|  |
| --- |
| Experiment No. 8 |
| Creating GUI with python containing widgets such as labels, textbox, radio, checkboxes and custom dialog boxes |
| Name: Krisha Chikka |
| Roll no. : 31 SE/Div1 |
| Date of Performance: 18/03/2024 |
| Date of Submission: 25/03/2024 |

**Experiment No. 8**

**Title:** Creating GUI with python containing widgets such as labels, textbox, radio, checkboxes and custom dialog boxes

**Aim:** To study and create GUI with python containing widgets such as labels, textbox, radio, checkboxes and custom dialog boxes

**Objective:** To introduce GUI, TKinter in python

**Theory:**

Python offers multiple options for developing GUI (Graphical User Interface). Out of all the GUI methods, tkinter is the most commonly used method. It is a standard Python interface to the Tk GUI toolkit shipped with Python. Python with tkinter is the fastest and easiest way to create the GUI applications. Creating a GUI using tkinter is an easy task.

To create a tkinter app:

Importing the module – tkinter

Create the main window (container)

Add any number of widgets to the main window

Apply the event Trigger on the widgets.

Importing tkinter is same as importing any other module in the Python code. Note that the name of the module in Python 2.x is ‘Tkinter’ and in Python 3.x it is ‘tkinter’.

**Program:**

import tkinter as tk

from tkinter import messagebox

def stop\_application():

root.destroy()

def submit\_form():

name = entry\_name.get()

email = entry\_email.get()

age = entry\_age.get()

if language\_var.get() == "":

messagebox.showerror("Error", "Please select your favorite programming language.")

return

language = language\_var.get()

gender = gender\_var.get()

hobbies = [hobby for hobby, var in hobbies\_vars.items() if var.get()]

message = f"Name: {name}\nEmail: {email}\nAge: {age}\nLanguage: {language}\nGender: {gender}\nHobbies: {', '.join(hobbies)}"

messagebox.showinfo("Form Submitted", message)

root = tk.Tk()

root.title("GUI CHERRY")

root.configure(background="Pink")

fontsize = ("Helvetica", 12)

# Labels

label\_name = tk.Label(root, text="Name:", font= fontsize)

label\_name.grid(row=0, column=0, padx=10, pady=5, sticky="e")

label\_email = tk.Label(root, text="Email:", font= fontsize)

label\_email.grid(row=1, column=0, padx=10, pady=5, sticky="e")

label\_age = tk.Label(root, text="Age:", font= fontsize)

label\_age.grid(row=2, column=0, padx=10, pady=5, sticky="e")

label\_language = tk.Label(root, text="Favorite Programming Language:", font= fontsize)

label\_language.grid(row=3, column=0, padx=10, pady=5, sticky="e")

label\_gender = tk.Label(root, text="Gender:", font= fontsize)

label\_gender.grid(row=4, column=0, padx=10, pady=5, sticky="e")

label\_hobbies = tk.Label(root, text="Hobbies:", font= fontsize)

label\_hobbies.grid(row=5, column=0, padx=10, pady=5, sticky="e")

# Entry Widgets

entry\_name = tk.Entry(root, width=25, font=("Helvetica", 15))

entry\_name.grid(row=0, column=1, padx=10, pady=5)

entry\_email = tk.Entry(root, width=25, font=("Helvetica", 15))

entry\_email.grid(row=1, column=1, padx=10, pady=5)

entry\_age = tk.Entry(root, width=25, font=("Helvetica", 15))

entry\_age.grid(row=2, column=1, padx=10, pady=5)

# Dropdown Menu

languages = ["Python", "Java", "C++", "JavaScript", "Ruby", "Other"]

language\_var = tk.StringVar()

language\_var.set("") # Default value

dropdown\_language = tk.OptionMenu(root, language\_var, \*languages)

dropdown\_language.config(font=fontsize)

dropdown\_language.grid(row=3, column=1, padx=10, pady=5, sticky="ew")

# Radio Buttons

gender\_var = tk.StringVar()

gender\_var.set("Male") # Default value

radio\_male = tk.Radiobutton(root, text="Male", variable=gender\_var, value="Male")

radio\_male.grid(row=4, column=1, padx=10, pady=5, sticky="w")

radio\_female = tk.Radiobutton(root, text="Female", variable=gender\_var, value="Female")

radio\_female.grid(row=4, column=1, padx=10, pady=5, sticky="e")

# Checkboxes

hobbies\_list = ["Reading", "Gaming", "Sketching", "Music", "Sports"]

hobbies\_vars = {}

for i, hobby in enumerate(hobbies\_list):

var = tk.BooleanVar()

checkbox = tk.Checkbutton(root, text=hobby, variable=var, font= fontsize)

checkbox.grid(row=6+i, column=1, padx=10, pady=2, sticky="w")

hobbies\_vars[hobby] = var

# Buttons

submit\_button = tk.Button(root, text="Submit", command=submit\_form, bg="#8f0647", fg= "white", height= 2, width= 20, borderwidth=7, relief=tk.RAISED)

