AZURE SKYNET
TECHNOLOGY | KNOWLEDGE | INNOVATION

<u>**ADDRESS:**</u> ***M-4, 1st Floor, Old DLF Colony, Sector 14, Gurugram, Haryana 122001***

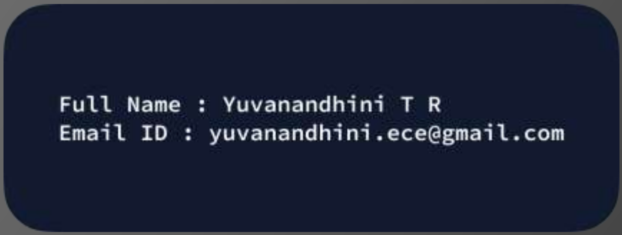
Project Report on Python Programming (Basic)

```
Full Name : Yuvanandhini T R
Email ID : yuvanandhini.ece@gmail.com
```

Acknowledgement

I would like to thank my Family who helped me a lot in finalizing this project within the limited time frame.

I would also like to express my special thanks of gratitude to my professors who motivated me to do this wonderful project on the topic "Python Programming (Basic)" which also helped me in knowing about so many new things and earned a skill or knowledge.



```
Full Name : Yuvanandhini T R
Email ID : yuvanandhini.ece@gmail.com
```

AZURE SKYNET
TECHNOLOGY | KNOWLEDGE | INNOVATION

Index

1. Creating A Login / Signup Interface To Send And Receive Messages.

Creating A Login / Signup Interface To Send And Receive Messages.

Aim Of The Project:

In this project, I have written lines of code in step wise & mentioned GUI code then, Server side code to create a login / signup Interface to send & receive messages. I have made tkinter as most important in GUI code. Final Outcome of the code is mentioned clearly for the basic understanding of the project.

Steps Performed During Development Of The Project:

GUI Code Of Creating A Login / Signup Interface To Send And Receive Messages:

```
from tkinter import *
import tkinter.ttk
import tkinter.messagebox
import requests


def get_message(from_username, to_username):
    req = requests.get('http://localhost:8080/navuy_messenger/retrieve_messages.py?from_username=' + from_username + '&to_username=' + to_username)
    data = req.json()
    return data['array']


def get_all_users():
    req = requests.get('http://localhost:8080/navuy_messenger/users_data.py')
    data = req.json()
    return data['array']


def check_user_exists(username):
    req = requests.get(url =
'http://localhost:8080/navuy_messenger/check_username_exists.py', params =
{'username': username})
    data = req.json()
    return data['boolean']


def check_pass_match_user(username, password):
    req = requests.get(url =
'http://localhost:8080/navuy_messenger/check_password_matches_username.py',
params = {'username': username, 'password': password})
    data = req.json()
    return data['boolean']


def chat_window(my_username, to_username):
    chat_window_tk = Tk()
    chat_window_tk.geometry("400x500")
    chat_window_tk.resizable(0, 0)
    chat_window_tk.title("Chat with " + to_username)


    try:
        error = False
        messages = get_message(my_username, to_username)
    except:
        error = True
    finally:

        chat = StringVar()

        def refresh_command():
            chat_window_tk.destroy()
            chat_window(my_username, to_username)

        def top():
            def your_chats_command():
                chat_window_tk.destroy()
                users_window(my_username)

            sep_frame = Frame(master=chat_window_tk)
            row_frame = Frame(master=sep_frame)
            Label(master=row_frame, text='<', font = ('Helvetica', 15), anchor = 'w').pack(side=LEFT, padx=(10, 0))
            Button(master=row_frame, text='Your Chats', fg = 'blue', font = ('Helvetica', 12), anchor = 'w', relief='flat', command=your_chats_command).pack(side=LEFT, pady=5)
            userlbl = Label(master=row_frame, text=to_username, font = ('Helvetica', 12), anchor = 'w')
            userlbl.pack(side=LEFT, padx=20, pady=5)
            Button(master=row_frame, text='Refresh Chat', fg = 'blue', font =
    ('Helvetica', 12), anchor = 'e', command=refresh_command).pack(side=RIGHT,
    pady=5, padx=(0, 10))

            row_frame.pack(fill = X, side=TOP)

            sep_frame.pack(fill = X, side=TOP)

            tkinter.ttk.Separator(orient=HORIZONTAL).pack(fill=X)


        def chat_ui():

            canvas = Canvas(chat_window_tk)

            sub_frame = Frame(master=canvas)

            scrollbar = Scrollbar(master=chat_window_tk, orient="vertical",
    command=canvas.yview)


            def on_vertical(event):

                direction = 0

                if event.num == 5 or event.delta == -120:

                    direction = 1

                if event.num == 4 or event.delta == 120:

                    direction = -1


                canvas.yview_scroll(direction, UNITS)


            def on_configure(event):
```

```
        canvas.configure(scrollregion=canvas.bbox('all'),
yscrollcommand=scrollbar.set)


    canvas.configure(yscrollcommand=scrollbar.set)

    scrollbar.pack(side='right', fill=Y)


    sub_frame.pack(fill=X)


    canvas.pack(side='top', fill=BOTH, expand=True)


    canvas.create_window((0, 0), width=int(canvas['width']),
window=sub_frame, anchor='n')


    canvas.bind_all('<MouseWheel>', on_vertical)

    canvas.bind('<Configure>', on_configure)

    sub_frame.bind("<Configure>", on_configure)


    def assemble_chat(message):

        if message["from_username"] == my_username:

            chat_frame = Frame(master=sub_frame)


            Label(master = chat_frame, text='< You', font = ('Helvetica', 12,
'bold')).pack(side=RIGHT)
```

```
                    Label(master = chat_frame, text=message['message'], font =
("Helvetica", 12)).pack(side=RIGHT)


                    chat_frame.pack(fill=X, side=TOP)
                else:
                    chat_frame = Frame(master=sub_frame)


                    Label(master = chat_frame, text=to_username + ' >', font =
('Helvetica', 12, 'bold')).pack(side=LEFT)
                    Label(master = chat_frame, text=message['message'], font =
("Helvetica", 12)).pack(side=LEFT)


                    chat_frame.pack(fill=X, side=TOP)


            for i in messages:
                assemble_chat(i)


        def chat_input():
            def send_command():
                if len(chat.get()) > 0:
                    requests.post(
                        'http://localhost:8080/navuy_messenger/send_message.py',
                        {
                            'from_username': my_username,
                            'to_username': to_username,
                            'message': str(chat.get())
                        }
                    )
                    refresh_command()

            input_frame = Frame(chat_window_tk)
            entry = Entry(master= input_frame, textvariable = chat, font = ('Helvetica', 12))
            entry.pack(fill = X, expand=True, side = LEFT, padx = 10, pady=(0,10))
            Button(master=input_frame, text='Send', fg = 'blue', font = ('Helvetica', 12), anchor = 'e', command=send_command).pack(side=RIGHT, padx=(0,10), pady=(0,10))
            input_frame.pack(fill=X, side=BOTTOM)


        def error_ui():
            Label(master=chat_window_tk, text="No Messages Sent / Received So Far!", font = ('Helvetica', 12)).pack(side = TOP, pady=(50,0))


        top()
        if not error:
            chat_ui()
        else:
            error_ui()
        chat_input()
        chat_window_tk.mainloop()



def users_window(my_username):
    users_window = Tk()
    users_window.geometry("375x400")
    users_window.resizable(0, 0)
    users_window.title("Navuy Messenger")

    try:
        error = False
        users = get_all_users()
    except:
        error = True
    finally:
        def form_label():
            Label(text ='Your Chats', width = '250', height = '3', font = ('Helvetica', 19, 'bold')).pack()
            tkinter.ttk.Separator(orient=HORIZONTAL).pack(fill=X)

        def canvas_ui():
```

```
    canvas = Canvas(users_window)
    sub_frame = Frame(master=canvas)
    scrollbar = Scrollbar(master=users_window, orient="vertical",
command=canvas.yview)

    def on_vertical(event):

        direction = 0
        if event.num == 5 or event.delta == -120:
            direction = 1
        if event.num == 4 or event.delta == 120:
            direction = -1

        canvas.yview_scroll(direction, UNITS)

    def on_configure(event):
        canvas.configure(scrollregion=canvas.bbox('all'),
yscrollcommand=scrollbar.set)

    canvas.configure(yscrollcommand=scrollbar.set)
    scrollbar.pack(side='right', fill=Y)

    sub_frame.pack(fill=X)
    canvas.pack(side='left', fill=BOTH, expand = True)

    canvas.create_window((0, 0), width=int(canvas['width'])-20,
window=sub_frame, anchor='n')

    canvas.bind_all('<MouseWheel>', on_vertical)
    canvas.bind('<Configure>', on_configure)
    sub_frame.bind("<Configure>", on_configure)

    def assemble_row(username):
        if username != my_username:
            def continue_chat_command():
                users_window.destroy()
                chat_window(my_username, username)

            sep_frame = Frame(master=sub_frame)
            row_frame = Frame(master=sep_frame)
            userlbl = Label(master=row_frame, text=username, font = ('Helvetica',
12), anchor = 'w')
            userlbl.pack(side=LEFT, padx=20, pady=5)
            Label(master=row_frame, text='>', font = ('Helvetica', 15), anchor =
'e').pack(side=RIGHT, padx=(0, 20))
```

```
                Button(master=row_frame, text='Continue Chat', fg = 'blue', font = ('Helvetica', 12), anchor = 'e', relief='flat', command = continue_chat_command).pack(side=RIGHT, pady=5)


                sep_spec_frame = Frame(sep_frame, width=int(canvas['width'])-40)

                separator = Frame(master = sep_spec_frame, height=2, width = int(canvas['width'])-40, bd=1, relief=SUNKEN)

                separator.pack(side=RIGHT)

                sep_spec_frame.pack(fill = X, side=BOTTOM)

                row_frame.pack(fill = X, side=TOP)

                sep_frame.pack(fill = X, side=TOP)


        for user in users:

            assemble_row(user)


    def error_ui():

        Label(master=users_window, text="No Users (Other Than You) Registered Yet!", font = ('Helvetica', 12)).pack(side = TOP, pady=(50,0))


    form_label()


    if not error:

        canvas_ui()

    else:
            error_ui()

        users_window.mainloop()

def main_login():
    main_screen = Tk()
    main_screen.geometry("375x375")
    main_screen.resizable(0,0)
    main_screen.title("Navuy Messenger")

    username = StringVar()
    pass_word = StringVar()

    def form_label():
        Label(text ='Login To Your Account', width = '250', height = '3', font = ('Helvetica', 19, 'bold')).pack()
        tkinter.ttk.Separator(orient=HORIZONTAL).pack(fill=X)
        Label(text = '').pack()

    def user_name():
        username_label = Label(text ='Username', font = ('Helvetica', 12), width = 30, anchor = 'w')
        username_label.pack()
        entry = Entry(textvariable = username, font = ('Helvetica', 12), width = 30)
        entry.pack()
        Label(text = ", font = ('Helvetica', 5)).pack()

    def password():
        password_label = Label(text ='Password', font = ('Helvetica', 12), width = 30, anchor = 'w')
        password_label.pack()
        entry = Entry(textvariable = pass_word, font = ('Helvetica', 12), width = 30)
        entry.pack()
        Label(text = ").pack()

    def login_button():

        def login_button_command():

            if username.get() == ":
                tkinter.messagebox.showerror("Invalid Input", 'Enter your username properly.')
                return

            if len(pass_word.get()) < 8:
                tkinter.messagebox.showerror("Invalid Input", 'The password must have a minimum of 8 characters.')
                pass_word.set(")

                return


            if not check_user_exists(username.get()):

                confirm = tkinter.messagebox.askyesno("Invalid Input", 'The username you entered doesn\'t exist. if you don\'t have an account, create a new one. Would you like to create a new account?')


                if confirm:

                    signup_screen_button_command()


                username.set(")

                pass_word.set(")

                return


            if not check_pass_match_user(username.get(), pass_word.get()):

                tkinter.messagebox.showerror("Invalid Input", 'Invalid username or password, please try again!')

                return


            main_screen.destroy()

            users_window(username.get())
```

```
        Button(text='Login', height ='2', width = 30, fg = 'blue', font = ('Helvetica', 12), command=login_button_command).pack()

        Label(text = '', font = ('Helvetica', 5)).pack()


    def signup_screen_button_command():

        main_screen.destroy()

        signup_window()


    def signup_screen():

        frame = Frame()

        Label(master = frame, text = "Don't have an account?", anchor = 'w', font = ('Helvetica', 10)).pack(side = LEFT)

        Button(master = frame, text = 'Sign up', relief = 'flat', fg = 'blue', font = ('Helvetica', 10), command = signup_screen_button_command).pack(side = LEFT)

        Label(master = frame, text = '                    ', font = ('Helvetica', 10)).pack()

        frame.pack()


    form_label()

    user_name()

    password()

    login_button()

    signup_screen()


    main_screen.mainloop()
```

```
def signup_window():
    main_signup = Tk()
    main_signup.geometry("375x435")
    main_signup.resizable(0,0)
    main_signup.title("Sign up Navuy")

    username_signup = StringVar(master = main_signup)
    password_signup = StringVar(master = main_signup)
    password_confirm_signup = StringVar(master = main_signup)

    def form_label():
        Label(master = main_signup, text ='Sign up', width = '250', height = '3', font =
('Helvetica', 19, 'bold')).pack()
        tkinter.ttk.Separator(master = main_signup,
orient=HORIZONTAL).pack(fill=X)
        Label(master = main_signup, text = '').pack()

    def signup_user_name():
        username_signuplbl = Label(master = main_signup, text ='Username', font =
('Helvetica', 12), width = 30, anchor = 'w')
        username_signuplbl.pack()
        entry = Entry(master = main_signup, textvariable = username_signup, font =
('Helvetica', 12), width = 30)
```

```
        entry.pack()
        Label(master = main_signup, text = '', font = ('Helvetica', 5)).pack()


    def signup_password():
        password_signuplbl = Label(master = main_signup, text ='Password', font = ('Helvetica', 12), width = 30, anchor = 'w')
        password_signuplbl.pack()
        entry = Entry(master = main_signup, textvariable = password_signup, font = ('Helvetica', 12), width = 30)
        entry.pack()
        Label(master = main_signup, text = '').pack()


    def confirm_password():
        password_confirmlbl = Label(master = main_signup, text ='Confirm Password', font = ('Helvetica', 12), width = 30, anchor = 'w')
        password_confirmlbl.pack()
        entry = Entry(master = main_signup, textvariable = password_confirm_signup, font = ('Helvetica', 12), width = 30)
        entry.pack()
        Label(master = main_signup, text = '').pack()


    def signup_button():
        def signup_button_command():
```

