

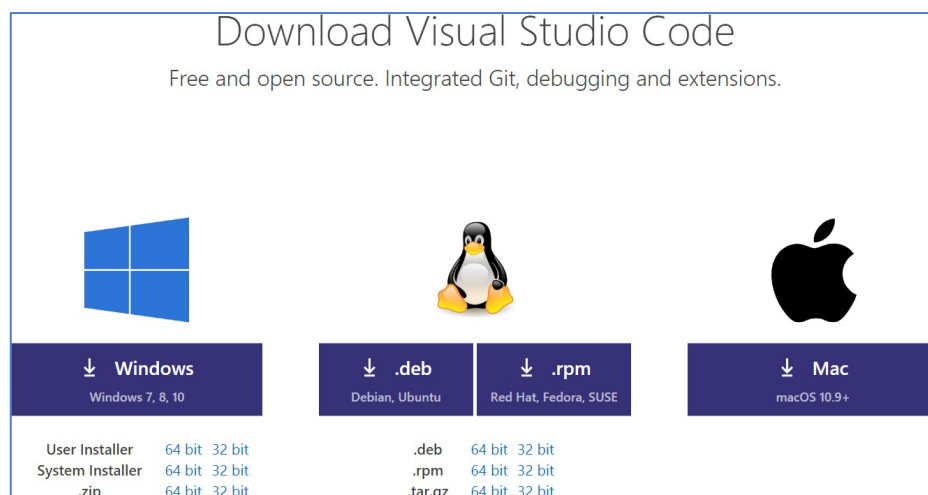
- Objectives:** By the end of this practical exercise, the students should be able to:
- Install and/or verify the setup of Python 3 and Visual Studio Code (VS Code).
 - Install and/or verify the setup of the build-in library and 3rd party packages:
 - hashlib, pycryptodome, cryptography.
 - Use VS code to do simple processing.

NOTE

- This installation is designed to work in conjunction with other modules.
- Skip this section on *VS code installation*, if you have already installed the program in another module (e.g. Programming in security).
- Skip this section on *Python 3 installation*, if you have already installed the language in another module (e.g. Programming in security).
- Please **install of 3rd party packages (libraries)**. All codes provided in the (Applied Cryptography) module will need these packages. These 3rd party packages should not affect your other modules (e.g. Programming in security).

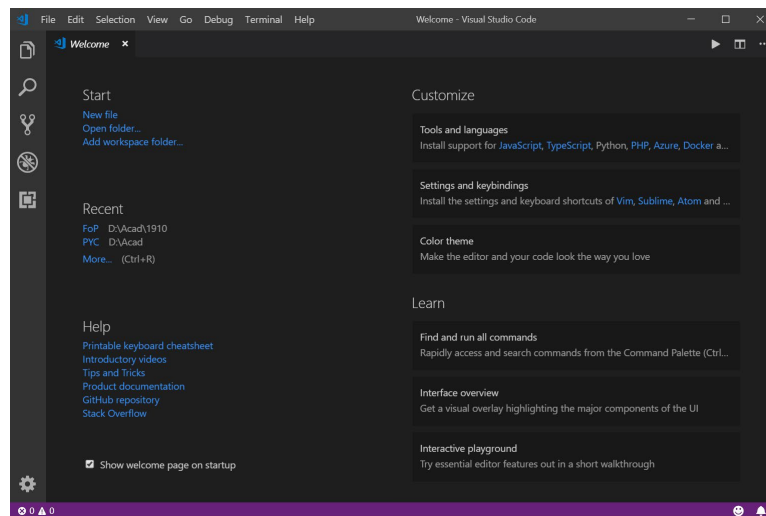
Visual Studio Code Installation

1. Download and install Visual Studio Code (if you don't have it)
 - a) Access this URL to download Visual Studio Code (choose the one that is compatible with your notebook): <https://code.visualstudio.com/download>



- b) Install VS Code. You can accept all default options during the installation process. For windows user, note that 'System Installer' is for the users who has full administrative right to their PC, whereas 'User Installer' is for those without.

