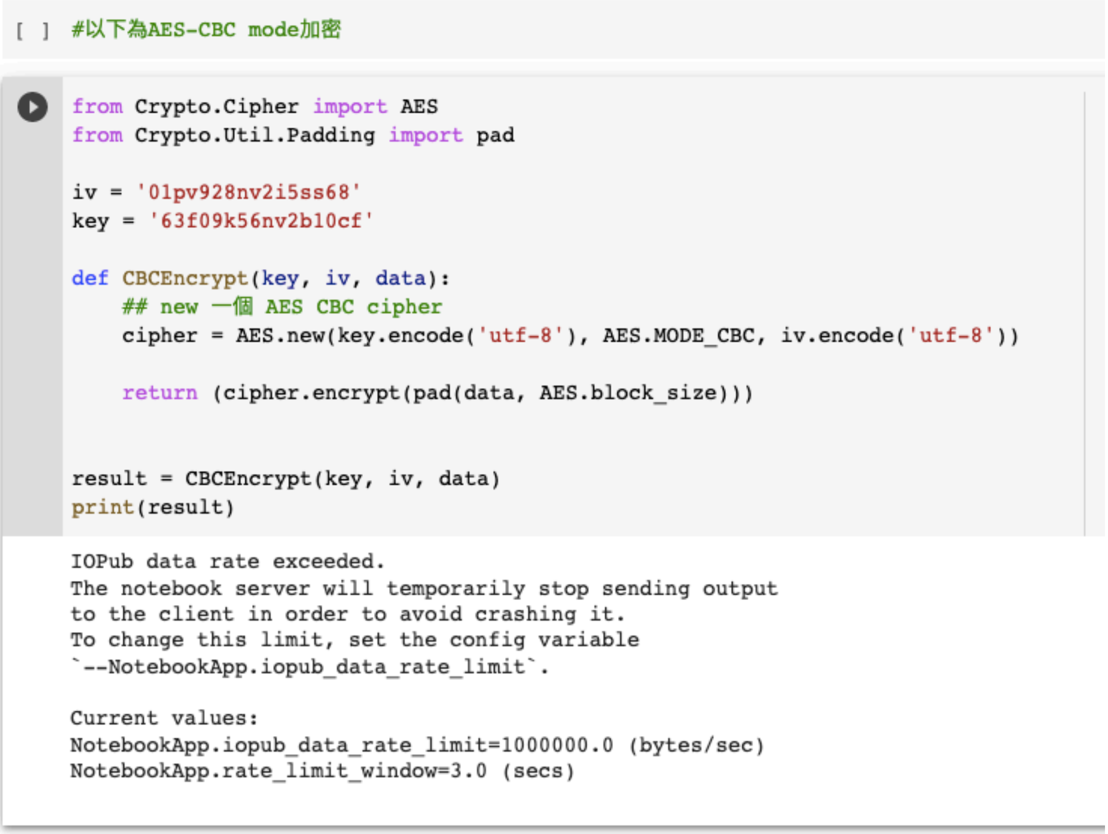
(1) 原始程式碼與說明被加密的檔案大小

原始程式碼以附件的方式一起繳交～
加密檔案大小：100MB
（python 語言，以下測試環境在 colab 執行，不清楚用什麼檔方便助教看，所以.py 與.ipynb 都有上傳附件，謝謝助教
檔案內含 3 種加密方式，照順序的程式碼

(2) 分別執行以上三種加密方式的速度 (每秒可加密多少 bytes)