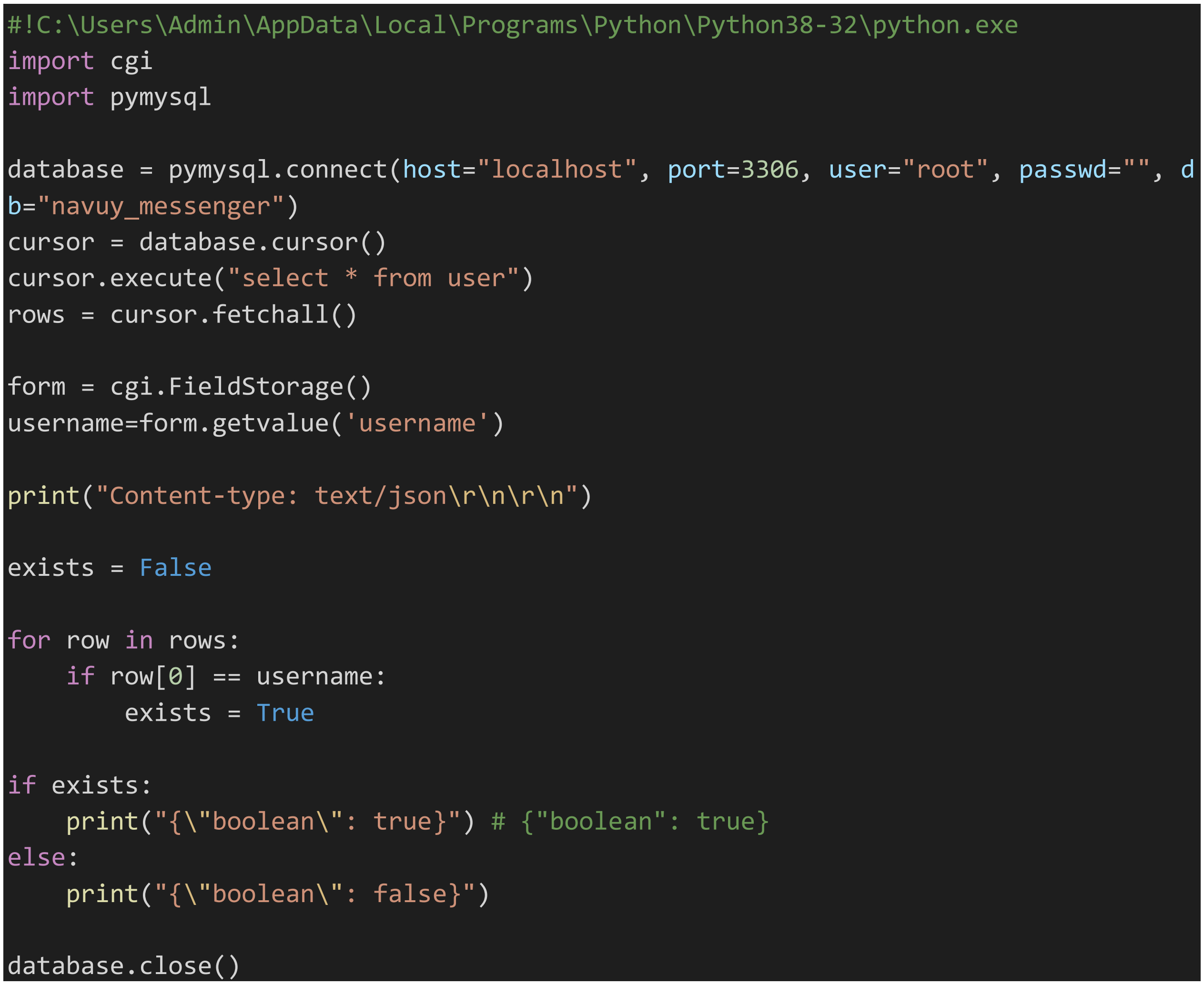
```
        if username_signup.get() == '':

            tkinter.messagebox.showerror("Invalid Input", 'Enter your username
properly.')

            return


        if check_user_exists(username_signup.get()):

            tkinter.messagebox.showerror("Invalid Input", 'The username you
entered is already taken please try a different one.')

            username_signup.set('')

            return


        if len(password_signup.get()) < 8:

            tkinter.messagebox.showerror("Invalid Input", 'The password must have
a minimum of 8 characters.')

            password_signup.set('')

            password_confirm_signup.set('')

            return


        if password_signup.get() != password_confirm_signup.get():

            tkinter.messagebox.showerror("Invalid Input", 'Enter your password
didn\'t match with confirm password.')

            password_signup.set('')

            password_confirm_signup.set('')

            return
        requests.post(
            'http://localhost:8080/navuy_messenger/signup_cgi.py',
            {'username': str(username_signup.get()),
            'password': str(password_signup.get())}
        )

        tkinter.messagebox.showinfo("Success", 'Your Account was created successfully. You can now login to your account.')

        login_screen_button_command()

    Button(master = main_signup, text='Sign up', height ='2', width = 30, fg = 'blue', font = ('Helvetica', 12), command = signup_button_command).pack()
    Label(master = main_signup, text = '', font = ('Helvetica', 5)).pack()

def login_screen_button_command():
    main_signup.destroy()
    main_login()



def back2login():
    new_frame = Frame(master = main_signup)
    Label(master = new_frame, text = "Do you want to go back to login?", anchor = 'w', font = ('Helvetica', 10)).pack(side = LEFT)

    Button(master = new_frame, text = 'Back', relief = 'flat', fg = 'blue', font = ('Helvetica', 10), command = login_screen_button_command).pack(side = LEFT)

    Label(master = new_frame, text = '        ', font = ('Helvetica', 10)).pack()

    new_frame.pack()


form_label()

signup_user_name()

signup_password()

confirm_password()

signup_button()

back2login()


main_signup.mainloop()


main_login()
```

