**DES AES Code (Python Language):**

*'''*

*Created on 6th Aug. 2018*

**@coder:** *Krishna Attal .*

*'''*

# Creating DES and AES Objects

# Converting binary to ASCII and Adding time delay to compare speed of DES and AES

from Crypto.Cipher import DES

from Crypto.Cipher import AES

import binascii

import time

# Command to enter any plain text of eight characters for Data Encryption Standard

# Command to enter any key of eight characters for Data Encryption Standard

msgDES = raw\_input("Please enter any plaintext for DES encryption (8 characters):")

keyDES = raw\_input("Please enter any DES key (8 characters)(type hex and enter for hex):")

# If statement to generate key for DES using binary to hex conversion .

# Getting DES key in output by adding key DES command.

if keyDES=="hex":

keyDES = raw\_input("Please enter any DES key in hex (16 characters):")

#print binascii.hexlify(keyDES)

keyDES = binascii.unhexlify(keyDES)

# Command to enter any plain text of eight characters for Advanced Encryption Standard

# Command to enter any key of eight characters for Advanced Encryption Standard

msgAES = raw\_input("Please enter any plaintext for AES encryption (16 characters):")

keyAES = raw\_input("Please enter any AES key (16 characters)(type hex and enter for hex):")

# If statement to generate key for AES using binary to hex conversion .

# Getting AES key in output by adding key AES command.

if keyAES=="hex":

keyAES = raw\_input("Please enter AES key in hex (32 characters):")

#print binascii.hexlify(keyAES)

keyAES = binascii.unhexlify(keyAES)

# Adding time command to check speed of DES with both plaintext and it’s key.

#Defining plaintext, ciphertext i.e. key, time for Data Encryption Standard.

startDES = time.time()

for i in range(1, 1000000):

cipher = DES.new(keyDES)

def encryptDES(plaintext):

global cipher

return cipher.encrypt(plaintext)

def decryptDES(ciphertext):

global cipher

dec=cipher.decrypt(ciphertext)

return dec

endDES = time.time()

# Adding time command to check speed of AES with both plaintext and it’s key.

#Defining plaintext, ciphertext i.e. key, time for Advanced Encryption Standard.

startAES = time.time()

for i in range(1, 1000000):

cipherAES = AES.new(keyAES)

def encryptAES(plaintextAES):

global cipherAES

return cipherAES.encrypt(plaintextAES)

def decryptAES(ciphertextAES):

global cipherAES

dec=cipherAES.decrypt(ciphertextAES)

return dec

endAES = time.time()

# Getting Output for encryption, decryption and time of DES using print function.

# Getting Output for encryption, decryption, and AES using print function.

if \_\_name\_\_ =='\_\_main\_\_':

print 'message DES:', msgDES

print 'message AES:', msgAES

encryptedDES = binascii.hexlify(encryptDES(msgDES))

decryptDES = decryptDES(binascii.unhexlify(encryptedDES))

print 'ecrypted DES:', encryptedDES

print 'decrypted DES:', decryptDES

print 'time DES:', endDES - startDES

encryptedAES = binascii.hexlify(encryptAES(msgAES))

decryptAES = decryptAES(binascii.unhexlify(encryptedAES))

print 'ecrypted AES:', encryptedAES

print 'decrypted AES:', decryptAES

print 'time AES:', endAES – startAES

**OUTPUT:**

Please enter plaintext for DES encryption (8 characters):

KRISHNAA

Please enter DES key (8 characters)type hex and enter for hex):GHIJKLMN

Please enter plaintext for AES encryption (16 characters):KRISHNAATTALKEY

Please enter AES key (16 characters)(type hex and enter for hex):GHIJKLMNOPQRSTUV

message DES: NIKHILVR

message AES: NIKHILVRODRIGUES

ecrypted DES: 60e5b62622880b6b

decrypted DES: NIKHILVR

time DES: 3.58026599884

ecrypted AES: 0693c26e5048a07df5b39cd7911eb0e7

decrypted AES: NIKHILVRODRIGUES

time AES: 1.46165800095