I. 使用 AES-CBC mode 加密
每秒可加密 1000000.0 bytes

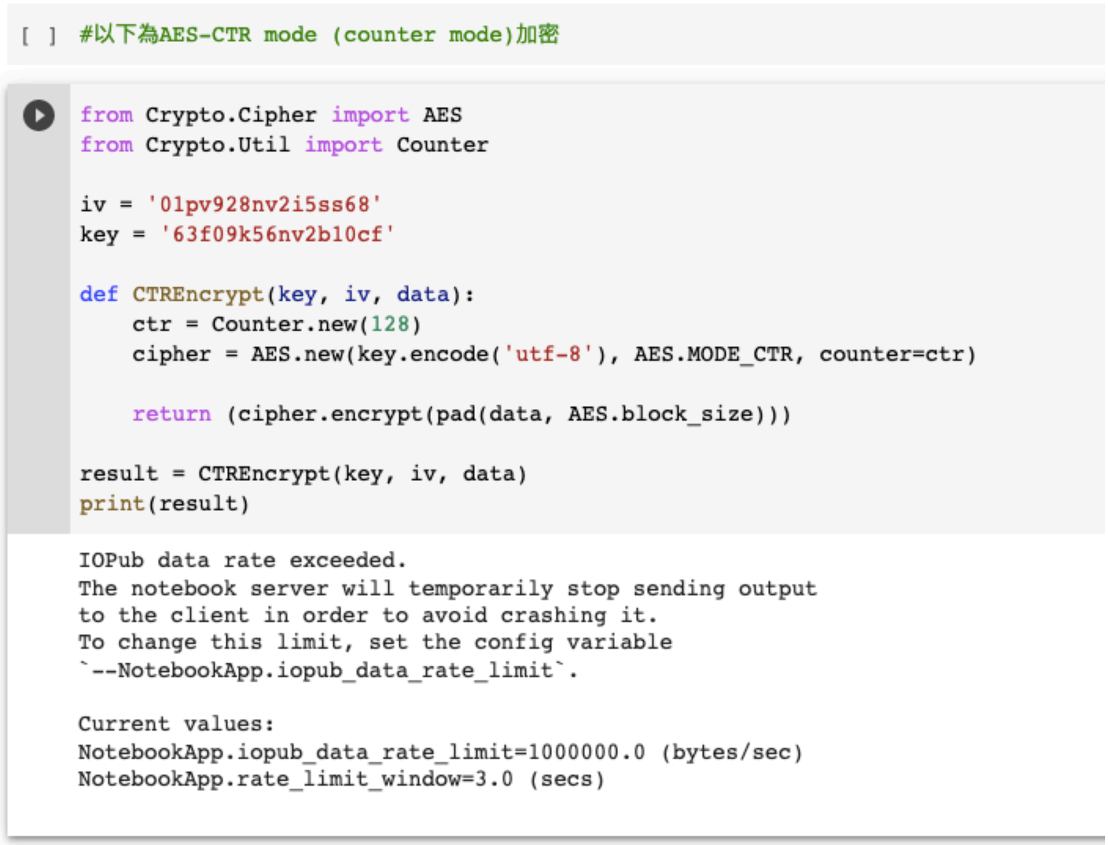


```
#以下為AES-CBC mode加密
```

```python
from Crypto.Cipher import AES
from Crypto.Util.Padding import pad

iv = '01pv928nv2i5ss68'
key = '63f09k56nv2b10cf'

def CBCEncrypt(key, iv, data):
    ## new 一個 AES CBC cipher
    cipher = AES.new(key.encode('utf-8'), AES.MODE_CBC, iv.encode('utf-8'))

    return (cipher.encrypt(pad(data, AES.block_size)))


result = CBCEncrypt(key, iv, data)
print(result)
```

```
IOPub data rate exceeded.
The notebook server will temporarily stop sending output
to the client in order to avoid crashing it.
To change this limit, set the config variable
`--NotebookApp.iopub_data_rate_limit`.

Current values:
NotebookApp.iopub_data_rate_limit=1000000.0 (bytes/sec)
NotebookApp.rate_limit_window=3.0 (secs)
```