Server Side Code Of Creating A Login / Signup Interface To Send And Receive Messages:

First Server Side:

```
#!C:\Users\Admin\AppData\Local\Programs\Python\Python38-32\python.exe
import cgi
import pymysql

database = pymysql.connect(host="localhost", port=3306, user="root", passwd="", d
b="navuy_messenger")
cursor = database.cursor()

form = cgi.FieldStorage()
username=form.getvalue('username')
password=form.getvalue('password')

cursor.execute("select * from user where username = %s", username)
row = cursor.fetchone()

print("Content-type: text/json\r\n\r\n")

exists = False

if row[0] == username and row[1] == password:
    exists = True

if exists:
    print("{\"boolean\": true}")
else:
    print("{\"boolean\": false}")

database.close()
```

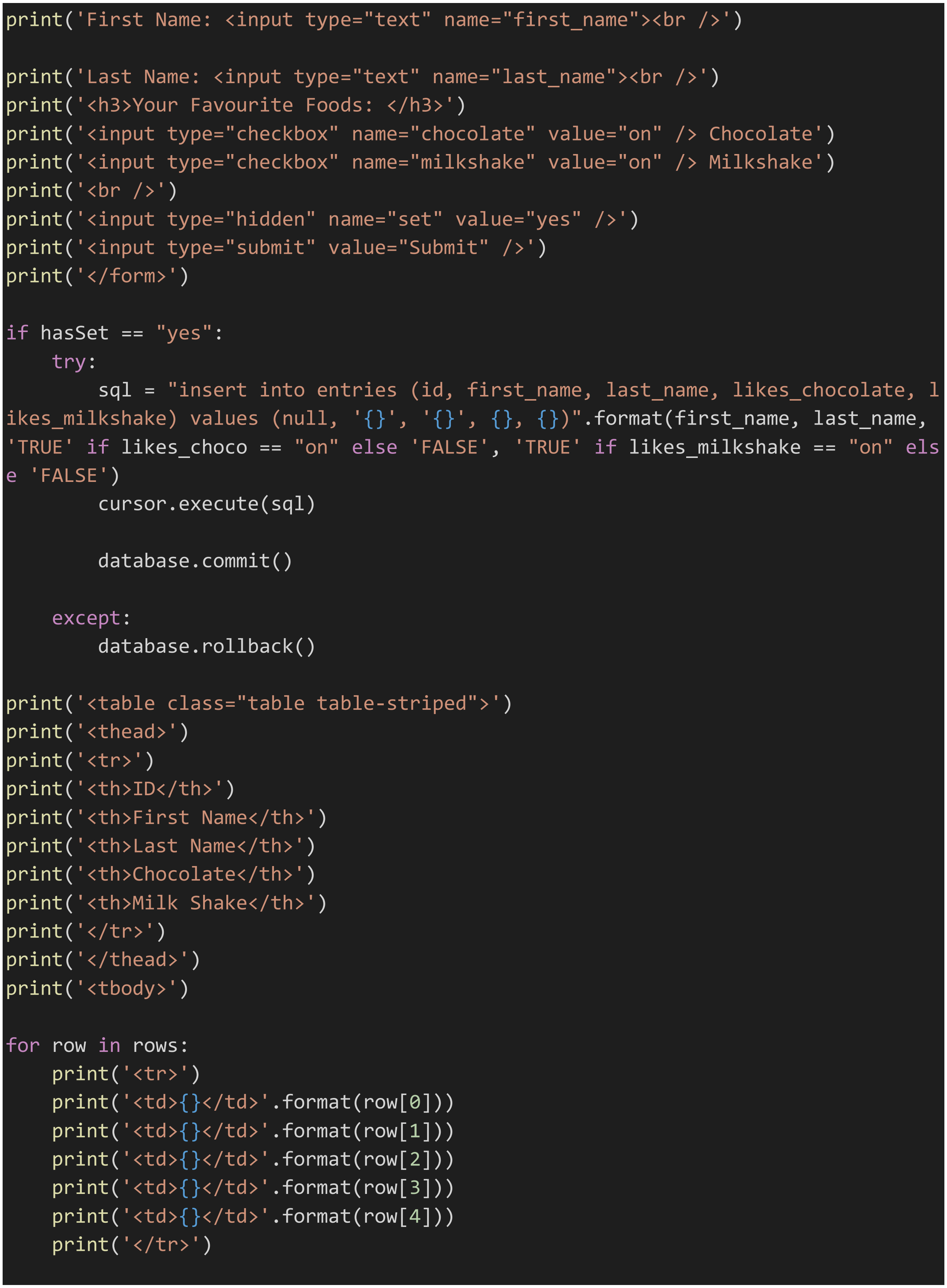
Second Server Side:



```
#!C:\Users\Admin\AppData\Local\Programs\Python\Python38-32\python.exe
import cgi
import pymysql

database = pymysql.connect(host="localhost", port=3306, user="root", passwd="", d
b="navuy_messenger")
cursor = database.cursor()
cursor.execute("select * from user")
rows = cursor.fetchall()

form = cgi.FieldStorage()
username=form.getvalue('username')

print("Content-type: text/json\r\n\r\n")

exists = False

for row in rows:
    if row[0] == username:
        exists = True

if exists:
    print("{\"boolean\": true}") # {"boolean": true}
else:
    print("{\"boolean\": false}")

database.close()
```

Third Server Side:

```
#!C:\Users\Admin\AppData\Local\Programs\Python\Python38-32\python.exe
import cgi
import pymysql

database = pymysql.connect(host="localhost", port=3306, user="root", passwd="", db="python_cgi_first")
cursor = database.cursor()

cursor.execute("select * from entries")

rows = cursor.fetchall()

form = cgi.FieldStorage()
hasSet = form.getvalue('set')
first_name=form.getvalue('first_name')
last_name=form.getvalue('last_name')

likes_choco = form.getvalue('chocolate')
if(likes_choco):
    choco="Chocolate"
else:
    choco=""

likes_milkshake = form.getvalue('milkshake')
if(likes_milkshake):
    milk="Milkshake"
else:
    milk=""

print('Content-type:text/html\r\n\r\n')
print('<html>')
print('<head>')
print('<title>CGI Python</title>')
print('<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-Vkoo8x4CGsO3+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh" crossorigin="anonymous">')
print('</head>')
print('<body>')
print('<h2>Survey Form</h2>')
print('<form action="/cgi-bin/python_cgi_first.py" method="post">')
```



```
print('First Name: <input type="text" name="first_name"><br />')

print('Last Name: <input type="text" name="last_name"><br />')
print('<h3>Your Favourite Foods: </h3>')
print('<input type="checkbox" name="chocolate" value="on" /> Chocolate')
print('<input type="checkbox" name="milkshake" value="on" /> Milkshake')
print('<br />')
print('<input type="hidden" name="set" value="yes" />')
print('<input type="submit" value="Submit" />')
print('</form>')

if hasSet == "yes":
    try:
        sql = "insert into entries (id, first_name, last_name, likes_chocolate, likes_milkshake) values (null, '{}', '{}', {}, {})".format(first_name, last_name, 'TRUE' if likes_choco == "on" else 'FALSE', 'TRUE' if likes_milkshake == "on" else 'FALSE')
        cursor.execute(sql)

        database.commit()

    except:
        database.rollback()

print('<table class="table table-striped">')
print('<thead>')
print('<tr>')
print('<th>ID</th>')
print('<th>First Name</th>')
print('<th>Last Name</th>')
print('<th>Chocolate</th>')
print('<th>Milk Shake</th>')
print('</tr>')
print('</thead>')
print('<tbody>')

for row in rows:
    print('<tr>')
    print('<td>{}</td>'.format(row[0]))
    print('<td>{}</td>'.format(row[1]))
    print('<td>{}</td>'.format(row[2]))
    print('<td>{}</td>'.format(row[3]))
    print('<td>{}</td>'.format(row[4]))
    print('</tr>')
print('</tbody>')
print('</table>')

print('<script src="https://code.jquery.com/jquery-3.4.1.slim.min.js" integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1yYfoRSJoZ+n" crossorigin="anonymous"></script>')
print('<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>')
print('<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj0O3uMBJnjuUD4Ih7YwaYd1iqfktj0Uod8GCExl3Og8ifwB6" crossorigin="anonymous"></script>')
print('</body>')
print('</html>')


database.close()
```

Fourth Server Side:

```
#!C:\Users\Admin\AppData\Local\Programs\Python\Python38-32\python.exe
import cgi
import pymysql

form = cgi.FieldStorage()
from_username=form.getvalue('from_username')
to_username=form.getvalue('to_username')

database = pymysql.connect(host="localhost", port=3306, user="root", passwd="", db="navuy_messenger")
cursor = database.cursor()
cursor.execute("SELECT * FROM `message` WHERE (`from_username` = \"" + from_username + "\" OR `from_username` = \"" + to_username + "\") AND (`to_username` = \"" + from_username + "\" OR `to_username` = \"" + to_username +"\")")
# "SELECT * FROM `message` WHERE (`from_username` = \"" + from_username + "\" OR `from_username` = \"" + to_username + "\") AND (`to_username` = \"" + from_username + "\" OR `to_username` = \"" + to_username +"\")"
rows = cursor.fetchall()

def construct_message(row):
    return '{"id":' + str(row[0]) + ',"from_username":"' + row[1] + '","to_username":"' + row[2] + '","message":"' + row[3] + '"}'

print("Content-type: text/json\r\n\r\n")
data = '{"array": ['
for i in range(0, len(rows)):
    if i + 1 == len(rows):
        data += construct_message(rows[i]) + ']}'
    else:
        data += construct_message(rows[i]) + ', '

print(data)

database.close()
```

