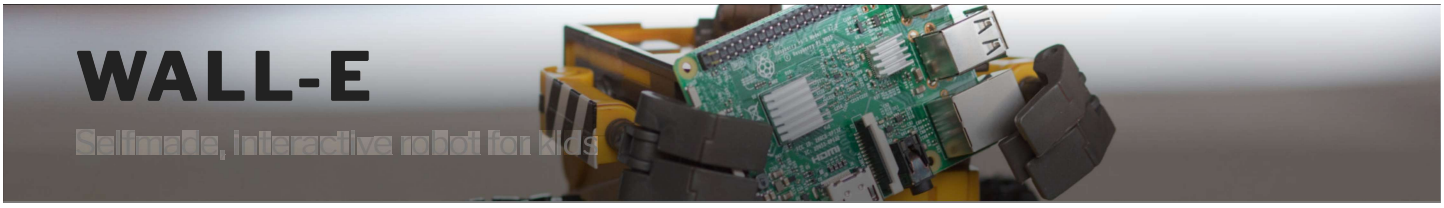


SETTING UP ECLIPSE

Eclipse is very useful for programming Python on Raspberry Pi:

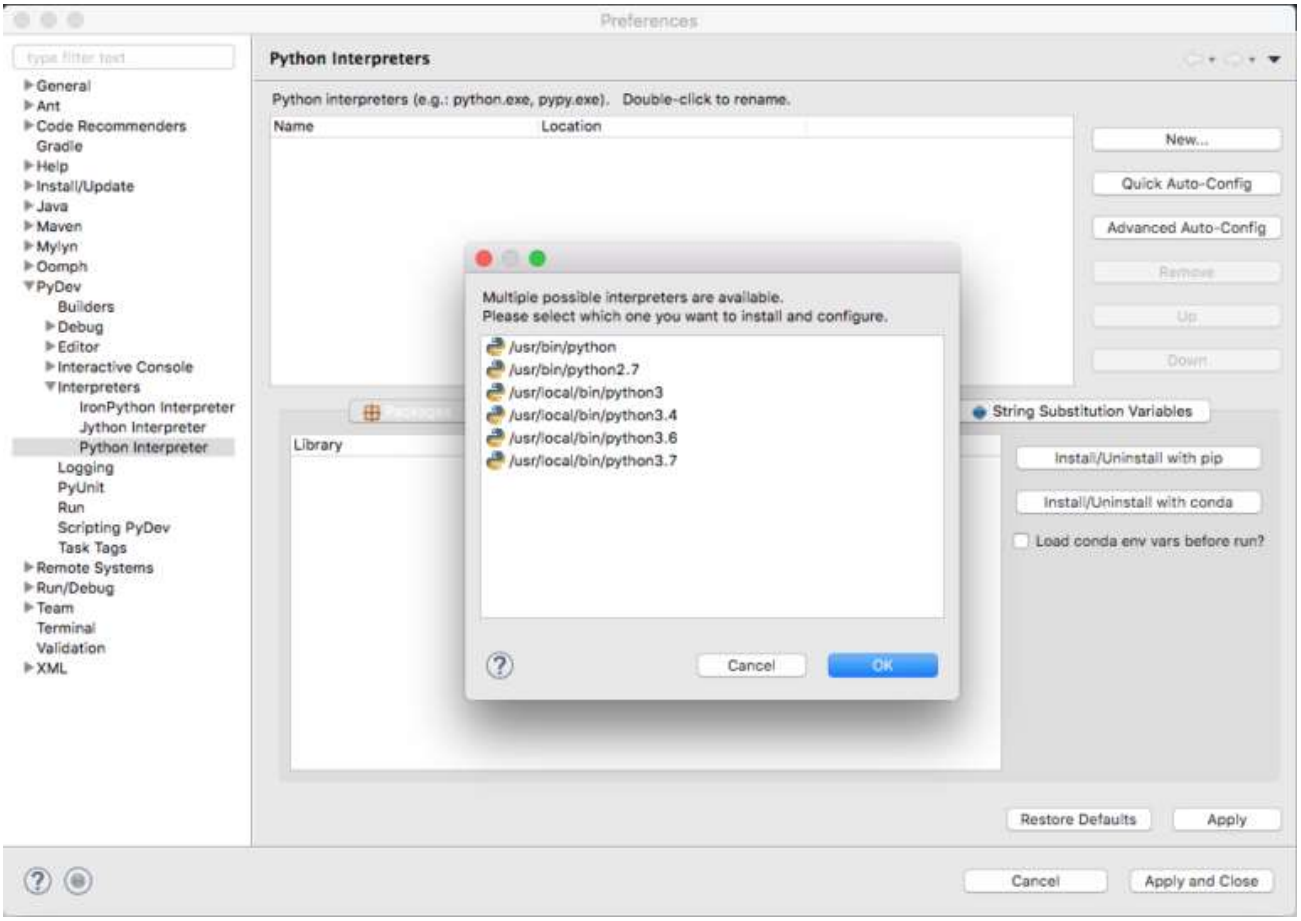
- It is fee of charge
- I comes with a remote access plug-in to edit, run and debug your code running on the Pi