II. 使用 AES-CTR mode (counter mode)加密
每秒可加密 1000000.0 bytes



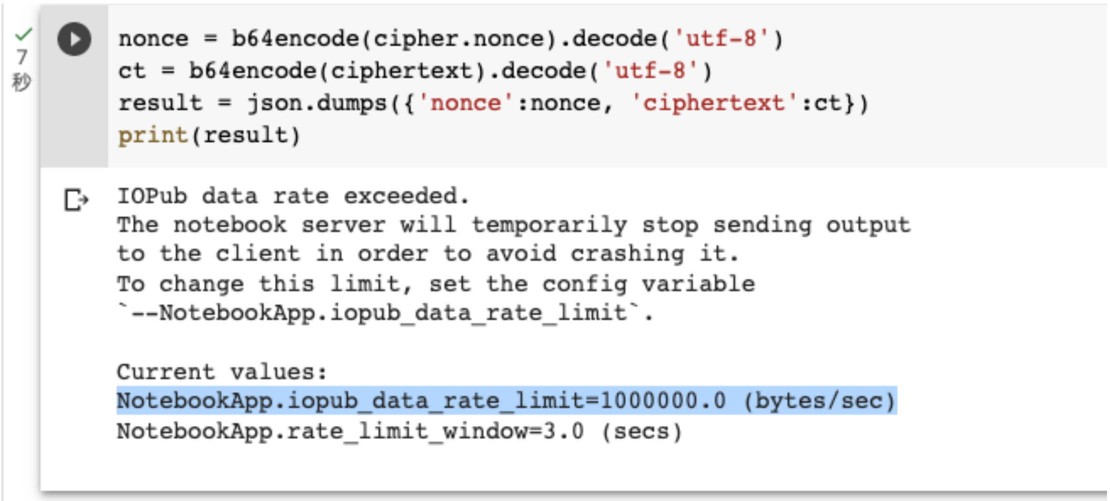
```
[ ]  #以下為AES-CTR mode (counter mode)加密
```

```
from Crypto.Cipher import AES
from Crypto.Util import Counter

iv = '01pv928nv2i5ss68'
key = '63f09k56nv2b10cf'

def CTREncrypt(key, iv, data):
    ctr = Counter.new(128)
    cipher = AES.new(key.encode('utf-8'), AES.MODE_CTR, counter=ctr)

    return (cipher.encrypt(pad(data, AES.block_size)))

result = CTREncrypt(key, iv, data)
print(result)
```

```
IOPub data rate exceeded.
The notebook server will temporarily stop sending output
to the client in order to avoid crashing it.
To change this limit, set the config variable
`--NotebookApp.iopub_data_rate_limit`.

Current values:
NotebookApp.iopub_data_rate_limit=1000000.0 (bytes/sec)
NotebookApp.rate_limit_window=3.0 (secs)
```

III. 使用 ChaCha20 加密

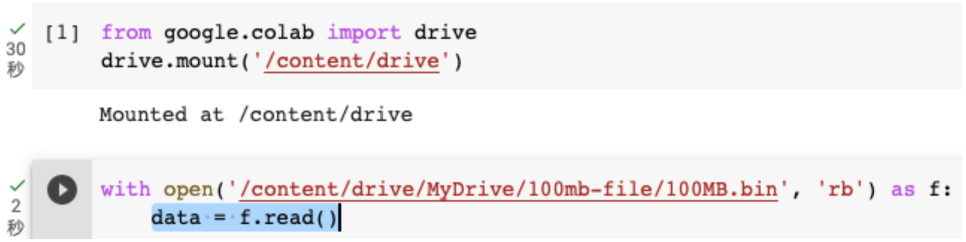
每秒可加密 1000000.0 bytes



```
nonce = b64encode(cipher.nonce).decode('utf-8')
ct = b64encode(ciphertext).decode('utf-8')
result = json.dumps({'nonce':nonce, 'ciphertext':ct})
print(result)
```

```
IOPub data rate exceeded.
The notebook server will temporarily stop sending output
to the client in order to avoid crashing it.
To change this limit, set the config variable
`--NotebookApp.iopub_data_rate_limit`.

Current values:
NotebookApp.iopub_data_rate_limit=1000000.0 (bytes/sec)
NotebookApp.rate_limit_window=3.0 (secs)
```