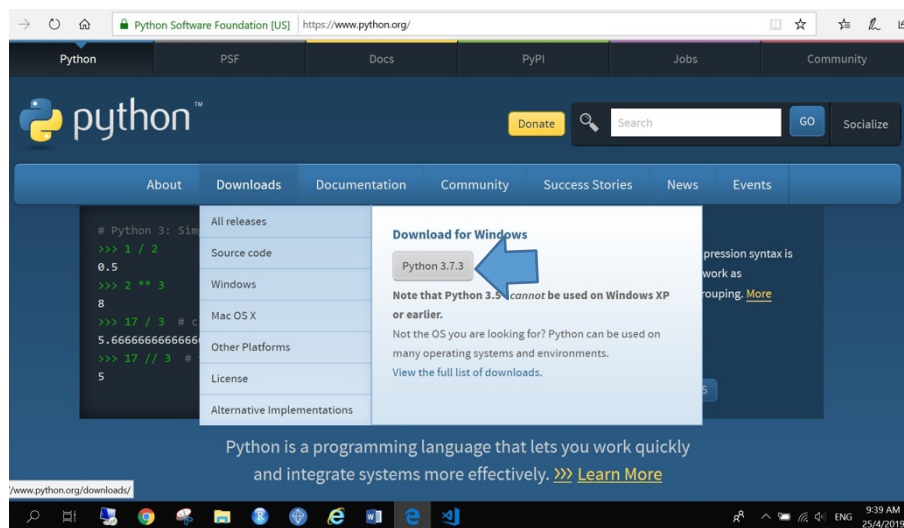
c) Launch VS Code. You may close any folders from your previous modules.



Python 3 installation

1. Installing Python 3

- a) We need to install Python before using VS Code for Python programming. Visit www.python.org to download Python 3.7.3 (or later). Note that this version of Python is not supported on earlier Windows platforms.



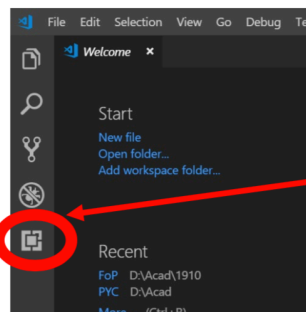
- b) **SELECT BOTH CHECKBOXES TO ADD PYTHON3 TO PATH**, then select **Install Now** to install Python 3.7.x This allows you to run Python from other locations.



Click [here](#) for instructions if you forgot to check it when installing.

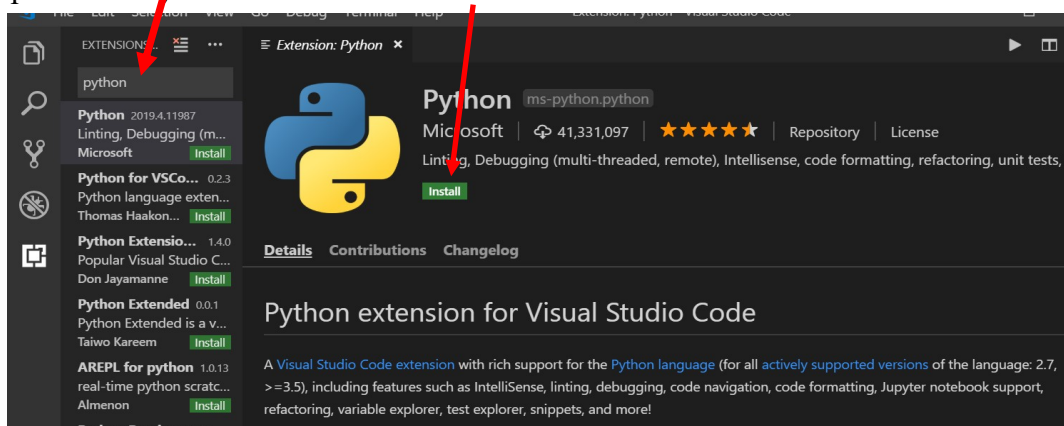
2. Visual Studio Code extensions

- a) VS Code extensions let you add languages, debuggers, and tools to your installation to support your development workflow. 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:

