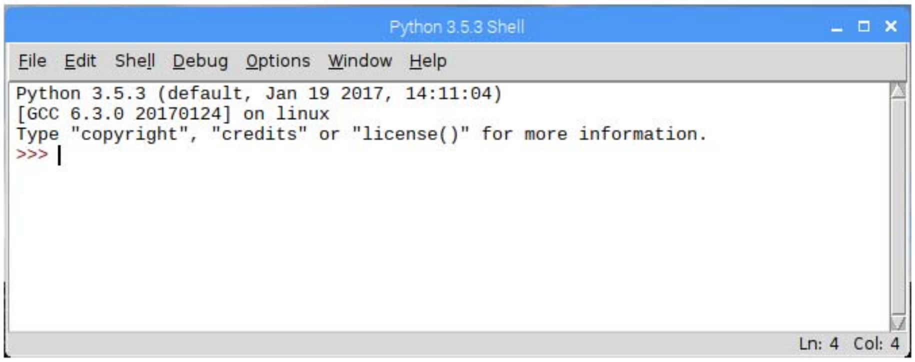
- The GUI of the RPi will then appear in the window, which you can then use to access and operate the RPi as in Practical Exercise #1.



3.2 You will now try using a simple Python IDE (Integrated Development Environment) software with build-in Python interpreter for Python programs development. This is the IDLE program (use the Python 3 version) available under the Application menu.

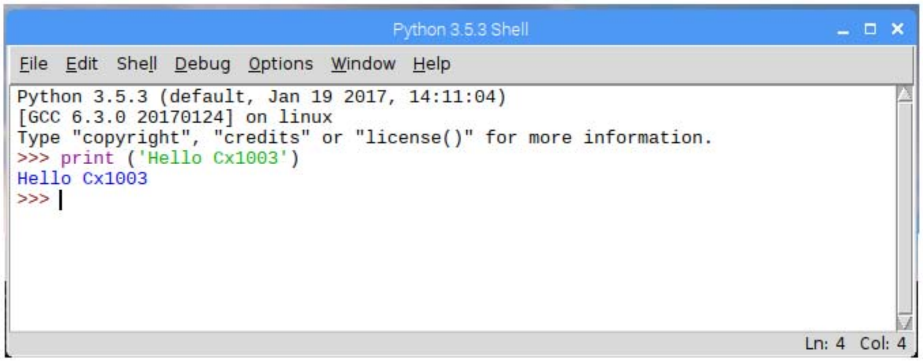


- It looks the same to the Python3 interpreter running in a terminal, and obviously will behave exactly the same way.



```
Python 3.5.3 (default, Jan 19 2017, 14:11:04)
[GCC 6.3.0 20170124] on linux
Type "copyright", "credits" or "license()" for more information.
>>> |
```

- Try the Hello message code like as shown below. Note that the interface is now displayed in colour to provide better interface to the user.



```
Python 3.5.3 (default, Jan 19 2017, 14:11:04)
[GCC 6.3.0 20170124] on linux
Type "copyright", "credits" or "license()" for more information.
>>> print ('Hello Cx1003')
Hello Cx1003
>>> |
```

- Close the VNC Viewer program before you proceed to the next part.

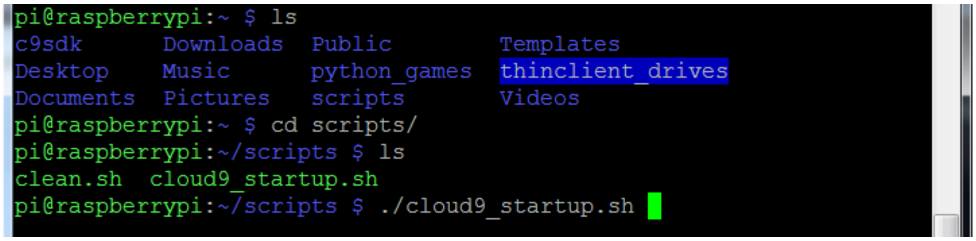
4 Cloud9 IDE

As can be observed, the features provided by the IDLE Python IDE is rather rudimentary, and will not be user friendly once your program grows and becomes more complex. In this course, we will be using a more sophisticated IDE, the Cloud9 IDE for you to practice developing the various programming exercises in subsequent practical sessions.

Cloud9 is a 'cloud-based' IDE that lets you write, run, and debug your code with just a web browser. This means that Cloud9 is a web server program that runs remotely on another computer (somewhere behind a 'cloud'), and allows the user to remotely access it using web browser through a network connection. In our case, you will run the Cloud9 Web server on the RPi, and then access it using a web browser on the desktop.