Install necessary packages

1. Download Eclipse for your system
2. Choose “IDE for Java developers” during installation (we never use Java then but the IDE configuration comes close to what we need for Python)
3. Install PyDev: Eclipse menu / Help / Eclipse Market Place:
   Search for “PyDev” and install it
4. Install Remote System Explorer: Eclipse menu / Help / Eclipse Market Place:
   Search for “Remote” and install it
5. Download and install the latest version of Python on your computer

Setting up Eclipse IDE to work with Python

In Eclipse navigate to Preferences and open the “Interpreter section”:



Select Advanced Auto-Config and select the latest version installed (in this picture I will install Python3.7). Apply and Close the dialog when import is finished.

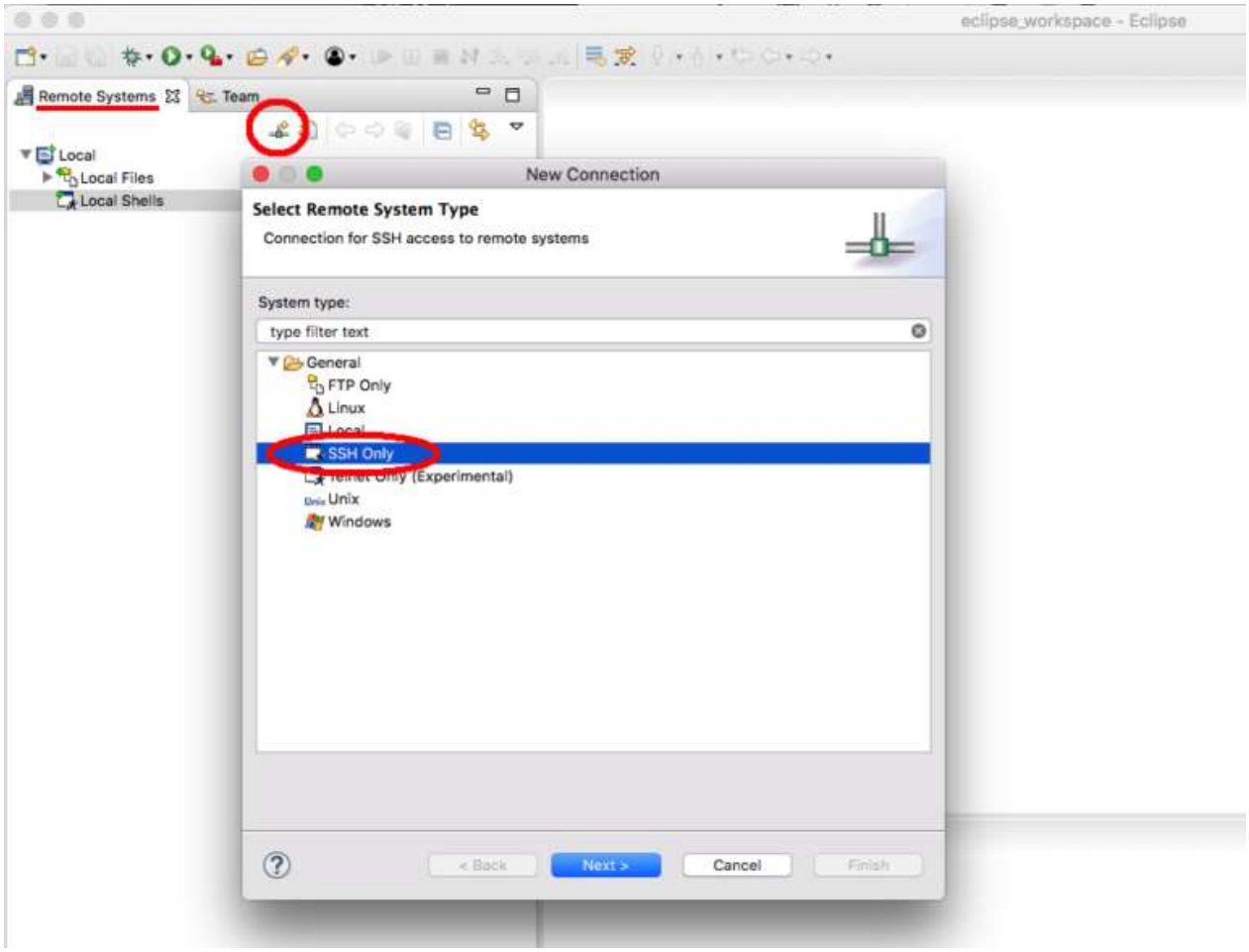
Create a new project

1. In the Eclipse menu select File / New / Project and select PyDev / PyDev Project
2. Give it a name

Get access to Raspberry Pi

1. In the Eclipse menu select Window / Perspective / Open Perspective / Remote System Explorer

2. Select “SSH only” as new connection



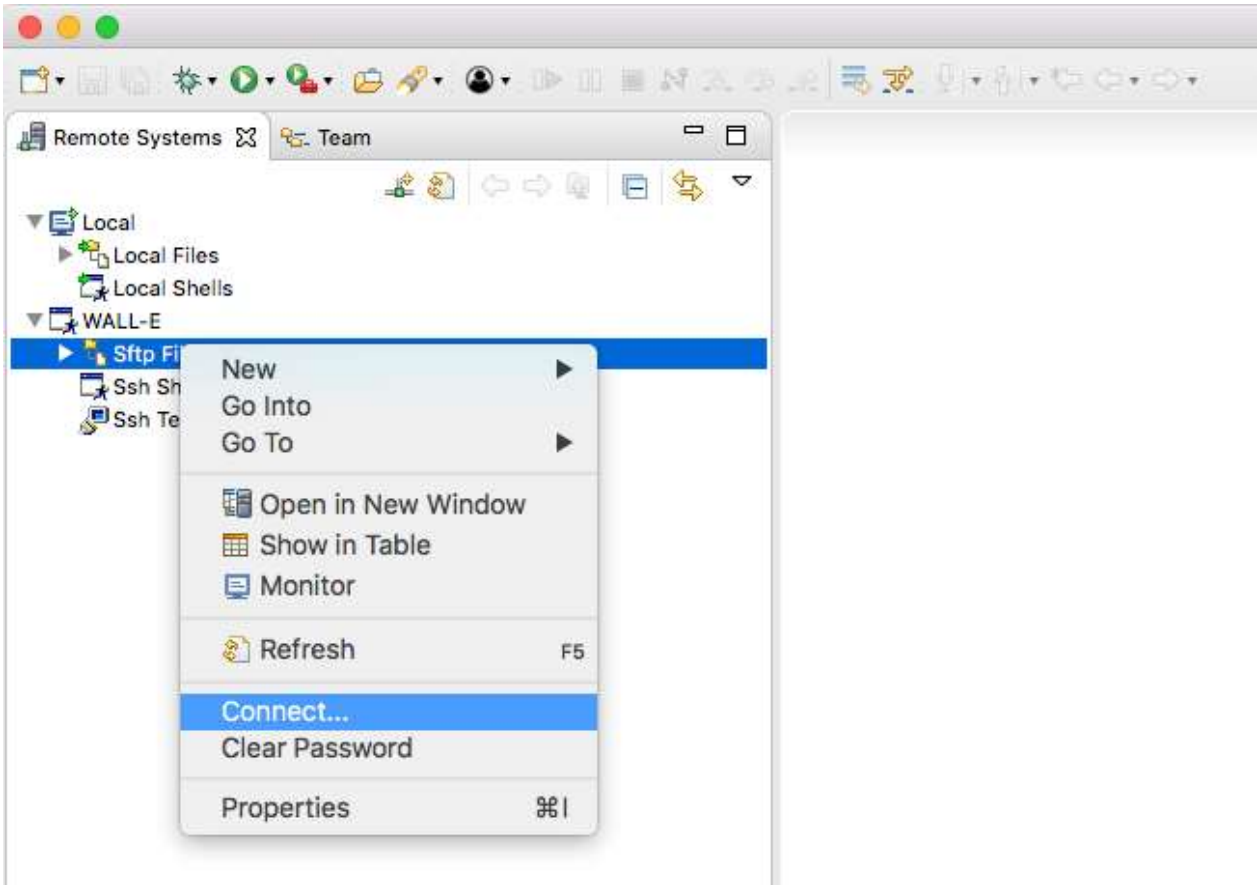
3. Define connection information

   Set host name to “192.168.0.3

   Set connection name to ‘WALL-E’

