deseo hacer un registro de asistencia en python y google sheetss mi ruta de acceso json es: "C:\Users\VENUS.SUCAPUCA\Desktop\Pythom\Registro de asistencia\liquid-agility-450012-u8-570c585e86ec.json".

En la pantalla principal deseo que aparezca como título: REGISTRO DE ASISTENCIA , TEMA: "EN ESTE ESPACIO ES PARA QUE YO MODIFIQUE DIARIAMENTE", LUEGO MÁS ABAJO NOMBRE DE EXPOSITOR, " AHI TAMBIEN LO VOY A CAMBIAR DIARIAMENT", LUGAR: “TAMBIEN SE CAMBIE DIARIAMENTE”, FECHA: "QUE SE LLENE CON LA FECHA DEL DÍA". LOS USUARIOS QUE VAN A REGISTRAR SU ASISTENCIA INGRESEN SU DNI Y AL COSTADO LE INDIQUE BUSCAR, UNA VEZ QUE INGRESEN SU DNI EL PROGRAMA BUSQUE EN MI HOJA DE GOOGLE SHEET CON EL NOMBRE:"BASE DE DATOS DE USUARIOS" PESTAÑA: "Usuarios" POR EL DNI Y SE VISUALICE LOS SIGUIENTES DATOS: “DNI:, NOMBRE:, Y CARGO:”, Y TENGA UNA PESTAÑA DONDE VALIDEN SU ASISTENCIA DE AHÍ QUE LES DIRIGA AL INCIO. ESTE REGISTRO DE ASISTENCIA SE VAYA ALMACENANDO EN MI OTRA HOJA DE GOOGLE SHEETS LLAMADA: "REGISTRO DE ASISTENCIA" PESTAÑA "Asistencia", que se registren los siguiente datos: DNI, NOMBRE, CARGO, FECHA, HORA DE REGISTRO, TEMA. LOS DATOS DONDE SE CAMBIARAN LOS VALORES DE TEMA:, EXPOSITOR:, Y LUGAR ESTAN EN LA HOJA DE SHHETS LLAMADA “DATOS DE REUNION” pestaña “Reunion”, pero dentro de esta pesataña también esta una columna llamada “Habilitar” si su valor es “habilitado” que se ejecute todo el comando si hay otro valor, que muestre el siguiente mensaje “Registro de asistencia deshabilitado”. SI LOGRAN REGISTRASE LES DE UN MENSAJE “GRACIAS POR REGISTRAR SU ASISTENCIA”

LOS TITULOS DE TEMA:, EXPOSITOR:, LUGAR: , Y FECHA: ESTEN EN NEGRITA mas no los valores. Que estos datos estén centrados y tengan un fondo armonioso pon letras llamativas.

Llamativo:

import gspread

from oauth2client.service\_account import ServiceAccountCredentials

import tkinter as tk

from tkinter import messagebox

import datetime

# Configurar acceso a Google Sheets

scope = ["https://spreadsheets.google.com/feeds", "https://www.googleapis.com/auth/drive"]

cred\_path = r"C:\Users\VENUS.SUCAPUCA\Desktop\Pythom\Registro de asistencia\liquid-agility-450012-u8-570c585e86ec.json"

creds = ServiceAccountCredentials.from\_json\_keyfile\_name(cred\_path, scope)

client = gspread.authorize(creds)

# Abrir hojas de cálculo

sheet\_users = client.open("BASE DE DATOS DE USUARIOS").worksheet("Usuarios")

sheet\_meeting = client.open("DATOS DE REUNION").worksheet("Reunion")

sheet\_attendance = client.open("REGISTRO DE ASISTENCIA").worksheet("Asistencia")

# Obtener datos de la reunión

meeting\_data = sheet\_meeting.get\_all\_records()

current\_meeting = next((row for row in meeting\_data if row["Habilitar"] == "habilitado"), None)

if current\_meeting:

    tema\_actual = current\_meeting["TEMA"]

    expositor\_actual = current\_meeting["EXPOSITOR"]

    lugar\_actual = current\_meeting["LUGAR"]

else:

    tema\_actual, expositor\_actual, lugar\_actual = "Registro de asistencia deshabilitado", "", ""

# Crear la ventana principal

root = tk.Tk()

root.title("Registro de Asistencia")

root.geometry("500x400")