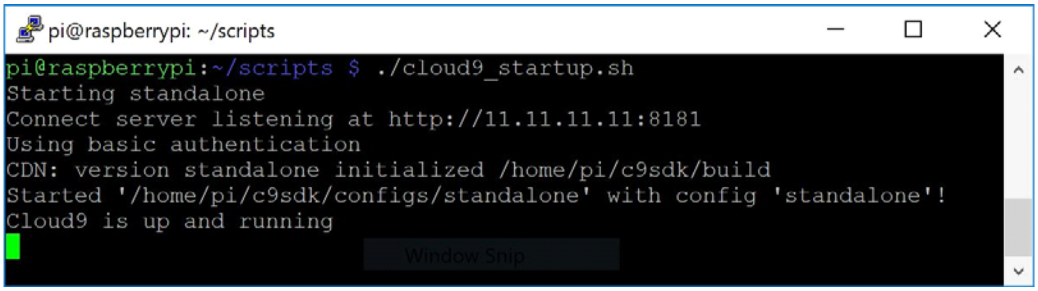
4.1 First you need to start the Cloud9 web server program on the RPi. This is to be done by running a (shell script) file that contain the instructions to launch the Cloud9 server program.

- **Start a PuTTY terminal program** on the desktop computer, login and change to the scripts directory as shown below.



```
pi@raspberrypi:~ $ ls
c9sdk      Downloads  Public        Templates
Desktop    Music      python_games  thinclient_drives
Documents  Pictures   scripts       Videos
pi@raspberrypi:~ $ cd scripts/
pi@raspberrypi:~/scripts $ ls
clean.sh  cloud9_startup.sh
pi@raspberrypi:~/scripts $ ./cloud9_startup.sh
```

- Execute the cloud9_startup.sh shell script file, and if everything is in order, a message will indicate that the Cloud9 server program is up and running, waiting for user to connect over the network.



```
pi@raspberrypi:~/scripts $ ./cloud9_startup.sh
Starting standalone
Connect server listening at http://11.11.11.11:8181
Using basic authentication
CDN: version standalone initialized /home/pi/c9sdk/build
Started '/home/pi/c9sdk/configs/standalone' with config 'standalone'!
Cloud9 is up and running
```

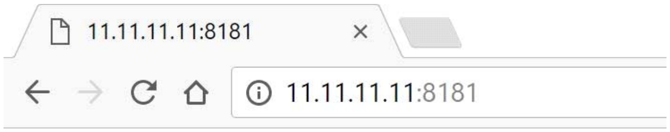
- Note that the program indicates that it is "listening at http://11.11.11.11:8181", in which the ***8181*** is the port number it is listening to.

- We will leave this server program running on the RPi, and now try to access it through our desktop computer's browser.

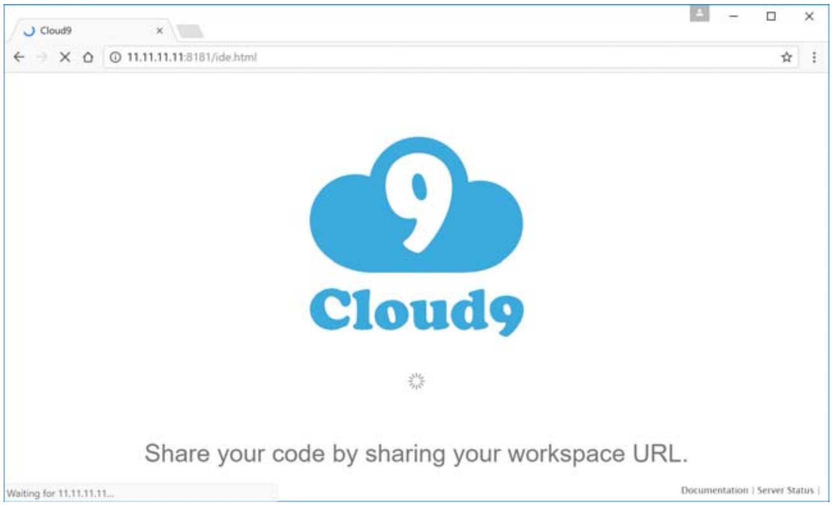
Ex. #2 – Network Access of RPi

4.2 We will now run a Web browser program on the desktop computer, such as the Firefox (or IE, or Chrome).

- Open a web browser on the desktop computer, key in the IP address and port number indicated by the server program