   Set description to ‘Raspberry Pi of WALL-E’

4. Connect to Pi



   Enter User ID: pi

   Enter Password: our chosen password for the Raspberry Pi

5. Open remote shell

   Navigate to Window / Show View / Other / Remote Systems / Remote Shell

Our first test script

1. Create a directory for our python project

   Navigate to “My Home” and create New / Folder in ‘/home/pi’ called ‘wall-e’

2. Create a Python script

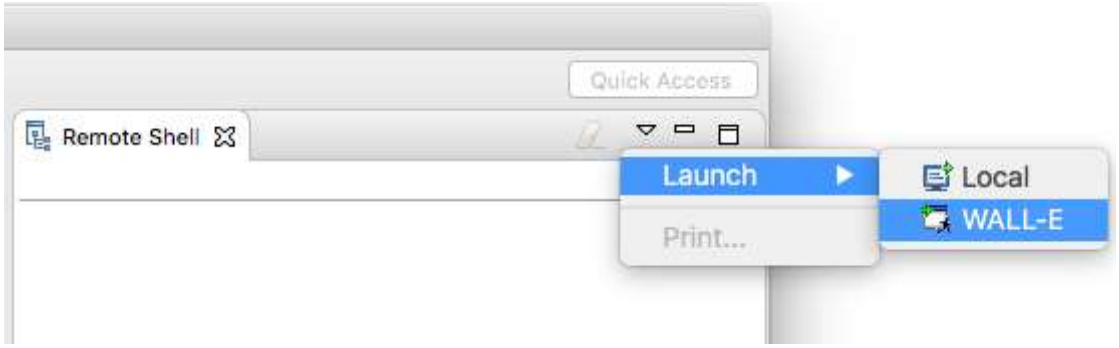
   Navigate to My Home / wall-e and create New / File called ‘main.py’

   Add the following lines of code:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
#python3 /home/pi/wall-e/main.py
import datetime, time
print("WALL-E started at %s"%(datetime.datetime.now().strftime("%Y-%m-%d %H:%M
```

3. Open the shell

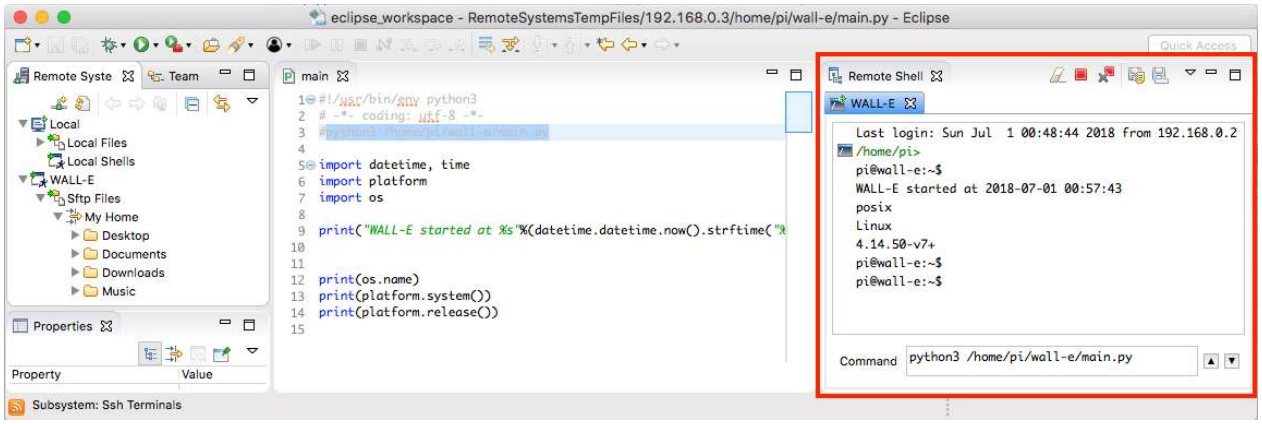
   As we are on the Pi, we can not use the run button of clips as it would executed locally. To avoid this we open a shell instance at the Pi:



4. Launch script

```
Command:  python3 /home/pi/wall-e/main.py
```

5. The result should look like this:



You did it! Your Pi is now ready to be programmed in Python.