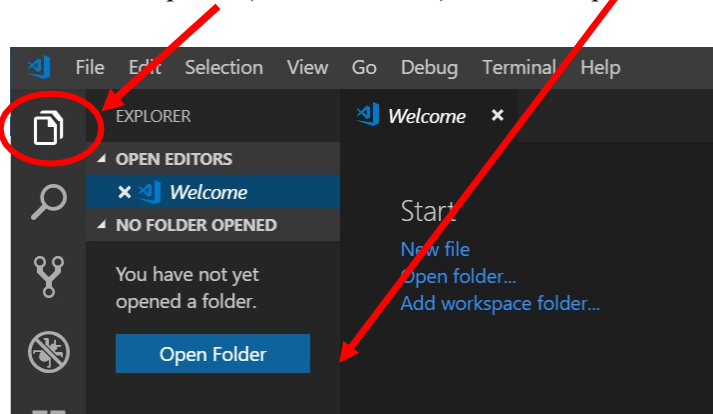


Extensions command (Ctrl+Shift+X).

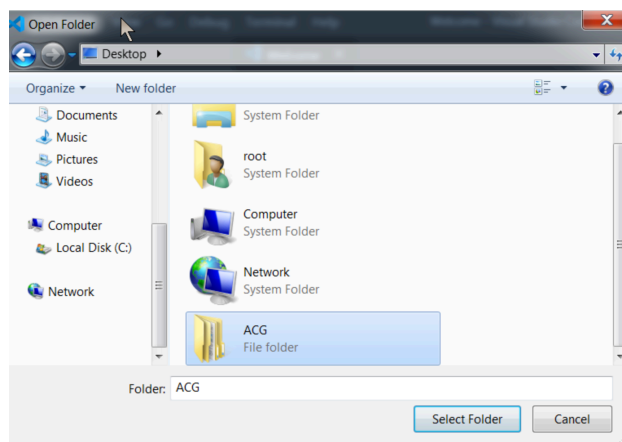
- b) Search for **Python** in the “Search Extensions” textbox. Select the first extension published by Microsoft and click **install**.



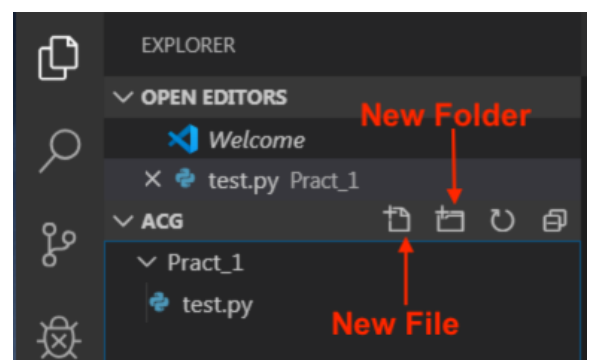
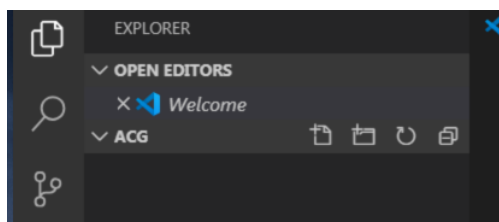
3. Create a directory to store all your Python codes for this module.
- a) Select the Explorer (or Ctrl+Shift+E) and click Open Folder.



- b) Create a new directory to store your files then click **Select Folder** bottom. (e.g. The **ACG** folder is on the Desktop. I selected **ACG** folder, then click **Select Folder** bottom).