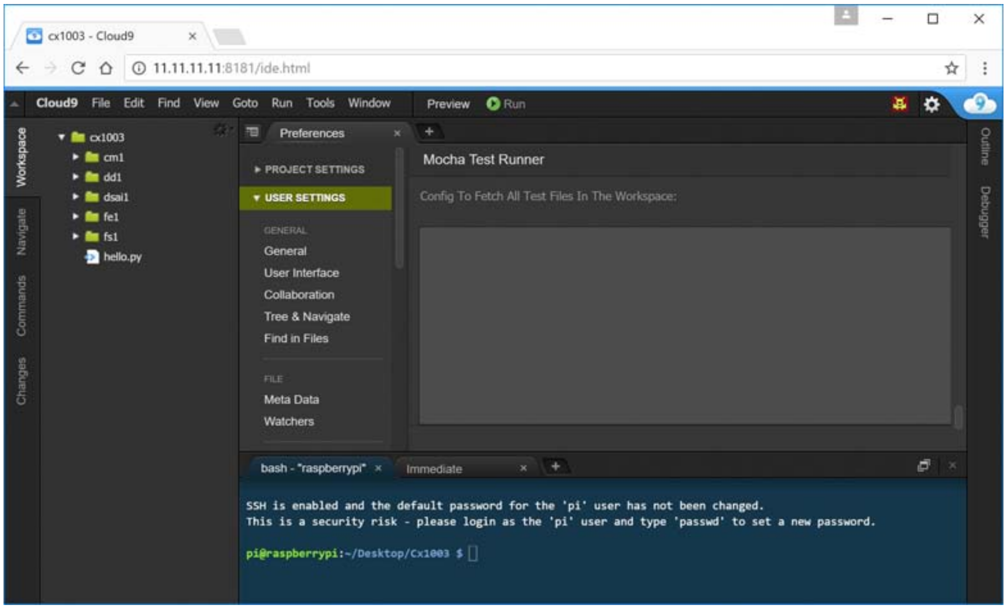


- The Cloud9 screen will appear briefly.


Cloud9
11.11.11.11:8181/ide.html
Cloud9
Waiting for 11.11.11.11...
Documentation | Server Status

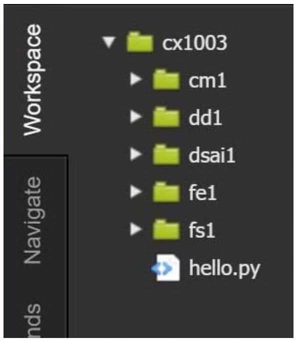
Share your code by sharing your workspace URL.

- And eventually the cloud9 IDE interface is launched



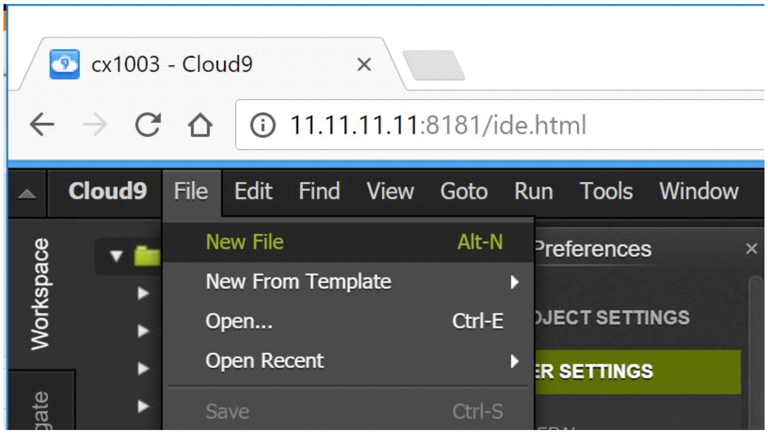
```
SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set a new password.

pi@raspberrypi:~/Desktop/Cx1003 $ 
```

4.3 On the left side of the IDE is the workspace, where you will store the program files you are going to create. The workspace has already been created with various sub-folders from ***Lab 1***, corresponding to the different lab groups that you belong to. You should hence store your program files in the appropriate folder.

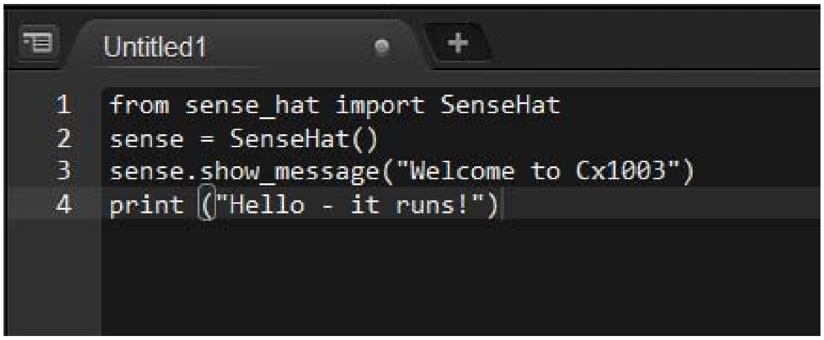


The top right window pane is for you to create your program code.