Fivth Server Side:

```
#!C:\Users\Admin\AppData\Local\Programs\Python\Python38-32\python.exe
import cgi
import pymysql

database = pymysql.connect(host="localhost", port=3306, user="root", passwd="", d
b="navuy_messenger")
cursor = database.cursor()

form = cgi.FieldStorage()
from_username=form.getvalue('from_username')
to_username=form.getvalue('to_username')
message=form.getvalue('message')

print("Content-type: text/html\r\n\r\n")
print("<html>")
print("<head></head>")
print("<body>")

sql = "INSERT INTO `message` (`id`, `from_username`, `to_username`, `message`) VA
LUES (NULL, '{}', '{}', '{}');".format(from_username, to_username, message)
print(sql)
cursor.execute(sql)
database.commit()

database.close()


print("</body>")
print("</html>")
```

Sixth Server Side:

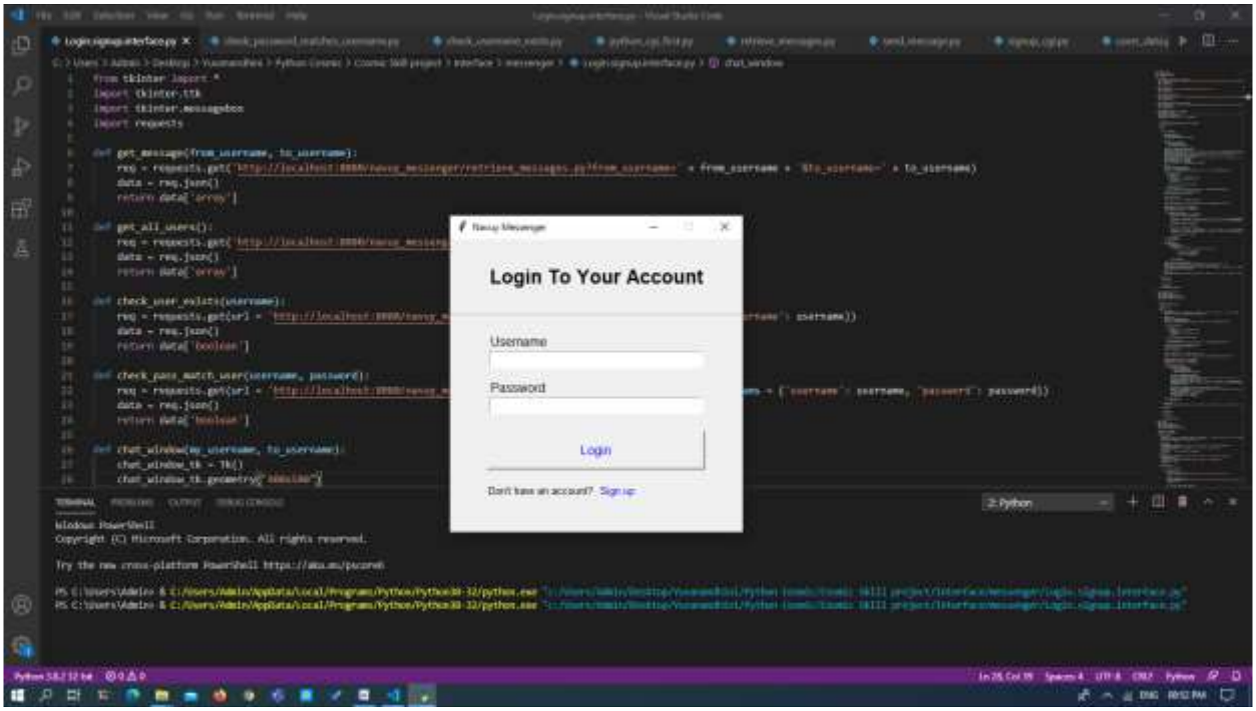
```
#!C:\Users\Admin\AppData\Local\Programs\Python\Python38-32\python.exe
import cgi
import pymysql

database = pymysql.connect(host="localhost", port=3306, user="root", passwd="", d
b="navuy_messenger")
cursor = database.cursor()

form = cgi.FieldStorage()
username=form.getvalue('username')
password=form.getvalue('password')

print("Content-type: text/html\r\n\r\n")
print("<html>")
print("<head></head>")
print("<body>")
print("")

sql = "insert into user (username, password, id) values ('{}', '{}', null)".forma
t(username, password)
print(sql)
cursor.execute(sql)
database.commit()

database.close()


print("</body>")
print("</html>")
```

Seventh Server Side:

```
#!C:\Users\Admin\AppData\Local\Programs\Python\Python38-32\python.exe
import cgi
import pymysql

print("Content-type: text/json\r\n\r\n")

database = pymysql.connect(host="localhost", port=3306, user="root", passwd="", db="navuy_messenger")
cursor = database.cursor()
cursor.execute("select * from user")
rows = cursor.fetchall()

data = '{"array": ['
for i in range(0, len(rows)):
    if i + 1 == len(rows):
        data += '"' + rows[i][0] + '"]}'
    else:
        data += '"{}", '.format(rows[i][0])


print(data)

database.close()
```

Output Of The Project:

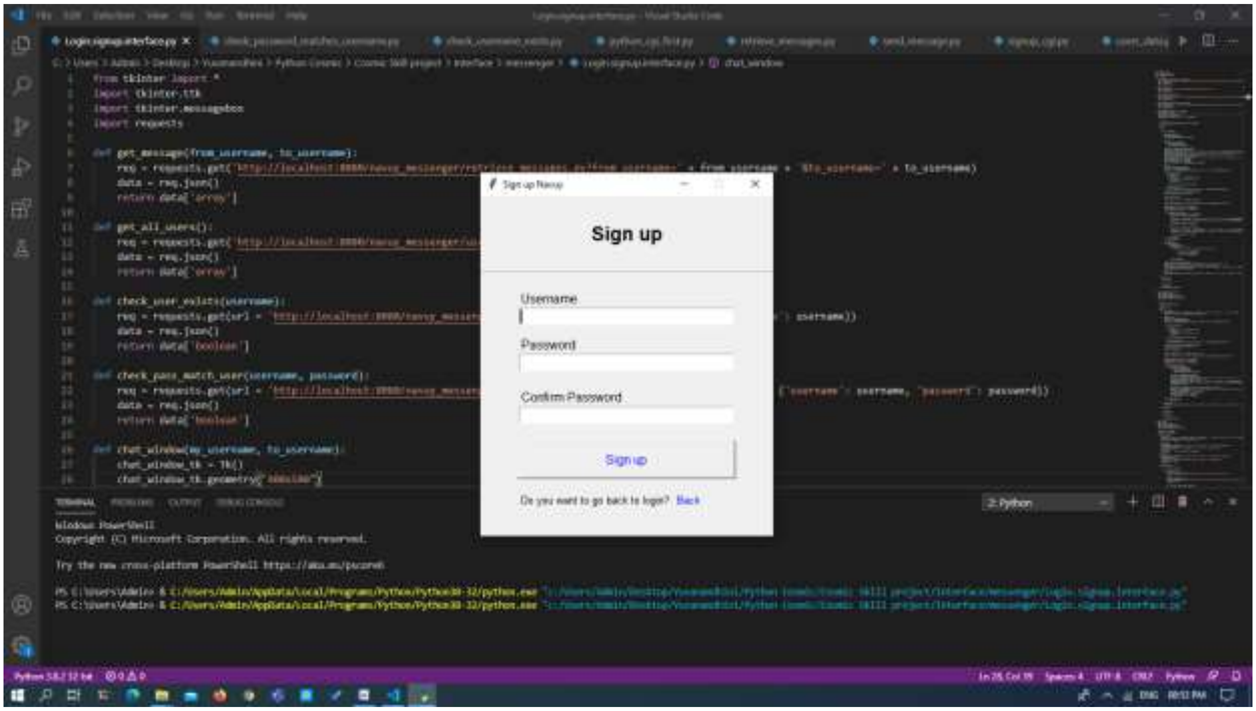
When we run this code, the very first output is “LOGIN WINDOW”. If we are already an user, we can login to it.



```
from tkinter import *
import tkinter.ttk
import tkinter.messagebox
import requests

def get_message(from_username, to_username):
    req = requests.get('http://localhost:8080/navy_messenger/retrieve_messages.py?from_username=' + from_username + '&to_username=' + to_username)
    data = req.json()
    return data['array']

def get_all_users():
    req = requests.get('http://localhost:8080/navy_messeng
    data = req.json()
    return data['array']

def check_user_exists(username):
    req = requests.get(url = 'http://localhost:8080/navy_m                    rname': username})
    data = req.json()
    return data['boolean']

def check_pass_match_user(username, password):
    req = requests.get(url = 'http://localhost:8080/navy_m                    ms = {'username': username, 'password': password})
    data = req.json()
    return data['boolean']

def chat_window(my_username, to_username):
    chat_window_tk = Tk()
    chat_window_tk.geometry("[illegible]")
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Admin> & C:/Users/Admin/AppData/Local/Programs/Python/Python38-32/python.exe "[illegible]/login_signup_interface.py"
PS C:\Users\Admin> & C:/Users/Admin/AppData/Local/Programs/Python/Python38-32/python.exe "[illegible]/login_signup_interface.py"
```

In this, we can click “sign up” if we aren’t a user before.