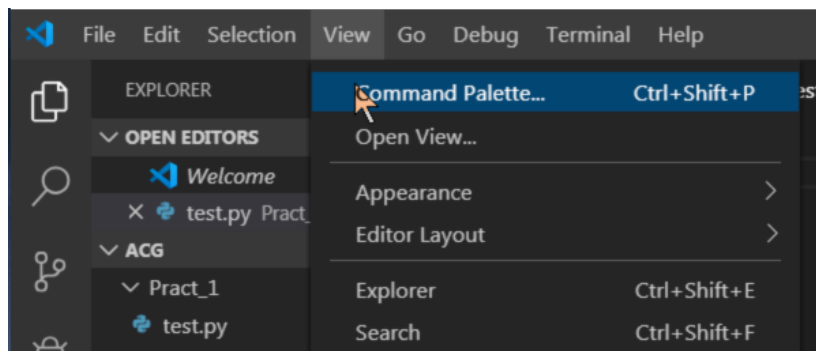


- c) Once the workspace folder is selected Create a new folder called **Pract_1** for your Practical 1 code files. Subsequently create a new python file within **Pract_1** folder called **test.py**.

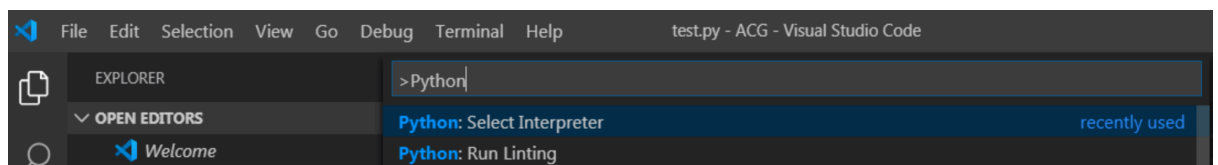


4. Select Python interpreter

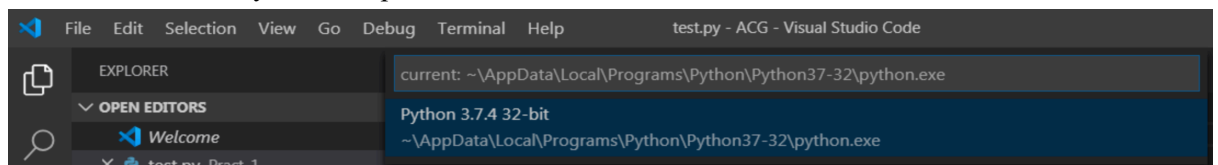
- a) Open the **Command** palette (or Ctrl+Shift+P) under the **View** menu.



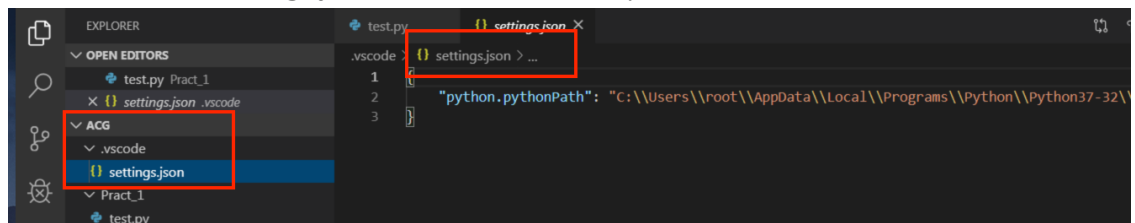
- b) Type **Python: Select Interpreter**.



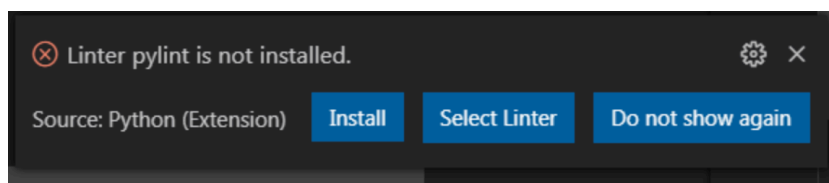
- c) Select the installed Python interpreter.



