# Automated Birthday Email Sender using PythonAnywhere

This document explains how to host, schedule, and monitor your birthday email script on PythonAnywhere.  
You can incorporate this as a reference or learning document in your project folder.

## 1. File Structure

Upload the following files to your PythonAnywhere project folder (/home/yourusername/birthday\_bot/):  
  
birthday\_bot/  
│  
├── birthdays.csv # Stores name, email, day, month, etc.  
├── letter\_templates/ # Folder with birthday message templates  
│ ├── letter\_1.txt  
│ ├── letter\_2.txt  
│ └── letter\_3.txt  
├── birthday\_mailer.py # Your main Python script  
└── README.md (optional) # Notes like these

## 2. Configure Your Script

Ensure your birthday\_mailer.py looks like this (using sendmail()):  
  
from datetime import datetime  
import pandas  
import random  
import smtplib  
  
MY\_EMAIL = 'your\_email@gmail.com'  
MY\_PASSWORD = 'your\_app\_password'  
  
today = datetime.now()  
today\_tuple = (today.month, today.day)  
  
data = pandas.read\_csv('/home/yourusername/birthday\_bot/birthdays.csv')  
birthdays\_dict = {(row["month"], row["day"]): row for (\_, row) in data.iterrows()}  
  
if today\_tuple in birthdays\_dict:  
 birthday\_person = birthdays\_dict[today\_tuple]  
 name = birthday\_person["name"]  
 file\_path = f"/home/yourusername/birthday\_bot/letter\_templates/letter\_{random.randint(1, 3)}.txt"  
 with open(file\_path) as file:  
 contents = file.read().replace("[NAME]", name)  
  
 message = f"Subject: Happy Birthday {name}!\n\n{contents}"  
  
 with smtplib.SMTP("smtp.gmail.com", 587) as conn:  
 conn.starttls()  
 conn.login(user=MY\_EMAIL, password=MY\_PASSWORD)  
 conn.sendmail(from\_addr=MY\_EMAIL, to\_addrs=birthday\_person["email"], msg=message)

## 3. Schedule a Daily Task (Using PythonAnywhere Scheduler)

1. Go to your PythonAnywhere dashboard.  
2. Click on 'Tasks' (left sidebar).  
3. Click 'Add a new scheduled task'.  
4. Set:  
 - Time: e.g., 09:00 UTC (India is UTC+5:30, so this = 14:30 IST)  
 **- Command:  
 python3.10 /home/yourusername/birthday\_bot/birthday\_mailer.py**  
5. Click 'Create'.

## 4. Checking Logs

To check for errors or outputs:  
  
1. Go to the Tasks page.  
2. Click on your scheduled task.  
3. View logs:  
 - output.log: Standard print output.  
 - error.log: Any errors or exceptions.  
 - log.txt: Scheduler logs.  
  
Tip: Use print() in your script for helpful logs, e.g.  
print(f"Sent birthday email to {birthday\_person['email']} on {today.strftime('%Y-%m-%d')}")

## 5. Gmail Security Notes

If using Gmail:  
- Enable 2-Factor Authentication.  
- Generate an App Password at https://myaccount.google.com/apppasswords.  
- Use this password in your script instead of your Gmail password.

## 6. Manual Testing from Bash Console

1. Go to the 'Consoles' tab in PythonAnywhere.  
2. Start a new 'Bash' console.  
3. Run the following:  
  
cd /home/yourusername/birthday\_bot/  
**python3 birthday\_mailer.py**

(In this project, we used the command as **python3 main\_model2.py**)  
  
Check the output in the console to confirm success or catch any errors.

## 7. Use Cases

- Automated birthday greetings  
- Sending discount coupons  
- Weekly/monthly email reports  
- Reminders and task notifications