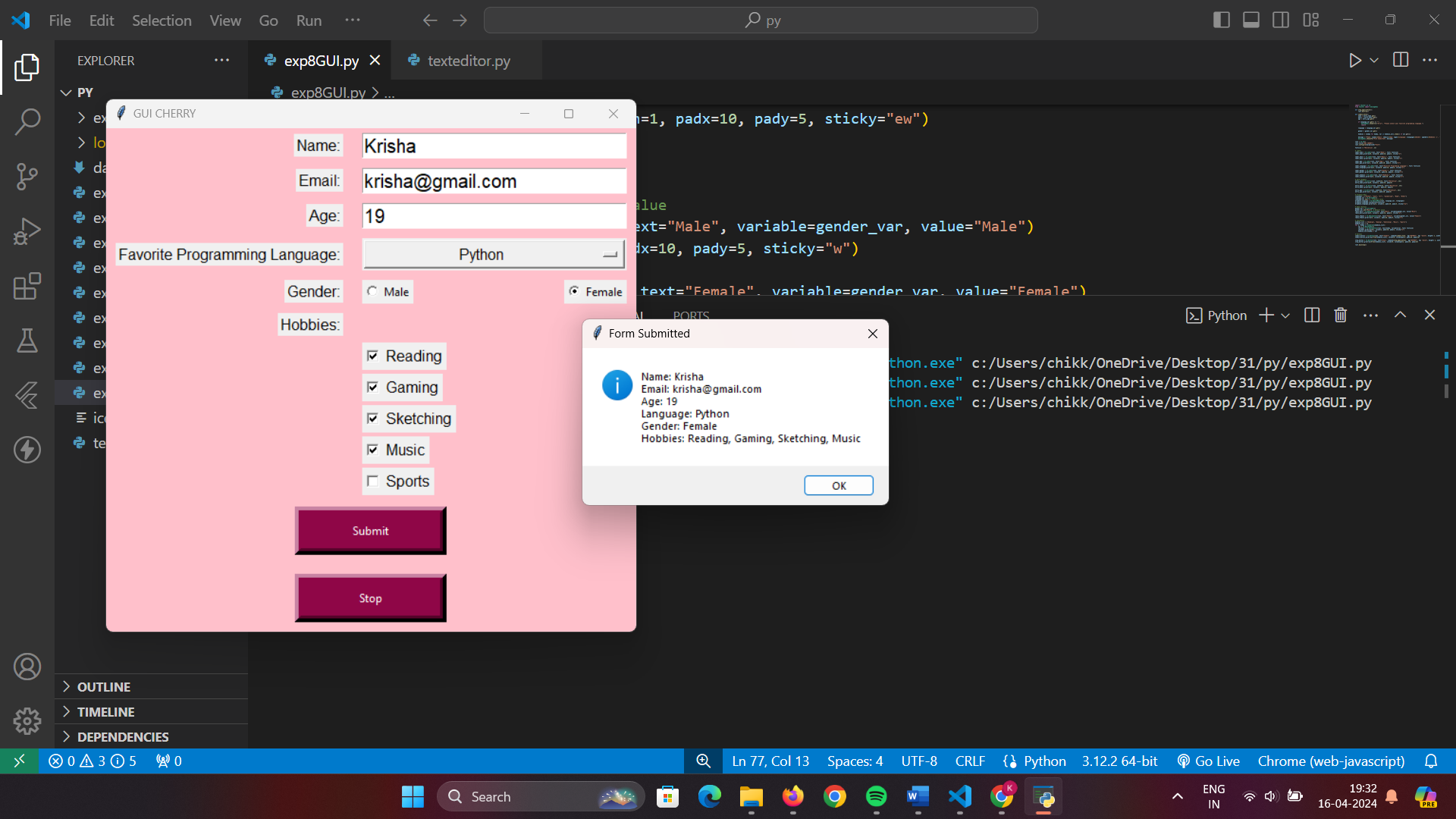
submit\_button.grid(row=7+len(hobbies\_list), column=0, columnspan=2, padx=10, pady=10)

stop\_button = tk.Button(root, text="Stop", command=stop\_application, bg="#8f0647", fg= "white", height= 2, width= 20, borderwidth=7, relief=tk.RAISED)

stop\_button.grid(row=8+len(hobbies\_list), column=0, columnspan=2, padx=10, pady=10)

root.mainloop()

**Output:**



**Conclusion:**

In this GUI application developed with Python's tkinter module, we explored various widgets such as labels, text boxes, radio buttons, checkboxes, and custom dialog boxes. By creating a user-friendly interface, users can input their information, select preferences, and submit the form. The application's aesthetic appeal was enhanced by customizing fonts, colors, and button styles. Through this project, we've gained a foundational understanding of tkinter's capabilities in building graphical user interfaces, reinforcing its significance in Python GUI development.