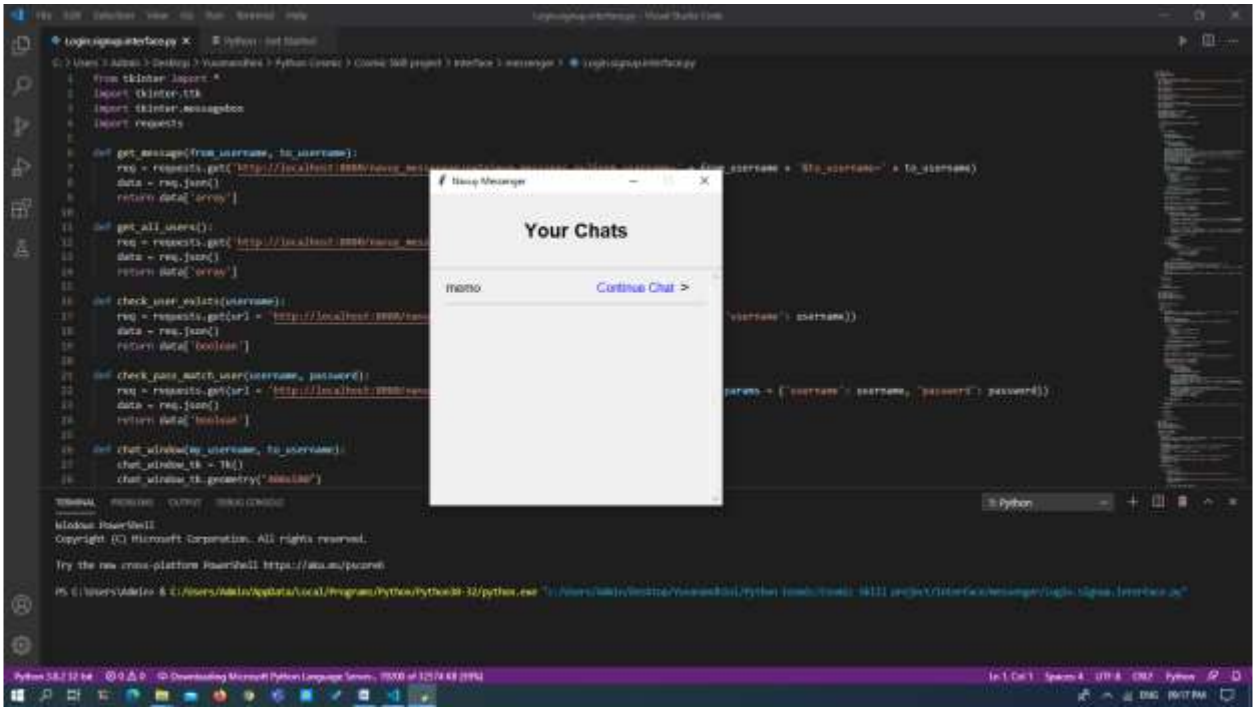


```
from tkinter import *
import tkinter.ttk
import tkinter.messagebox
import requests

def get_message(from_username, to_username):
    req = requests.get('http://localhost:8888/navoy_messenger/ret[illegible]  + from_username + '&to_username=' + to_username)
    data = req.json()
    return data['array']

def get_all_users():
    req = requests.get('http://localhost:8888/navoy_messenger/u[illegible]
    data = req.json()
    return data['array']

def check_user_exists(username):
    req = requests.get(url = 'http://localhost:8888/navoy_messen[illegible] ': username})
    data = req.json()
    return data['boolean']

def check_pass_match_user(username, password):
    req = requests.get(url = 'http://localhost:8888/navoy_messen[illegible] {'username': username, 'password': password})
    data = req.json()
    return data['boolean']

def chat_window(my_username, to_username):
    chat_window_tk = Tk()
    chat_window_tk.geometry('[illegible]')
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Admin> & C:/Users/Admin/AppData/Local/Programs/Python/Python38-32/python.exe "[illegible]/Login_signup_interface.py"
PS C:\Users\Admin> & C:/Users/Admin/AppData/Local/Programs/Python/Python38-32/python.exe "[illegible]/Login_signup_interface.py"
```

After clicking sign up, we can enter the details asked above & “sign up”.

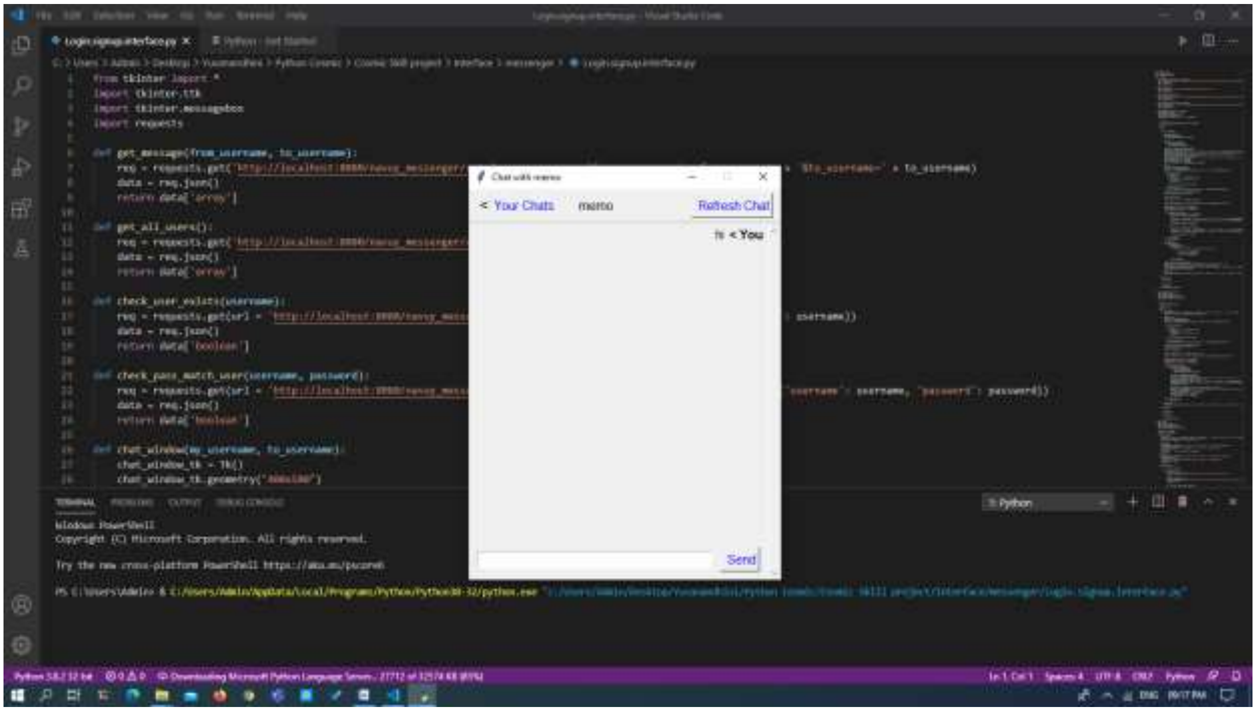
If you want to go back, you can click “Back”



```
from tkinter import *
import tkinter.ttk
import tkinter.messagebox
import requests

def get_message(from_username, to_username):
    req = requests.get('http://localhost:8888/navoy_mes[illegible] from_username + '&to_username=' + to_username)
    data = req.json()
    return data['array']

def get_all_users():
    req = requests.get('http://localhost:8888/navoy_mes[illegible]
    data = req.json()
    return data['array']

def check_user_exists(username):
    req = requests.get(url = 'http://localhost:8888/navo[illegible] 'username': username})
    data = req.json()
    return data['boolean']

def check_pass_match_user(username, password):
    req = requests.get(url = 'http://localhost:8888/navo[illegible] params = {'username': username, 'password': password})
    data = req.json()
    return data['boolean']

def chat_window(my_username, to_username):
    chat_window_tk = Tk()
    chat_window_tk.geometry('[illegible]')
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Admin> & C:/Users/Admin/AppData/Local/Programs/Python/Python38-32/python.exe "[illegible]/Login_signup_interface.py"
```

After clicking “sign up”, we can access our chats If already your friend is a user of this Navuy Messenger, You can text them your messeage. Above you can see it. After clicking “Continue Chat”, we can start texting them in personal. I have texed “hi” to the user named “memo”.

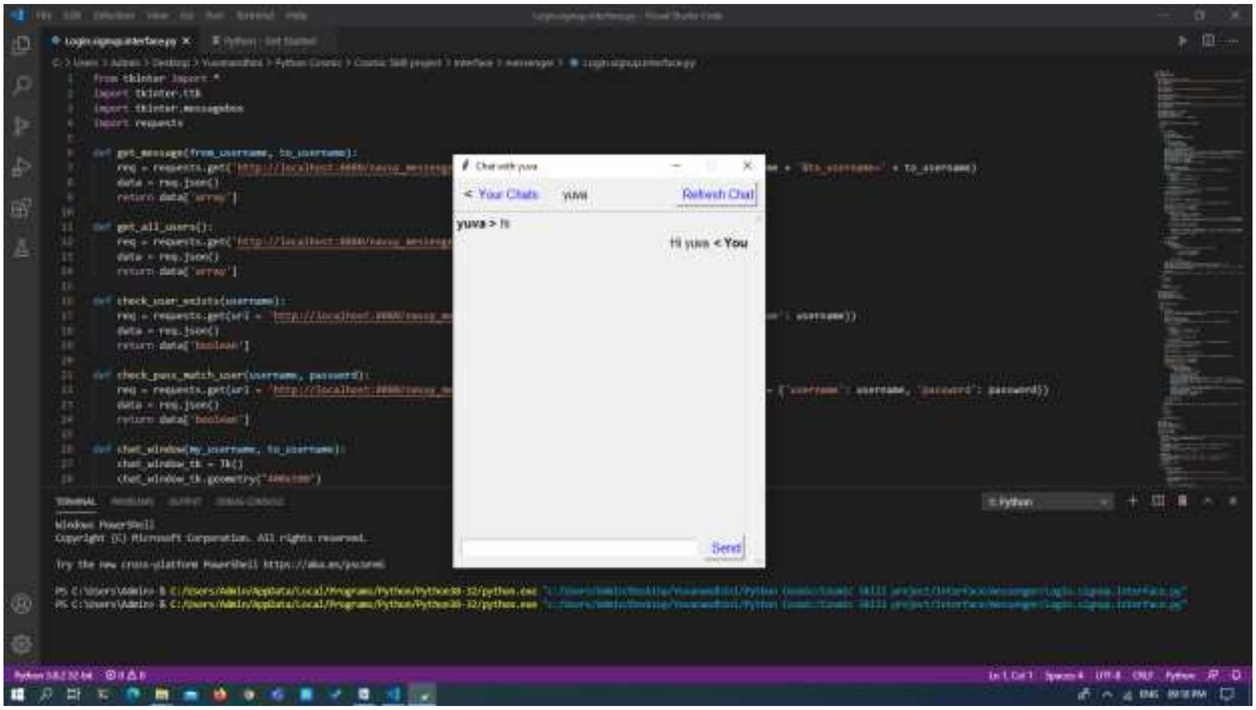


```
from tkinter import *
import tkinter.ttk
import tkinter.messagebox
import requests

def get_message(from_username, to_username):
    req = requests.get('http://localhost:8080/navuy_messenger/      + '&to_username=' + to_username)
    data = req.json()
    return data['array']

def get_all_users():
    req = requests.get('http://localhost:8080/navuy_messenger/
    data = req.json()
    return data['array']

def check_user_exists(username):
    req = requests.get(url = 'http://localhost:8080/navuy_mess      : username})
    data = req.json()
    return data['boolean']

def check_pass_match_user(username, password):
    req = requests.get(url = 'http://localhost:8080/navuy_mess      'username': username, 'password': password})
    data = req.json()
    return data['boolean']

def chat_window(my_username, to_username):
    chat_window_tk = Tk()
    chat_window_tk.geometry('400x500')
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Admin> & C:/Users/Admin/AppData/Local/Programs/Python/Python38-32/python.exe "c:/Users/Admin/Desktop/Yuvanandhini/Python Cosmic/Cosmic Skill project/interface/messenger/login_signup_interface.py"
```