- Click on File→New File option to open a new file for coding

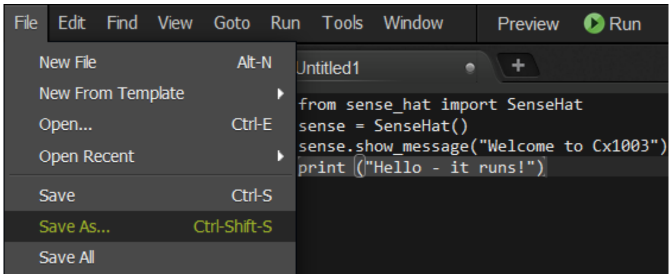


- Type the code such as the following

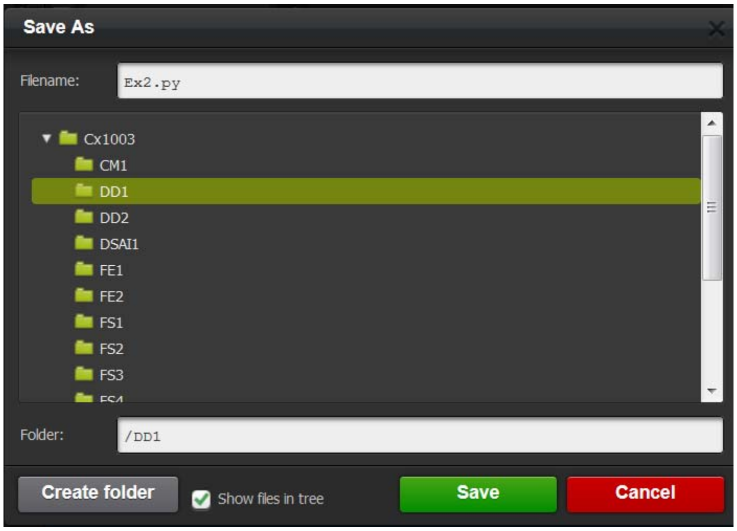


```
from sense_hat import SenseHat
sense = SenseHat()
sense.show_message("Welcome to Cx1003")
print ("Hello - it runs!")
```

- Save the file as “**Ex2.py**” in your folder (e.g. DD1 folder here)



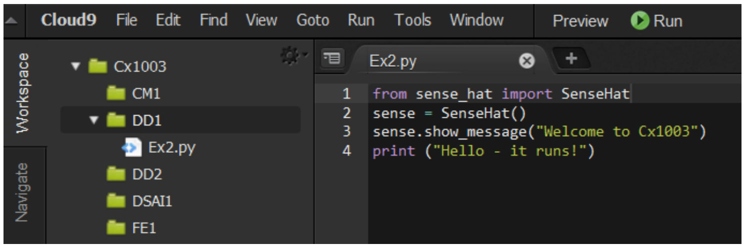
```
from sense_hat import SenseHat
sense = SenseHat()
sense.show_message("Welcome to Cx1003")
print ("Hello - it runs!")
```



Filename: Ex2.py

Folder: /DD1

☑ Show files in tree

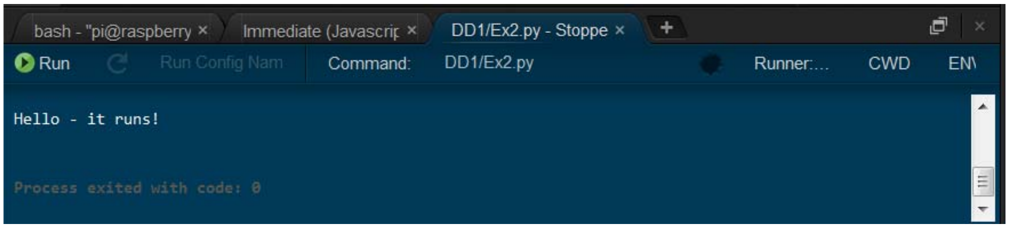


```
from sense_hat import SenseHat
sense = SenseHat()
sense.show_message("Welcome to Cx1003")
print ("Hello - it runs!")
```

Notice that the IDE now highlights the code in different colours once it knows that this is a Python program. (Q. How does it know that?)

Next you can run the program by clicking the Run icon, Run on the top panel.

- If everything is in order, you should see the message displays on the SenseHat module, as well as the message in the output panel at the bottom of the IDE.



Command: DD1/Ex2.py

```
Hello - it runs!


Process exited with code: 0
```

Q. Does the program execution exhibit certain behaviour expected of an interpreted language?

4.4 Configure cloud9 IDE for Python 3

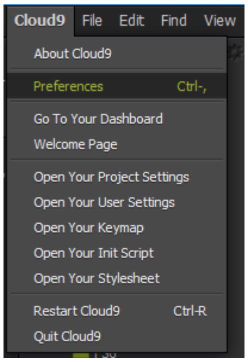
Python programming language was conceived in the late 1980s, and has since gone through several revisions, with version 3 being the latest (specifically, version 3.7 that was released on June 27, 2018). (https://en.wikipedia.org/wiki/History_of_Python). But Python 2 (2.7) can still be found in many existing/legacy applications.

There are plenty of differences between version 2 and version 3 of the Python language. Two of the differences you will notice when you code programs in this course are the **print()** and **input()** functions.

