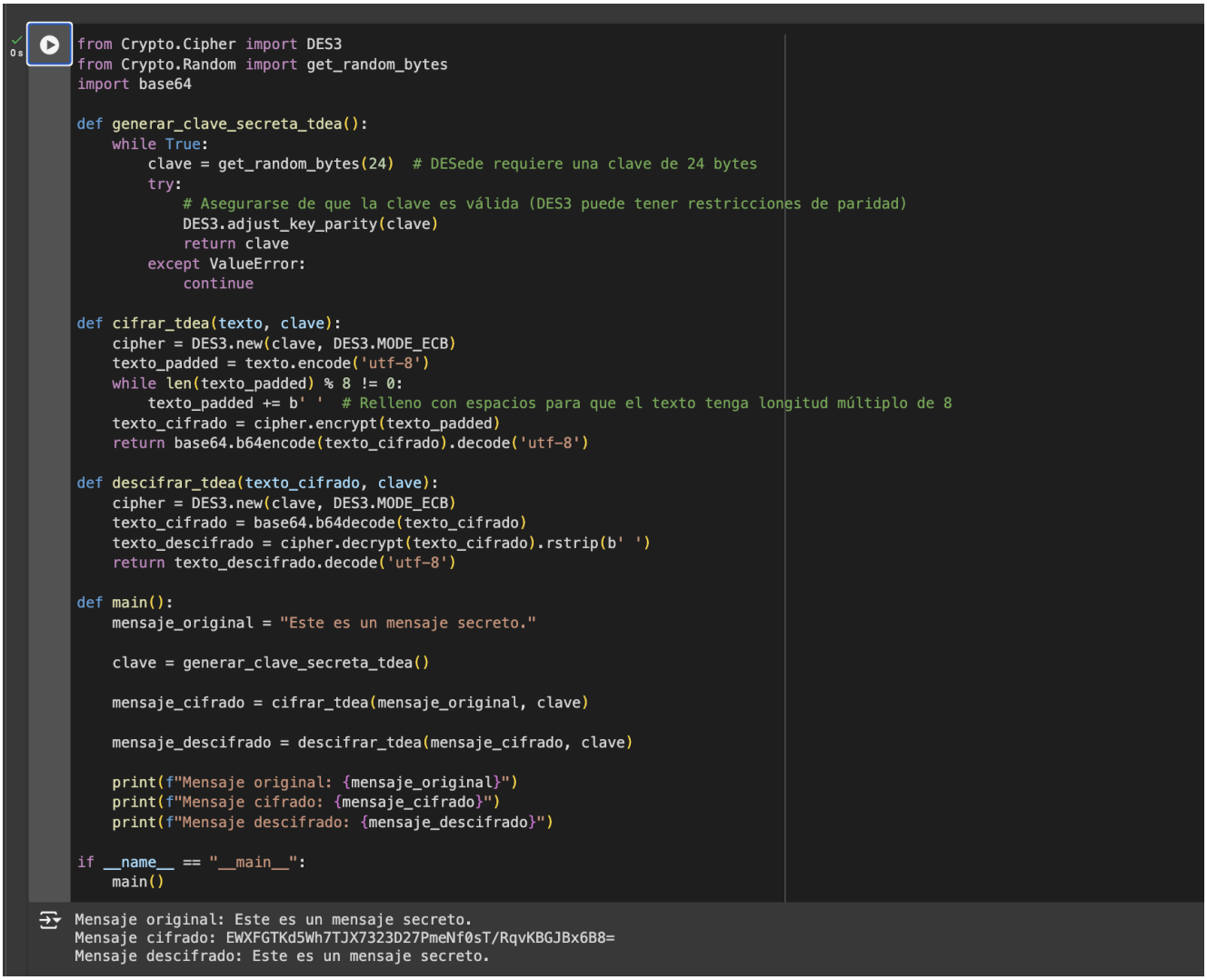
Cifrado simétrico TDEA (Triple Data Encryption Algorithm) en Python

[crayon-6a9d5f3a82c5f505105798/]



```python
from Crypto.Cipher import DES3
from Crypto.Random import get_random_bytes
import base64

def generar_clave_secreta_tdea():
    while True:
        clave = get_random_bytes(24)  # DESede requiere una clave de 24 bytes
        try:
            # Asegurarse de que la clave es válida (DES3 puede tener restricciones de paridad)
            DES3.adjust_key_parity(clave)
            return clave
        except ValueError:
            continue

def cifrar_tdea(texto, clave):
    cipher = DES3.new(clave, DES3.MODE_ECB)
    texto_padded = texto.encode('utf-8')
    while len(texto_padded) % 8 != 0:
        texto_padded += b' '  # Relleno con espacios para que el texto tenga longitud múltiplo de 8
    texto_cifrado = cipher.encrypt(texto_padded)
    return base64.b64encode(texto_cifrado).decode('utf-8')

def descifrar_tdea(texto_cifrado, clave):
    cipher = DES3.new(clave, DES3.MODE_ECB)
    texto_cifrado = base64.b64decode(texto_cifrado)
    texto_descifrado = cipher.decrypt(texto_cifrado).rstrip(b' ')
    return texto_descifrado.decode('utf-8')

def main():
    mensaje_original = "Este es un mensaje secreto."

    clave = generar_clave_secreta_tdea()

    mensaje_cifrado = cifrar_tdea(mensaje_original, clave)

    mensaje_descifrado = descifrar_tdea(mensaje_cifrado, clave)

    print(f"Mensaje original: {mensaje_original}")
    print(f"Mensaje cifrado: {mensaje_cifrado}")
    print(f"Mensaje descifrado: {mensaje_descifrado}")

if __name__ == "__main__":
    main()
```

```
Mensaje original: Este es un mensaje secreto.
Mensaje cifrado: EWXFGTKd5Wh7TJX7323D27PmeNf0sT/RqvKBGJBx6B8=
Mensaje descifrado: Este es un mensaje secreto.
```