Python Gmail Email Program

Contents

Time required: 30 minutes

Let's learn how to send email using Python and Gmail. Being able to send an email from Python opens up all sorts of possibilities. We will use this email program in later programs.

Setting up a Gmail Account for Email

If you don't have a Gmail account, you will want to create one.

Let's enable your Gmail account to receive connections from external programs, like Python.

1. Open your browser and access your Gmail account.

2. On the login screen → enter your Gmail username and password.

3. After the login → access the following URL:
   https://myaccount.google.com/signinoptions/two-step-verification
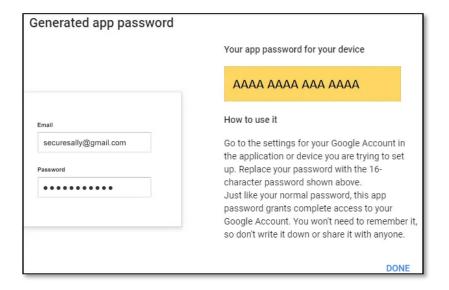
4. Enable the two-step verification on this account.

Create an Application Password

1. Access the following URL:
   https://security.google.com/settings/security/apppasswords

2. Select Gmail application and the type of device: **Other**.

3. Name the device: **Python**

4. Click on the Generate button and take note of the randomly generated password.

You have finished the required steps for the Gmail integration.

Tutorial: Send Email with Python

Create a Python program named: **gmail_credentials.py**

```python
"""
    Name: gmail_credentials.py
    Author:
    Created:
    Purpose: Credentials to send email through Python using Gmail
"""

SMTP_SERVER = "smtp.gmail.com"
# Secure SMTP port
PORT = 587

#---------- REPLACE WITH YOUR INFORMATION ------------#

LOGIN = "youremailaddress@gmail.com"
APP_PASSWORD = ""

#----------------------------------------------------#
```

Create a Python program named: **send_gmail.py**

```python
     Author:
     Created:
     Purpose: Send email with Python and Gmail
"""
# Library to manage communcation with an SMTP server
import smtplib
# Library to create an email message
from email.message import EmailMessage
# Import gmail credentials and SMTP server settings
import gmail_credentials


def main():
    # -------------- REPLACE WITH YOUR INFORMATION ------------------#
    email_from = "William Loring <williamaloring@gmail.com>"
    # A list containing one or more email addresses
    email_dst = [
        "loringw@wncc.edu",
        "williamloring@hotmail.com"
    ]
    subject = "Email from Python Email Program"
    content = """
    This test message is sent from Python Bill.
    """

    # -------------------- CREATE EMAIL MESSAGE --------------------#
    message = EmailMessage()
    message["From"] = email_from
    message.set_content(content)
    message["Subject"] = subject
    message["To"] = email_dst
```

```
36    # ------------------ SEND EMAIL MESSAGE ----------------------#
37    try:
38        # Use with context manager to create an smtp_server object
39        with smtplib.SMTP(
40            gmail_credentials.SMTP_SERVER,
41            gmail_credentials.PORT
42        ) as smtp_server:
43            # Show all communication with the server
44            # This line can be commented out
45            smtp_server.set_debuglevel(True)
46            # Say enhanced hello to the smtp server
47            smtp_server.ehlo()
48            # Request a TLS connection with the SMTP server
49            smtp_server.starttls()
50            # Login to the SMTP server
51            smtp_server.login(
52                gmail_credentials.LOGIN,
53                gmail_credentials.PASSWORD
54            )
55            # Ask smtp_server to send out message
56            smtp_server.send_message(
57                message
58            )
59        print()
60        print(25*"**")
61        print("    Email message successfully sent.")
62        print(25*"**")
63        print()
64    except Exception as e:
65        print(25*"**")
66        print(f"Message not sent.")
67        print(e)
68        print(25*"**")
69
70
71 main()
```

The example run contains all the chatter back and forth between your Python program and the mail server. The debug level is set to True so we can see how much background communication occurs sending a simple email message. The debug level line can be commented out.

Example run:

```
reply: retcode (250); Msg: b'2.1.0 OK ep25-20020a056870a99900b0011f390fdb0asm361
9396oab.12 - gsmtp'
send: 'rcpt TO:<loringw@wncc.edu>\r\n'
reply: b'250 2.1.5 OK ep25-20020a056870a99900b0011f390fdb0asm3619396oab.12 - gsm
tp\r\n'
reply: retcode (250); Msg: b'2.1.5 OK ep25-20020a056870a99900b0011f390fdb0asm361
9396oab.12 - gsmtp'
send: 'rcpt TO:<williamloring@hotmail.com>\r\n'
reply: b'250 2.1.5 OK ep25-20020a056870a99900b0011f390fdb0asm3619396oab.12 - gsm
tp\r\n'
reply: retcode (250); Msg: b'2.1.5 OK ep25-20020a056870a99900b0011f390fdb0asm361
9396oab.12 - gsmtp'
send: 'data\r\n'
reply: b'354  Go ahead ep25-20020a056870a99900b0011f390fdb0asm3619396oab.12 - gs
mtp\r\n'
reply: retcode (354); Msg: b'Go ahead ep25-20020a056870a99900b0011f390fdb0asm361
9396oab.12 - gsmtp'
data: (354, b'Go ahead ep25-20020a056870a99900b0011f390fdb0asm3619396oab.12 - gs
mtp')
send: b'From: William Loring <williamaloring@gmail.com>\r\nContent-Type: text/pl
ain; charset="utf-8"\r\nContent-Transfer-Encoding: 7bit\r\nMIME-Version: 1.0\r\n
Subject: Email from Python Email Program\r\nTo: loringw@wncc.edu, williamloring@
hotmail.com\r\n\r\n\r\nThis test message is sent from Python Bill.\r\n.\r\n'
reply: b'250 2.0.0 OK  1662293307 ep25-20020a056870a99900b0011f390fdb0asm3619396
oab.12 - gsmtp\r\n'
reply: retcode (250); Msg: b'2.0.0 OK  1662293307 ep25-20020a056870a99900b0011f3
90fdb0asm3619396oab.12 - gsmtp'
data: (250, b'2.0.0 OK  1662293307 ep25-20020a056870a99900b0011f390fdb0asm361939
6oab.12 - gsmtp')
send: 'QUIT\r\n'
reply: b'221 2.0.0 closing connection ep25-20020a056870a99900b0011f390fdb0asm361
9396oab.12 - gsmtp\r\n'
reply: retcode (221); Msg: b'2.0.0 closing connection ep25-20020a056870a99900b00
11f390fdb0asm3619396oab.12 - gsmtp'

**************************************************
    Email message successfully sent.
**************************************************
```

Test your program.

Assignment Submission

- Send the instructor an email at loringw@wncc.edu
- Insert a screenshot of the program run
- Attach the completed program files
- Submit the assignment in Blackboard.