# Encabezado

tk.Label(root, text="REGISTRO DE ASISTENCIA", font=("Arial", 14, "bold")).pack(pady=10)

# Datos de la reunión

tk.Label(root, text=f"TEMA: {tema\_actual}", font=("Arial", 10, "bold")).pack()

tk.Label(root, text=f"EXPOSITOR: {expositor\_actual}", font=("Arial", 10, "bold")).pack()

tk.Label(root, text=f"LUGAR: {lugar\_actual}", font=("Arial", 10, "bold")).pack()

tk.Label(root, text=f"FECHA: {datetime.date.today()}", font=("Arial", 10, "bold")).pack(pady=5)

# Entrada de DNI

tk.Label(root, text="Ingrese su DNI:", font=("Arial", 10)).pack()

dni\_entry = tk.Entry(root, font=("Arial", 12))

dni\_entry.pack(pady=5)

def buscar\_dni():

    """Busca el DNI en la base de datos y muestra los datos del usuario."""

    dni = dni\_entry.get().strip()

    if not dni:

        messagebox.showwarning("Advertencia", "Ingrese un DNI válido")

        return

    users\_data = sheet\_users.get\_all\_records()

    user = next((row for row in users\_data if str(row["DNI"]) == dni), None)

    if user:

        nombre.set(f"NOMBRE: {user['Nombre']}")

        cargo.set(f"CARGO: {user['Cargo']}")

    else:

        messagebox.showerror("Error", "DNI no encontrado")

# Botón para buscar DNI

tk.Button(root, text="Buscar", command=buscar\_dni, font=("Arial", 10, "bold"), bg="lightblue").pack(pady=5)

# Etiquetas de datos del usuario

nombre = tk.StringVar()

cargo = tk.StringVar()

tk.Label(root, textvariable=nombre, font=("Arial", 10)).pack()

tk.Label(root, textvariable=cargo, font=("Arial", 10)).pack()

def registrar\_asistencia():

    """Registra la asistencia en Google Sheets."""

    if tema\_actual == "Registro de asistencia deshabilitado":

        messagebox.showerror("Error", "Registro de asistencia deshabilitado")

        return

    dni = dni\_entry.get().strip()

    if not dni or not nombre.get():

        messagebox.showwarning("Advertencia", "Debe buscar su DNI primero")

        return

    fecha\_hora = datetime.datetime.now().strftime("%Y-%m-%d %H:%M:%S")

    sheet\_attendance.append\_row([dni, nombre.get().replace("NOMBRE: ", ""), cargo.get().replace("CARGO: ", ""),

                                 datetime.date.today().strftime("%Y-%m-%d"), fecha\_hora, tema\_actual])

    messagebox.showinfo("Éxito", "GRACIAS POR REGISTRAR SU ASISTENCIA")

    dni\_entry.delete(0, tk.END)

    nombre.set("")

    cargo.set("")

# Botón de validación de asistencia

tk.Button(root, text="Validar Asistencia", command=registrar\_asistencia, font=("Arial", 10, "bold"), bg="lightgreen").pack(pady=10)

root.mainloop()

PRIMERA:

from flask import Flask, render\_template, request, redirect, url\_for, flash

from oauth2client.service\_account import ServiceAccountCredentials

import gspread

import datetime

# Inicializa la aplicación Flask

app = Flask(\_\_name\_\_)

app.secret\_key = "mi\_clave\_secreta"  # Asegúrate de usar una clave secreta para las sesiones

# Alcance de permisos para Google Sheets y Google Drive

scope = ["https://spreadsheets.google.com/feeds", "https://www.googleapis.com/auth/drive"]

# Ruta a tu archivo JSON de credenciales

creds = ServiceAccountCredentials.from\_json\_keyfile\_name(r'C:\Users\VENUS.SUCAPUCA\Desktop\Pythom\Registro de asistencia\liquid-agility-450012-u8-570c585e86ec.json', scope)

# Autenticación y conexión con Google Sheets

client = gspread.authorize(creds)

# Accede a la hoja "DATOS DE REUNION"

spreadsheet\_reunion = client.open("DATOS DE REUNION")

worksheet\_reunion = spreadsheet\_reunion.sheet1  # Ajusta según la pestaña que necesites

# Accede a la hoja "BASE DE DATOS DE USUARIOS"

spreadsheet\_usuarios = client.open("BASE DE DATOS DE USUARIOS")

worksheet\_usuarios = spreadsheet\_usuarios.sheet1  # Ajusta según la pestaña que necesites

