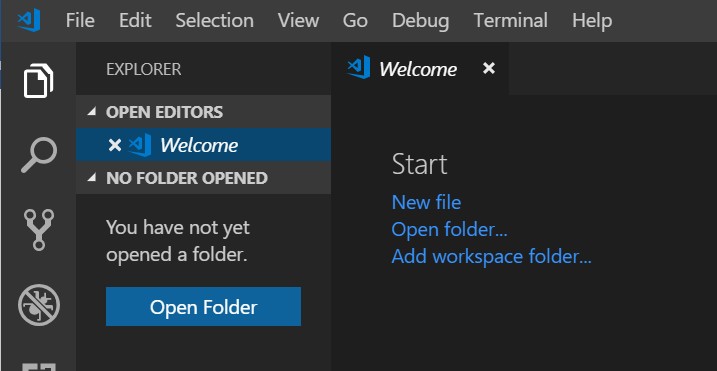
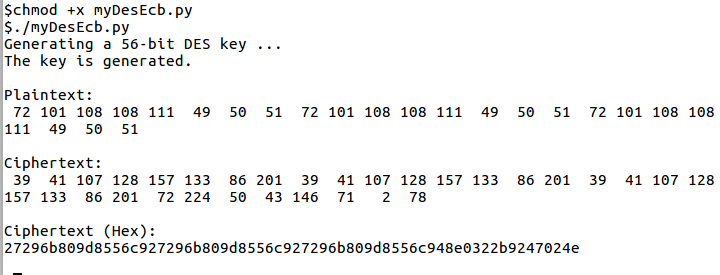
**Objectives**: By the end of this practical exercise, the students should be able to:

* Call the encrypt() and decrypt() functions of pycryptodome package.
* Understand the application aspect of symmetric ciphers .

1. Create a **Pract\_2** folder in your ACG folder. (Assuming ACG folder is on your Desktop, the full file path will be **C:\Users\<<user\_name>>\Desktop\ACG\Pract\_2** where <<user\_name>> is your login name)
2. Download the python program source, **myDesEcb.py,** from the Black Board to your **Pract\_2** folder.
3. Open the **Pract\_2** folder with your Visual Studio Code.



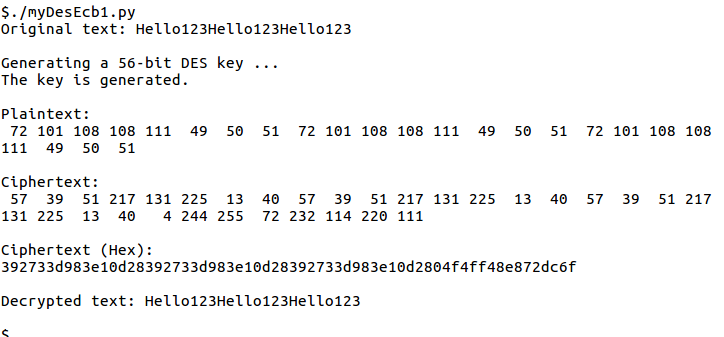
1. Run the **myDesEcb.py**" with the Play button  .
2. The following will be printed



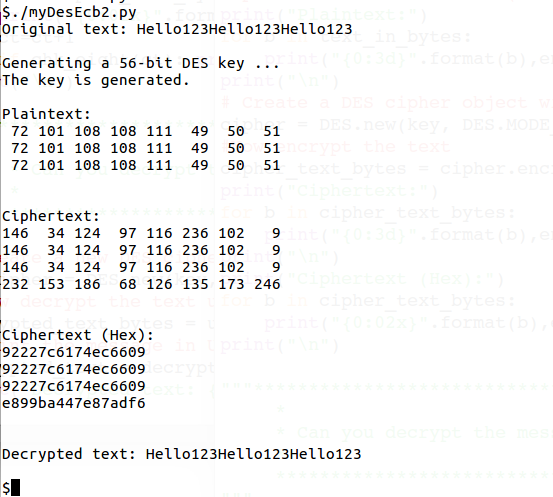
1. Try running the **myDesEcb.py** program again.
2. Did you observe the ciphertexts generated from the 1st run and 2nd run are different? What is the reason?
3. The encryption scheme of this sample program is DES with a random 56-bit key; ECB mode, block size is 32 bytes; padding style:pkcs7 (default).

Task Requirements:

1. Study the python code in the "**myDesEcb.py**" program. You are required to modify from it and create the "**myDesEcb1.py**" program to produce additional lines to display the decrypted cipher text. (It equals to the original text).



1. To improve the readability of the output of the program, modify your completed program in question 1 to include a function which checks for the number of printed bytes. If it is in a multiple of eight bytes (64 bits), insert a “carriage return” (or "\n" character) to move the next block of output to the next line. A sample of the output is shown below:



Hint: Use the **chk\_eight** function (line 8-11) given.

~ The End ~