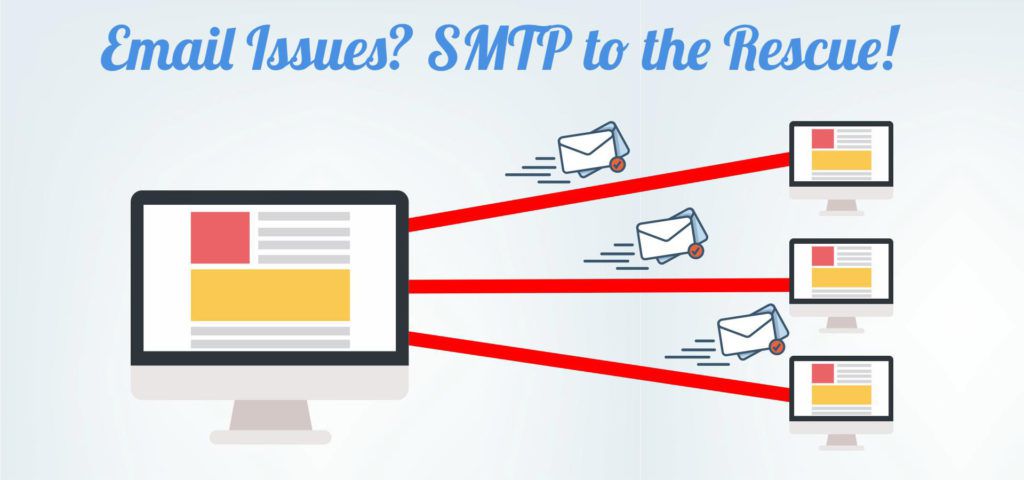
**Automatic Email Sending with Python**



* Objective
* Languages Used
* Code
* Challenges
* Modifications can be done in future

**Objective**

* Automation of Email Sending using Python and SMTP server
* Everything start with a computer which can send an e-mail by itself. Now robots are the master of the universe…

****

**Language Used**

* Python
* **pandas**
* **email**
* **smtplib**
* **ssl**
* **win32com.client**
* **psutil**
* **os**
* **subprocess**

**Code**

*# Python code to illustrate Sending mail from*

*# your Gmail account*

import pandas as pd

import email, smtplib, ssl

from email import encoders

from email.mime.base import MIMEBase

from email.mime.multipart import MIMEMultipart

from email.mime.text import MIMEText

userid=”your Emailid”

password="your password"

gmailid="@gmail.com"

outlookid="@outlook.com"

df = pd.read\_csv (r'E:\project\bit and byte\grades.csv')

email=(df.email[0:len(df)])

name=(df.name[0:len(df)])

roll=(df.Roll\_No[0:len(df)])

grade=(df.grade[0:len(df)])

subject\_for\_PG="Welcome to PG"

subject\_for\_MSC="Welcome to MSC"

def gmail():

import smtplib

*# creates SMTP session*

s = smtplib.SMTP('smtp.gmail.com', 587)

*# start TLS for security*

s.starttls()

*# Authentication*

s.login(userid, password)

msg = MIMEMultipart()

msg['Subject'] = subject\_for\_PG

*#newmsg.append(message)*

msg1 = MIMEMultipart()

msg1['Subject'] = subject\_for\_MSC

filename = r"E:\Assignment\semester2\python\pandascheat.pdf"

for i in range (len(df)):

if ("PG" in roll[i]):

*#this is for adding subject*

TEXT = "hey"+' '+name[i]+" you are in PG"+" "+'\n'

msg.attach(MIMEText(TEXT, 'plain'))

*#message= 'Subject: {}\n\n{}'.format(SUBJECT, TEXT)*

*# open the file to be sent*

filename = "pandascheat.pdf"

attachment = open(r"E:\Assignment\semester2\python\pandascheat.pdf", "rb")

*# instance of MIMEBase and named as p*

p = MIMEBase('application', 'octet-stream')

*# To change the payload into encoded form*

p.set\_payload((attachment).read())

*# encode into base64*

encoders.encode\_base64(p)

p.add\_header('Content-Disposition', "attachment; filename= %s" % filename)

*# attach the instance 'p' to instance 'msg'*

msg.attach(p)

text = msg.as\_string()

s.sendmail("\*\*",email[i],text)

elif ("MSC" in roll[i]):

*#this is for adding subject*

TEXT = "hey"+' '+name[i]+" you are in MSC"

msg1.attach(MIMEText(TEXT, 'plain'))

*#message= 'Subject: {}\n\n{}'.format(SUBJECT, TEXT)*

*# open the file to be sent*

filename = "pandascheat.pdf"

attachment = open(r"E:\Assignment\semester2\python\pandascheat.pdf", "rb")

*# instance of MIMEBase and named as p*

p = MIMEBase('application', 'octet-stream')

*# To change the payload into encoded form*

p.set\_payload((attachment).read())

*# encode into base64*

encoders.encode\_base64(p)

p.add\_header('Content-Disposition', "attachment; filename= %s" % filename)

*# attach the instance 'p' to instance 'msg'*

msg1.attach(p)

text = msg1.as\_string()

s.sendmail("\*\*",email[i],text)

*# terminating the session*

s.quit()

*#gmail()*

def outlook():

import win32com.client as win32

import psutil

import os

import subprocess

import pandas as pd

df = pd.read\_csv (r'E:\oulook.csv')

df

i=len(df)

i=len(df)

email=(df.email[0:len(df)])

name=(df.name[0:len(df)])

status=(df.status[0:len(df)])

*# Drafting and sending email notification to senders. You can add other senders' email in the list*

def send\_notification():

for i in range (len(df)):

outlook = win32.Dispatch('outlook.application')

mail = outlook.CreateItem(0)

mail.To = email[i]

mail.Subject = subj

mail.body = 'Welcome to Data Science”

mail.send

#open\_outlook()

*# Open Outlook.exe. Path may vary according to system config*

*# Please check the path to .exe file and update below*

def open\_outlook():

try:

subprocess.call(['"C:\Program Files (x86)\Microsoft Office\root\Office16\OUTLOOK.EXE"'])

os.system("C:\Program Files (x86)\Microsoft Office\root\Office16\OUTLOOK.EXE");

except:

print("Outlook didn't open successfully")

*#send\_notification()*

*# Checking if outlook is already opened. If not, open Outlook.exe and send email*

def kuchbhi():

for item in psutil.pids():

p = psutil.Process(item)

if p.name() == "OUTLOOK.EXE":

flag = 1

break

else:

flag = 0

if (flag == 1):

send\_notification()

else:

open\_outlook()

send\_notification()

*#send\_notification()*

kuchbhi()

def decide():

if gmailid in userid:

gmail()

else:

outlook()

decide()

**Challenges Faced**

* Integrate the Gmail code and Outlook code was time taking and need troubleshooting
* Attaching the pdfs are challenging with text message.
* Automate the thing was also a challenge. Scheduling with the taskbar was not so easy.

**Modification in Future**

* We can directly make pdf from data frame in python and send it to receiver.
* Ex. If there is result dataset in excel for different department, we can make a data frame in python(pandas) and make it to pdf and send it to each department.
* We can make the project in server so that it can be distributed.