Send Email to a mailing list in Python via Gmail

Include attachments for each group of recipients

In [1]:

```python
# Create a file ("config.py") containing your credentials and the HTML email content (model below)
EMAIL_ADDRESS = "email_address"
PASSWORD = "pass"
ALIAS = "alias_name <email_address>"



EMAIL_HTML = """


Insert your message here - HTML code (mother tongue)

"""

EMAIL_HTML_EN = """

Insert your message here - HTML code (English)

"""
```

In [2]:

```python
import config
import smtplib
import pandas as pd
from datetime import datetime
import os
```

In [3]:

```python
# Parametros
root_path = '2018-10/'
csv_file = "mailing_list/mail_list.csv"
periodo = 'Outubro-2018'
period = 'October-2018'
log_file = "log.csv"
```

In [4]:

```python
# Email module
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart
from email.mime.base import MIMEBase
from email import encoders
```

In [5]:

```python
# Import your mailing list csv
df = pd.read_csv(csv_file)
print("\n")
print("CSV '%s' loaded" % csv_file)
print("\n")
print("Number of rows: "+str(df.shape[0]))
```

CSV 'mailing_list/mail_list.csv' loaded

Number of rows: 1

In [6]:

```python
# Check integrity of columns at csv file
lista=df.columns
if 'Pessoa - E-mail' and 'Pessoa - Organização' and 'Organização - Idioma' in lista:
    print("\n")
    print("Your csv_file contain the required colums to continue")
else:
    print("\n")
    print("Please, check if the required colums exists at the csv_file")
    exit()
```

Your csv_file contain the required colums to continue

In [7]:

```python
# Drop recipients without email address
antes = int(df.shape[0])
df.dropna(subset=['Pessoa - E-mail'], inplace=True)
depois = int(df.shape[0])
print("\n")
print("Number of recipients without email: "+str(antes-depois))
print("\n")
print("Total number of valid recipients: "+str(df.shape[0]))
print("\n")
```

Number of recipients without email: 0

Total number of valid recipients: 1

In [8]:

```python
# Rename columns
df.rename(columns={"Pessoa - Organização": "Ambiente", "Pessoa - E-mail": "Email",
"Organização - Idioma": "Idioma"}, inplace=True)

# Create recipient list according to group of company
ambientes = list(df.Ambiente.unique())
for nome_ambiente in ambientes:

    email_send = ', '.join(df['Email'][df['Ambiente'] == nome_ambiente])


    idioma = (df['Idioma'][df['Ambiente'] == nome_ambiente]).unique()
```

In [9]:

```python
    # Sender information
    email_user = config.EMAIL_ADDRESS
    email_password = config.PASSWORD
    email_alias = config.ALIAS
```

In [10]:

```python
    # You can add bcc email address
    email_bcc = ['email_bcc@domain.com']
    # more than one: email_bcc = ['bob@rocks.com', 'julian_is@awesome.com']
```

In [11]:

```python
    # Code
    diretorio = root_path+nome_ambiente+'/'


    ficheiros = os.listdir(diretorio)
```

In [12]:

```python
    if idioma == 'Português':
        mes = periodo
        subject = 'Subject Test '+mes
        html = config.EMAIL_HTML
        body = 'Prezados Senhores\nEncaminho em anexo o relatório.\nMeus melhore
s cumprimentos'
    else:
        mes = period
        subject = 'Subject Test '+mes
        html = config.EMAIL_HTML_EN
        body = 'Dear Sir / Madam\nYou are receiving our report\nBest regards'
```

In [13]:

```python
    # Code
    msg  = MIMEMultipart('alternative')
    #msg  = MIMEMultipart()
    msg['From'] = config.ALIAS
    msg['To'] = email_send
    msg['Subject'] = subject
```

```python
        msg.attach(MIMEText(body,'plain'))
        msg.attach(MIMEText(html,'html'))
```

```python
        for x in ficheiros:
            filename = diretorio+x
            attachment = x


            part = MIMEBase('application','octet-stream')
            part.set_payload(open(filename,'rb').read())
            encoders.encode_base64(part)
            part.add_header('Content-Disposition','attachment', filename= attachment
)
            msg.attach(part)
            text = msg.as_string()

        try:

            server = smtplib.SMTP('smtp.gmail.com',587)
            server.starttls()
            server.login(email_user,email_password)

            server.sendmail(email_user,[email_send]+email_bcc,text)
            server.quit()
            print("Success: Email was sent to %s." % nome_ambiente)

            #print("Sucess: Email sent!")
            #print("Success: Email was sent to %s." % email_send)

            # cria variavel tempo
            now = str(datetime.now())

            # Acrescenta info no arquivo Log
            log = open(log_file, "a")
            log.write(now+";"+str(nome_ambiente)+";"+str(email_send)+"\n")
            log.close()
        except:
            print("Email failed to send to %s."  % nome_ambiente)
            #print("Email failed to send.")
```

Success: Email was sent to Company.