OTP Verification System with GUI

Documentation

Functions:

1) send_mail():

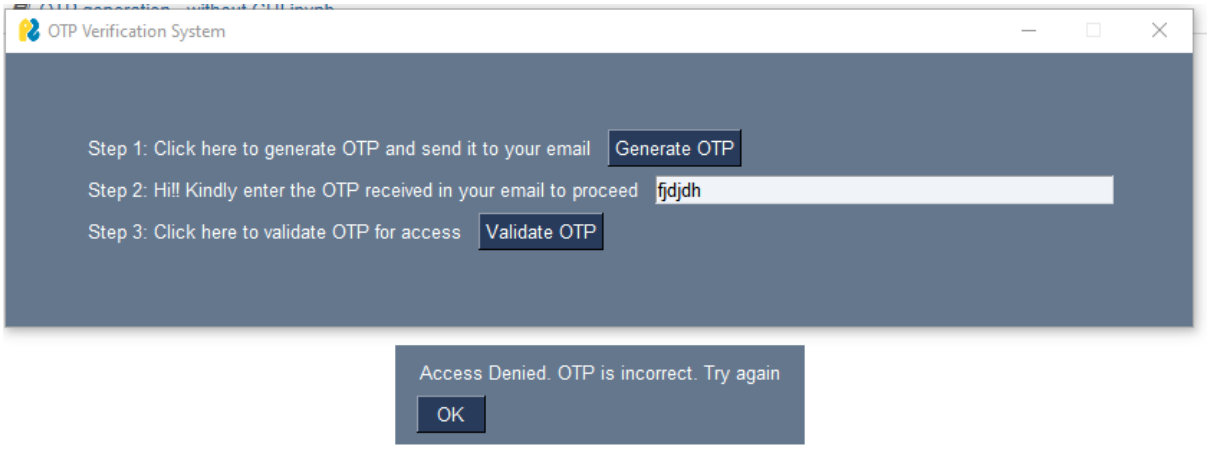
   Creates an SMTP session to send the OTP as email. TLS is used for security.

2) otp_check()

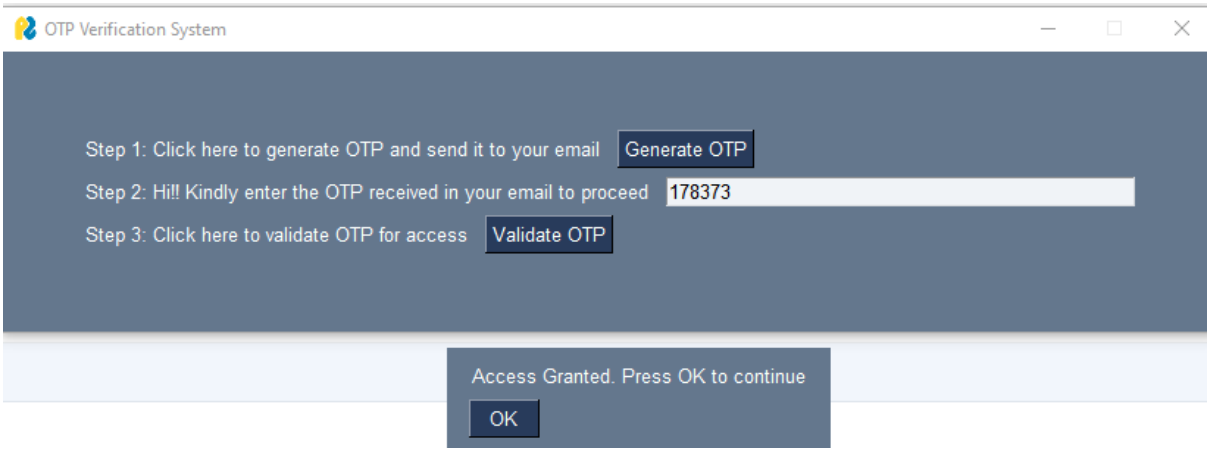
   To check if the entered OTP is same as the OTP shared through email. Pops up an 'Access Granted' message if the OTP is correct. Incase of wrong OTP, it pops up an 'Access Denied. OTP is incorrect. Try again' message.

Test cases:

1) In case of entering wrong OTP, it denies access to the user.



2) In case of entering correct OTP as received in the email, it grants access to the user.



GUI:

1) Generate OTP button is used to generate a random 6 digit number and calls the send_mail function
2) After entering the received OTP in the relevant input space, Validate OTP button is used to check if the OTP is correct by calling the check_otp function

3) The program ends when either the correct OTP is entered and validated or if the GUI window is closed.
4) Incase the entered OTP is incorrect, the user has the option to generate a new OTP or enter the correct OTP and validate it.

Program:

```
import random
import smtplib
import PySimpleGUI as sg

def send_mail():
    # creates SMTP session
    s = smtplib.SMTP('smtp.gmail.com', 587)
    # start TLS for security
    s.starttls()
    # Authentication
    s.login("xxxxxxxxxx@gmail.com", "password") # Specify sender's email id and application password
    # message to be sent
    message = "\nHi there!! :) \n\nYour OTP is: "+str(otp)
    # sending the mail
    s.sendmail("xxxxxxxxxx@gmail.com", "yyyyyyyyyy@gmail.com", message) # Specify sender's and receiver's email addresses
    # terminating the session
    s.quit()

def otp_check(otp_rec): # Function to prompt the user to enter the OTP received in their email
    if otp_rec==str(otp) and otp!=0: # Access should not be granted if otp entered is equal to initialization value
        global flag
        flag=True
        sg.popup_no_titlebar("Access Granted. Press OK to continue")
    else:
        sg.popup_no_titlebar("Access Denied. OTP is incorrect. Try again")

layout = [[sg.Text("Step 1: Click here to generate OTP and send it to your email"), sg.Button("Generate OTP")],
          [sg.Text("Step 2: Hi!! Kindly enter the OTP received in your email to proceed"), sg.Input(key='-otp-')],
          [sg.Text("Step 3: Click here to validate OTP for access"), sg.Button("Validate OTP")]]

# Create the window
window = sg.Window("OTP Verification System", layout,margins=(50,50))

# Initializing the otp value
otp=0
flag=False # flag is used to exit out of the GUI only if entered otp is correct after validation

# Create an event loop
while True:

    event, values = window.read()

    # Create and send otp to mail when user clicks 'generate otp' button
    if event=='Generate OTP':
        otp=random.randint(100000,999999) #Generating the 6 digit otp using randint from random module
        send_mail() # Calling function send_mail to send the generated otp

    # End program if user closes window
    elif event == sg.WIN_CLOSED:
        break

    # End program if entered otp is correct
    elif values['-otp-']!='':
        otp_check(values['-otp-']) # Calling function otp_check to validate the otp entered by user
        if event == "Validate OTP" and flag==True:
            break

window.close()
```