(3)比較解密後的檔案與原始檔案，證明實作正確

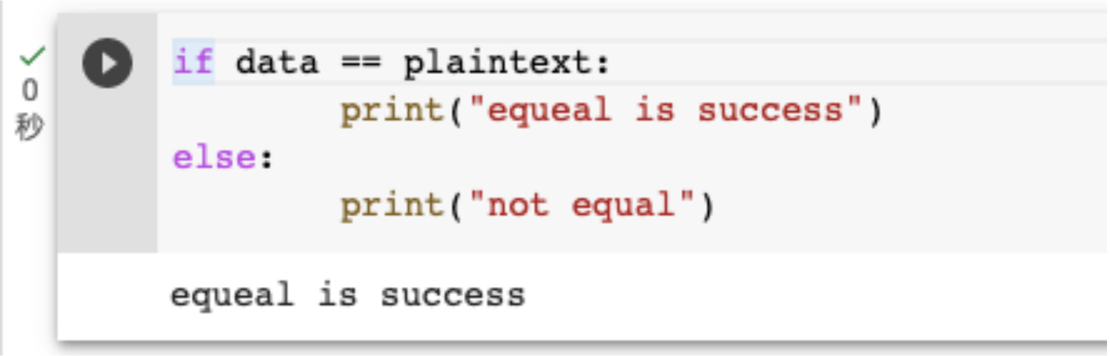


```
[1]  from google.colab import drive
     drive.mount('/content/drive')

     Mounted at /content/drive
```

```
with open('/content/drive/MyDrive/100mb-file/100MB.bin', 'rb') as f:
    data = f.read()
```

程式中，data 為原始的檔案二進位內容、plaintext 為解密後內容

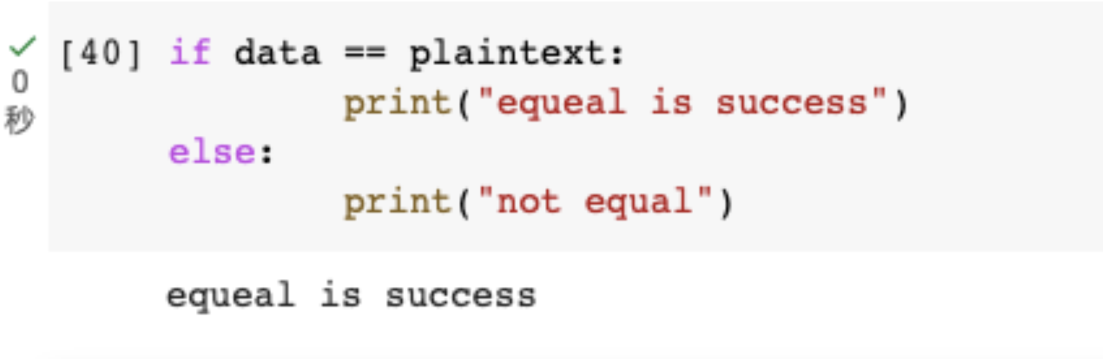
I. 使用 AES-CBC mode 加密



```
if data == plaintext:
        print("equeal is success")
else:
        print("not equal")
```

```
equeal is success
```

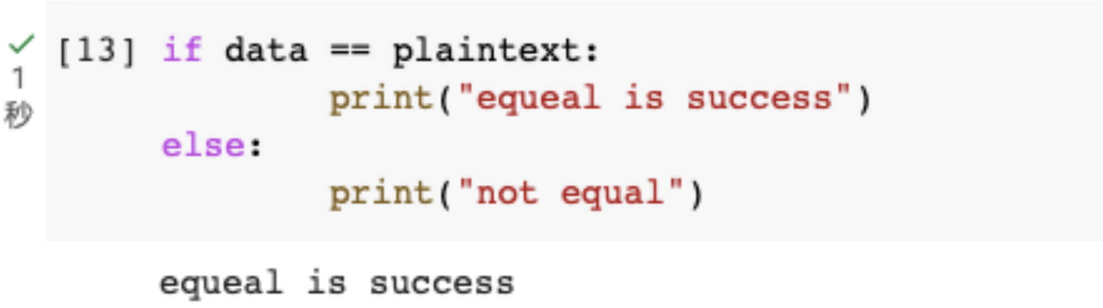
II. 使用 AES-CTR mode (counter mode)加密



```
[40] if data == plaintext:
             print("equeal is success")
     else:
             print("not equal")
```

```
equeal is success
```

III. 使用 ChaCha20 加密



```
[13] if data == plaintext:
             print("equeal is success")
     else:
             print("not equal")
```

```
equeal is success
```