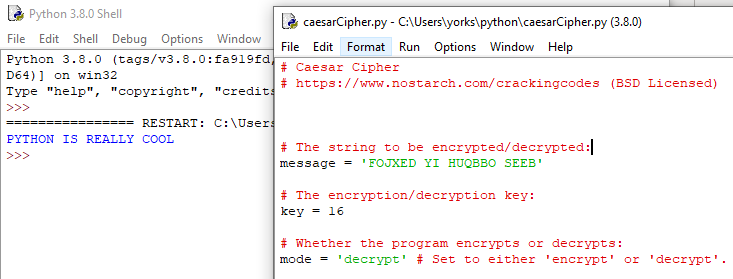
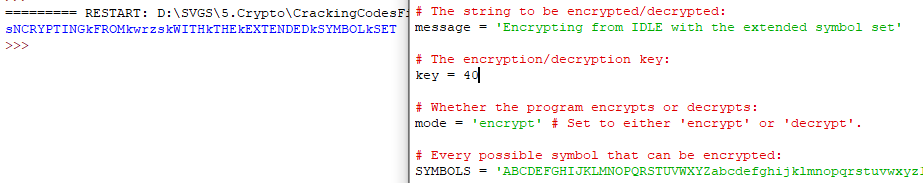
# Cryptography Homework 2a—Create your Python Environment—KEY

## For Turn in

1. Open the file from CrackingCodes.zip, caesarCipher.py and run it. If you haven’t already, remove the pyperclip mentions from the file to prevent errors.
   1. To show that you have a working version of Python, execute caesarCipher.py either from Idle or a terminal (either Linux or Windows, your choice) and hand in a screenshot.  
      
   2. Edit the message and the key in caesarCipher.py to encrypt a message of your choosing. Then change to decrypt mode and decrypt your message. Hand in a screenshot.   
      See above.
   3. In caesarCipher.py, the symbol set is expanded to include upper and lower case, numbers, and some punctuation symbols.   
      SYMBOLS = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890 !?.'  
      The caesarCipher.py code has a problem, which you can see by encrypting a message with the key = 40. What happened? How can you fix it?  
        
      When the key is 40, the lower-case letters in the plaintext become upper-case versions of the same letters in ciphertext. A similar thing happens with key 26. One solution is not to use keys 26 or 40. Another solution would be to jumble the order of the characters in the SYMBOLS variable; it becomes a substitution cipher and is no longer a shift cipher, though.
2. The affineCipher.py script also uses an expanded symbol set.  
   SYMBOLS = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890 !?.'
   1. This symbol set is not a field, because not all elements have a multiplicative inverse. What caused that to happen? (Hint: Look at the length of SYMBOLS.)  
        
      There are 66 characters in the symbol set, so any key that contains the factors of 66 (2, 3, and 11) will not have an inverse and will fail.
   2. Add or remove characters from SYMBOLS so that all elements of SYMBOLS have multiplicative inverses. Again, len(SYMBOLS) is the key.
      1. What SYMBOLS set did you choose, and what is its length?
      2. Encrypt a message. Hand in the encrypted message and the key you used.  
         We need to make the length of SYMBOLS equal a prime number. The closest primes to 66 are 59, 61, 67, and 71. I’ll choose 67, and add one character, a colon, to the list.