# Accede a la hoja "REGISTRO DE ASISTENCIA"

spreadsheet\_asistencia = client.open("REGISTRO DE ASISTENCIA")

worksheet\_asistencia = spreadsheet\_asistencia.sheet1  # Ajusta según la pestaña que necesites

@app.route('/', methods=['GET', 'POST'])

def registro():

    if request.method == 'POST':

        dni = request.form.get("dni")

        # Busca el usuario en la hoja "Usuarios" por el DNI

        try:

            usuario = worksheet\_usuarios.find(dni)

            usuario\_data = worksheet\_usuarios.row\_values(usuario.row)

            nombre = usuario\_data[1]  # Ajusta según la columna correcta

            cargo = usuario\_data[2]   # Ajusta según la columna correcta

        except:

            flash("Usuario no encontrado, por favor verifica el DNI", "error")

            return redirect(url\_for('registro'))

        # Verifica si el registro está habilitado en la hoja "DATOS DE REUNION"

        reunion\_data = worksheet\_reunion.get\_all\_records()

        habilitado = reunion\_data[0]["Habilitar"] if reunion\_data else "deshabilitado"

        if habilitado.lower() != "habilitado":

            flash("Registro de asistencia deshabilitado, solicita a tu administrador que lo habilite", "error")

            return redirect(url\_for('registro'))

        # Si todo está bien, registrar la asistencia

        fecha = datetime.datetime.now().strftime("%Y-%m-%d")

        hora = datetime.datetime.now().strftime("%H:%M:%S")

        tema = reunion\_data[0]["Tema"] if reunion\_data else "Tema no disponible"

        # Registrar la asistencia en la hoja "REGISTRO DE ASISTENCIA"

        worksheet\_asistencia.append\_row([dni, nombre, cargo, fecha, hora, tema])

        flash("Gracias por registrar tu asistencia", "success")

        return redirect(url\_for('registro'))

    # Cargar datos de la reunión desde la hoja "DATOS DE REUNION"

    reunion\_data = worksheet\_reunion.get\_all\_records()

    tema = reunion\_data[0]["Tema"] if reunion\_data else "Tema no disponible"

    expositor = reunion\_data[0]["Expositor"] if reunion\_data else "Expositor no disponible"

    lugar = reunion\_data[0]["Lugar"] if reunion\_data else "Lugar no disponible"

    fecha = datetime.datetime.now().strftime("%Y-%m-%d")

    return render\_template('registro.html', tema=tema, expositor=expositor, lugar=lugar, fecha=fecha)

if \_\_name\_\_ == '\_\_main\_\_':

    app.run(debug=True)

HTML:

<!DOCTYPE html>

<html lang="es">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Registro de Asistencia</title>

    <style>

        body {

            font-family: Arial, sans-serif;

            background-color: #f0f8ff; /\* Fondo armonioso \*/

            color: #333;

            text-align: center;

        }

        h1 {

            color: #005f73; /\* Título en color llamativo \*/

            font-size: 3em;

            margin-top: 20px;

        }

        h2 {

            color: #0a9396;

            font-size: 1.8em;

            margin-top: 20px;

        }

        .titulo {

            font-weight: bold;

        }

        .form-container {

            background-color: #ffffff;

            border-radius: 10px;

            padding: 20px;

            margin-top: 30px;

            box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);

            display: inline-block;

            width: 60%;

            text-align: left;

        }

        .form-container input, .form-container button {

            font-size: 1.2em;

            padding: 10px;

            margin: 10px;

            border-radius: 5px;

            border: 1px solid #ddd;

        }

        .form-container input[type="text"] {

            width: 100%;

        }

        .form-container button {

            background-color: #0a9396;

            color: white;

            border: none;

            cursor: pointer;

        }

        .form-container button:hover {

            background-color: #005f73;

        }

        .resultado {

            margin-top: 20px;

        }

        .mensaje {

            margin-top: 20px;

            color: red;

            font-weight: bold;

        }

        .exito {

            margin-top: 20px;

            color: green;

            font-weight: bold;

        }

        .titulo-dato {

            font-weight: bold;

        }

    </style>

</head>

<body>

    <h1>REGISTRO DE ASISTENCIA</h1>

    <!-- Información de la reunión -->

    <h2>

        <span class="titulo">TEMA:</span> En este espacio es para que yo modifique diariamente