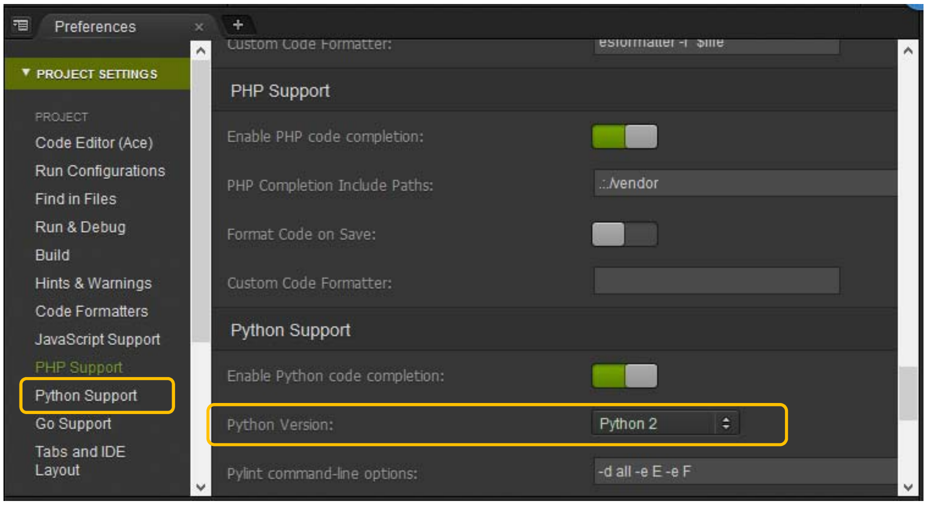
- In Python 2, the print statement is written as **print "Hello world"**, while in Python 3, print statement is replaced as a function and is written as **print ("Hello world")**.
- In Python 2, the **input()**function evaluates the data it receives and returns the data type based on what it 'thinks' it should be. In Python 3, **input()** function always returns the data as str (i.e. string) data type object.

Cloud9 IDE supports both *Python 2 and Python 3*, but it is obvious that we should learn the latest, which is Python 3. As such, you will configure Cloud9 IDE for Python 3 in this course, using the steps as described below. (The preferences may have already set to Python3 by your classmate)

i) Select the **Preferences** option under the Cloud9 menu



ii) The **Preferences** window pane will be shown. Look for the **Python Version** setting.



Custom Code Formatter: esformatter -i $file

PHP Support

Enable PHP code completion:

PHP Completion Include Paths: ::/vendor

Format Code on Save:

Custom Code Formatter:

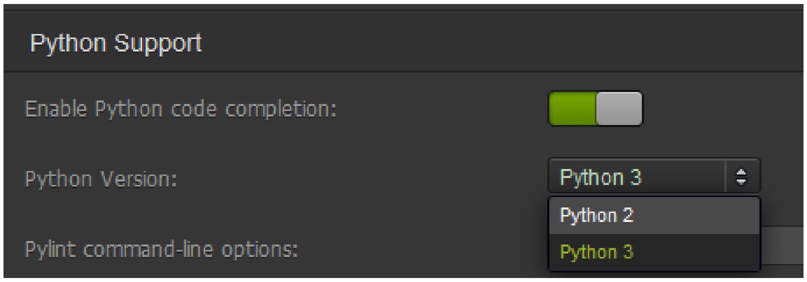
Python Support

Enable Python code completion:

Python Version: Python 2

Pylint command-line options: -d all -e E -e F

iii) If the **Python Version** is shown as **Python 2**, change it to **Python 3**.

