In [ ]:
```
Practical No. : 03
Title : Write a Java/C/C++/Python program to implement DES algorithm.
Program Code with output
```

In [1]:
```
pip install PyCryptodome
```

```
Collecting PyCryptodome
  Downloading pycryptodome-3.17-cp35-abi3-win_amd64.whl (1.7 MB)
Installing collected packages: PyCryptodome
Successfully installed PyCryptodome-3.17
Note: you may need to restart the kernel to use updated packages.
```

In [1]:
```
from Crypto.Cipher import DES
```

In [2]:
```
from secrets import token_bytes
```

In [3]:
```
def encrypt(msg,key):

  cipher = DES.new(key,DES.MODE_EAX)
  nonce = cipher.nonce

  cipher_text , tag = cipher.encrypt_and_digest(msg.encode("ascii"))

  return cipher_text , tag , nonce
```

In [4]:
```
def decrypt(cipher_text , tag , nonce):

  cipher = DES.new(key,DES.MODE_EAX,nonce = nonce)

  plain_text = cipher.decrypt(cipher_text)

  try:
    cipher.verify(tag)
    return plain_text.decode("ascii")
  except:
    return False
```

In [5]:
```
key = token_bytes(8) # 8 byte = 64 bit key
```

In [6]:
```
cipher_text,tag,nonce=encrypt(input("Enter plain text : "),key)

plain_text = decrypt(cipher_text,tag,nonce)
```

```
Enter plain text : Hello Programmer
```

In [7]:
```
if(plain_text==False):
  print("Message has been corrupted")
else:
  print(f'\nPlaintext : {plain_text}')
  print(f'Key : {key}')
  print(f'Ciphertext : {cipher_text}')
```

```
Plaintext : Hello Programmer
Key : b'\x17\xe5Q\xdb\x00\xc8\xb83'
Ciphertext : b'\xacD\xe8\xbc\xc1\xf3\xf6(\xaa\xde\xda\xb8}\x1c\x10w'
```