

# Python 寄email

# 為甚麼要自動寄email?

如果我們需要大量的寄送郵件  
總不能一個一個複製

# 設定寄出郵件的google帳號



<https://myaccount.google.com/security?pli=1>

# 設定完成



首頁



個人資訊



資料和隱私權



安全性



使用者和分享內容

## 您登入 Google 的方式

請定期更新這些資訊，確保您隨時都能順利登入自己的 Google 帳戶



兩步驟驗證



啟用時間：清晨7:20



密碼金鑰

開始使用密碼金鑰



# 設定密碼

## ← 應用程式密碼

應用程式密碼可讓您在支援兩步驟驗證的裝置上，透過應用程式登入您的 Google 帳戶。應用程式密碼只要輸入一次即可，因此您不必記住這組密碼。 [瞭解詳情](#)

您沒有任何應用程式密碼。

選取您要產生應用程式密碼的應用程式和裝置。

選取應用程式

郵件

日曆

聯絡人

YouTube

其他 (自訂名稱)

選取裝置

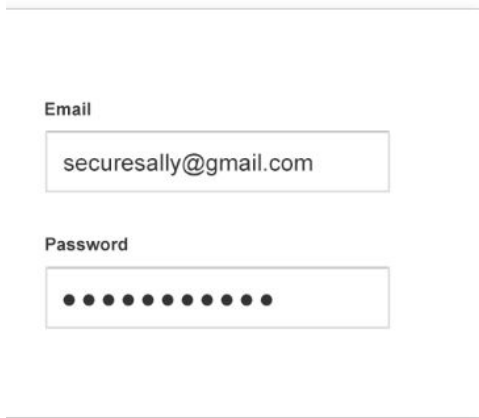


產生

<https://myaccount.google.com/apppasswords>

# 獲得密碼

## 系統產生的應用程式密碼



Email

securesally@gmail.com

Password

● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●

您裝置專用的應用程式密碼



使用方式

在您想設定 Google 帳戶的應用程式或裝置上前往帳戶的「設定」頁面，然後將您的密碼替換成上方的 16 字元密碼。

這個應用程式密碼就如同您平常使用的密碼，可授予完整的 Google 帳戶存取權限。您不需要記住這組密碼，因此，請勿將密碼寫下或透露給任何人知道。

完成

## 接下來，送email的時候到了

```
import smtplib, ssl
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart
from email.header import Header

sender_email = "jufengwul02000@gmail.com"
receiver_email = "leowul02000@gmail.com"
password = " "

msg = MIMEMultipart()
msg['From'] = sender_email
msg['To'] = receiver_email
msg['Subject'] = Header('Test Send Email', 'utf-8').encode()

body = 'This is email from python'

msg_content = MIMEText(body)
msg.attach(msg_content)

# Create secure connection with server and send email
context = ssl.create_default_context()
with smtplib.SMTP_SSL("smtp.gmail.com", 465, context=context) as server:
    # 465是標準的SMTP-over-SSL端口
    server.login(sender_email, password)
    server.sendmail(sender_email, receiver_email, msg.as_string())

print("success send email")
```

send\_email.py

## 接下來，我們希望可以增加附件

```
with open(pdf_path, 'rb') as f:
    part = MIMEBase('application', 'octet-stream') #MIMEBase 類別允許你創建包含不同類型附件 (例
    #"application" 是主類型，用於表示應用程序類型的文件。
    #"octet-stream" 是子類型，表示數據是未知類型的二進制數據。
    part.set_payload(f.read())
    encoders.encode_base64(part)
    part.add_header('Content-Disposition', 'attachment; filename="test_pdf.pdf"')
    msg.attach(part)

txt_path = "test.txt"
with open(txt_path, 'rb') as f:
    part = MIMEBase('application', 'octet-stream')
    part.set_payload(f.read())
    encoders.encode_base64(part)
    part.add_header('Content-Disposition', 'attachment; filename="test.txt"')
    msg.attach(part)

msg_content = MIMEText(body)
msg.attach(msg_content)

# Create secure connection with server and send email
context = ssl.create_default_context()
with smtplib.SMTP_SSL("smtp.gmail.com", 465, context=context) as server:
    server.login(sender_email, password)
    server.sendmail(sender_email, receiver_email, msg.as_string())
```



# 寄送word的秘訣

直接寄送也可以

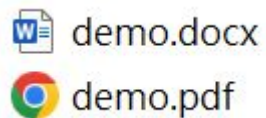
```
body = 'This is email from python'

doc_path = "this_is_test_word.docx"
with open(doc_path, 'rb') as f:
    part = MIMEBase('application', 'octet-stream') #MIMEBase 類別允許你創建包含不同類型附件（例如文本、圖
    #"application" 是主類型，用於表示應用程序類型的文件。
    #"octet-stream" 是子類型，表示數據是未知類型的二進制數據。
    part.set_payload(f.read())
    encoders.encode_base64(part)
    part.add_header('Content-Disposition', 'attachment; filename="docx.docx"')
    msg.attach(part)
```

但是會建議轉成pdf

# word轉pdf

```
from docx2pdf import convert  
convert("demo.docx", "demo.pdf")
```



pip install docx2pdf

## 我們希望可以讓內容更多元

```
body = """
<p>Python 測試email...</p>
<p><a href="http://https://tw.fukuro-art-tech.com/">這是連結</a></p>
"""

msg_content = MIMEText(body, 'html')
msg.attach(msg_content)

# Create secure connection with server and send email
context = ssl.create_default_context()
with smtplib.SMTP_SSL("smtp.gmail.com", 465, context=context) as server:
    server.login(sender_email, password)
    server.sendmail(sender_email, receiver_email, msg.as_string())

print("success send email")
```