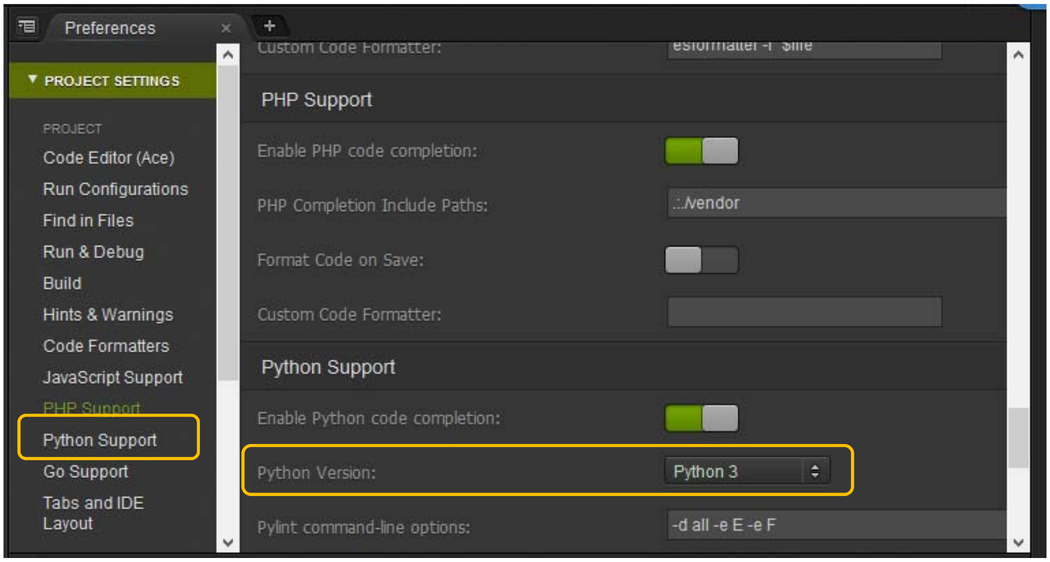


Python Support

Enable Python code completion:

Python Version: Python 3

Pylint command-line options:



Custom Code Formatter: esformatter -i $file

PHP Support

Enable PHP code completion:

PHP Completion Include Paths: .:./vendor

Format Code on Save:

Custom Code Formatter:

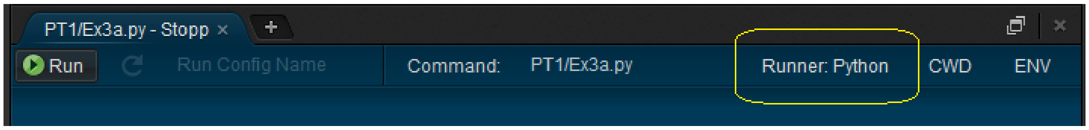
Python Support

Enable Python code completion:

Python Version: Python 3

Pylint command-line options: -d all -e E -e F

iv) In addition, when you later run your program, check that the setting for **Runner** is set to **Python** (and NOT Python 2 Runner: Python 2



Command: PT1/Ex3a.py

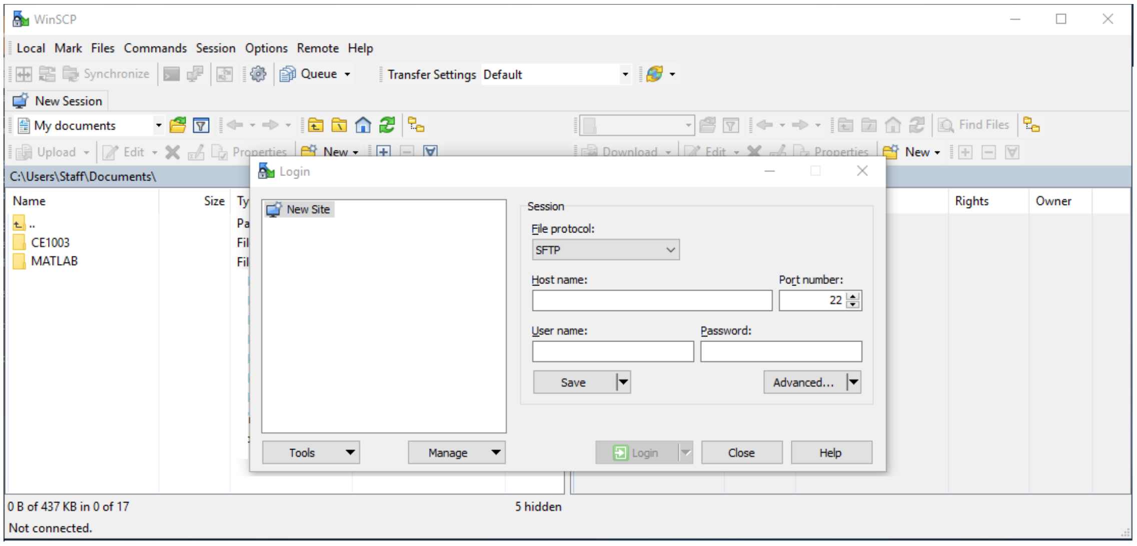
Runner: Python

5. Back-up your program file

At the end of each practical session, you may want to copy your program files on the RPi to your own USB thumb drive. ***There are several ways to do this***. For example, you can use the WinSCP program as describe here.

- Open the WinSCP program on the desktop computer.

- The program will run and prompt for the detail of the RPi board that you want it to connect to.