    </h2>

    <h2>

        <span class="titulo">EXPOSITOR:</span> Ahi tambien lo voy a cambiar diariamente

    </h2>

    <h2>

        <span class="titulo">LUGAR:</span> También se cambia diariamente

    </h2>

    <h2>

        <span class="titulo">FECHA:</span> {{ fecha\_actual }}

    </h2>

    <!-- Formulario para buscar el DNI del usuario -->

    <div class="form-container">

        <form action="/registro" method="POST">

            <label for="dni">Ingrese su DNI:</label><br>

            <input type="text" id="dni" name="dni" placeholder="DNI del usuario" required><br>

            <button type="submit">Buscar</button>

        </form>

    </div>

    <!-- Si hay resultados, mostrar los datos del usuario -->

    {% if usuario %}

        <div class="resultado">

            <h3><span class="titulo-dato">DNI:</span> {{ usuario['DNI'] }}</h3>

            <h3><span class="titulo-dato">NOMBRE:</span> {{ usuario['Nombre'] }}</h3>

            <h3><span class="titulo-dato">CARGO:</span> {{ usuario['Cargo'] }}</h3>

            <form action="/registrar\_asistencia" method="POST">

                <input type="hidden" name="dni" value="{{ usuario['DNI'] }}">

                <input type="hidden" name="nombre" value="{{ usuario['Nombre'] }}">

                <input type="hidden" name="cargo" value="{{ usuario['Cargo'] }}">

                <input type="hidden" name="tema" value="{{ tema }}">

                <input type="hidden" name="expositor" value="{{ expositor }}">

                <input type="hidden" name="lugar" value="{{ lugar }}">

                <input type="hidden" name="fecha" value="{{ fecha\_actual }}">

                <button type="submit">Validar Asistencia</button>

            </form>

        </div>

    {% endif %}

    <!-- Mensaje de error o éxito -->

    {% if mensaje\_error %}

        <div class="mensaje">{{ mensaje\_error }}</div>

    {% endif %}

    {% if mensaje\_exito %}

        <div class="exito">{{ mensaje\_exito }}</div>

    {% endif %}

</body>

</html>

TENGO ESTE CODIGO POYTHON:

from flask import Flask, render\_template, request, redirect, url\_for, flash

from oauth2client.service\_account import ServiceAccountCredentials

import gspread

import datetime

app = Flask(\_\_name\_\_)

app.secret\_key = "mi\_clave\_secreta"

# Configuración de Google Sheets

scope = ["https://spreadsheets.google.com/feeds", "https://www.googleapis.com/auth/drive"]

creds = ServiceAccountCredentials.from\_json\_keyfile\_name(

    r'C:\Users\VENUS.SUCAPUCA\Desktop\Pythom\Registro de asistencia\liquid-agility-450012-u8-570c585e86ec.json',

    scope

)

client = gspread.authorize(creds)

# Acceso a hojas de cálculo

spreadsheet\_usuarios = client.open("BASE DE DATOS DE USUARIOS")

worksheet\_usuarios = spreadsheet\_usuarios.worksheet("Usuarios")

spreadsheet\_asistencia = client.open("REGISTRO DE ASISTENCIA")

worksheet\_asistencia = spreadsheet\_asistencia.worksheet("Asistencia")

spreadsheet\_reunion = client.open("DATOS DE REUNION")

worksheet\_reunion = spreadsheet\_reunion.worksheet("Reunion")

@app.route('/', methods=['GET', 'POST'])

def registro():

    if request.method == 'POST':

        dni = request.form.get("dni")

        # Buscar usuario por DNI

        try:

            usuario = worksheet\_usuarios.find(dni)

            usuario\_data = worksheet\_usuarios.row\_values(usuario.row)

            if len(usuario\_data) < 3:

                flash("Error en la estructura de datos del usuario.", "error")

                return redirect(url\_for('registro'))

            nombre = usuario\_data[1]  # Asegúrate de que está en la columna correcta

            cargo = usuario\_data[2]

        except:

            flash("Usuario no encontrado, por favor verifica el DNI", "error")

            return redirect(url\_for('registro'))