- d) A new file called **settings.json** will be included into your folder.



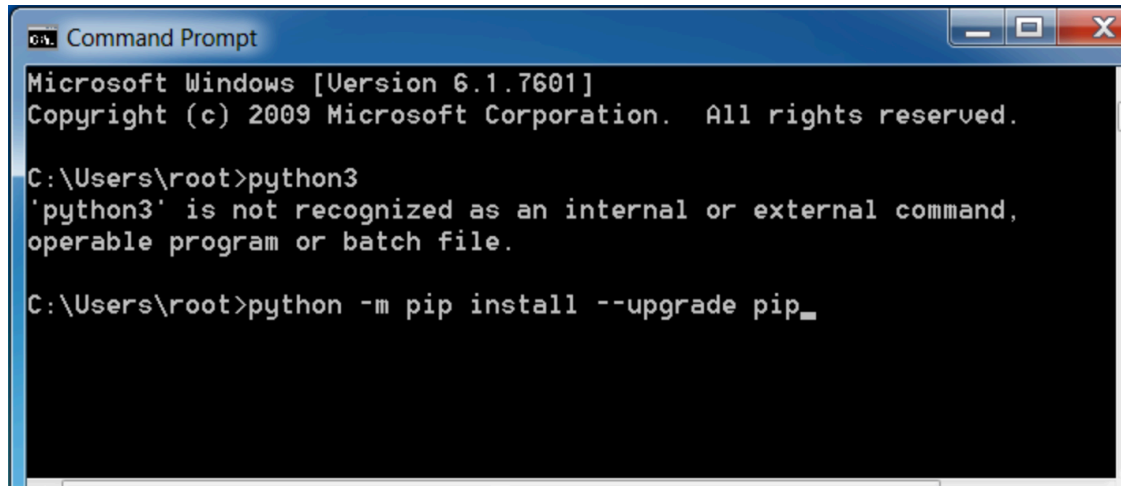
- e) Linting highlights some syntactical problems in the Python source code to help identify and correct subtle programming errors. If you see this message below click **Install**.



You have installed support for Python coding on VS Code.

Install 3rd party packages

- Open a Command Prompt (cmd), type the command **python -m pip install --upgrade pip** to update the pip tool.



```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\root>python3
'python3' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\root>python -m pip install --upgrade pip_
```

- Type the command **pip3 install pycryptodomex** to install the pycryptodomex library.
- Type the command **python -m Cryptodome.SelfTest** to test if the pycryptodomex library is installed correctly. The testing will take a while and ends with the following messages.



```
.....
-----
Ran 40277 tests in 290.170s

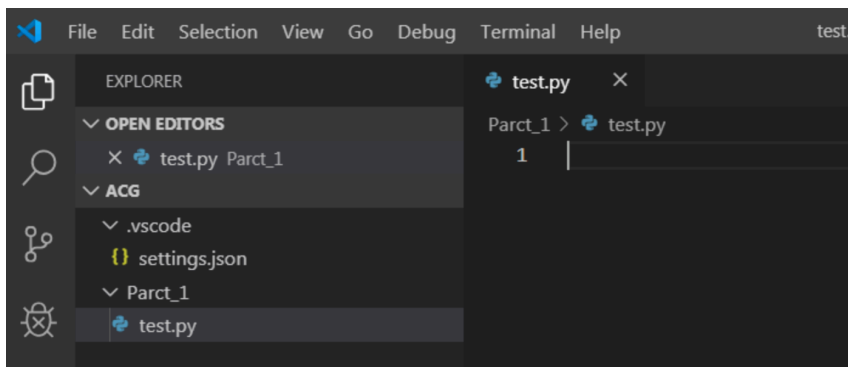
OK

C:\Users\root>
```

- Type the command **pip3 install cryptography** to install the **cryptography** library.
- Close the command prompt and you are done 3rd party package installation.