Chat with memo

hi < You



```
from tkinter import *
import tkinter.ttk
import tkinter.messagebox
import requests

def get_message(from_username, to_username):
    req = requests.get('http://localhost:8080/navuy_messeng      + '&to_username=' + to_username)
    data = req.json()
    return data['array']

def get_all_users():
    req = requests.get('http://localhost:8080/navuy_messeng
    data = req.json()
    return data['array']

def check_user_exists(username):
    req = requests.get(url = 'http://localhost:8080/navuy_m      : username})
    data = req.json()
    return data['boolean']

def check_pass_match_user(username, password):
    req = requests.get(url = 'http://localhost:8080/navuy_m      {'username': username, 'password': password})
    data = req.json()
    return data['boolean']

def chat_window(my_username, to_username):
    chat_window_tk = Tk()
    chat_window_tk.geometry('400x500')
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Admin> & C:/Users/Admin/AppData/Local/Programs/Python/Python38-32/python.exe "c:/Users/Admin/Desktop/Yuvanandhini/Python Cosmic/Cosmic Skill project/interface/messenger/login_signup_interface.py"
PS C:\Users\Admin> & C:/Users/Admin/AppData/Local/Programs/Python/Python38-32/python.exe "c:/Users/Admin/Desktop/Yuvanandhini/Python Cosmic/Cosmic Skill project/interface/messenger/login_signup_interface.py"
```

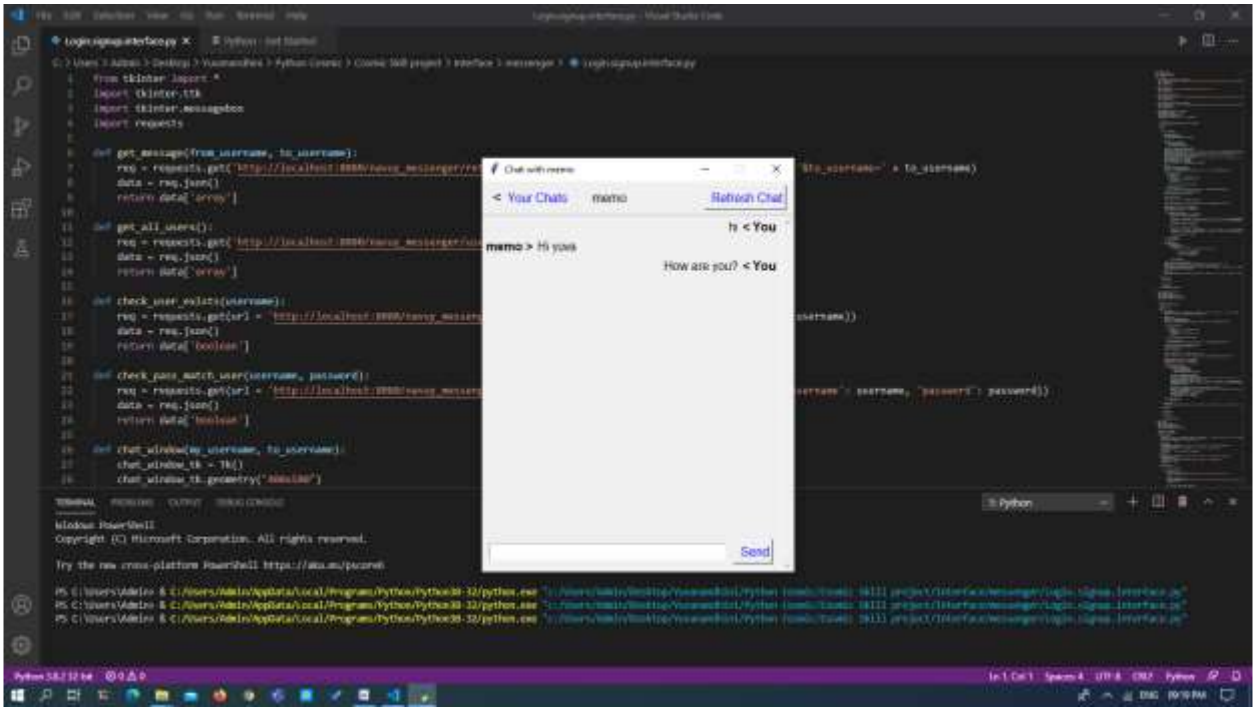
Chat with yuva

yuva > hi

Hi yuva < You

Here we can see our reply from our side to the user named “memo”.

Here we can see the reply from the user "memo"



```
from tkinter import *
import tkinter.ttk
import tkinter.messagebox
import requests

def get_message(from_username, to_username):
    req = requests.get('http://localhost:8080/navoy_messenger/re[illegible] '&to_username=' + to_username)
    data = req.json()
    return data['array']

def get_all_users():
    req = requests.get('http://localhost:8080/navoy_messenger/us[illegible]
    data = req.json()
    return data['array']

def check_user_exists(username):
    req = requests.get(url = 'http://localhost:8080/navoy_messeng[illegible]username))
    data = req.json()
    return data['boolean']

def check_pass_match_user(username, password):
    req = requests.get(url = 'http://localhost:8080/navoy_messeng[illegible]username': username, 'password': password))
    data = req.json()
    return data['boolean']

def chat_window(my_username, to_username):
    chat_window_tk = Tk()
    chat_window_tk.geometry('[illegible]')
```

```
hi < You
memo > Hi yuva
How are you? < You
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Admin> & C:/Users/Admin/AppData/Local/Programs/Python/Python38-32/python.exe "c:/Users/Admin/Desktop/[illegible]/Python Cosmic/Cosmic Skill project/Interface/messenger/Login_signup_interface.py"
PS C:\Users\Admin> & C:/Users/Admin/AppData/Local/Programs/Python/Python38-32/python.exe "c:/Users/Admin/Desktop/[illegible]/Python Cosmic/Cosmic Skill project/Interface/messenger/Login_signup_interface.py"
PS C:\Users\Admin> & C:/Users/Admin/AppData/Local/Programs/Python/Python38-32/python.exe "c:/Users/Admin/Desktop/[illegible]/Python Cosmic/Cosmic Skill project/Interface/messenger/Login_signup_interface.py"
```

Then I have replied "How are you?". Here the conversion is succeed through this lines of code.

Conclusion Of The Project:

The lines of code which I have written is successfully came out well. Here the one to one conversion between two people is done. Hence, **Created A Login / Signup Interface To Send And Receive Messages.**

Reference Of The Project:

Innovate Yourself

Tkinter

About tkinter

Bibliography Of The Project:

https://www.geeksforgeeks.org/python-gui-tkinter/

http://effbot.org/tkinterbook/

https://www.datacamp.com/community/tutorials/gui-tkinter-python?utm_source=adwords_ppc&utm_campaignid=1455363063&utm_adgroupid=65083631748&utm_device=c&utm_keyword=&utm_matchtype=b&utm_network=g&utm_adpostion=&utm_creative=278443377086&utm_targetid=aud-299261629574:dsa-429603003980&utm_loc_interest_ms=&utm_loc_physical_ms=9299627&gclid=Cj0KCQjwlvT8BRDeARIsAACRFiVmnCGl3RPGTPgaKLA9vb_PAE4ILwarxMUChwgOKjAd12hGrFIUbvwaAp8YEALw_wcB

2. Creating A Simple Chatter Bot To Chat Like A Human.

Creating A Simple Chatter Bot To Chat Like A Human.

Aim Of The Project:

In this project, I have written lines of code in step wise & mentioned GUI code to create a Simple Chatter Bot To Chat Like A Human. I have made tkinter as most important in GUI code. Final Outcome of the code is mentioned clearly for the basic understanding of the project.