        # Obtener datos de la reunión

        reunion\_data = worksheet\_reunion.get\_all\_records()

        if not reunion\_data:

            flash("No hay datos de reunión disponibles.", "error")

            return redirect(url\_for('registro'))

        habilitado = reunion\_data[0].get("Habilitar", "deshabilitado").strip().lower()

        if habilitado != "habilitado":

            flash("Registro de asistencia deshabilitado", "error")

            return redirect(url\_for('registro'))

        # Guardar asistencia

        fecha = datetime.datetime.now().strftime("%Y-%m-%d")

        hora = datetime.datetime.now().strftime("%H:%M:%S")

        tema = reunion\_data[0].get("Tema", "Tema no disponible")

        worksheet\_asistencia.append\_row([dni, nombre, cargo, fecha, hora, tema])

        flash("Gracias por registrar tu asistencia", "success")

        return redirect(url\_for('registro'))

    # Obtener datos de la reunión

    reunion\_data = worksheet\_reunion.get\_all\_records()

    if not reunion\_data:

        reunion\_data = [{"Tema": "No definido", "Expositor": "No definido", "Lugar": "No definido"}]

    tema = reunion\_data[0].get("Tema", "No definido")

    expositor = reunion\_data[0].get("Expositor", "No definido")

    lugar = reunion\_data[0].get("Lugar", "No definido")

    fecha = datetime.datetime.now().strftime("%Y-%m-%d")

    return render\_template('registro.html', tema=tema, expositor=expositor, lugar=lugar, fecha=fecha)

if \_\_name\_\_ == '\_\_main\_\_':

    app.run(debug=True)

Y ESTE HTML:

<<!DOCTYPE html>

<html lang="es">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Registro de Asistencia</title>

    <style>

        body { font-family: Arial, sans-serif; text-align: center; background-color: #f4f4f4; }

        h1 { color: #333; }

        label { font-weight: bold; }

        input, button { padding: 10px; margin: 10px; }

        .container { background: white; padding: 20px; border-radius: 10px; width: 50%; margin: auto; box-shadow: 0px 0px 10px 0px gray; }

    </style>

</head>

<body>

    <div class="container">

        <h1>REGISTRO DE ASISTENCIA</h1>

        <p><b>TEMA:</b> <span id="tema">Cargando...</span></p>

        <p><b>EXPOSITOR:</b> <span id="expositor">Cargando...</span></p>

        <p><b>LUGAR:</b> <span id="lugar">Cargando...</span></p>

        <p><b>FECHA:</b> <span id="fecha"></span></p>

        <label for="dni">Ingrese su DNI:</label>

        <input type="text" id="dni" placeholder="DNI">

        <button onclick="buscarDNI()">Buscar</button>

        <div id="datos\_usuario"></div>

        <button onclick="registrarAsistencia()">Registrar Asistencia</button>

        <p id="mensaje"></p>

    </div>

    <script>

        document.getElementById("fecha").innerText = new Date().toLocaleDateString();

        // Función para buscar usuario por DNI

        function buscarDNI() {

            let dni = document.getElementById("dni").value;

            fetch(`/buscar\_dni?dni=${dni}`)

            .then(response => response.json())

            .then(data => {

                if (data.error) {

                    document.getElementById("datos\_usuario").innerHTML = `<p style="color:red;">${data.error}</p>`;

                } else {

                    document.getElementById("datos\_usuario").innerHTML = `<p><b>DNI:</b> ${data.dni}</p><p><b>Nombre:</b> ${data.nombre}</p><p><b>Cargo:</b> ${data.cargo}</p>`;

                }

            });

        }

        // Función para registrar asistencia

        function registrarAsistencia() {

            let dni = document.getElementById("dni").value;

            fetch('/registrar\_asistencia', {

                method: 'POST',

                headers: { 'Content-Type': 'application/json' },

                body: JSON.stringify({ dni: dni })

            })

            .then(response => response.json())

            .then(data => {

                document.getElementById("mensaje").innerText = data.mensaje;

            });

        }

        // Obtener los datos de la reunión (Tema, Expositor, Lugar)

        fetch("/obtener\_datos\_reunion")

        .then(response => response.json())

        .then(data => {

            if (data.error) {

                document.getElementById("mensaje").innerText = data.error;

            } else {

                document.getElementById("tema").innerText = data.tema;

                document.getElementById("expositor").innerText = data.expositor;

                document.getElementById("lugar").innerText = data.lugar;

            }

        });

    </script>

</body>

</html>

VERIFICA SI HAY ERRORES Y DAME LOS CODIGOS CORRECTOS