

Integrating SendGrid

Service

SendGrid Integration with Python Code

Date	08 November, 2022
Team ID	PNT2022TMID00943
Project Name	Plasma Donor Application

Creating an API Key

The screenshot shows a web browser window with the URL `app.sendgrid.com/settings/api_keys`. The browser's tab bar includes several open tabs: "Meet - bnk-fdkr-yfd", "Installing the stand-alone IBM Cl...", "Inbox (4,368) - godsonraj@gmail.com", and "SendGrid". A notification banner at the top of the page states: "Before sending your first email, you'll need to create a sender identity. [Create a sender identity](#)".

The main content area is titled "API Keys" and features a "Create API Key" button in the top right corner. Below the title is a table with the following structure:

NAME	API KEY	ACTION
raj API Key ID: D8Y53524SymJBmPNK-oE5w	▾

A left-hand sidebar contains navigation links: "Dashboard", "Email API", "Marketing", "Design Library", "Stats", "Activity", "Suppressions", and "Settings". The user's profile "GODSON RAJ R" is visible at the top of this sidebar.

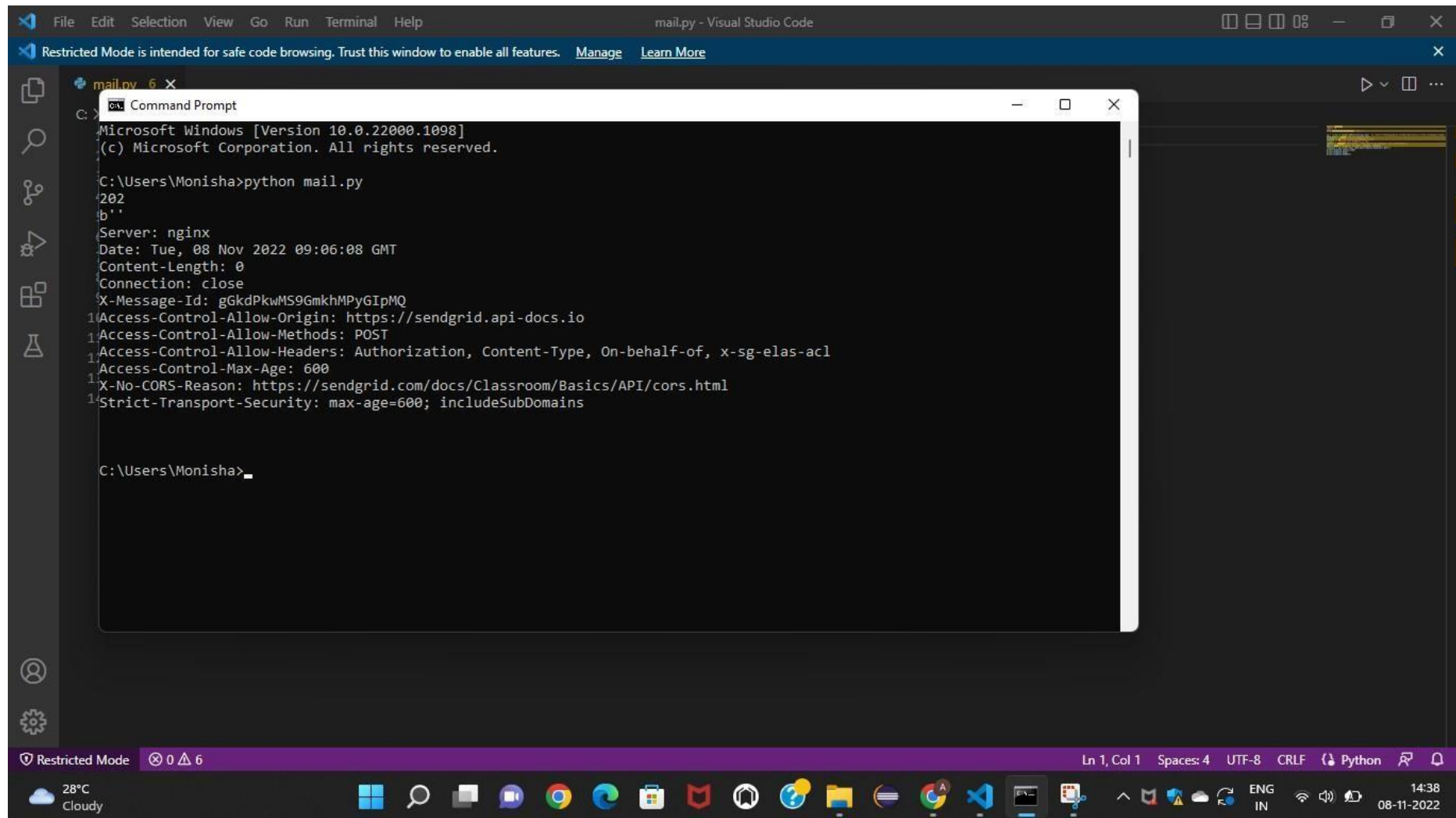
The Windows taskbar at the bottom of the screen shows the Start button, search icon, Dell logo, File Explorer, Google Chrome, and Microsoft Word. System status icons on the right include volume, network, and battery, along with the date and time: "11:10 PM 08-11-2022".

Creating a Virtual Environment “Email Test” and creating mail.py file with the following code inside the virtual environment

```
import sendgrid
import os
from sendgrid.helpers.mail import *

sg = sendgrid.SendGridAPIClient(api_key= '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' )
from_email = Email("godsonrajr@gmail.com")
to_email = To("godsonrajr01@gmail.com")
subject = "IBM project is in phase of final stage"
content = Content("text/plain", "and easy to do anywhere, even with Python")
mail = Mail(from_email, to_email, subject, content)
response = sg.client.mail.send.post(request_body=mail.get())
print(response.status_code)
print(response.body)
print(response.headers)
```

Running the python file through the terminal



The screenshot shows the Visual Studio Code interface with a file named `mail.py` open. A Command Prompt window is overlaid on the editor, displaying the output of running `python mail.py`. The output shows a 202 status code and various headers, including CORS-related headers. The status bar at the bottom indicates the file is in Restricted Mode, and the system tray shows the date and time as 14:38 on 08-11-2022.

```
mail.py - Visual Studio Code
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

mail.py 6 X
C:\Users\Monisha>python mail.py
Microsoft Windows [Version 10.0.22000.1098]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Monisha>python mail.py
202
b''
Server: nginx
Date: Tue, 08 Nov 2022 09:06:08 GMT
Content-Length: 0
Connection: close
X-Message-Id: gGkdPkwMS9GmkhMPyGIpMQ
1Access-Control-Allow-Origin: https://sendgrid.api-docs.io
1Access-Control-Allow-Methods: POST
1Access-Control-Allow-Headers: Authorization, Content-Type, On-behalf-of, x-sg-elas-acl
1Access-Control-Max-Age: 600
1X-No-CORS-Reason: https://sendgrid.com/docs/Classroom/Basics/API/cors.html
14Strict-Transport-Security: max-age=600; includeSubDomains

C:\Users\Monisha>
```

Test mail received via SendGrid