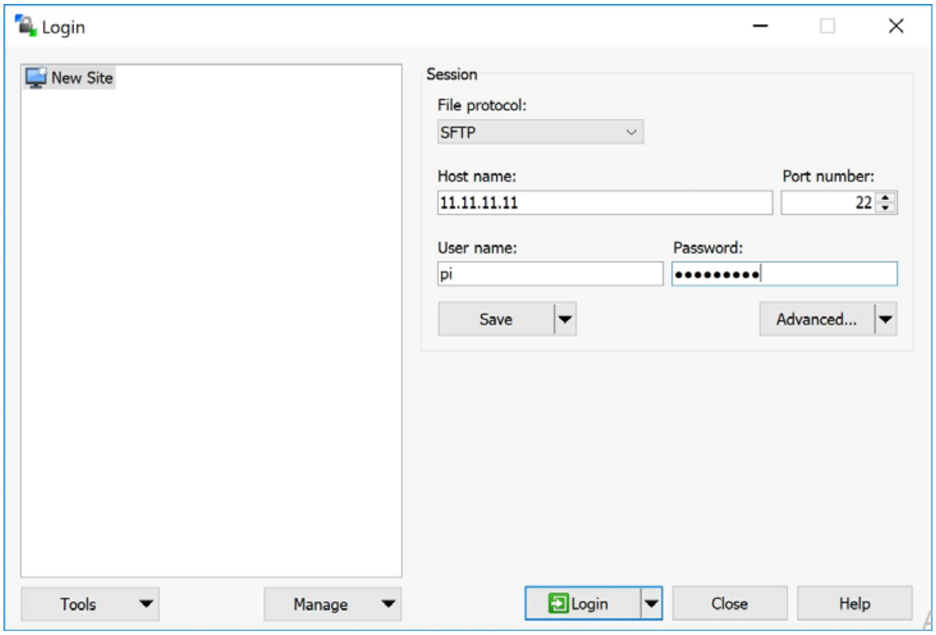


0 B of 437 KB in 0 of 17

5 hidden

Not connected.

- Key in the information requested



File protocol: SFTP

Host name: 11.11.11.11

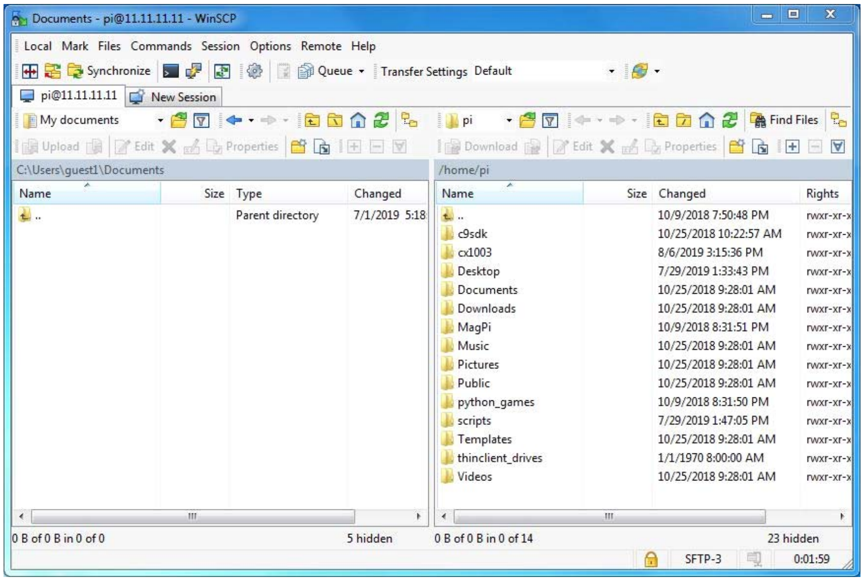
Port number: 22

User name: pi

Password: ●●●●●●●●●

Ex. #2 – Network Access of RPi

- Once the connection is established, the WinSCP program will display the default directories of both the desktop computer (on the left window pane) and RPi (on the right window pane).


Documents - pi@11.11.11.11 - WinSCP
Local Mark Files Commands Session Options Remote Help
Synchronize
Queue
Transfer Settings Default
pi@11.11.11.11
New Session
My documents
Upload
Edit
Properties
pi
Find Files
Download
Edit
Properties

