Software Testing Assignment -1

```python
# THIS IS MAIN FILE I NAMED AS Send_OTP_Using_Mail_2, this is version 2.
import GetPass #Importing Password,Email From another file.
import random
import smtplib #This library is used for sending message using email.
import time

password = GetPass.pwd
Sender_Mail = GetPass.email

def EmailValidation(Email):
    True_Str1,True_Str2 = "yahoo" in Email,"gmail" in Email  # This will store boolean values.

    if (True_Str1 or True_Str2) and( "@" in Email and "." in Email and "com" in Email):
        print("\nNo Error Found in Email!")
    else:
        raise AssertionError("Please enter valid Domain Name!")

def genrateOtp():
    Length = int(input("Enter Length of OTP: "))
    otp = ''.join([str(random.randint(0,9)) for i in range(Length)]) #Generated OTP using
random.randint()
    return otp




def sendMail(Name,Email,otp):
    server = smtplib.SMTP('smtp.gmail.com',587) #Created gmail's server, and connected to gmail API
    # Adding transfer layered security
    server.starttls()
    server.login(Sender_Mail,password) # Email, App password are inserted.
    if True:
        msg = 'Subject: Sending Mail using Python (smtplib)!\n\nHello '+Name+', Your OTP is
'+str(otp)+'\n\nYou Have 30 Seconds to enter OTP!'

        #Inserted Sender email ID, Recevier email ID.
        server.sendmail(Sender_Mail,Email,msg)
        print("Email Sent!")
    server.quit()




def validateOTP(OT):
    # This function will check entered otp is valid or not!
    # This function also have Time Limit of 30 Sec
    test_time = 30
    beg_time = time.time()
    now_time = time.time()
    otp = OT
    input_otp = 0
    if input_otp == otp:
        pass
    else:
        while input_otp != otp and int(now_time)-int(beg_time) <= test_time:
            if now_time-beg_time <=test_time:
                input_otp = input("Enter Valid OTP: ")
            now_time = time.time()
```

Software Testing Assignment -1

```python
if input_otp == otp:
    print("OTP IS VALID!")
else:
    raise AssertionError("Out Of Time!")
```

Software Testing Assignment -1

```python
#THIS IS ANOTHER FILE I NAMED AS OTP.
import unittest
import smtplib
import Send_OTP_Using_Mail_2 as O

class BetweenAssertMixin(object):
    def assertBetween(self, x, low, hi):
        if not (low <= x <= hi):
            raise AssertionError('Length of OTP is %r should be in between %r and %r' % (x, low, hi))



class OTP(unittest.TestCase,BetweenAssertMixin):
    def testcase1(self):
        print("-----------------------------------TestCase No.1-----------------------------------")
        # This is valid TestCase
        Name = "Sahiil"
        Email = "Sahiilshriwardhankar@gmail.com"

        #Validation of Email
        O.EmailValidation(Email)

        #Checking OTP
        otp = O.genrateOtp()
        self.assertBetween(len(otp),4,8)

        #Calling Sendmail Function
        O.sendMail(Name,Email,otp)




        #Validation of OTP
        O.validateOTP(otp)



    def testcase2(self):
        print("-----------------------------------TestCase No.2-----------------------------------\n")

        # Email Validation
        # Here i provided, incorrect Email ID.
        Name = "Sahiil"
        Email = "Sahiilshriwardhankargmail.com"

        #Validation of Email
        O.EmailValidation(Email)


        #Checking OTP
        otp = O.genrateOtp()
        self.assertBetween(len(otp),4,8)

        #Calling Sendmail Function
        O.sendMail(Name,Email,otp)

        #Validation of OTP
```

Software Testing Assignment -1

```python
        O.validateOTP(otp)

        print("-------------------------------------------------------------------------------\n")


    def testcase3(self):
        print("----------------------------------TestCase No.3----------------------------------\n")

        # There is no
        Name = "Sahiil"
        Email = "Sahiilshriwardhankar@gmail.com"

        #Validation of Email
        O.EmailValidation(Email)

        # Checking OTP
        # Here i will Enter invalid otp length
        otp = O.genrateOtp()
        self.assertBetween(len(otp),4,8)

        #Calling Sendmail Function
        O.sendMail(Name,Email,otp)

        #Validation of OTP
        O.validateOTP(otp)


    def testcase4(self):
        print("----------------------------------TestCase No.4----------------------------------\n")

        #Checking Email
        Name = "Sahiil"
        Email = "Sahiilshriwardhankar@gmail.com"

        #Validation of Email
        O.EmailValidation(Email)

        #Checking OTP
        otp = O.genrateOtp()
        self.assertBetween(len(otp),4,8)

        #Calling Sendmail Function
        O.sendMail(Name,Email,otp)

        #Validation of OTP and I will take time greater then 30sec.
        O.validateOTP(otp)



        print("-------------------------------------------------------------------------------\n")


unittest.main()
```

Software Testing Assignment -1

Sending Mail using Python (smtplib)! Inbox ×

sahiilshriwardhankar@gmail.com   9:40 PM (2 minutes ago)
to bcc: Sahiilshriwardhankar

Hello Sahiil, Your OTP is 1602

You Have 30 Seconds to enter OTP!

↩ Reply   → Forward

---

sahiilshriwardhankar@gmail.com   9:40 PM (2 minutes ago)
to bcc: Sahiilshriwardhankar

Hello Sahiil, Your OTP is 1602

You Have 30 Seconds to enter OTP!