

# **PYTHON CAPSTONE PROJECT**

## OTP Verification System

## AIM

- ✓ To generate a 6-digit OTP number and send it to the user's email address for verification.
- ✓ The user should receive the OTP and enter the received OTP into the system for validation.
- ✓ If the entered OTP matches the generated OTP validation is successful and if it is incorrect, 3 attempts are given for the user to retry.

IDE used:- Jupyter Notebook

## MODULES USED

```
In [1]: # Importing important libraries  
import random  
import smtplib  
from email.message import EmailMessage
```

RANDOM : Used to generate random 6-digit numbers for OTP.

SMTP : Safe Mail Transfer Protocol - Used to send email to the user.

EMAIL.MESSAGE : Used to send message in the email.

# EXPLANATION OF CODE

## I. Generating random 6-digit OTP:-

```
# Generating random 6-digit OTP  
def generate_otp():  
    return random.randint(100000, 999999)
```

- Here function generate\_otp() is used.
- This function returns randomly generated 6-digit otp using random.randint.

## 2. Sending OTP via Email:-

```
# Send OTP via Email
def send_otp(otp, to_email):
    #Set up the SMTP Server
    try:
        server = smtplib.SMTP('smtp.gmail.com', 587)
        server.starttls()

        # Define Sender's Email credentials
        from_mail = 'purva0377@gmail.com'
        password = 'unyo yyvs orzt dxul'
        server.login(from_mail, password)

        # Create Email message to be sent to recipient
        msg = EmailMessage()
        msg['Subject'] = "OTP Verification"
        msg['From'] = from_mail
        msg['To'] = to_email
        msg.set_content(f"Your OTP for verification is: {otp}")

        # Send mail
        server.send_message(msg)
        print(f"OTP has been sent to: {to_email}.")

    except Exception as e:
        print(f"An error occurred: {e}")
    finally:
        server.quit()
```

- The send\_otp(otp, to\_email) function is used to send the otp to the user's email address.
- The code to send the email is written in try block to handle any potential errors that may occur (such as network issues or incorrect login credentials)
- Here, smtplib.SMTP is used to connect to Gmail's SMTP server.
- EmailMessage is used for the content /body of the email message to be sent.
- The except block is used to catch and handle any exceptions(errors) that occur during the execution of the code within the try block.
- The finally block ensures that the SMTP server connection is closed properly, regardless of any error occurs.

### 3. Prompt user to enter the OTP received:-

```
# Prompt the user to enter the OTP received
def get_user_otp():
    return input("Enter the OTP you received:")
```

### 4. Verification of OTP:-

```
#Verification of OTP
def verify_otp(generated_otp):
    attempts = 3 #Allow up tp 3 attempts
    while attempts > 0:
        user_otp = get_user_otp()
        if user_otp == str(generated_otp):
            print("Validation Successfull!")
            return True
        else:
            attempts -= 1
            print(f"Incorrect OTP. {attempts} attempts are remaining.")
    print("Invalid OTP. Please try again.")
    return False
```

- The function `get_user_otp()` prompts the user to enter the OTP that they received in their email.
- The function `verify_otp(generated_otp)` is used to verify the user otp.
- Here, while loop is used to give the user 3 attempts for retry if they enter incorrect otp.
- If statement is used to compare the user otp with the generated otp. If True, it prints 'Validation Successful'.
- Else decrements the attempts by one.
- Lastly, if all attempts are used it prints 'Invalid OTP. Please try again' and returns false.

## 5. Main Function:-

```
# Main Function
def main():
    print("Welcome to OTP verification syatem!")

    #Generate OTP
    otp = generate_otp()

    #Get recipient's email
    to_email = input("Please enter your email address:")

    #Send the OTP
    send_otp(otp, to_email)

    #Verify OTP
    if verify_otp(otp):
        print("Otp verification complete. You may proceed.")
    else:
        print("OTP verification failed.")

# Run the Program
if __name__ == "__main__":
    main()
```

- The main function is used to integrate and execute all the components of program in a logical sequence.
- First the OTP is generated.
- Second user is prompted to enter their email address.
- Third the OTP is sent to the user's email address.
- Fourth is the verification of the OTP.
- if `__name__ == "__main__"`: block ensures that the program runs the `main()` function only when the script is executed directly, not when imported.

## EXAMPLE SCENARIOS

```
Welcome to OTP verification syatem!  
Please enter your email address:pkv26ap@gmail.com  
OTP has been sent to: pkv26ap@gmail.com.  
Enter the OTP you received:825578  
Validation Successfull!  
Otp verification complete. You may proceed.
```

- The user gives correct OTP and the validation becomes successful.

- The user gives incorrect OTP and they are given attempts to retry.

```
Welcome to OTP verification syatem!  
Please enter your email address:pkv26ap@gmail.com  
OTP has been sent to: pkv26ap@gmail.com.  
Enter the OTP you received:4556  
Incorrect OTP. 2 attempts are remaining.
```

Enter the OTP you received:

```
Welcome to OTP verification syatem!  
Please enter your email address:pkv26ap@gmail.com  
OTP has been sent to: pkv26ap@gmail.com.  
Enter the OTP you received:4556  
Incorrect OTP. 2 attempts are remaining.  
Enter the OTP you received:sfsf  
Incorrect OTP. 1 attempts are remaining.  
Enter the OTP you received:678905  
Incorrect OTP. 0 attempts are remaining.  
Invalid OTP. Please try again.  
OTP verification failed.
```

- User's all three attempts are used and the validation failed. So, they have to try again.