Steps Performed During Development Of The Project:

GUI Code Of Creating A Simple Chatter Bot To Chat Like A Human:

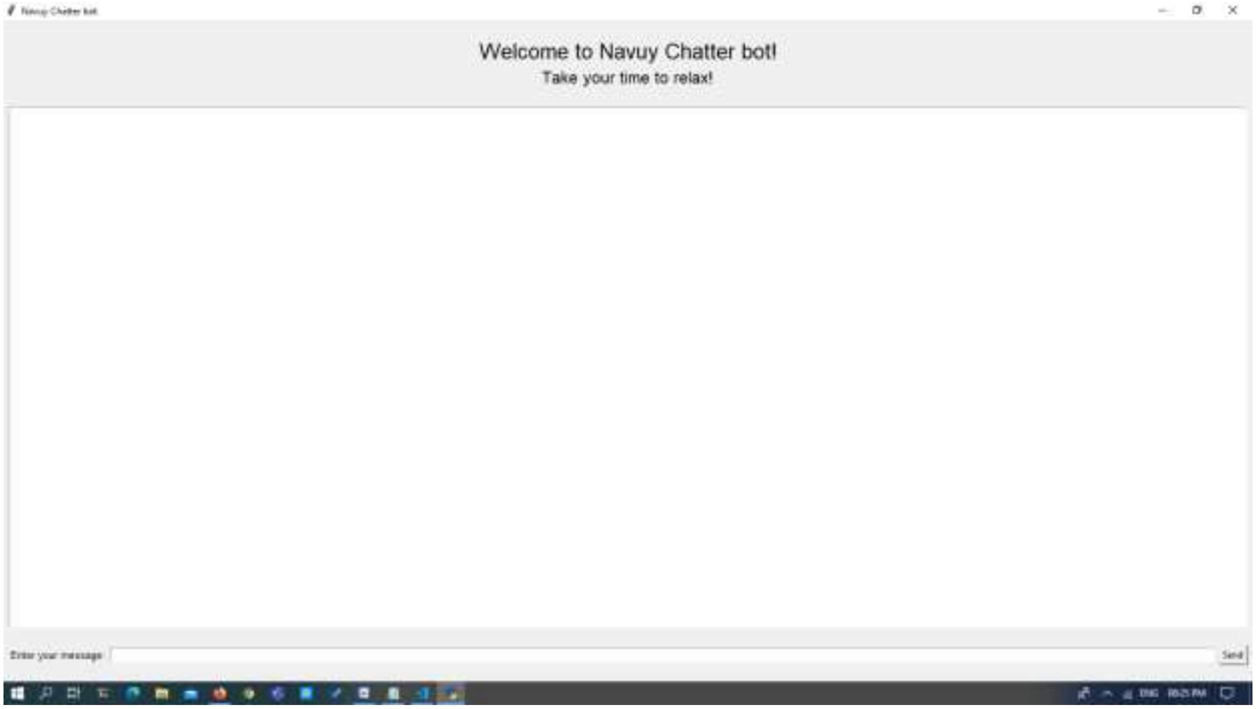
```
from tkinter import *
import tkinter as tk
from tkinter import messagebox
from tkinter import simpledialog
import tkinter.font as tkFont

import json
import os.path

sentence_pairs = {
    "what is your name?": "I'm Navuy!",
    "__first_launch__": True
}

if not os.path.exists('chat_text.txt'):
    f = open('chat_text.txt', 'w+')
    f.write(json.dumps(sentence_pairs))

    f.close()

f = open('chat_text.txt', 'r')
json_string = f
sentence_pairs = json.load(json_string)
f.close()

window = Tk()

window.title("Navuy Chatter bot")

space = Label(window, height=1)
space.pack()

headFontStyle = tkFont.Font(family="Lucida Grande", size=20)
bodyFontStyle = tkFont.Font(family="Lucida Grande", size=16)
textFontStyle = tkFont.Font(family="Lucida Grande", size=10)

welcomelbl = Label(window, text="Welcome to Navuy Chatter bot!",
font=headFontStyle)
welcomelbl.pack()
```

```
welcomeDescriptionlbl = Label(window, text="Take your time to relax!",
font=bodyFontStyle)

welcomeDescriptionlbl.pack()


space = Label(window, height=1)

space.pack()


messagesListFrame = Frame(window)


space = Label(messagesListFrame, height=1)

space.pack(side = LEFT)


messagesList = Listbox(messagesListFrame)

messagesList.pack(side = LEFT, fill = BOTH, expand = True)


space = Label(messagesListFrame, height=1)

space.pack(side = LEFT)


messagesListFrame.pack(fill = BOTH, expand = True)


space = Label(window, height=1)

space.pack()
```

```
messagesEntryFrame = Frame(window)

space = Label(messagesEntryFrame, height=1)
space.pack(side = LEFT)

messageEntryInnerFrame = Frame(messagesEntryFrame)

messageEntryLabel = Label(messageEntryInnerFrame, text="Enter your message: ", font=textFontStyle, anchor="w")
messageEntryLabel.pack(fill = BOTH, side = LEFT)

messageEntry = Entry(messageEntryInnerFrame)
messageEntry.focus()
messageEntry.pack(side= LEFT, fill=X, expand = True)

space = Label(messageEntryInnerFrame, height=1)
space.pack(side = LEFT)

def send():
    user_input = messageEntry.get()
    if not (user_input == "" or user_input == None):
        messagesList.insert(END, "You: %s" % user_input)
        if user_input.lower() in sentence_pairs.keys():
            messagesList.insert(END, 'Navuy: %s' % sentence_pairs[user_input.lower()])
        else:
            answer = ""

            while answer == "" or answer == None:
                answer = simpledialog.askstring("Navuy Says...", "Oops! I don't know what to say! Please text my reply yourself! So that I can reply to you next time without fail!", parent=window)

            sentence_pairs[user_input.lower()] = answer

            if sentence_pairs['__first_launch__'] == True:
                messagebox.showinfo("Navuy Says...","Thanks for your reply. By the way, you have contributed to Navuy by helping it to reply this message. You can now continue to message to Navuy!")
                sentence_pairs['__first_launch__'] = False

            messagesList.insert(END, 'Navuy: %s' % "I don't know to reply! (You Entered The Reply Yourself)")

        messageEntry.delete(0, 'end')
        messageEntry.focus()
    else:
        messagebox.showerror("Navuy Says...", "Please enter a valid message!")

button = Button(messageEntryInnerFrame, text="Send", command = send)
button.pack(side=LEFT)

space = Label(messageEntryInnerFrame, height=1)
space.pack(side = LEFT)

messageEntryInnerFrame.pack(fill = X)

messagesEntryFrame.pack(fill = X, )

space = Label(window, height=1)
space.pack()

def on_closing():
    if messagebox.askokcancel("Quit", "Do You Want To Quit Navuy Chatter Bot?"):
        window.destroy()

        f = open('chat_text.txt', 'w+')
        f.write(json.dumps(sentence_pairs))

        f.close()

window.protocol("WM_DELETE_WINDOW", on_closing)

def onEnter(event):
    send()

window.bind("<Return>", onEnter)
window.mainloop()
```

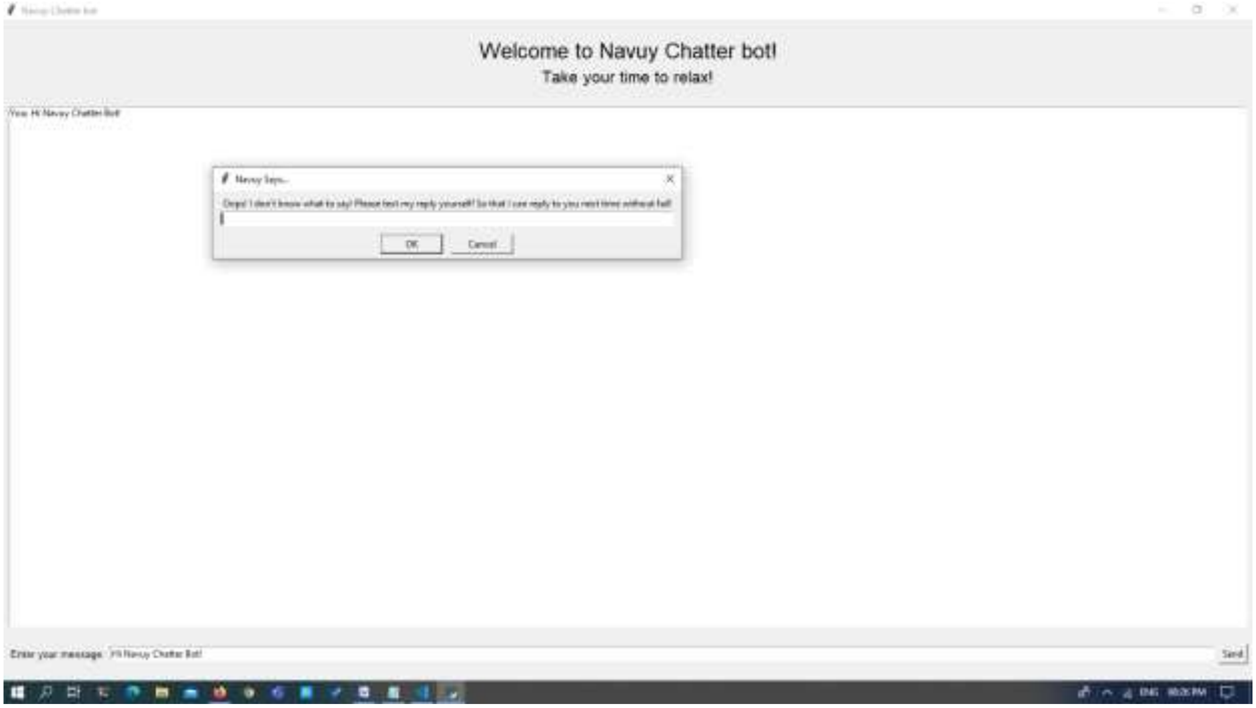
Output Of The Project:

Here after we run the complete code, we will get this chatter bot window.


Welcome to Navuy Chatter bot!

Take your time to relax!