C:\Users\guest1\Documents

Name	Size	Type	Changed
..		Parent directory	7/1/2019 5:18

0 B of 0 B in 0 of 0 5 hidden

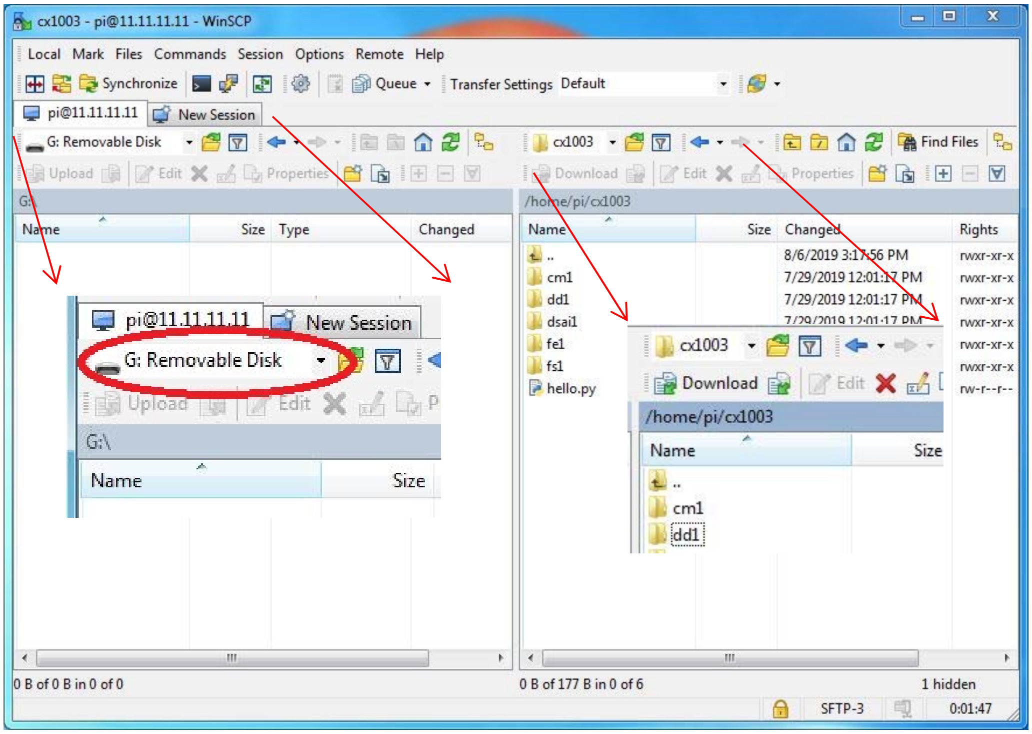
/home/pi

Name	Size	Changed	Rights
..		10/9/2018 7:50:48 PM	rwxr-xr-x
c9sdk		10/25/2018 10:22:57 AM	rwxr-xr-x
cx1003		8/6/2019 3:15:36 PM	rwxr-xr-x
Desktop		7/29/2019 1:33:43 PM	rwxr-xr-x
Documents		10/25/2018 9:28:01 AM	rwxr-xr-x
Downloads		10/25/2018 9:28:01 AM	rwxr-xr-x
MagPi		10/9/2018 8:31:51 PM	rwxr-xr-x
Music		10/25/2018 9:28:01 AM	rwxr-xr-x
Pictures		10/25/2018 9:28:01 AM	rwxr-xr-x
Public		10/25/2018 9:28:01 AM	rwxr-xr-x
python_games		10/9/2018 8:31:50 PM	rwxr-xr-x
scripts		7/29/2019 1:47:05 PM	rwxr-xr-x
Templates		10/25/2018 9:28:01 AM	rwxr-xr-x
thinclient_drives		1/1/1970 8:00:00 AM	rwxr-xr-x
Videos		10/25/2018 9:28:01 AM	rwxr-xr-x

0 B of 0 B in 0 of 14 23 hidden

SFTP-3 0:01:59

- You can plug in your USB thumb drive to the desktop computer, and then select it for the left window pane. Similarly select the appropriate directory of the RPi on the right window pane.



G:\

Name	Size	Type	Changed

0 B of 0 B in 0 of 0

/home/pi/cx1003

Name	Size	Changed	Rights
..		8/6/2019 3:17:56 PM	rwxr-xr-x
cm1		7/29/2019 12:01:17 PM	rwxr-xr-x
dd1		7/29/2019 12:01:17 PM	rwxr-xr-x
dsai1		7/29/2019 12:01:17 PM	rwxr-xr-x
fe1			rwxr-xr-x
fs1			rwxr-xr-x
hello.py			rw-r--r--

0 B of 177 B in 0 of 6 1 hidden

SFTP-3 0:01:47

- You can then copy your file on the RPi to your USB thumb drive by dragging it from the right window pane to the left window pane, and vice versa.

(Alternatively, you can copy your file on the RPi to the desktop computer, and then email it to yourself.)

6. Summary

In this practical exercise, you had seen how the RPi can be setup in a headless mode. You then learnt how it can be remotely accessed through the network using software programs such as PuTTY and VNC Viewer. You had also learnt how to use the Cloud9 IDE to develop simple Python program, and how to save your work to a USB thumb drive using the WinSCP program.

Lastly, issue the following commands in the PuTTY Terminal to properly shut down the RPi computer.

- Terminate the Cloud9 IDE – select the PuTTY Terminal that you started the Cloud9 server (you would have observed several messages in the Terminal, correspond to what you had performed earlier in your browser Cloud9 IDE), press **Crtl-C** to stop the server program
- Delete the file that you have created by running the script file: **`./clean.sh`**
  Make sure that you have saved your work (if you want) to your thumb drive before you do this.
- Shutdown the RPi by issuing the command: **`sudo shutdown –h now`**
- **!! Please wait for about 10 seconds for the Rpi to complete the shutdown process before power off the Rpi power (removing usb cables from PC), otherwise system files corruption may occur.**