Your 1st ACG program

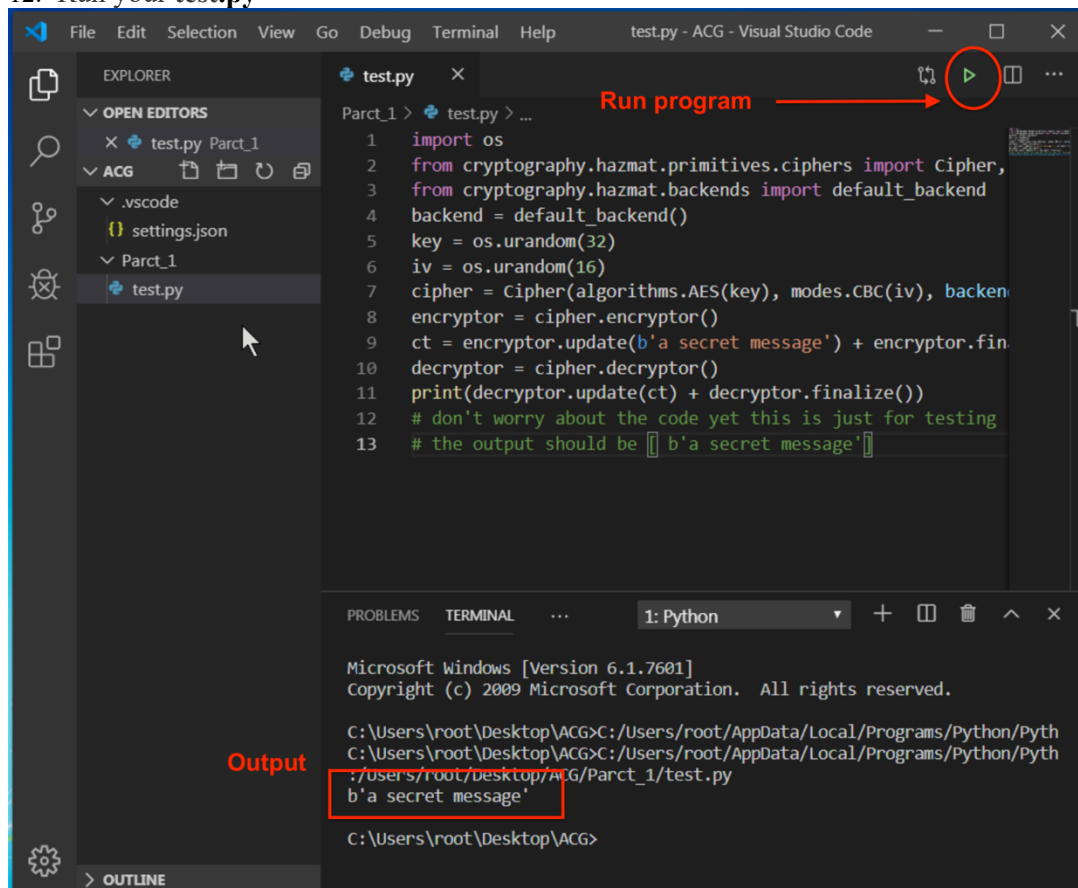
- Start up your VS code, **Open Folder**, select your ACG folder.



11. Copy and paste the following code into **test.py**

```
import os
from cryptography.hazmat.primitives.ciphers import Cipher, algorithms, modes
from cryptography.hazmat.backends import default_backend
backend = default_backend()
key = os.urandom(32)
iv = os.urandom(16)
cipher = Cipher(algorithms.AES(key), modes.CBC(iv), backend=backend)
encryptor = cipher.encryptor()
ct = encryptor.update(b'a secret message') + encryptor.finalize()
decryptor = cipher.decryptor()
print(decryptor.update(ct) + decryptor.finalize())
# don't worry about the code yet this is just for testing
# the output should be [ b'a secret message']
```

12. Run your **test.py**



End of the practical