In [ ]:

```
import smtplib
from getpass import getpass
```

Connecting Server

In [ ]:

```
def create_server(host_address,port_number):
    global svr
    svr = smtplib.SMTP(host=host_address,port=port_number)
    svr.starttls() #converting insecure connection to secure connection
create_server("smtp.gmail.com",587)
```

User Mail and Pasword

In [ ]:

```
def user():
    global Mail_address,Password
    Mail_address = input("Enter Mail Address : ")
    Password = getpass("Enter Password : ")
    svr.login("{ad}".format(ad=Mail_address),"{pd}".format(pd = Password))
user()
```

Contacts list function

In [ ]:

```
def contacts(contacts_file):
    global Names
    Names = []
    global Mails
    Mails = []
    if contacts_file[-4:] == ".csv":
        with open(contacts_file,'r') as contact  :
            for each_contact in contact:
                data =  each_contact.split(",")
                Names.append(data[0])
                Mails.append(data[1])
        return Names,Mails
    elif contacts_file[-4:] == ".txt":
        with open(contacts_file,'r') as contact  :
            for each_contact in contact:
                data = each_contact.split("  ")
                Names.append(data[0])
                Mails.append(data[1])
        return Names,Mails
    else:
        print("Only .csv or .txt file")
#contacts("contacts.csv")
```

Body Message

In [ ]:

```
#Template for custom name and address
from string import Template
def body_msg(body_file):
    if body_file[-4:] == ".txt":
        with open(body_file,'r',encoding = 'utf-8') as message:
            msg_content = message.read()
            global template_msg
            template_msg = Template(msg_content)
        #return template_msg
    else :
        print("Only text file")



def body_msg_htm(body_file_htm):
    if ((body_file_htm[-4:] == ".htm") or (body_file_htm[-5:] == ".html")) :
        with open(body_file_htm,'r',encoding = 'utf-8') as message:
            global html_msg_content
            html_msg_content = message.read()
            #return html_msg_content
    else :
        print("Only html file")


#body_msg("msg.txt")
#body_msg_htm("msg.html")
```

Function for Sending .txt message

In [ ]:

```
from email.mime.multipart import MIMEMultipart
from email.mime.text import MIMEText

def final_mail_and_send(contacts_file):
    Names,Mails = contacts(contacts_file)
    for Name,Mail in zip(Names,Mails):

        #Creating a message
        global msg
        msg = MIMEMultipart()

        #body message from function body_msg
        body_message = template_msg

        #Customising the body message
        global message
        final_message = body_message.substitute(PERSON = Name.title(),MAIL_ID = Mail)

        #From ,to , Subject
        msg['From'] = Mail_address
        msg['To'] = Mail
        msg['Subject'] = "Hi Hello"
        #Body
        msg.attach(MIMEText(final_message,'plain'))

        #Sending
        svr.send_message(msg)
        del msg
#final_mail_and_send("contacts.csv")
```

Function for sending .html Message

In [ ]:

```
def final_mail_and_send_html(contacts_file):
    Names,Mails = contacts(contacts_file)
    for Name,Mail in zip(Names,Mails):

        #Creating a message
        global msg
        msg = MIMEMultipart()

        #body message from function body_msg
        body_htm_message = html_msg_content #for html message

        #From ,to , Subject
        msg['From'] = Mail_address
        msg['To'] = Mail
        msg['Subject'] = "Hi Hello"

        #Body
        msg.attach(MIMEText(body_htm_message,'html'))

        #Sending
        svr.send_message(msg)
        del msg
    svr.quit()


#final_mail_and_send_html("contacts.csv")
```