I have texted the message to bot as “Hi navuy chatter Bot”. It will be showing the message box like this below. We can type chatterbot’s own replies

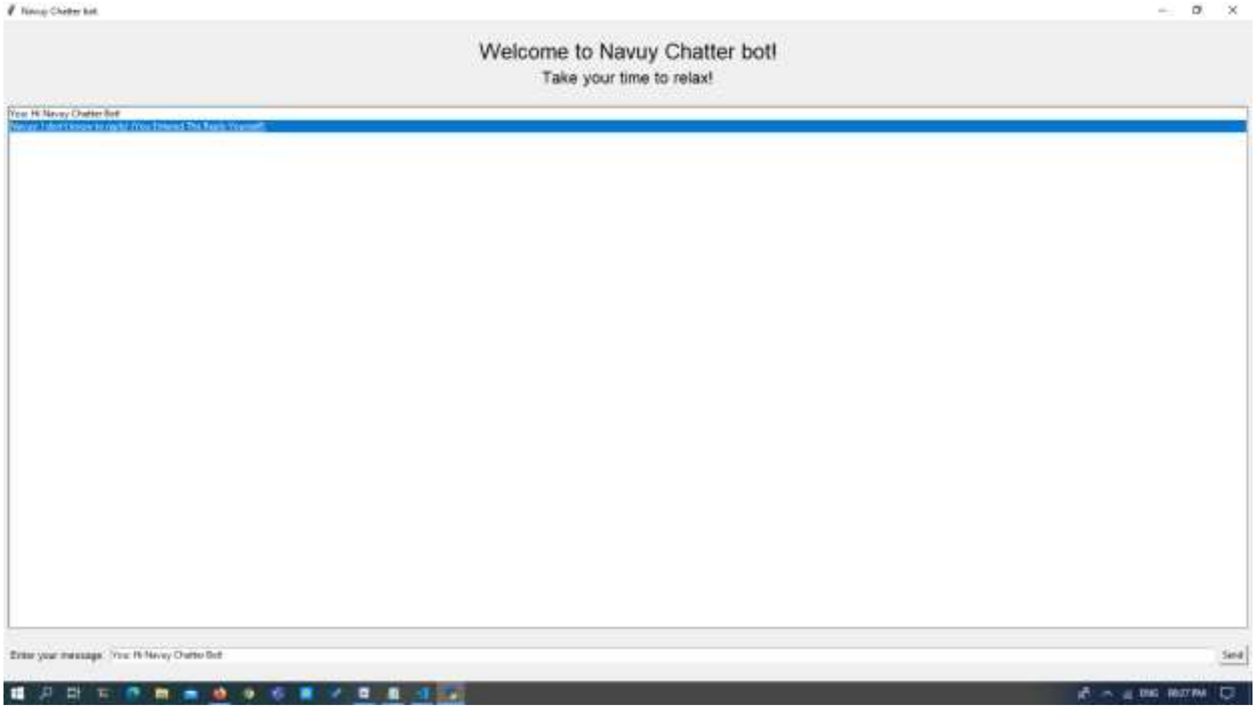

Welcome to Navuy Chatter bot!

Take your time to relax!

You: Hi Navuy Chatter Bot

Oops! I don't know what to say! Please text my reply yourself! So that I can reply to you next time without fail!

Enter your message: Hi Navuy Chatter Bot

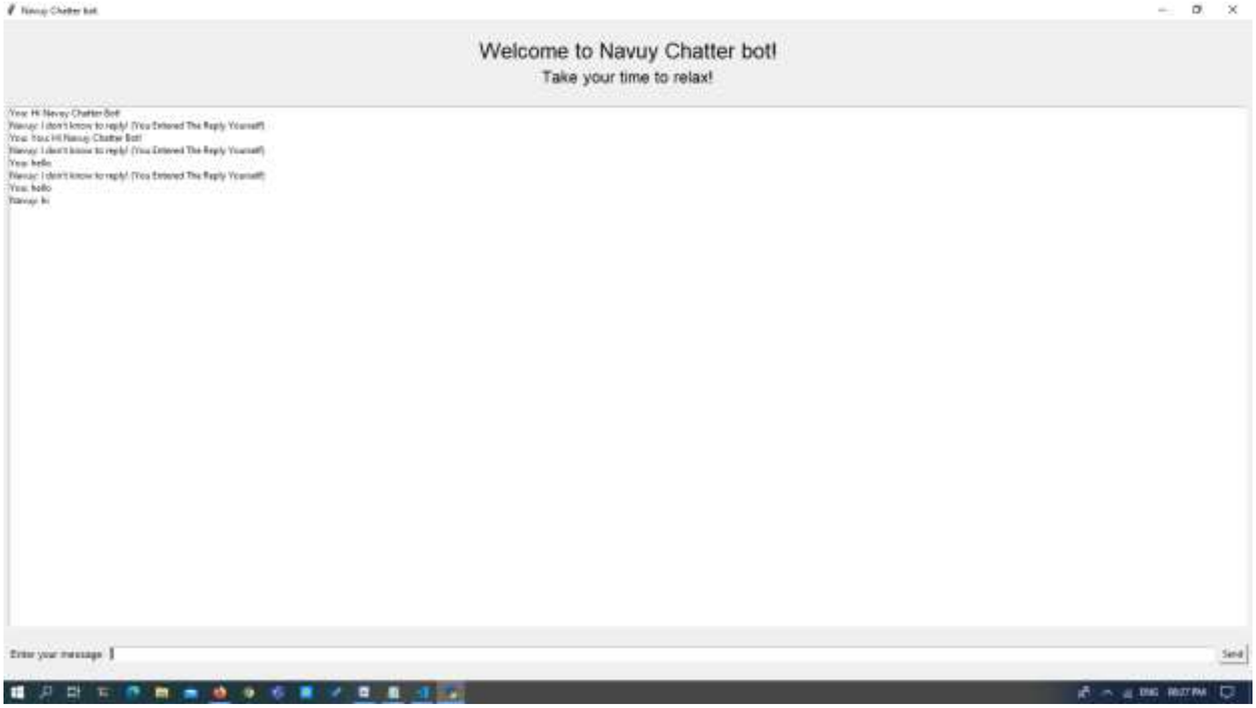


Welcome to Navuy Chatter bot!
Take your time to relax!

```
You: Hi Navuy Chatter Bot!
Navuy: I don't know to reply! (You Entered The Reply Yourself)
```

Enter your message: You: Hi Navuy Chatter Bot!

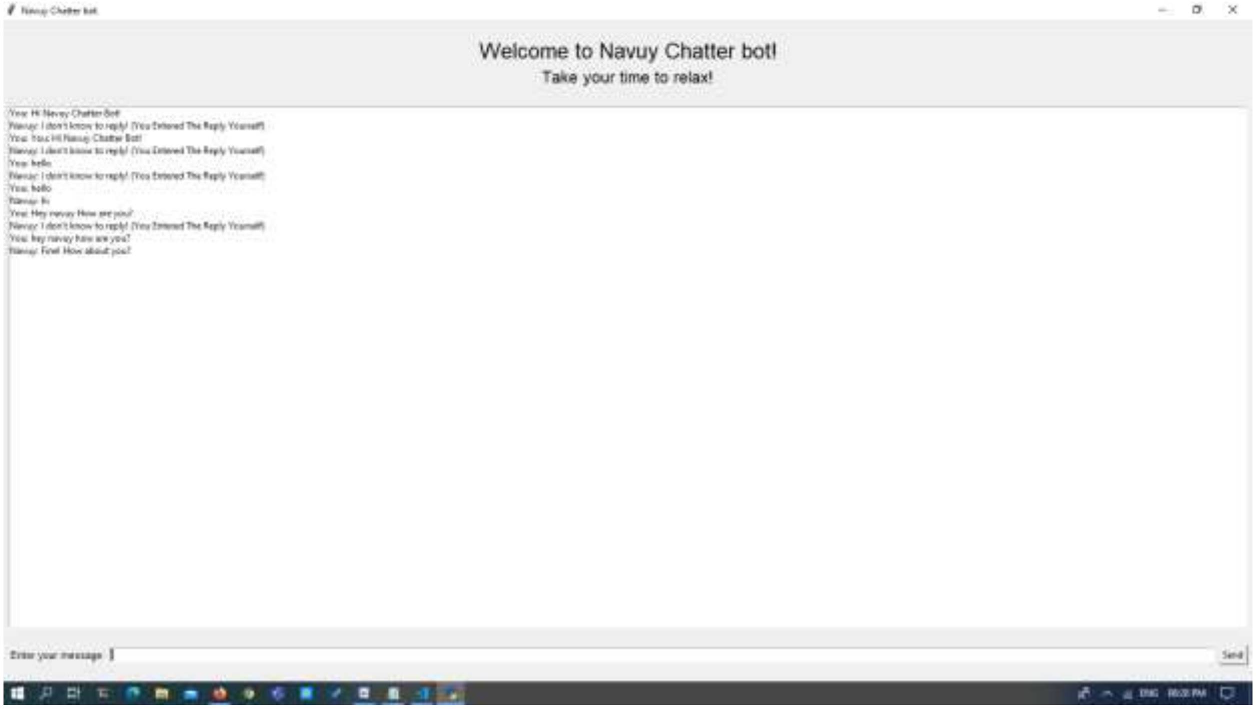
Here we can see Chatterbot’s reply here.



Welcome to Navuy Chatter bot!
Take your time to relax!

```
You: Hi Navuy Chatter Bot!
Navuy: I don't know to reply! (You Entered The Reply Yourself)
You: You: Hi Navuy Chatter Bot!
Navuy: I don't know to reply! (You Entered The Reply Yourself)
You: hello
Navuy: I don't know to reply! (You Entered The Reply Yourself)
You: hello
Navuy: hi
```

Enter your message:



```
You: Hi Navuy Chatter Bot!
Navuy: I don't know to reply! (You Entered The Reply Yourself)
You: You: Hi Navuy Chatter Bot!
Navuy: I don't know to reply! (You Entered The Reply Yourself)
You: hello
Navuy: I don't know to reply! (You Entered The Reply Yourself)
You: hello
Navuy: hi
You: Hey navuy How are you?
Navuy: I don't know to reply! (You Entered The Reply Yourself)
You: hey navuy how are you?
Navuy: Fine! How about you?
```

It replies to the text correctly when, we store it before.

Conclusion Of The Project:

The lines of code which I have written is successfully came out well. Here the conversion between Chatterbot & user is done. Hence, **Created A Simple Chatter Bot To Chat Like A Human.**

Reference Of The Project:

Innovate Yourself

Tkinter

About tkinter

Bibliography Of The Project:

https://www.geeksforgeeks.org/python-gui-tkinter/

http://effbot.org/tkinterbook/

https://www.datacamp.com/community/tutorials/gui-tkinter-python?utm_source=adwords_ppc&utm_campaignid=1455363063&utm_adgroupid=65083631748&utm_device=c&utm_keyword=&utm_matchtype=b&utm_network=g&utm_adpostion=&utm_creative=278443377086&utm_targetid=aud-299261629574:dsa-429603003980&utm_loc_interest_ms=&utm_loc_physical_ms=9299627&gclid=Cj0KCQjwlvT8BRDeARIsAACRFiVmnCGl3RPGTPgaKLA9vb_PAE4ILwarxMUChwgOKjAd12hGrFIUbvwaAp8YEALw_wcB

Thank You !