

Q1. Apply regular expression for form validation. Create your domain-form using Tkinter Module. → Form should contain Text box [For Name, Email Id, Phone number], Dropdown [for Gender], Spinbox [for Year/DoB] and other necessary widgets required for your domain. → Validate Your Name, Email Id, Phone number in the form.

```
In [ ]: import tkinter as tk
import re
from tkinter import filedialog
import os
import subprocess

root = tk.Tk()
root.title("User Information Form")

window_width = 600
window_height = 400
root.geometry(f"{window_width}x{window_height}")

def validate_email(email):
    pattern = r'^[\w\.-]+@[\w\.-]+$'
    return re.match(pattern, email)

def validate_phone(phone):
    pattern = r'^\d{10}$'
    return re.match(pattern, phone)

documents = []

def add_user():
    name = name_entry.get()
    email = email_entry.get()
    phone = phone_entry.get()
    gender = gender_var.get()
    dob = dob_spinbox.get()
    if validate_email(email) and validate_phone(phone):
        user_info = f"Name: {name}\nEmail: {email}\nPhone: {phone}\nGender: {gender}"
        document_list.insert(tk.END, user_info)
    else:
        error_label.config(text="Invalid Email or Phone")

def add_document():
    file_path = filedialog.askopenfilename(title="Select a document")
    if file_path:
        documents.append(file_path)
        listbox.insert(tk.END, os.path.basename(file_path))

def view_document():
    selected_index = listbox.curselection()
    if selected_index:
        index = selected_index[0]
```

```
document_path = documents[index]

if os.name == 'nt': # Check if the OS is Windows
    os.startfile(document_path)
else:
    try:
        subprocess.Popen(['open', document_path]) # For macOS
    except FileNotFoundError:
        # Handle other platforms or errors
        pass

def delete_document():
    selected_index = listbox.curselection()
    if selected_index:
        index = selected_index[0]
        documents.pop(index)
        listbox.delete(index)

name_label = tk.Label(root, text="Name")
name_label.pack()
name_entry = tk.Entry(root)
name_entry.pack()

email_label = tk.Label(root, text="Email")
email_label.pack()
email_entry = tk.Entry(root)
email_entry.pack()

phone_label = tk.Label(root, text="Phone")
phone_label.pack()
phone_entry = tk.Entry(root)
phone_entry.pack()

gender_label = tk.Label(root, text="Gender")
gender_label.pack()
gender_var = tk.StringVar()
gender_dropdown = tk.OptionMenu(root, gender_var, "Male", "Female", "Other")
gender_dropdown.pack()

dob_label = tk.Label(root, text="Year of Birth")
dob_label.pack()
dob_spinbox = tk.Spinbox(root, from_=1900, to=2023)
dob_spinbox.pack()

add_button = tk.Button(root, text="Add", command=add_user)
add_button.pack()

error_label = tk.Label(root, text="", fg="red")
error_label.pack()

document_list = tk.Listbox(root)
document_list.pack()
```

```
delete_button = tk.Button(root, text="Delete", command=delete_document)
delete_button.pack()

add_button = tk.Button(root, text="Add Document", command=add_document)
view_button = tk.Button(root, text="View Document", command=view_document)
listbox = tk.Listbox(root, selectmode=tk.SINGLE)

add_button.pack()
view_button.pack()
listbox.pack()

root.mainloop()
```