

Experiment No. 8

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

Date of Performance: 20/03/2024

Date of Submission: 27/03/2024

Vidyavardhini's College of Engineering & Technology

Department of Computer Engineering

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()
  # Check if any language is selected
  if language var.get() == "":
    messagebox.showerror("Error", "Please select your favorite programming language.")
    return
  # Get the selected language
  language = language var.get()
  # Get the selected gender
  gender = gender var.get()
  # Get the checked hobbies
  hobbies = [hobby for hobby, var in hobbies vars.items() if var.get()]
  # Display submitted information
```



```
f"Name:
                                    {name}\nEmail:
                                                       {email}\nAge:
                                                                       {age}\nLanguage:
{language}\nGender: {gender}\nHobbies: {', '.join(hobbies)}"
  messagebox.showinfo("Form Submitted", message)
root = tk.Tk()
root.title("Comprehensive GUI Application")
# Labels
label name = tk.Label(root, text="Name:")
label name.grid(row=0, column=0, padx=10, pady=5, sticky="e")
label email = tk.Label(root, text="Email:")
label email.grid(row=1, column=0, padx=10, pady=5, sticky="e")
label age = tk.Label(root, text="Age:")
label age.grid(row=2, column=0, padx=10, pady=5, sticky="e")
label language = tk.Label(root, text="Favorite Programming Language:")
label language.grid(row=3, column=0, padx=10, pady=5, sticky="e")
label gender = tk.Label(root, text="Gender:")
label_gender.grid(row=4, column=0, padx=10, pady=5, sticky="e")
```



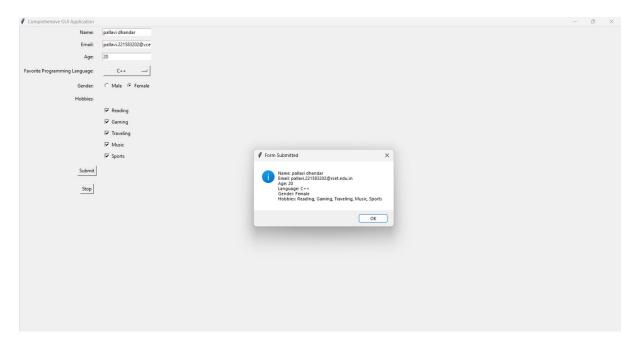
```
label hobbies = tk.Label(root, text="Hobbies:")
label hobbies.grid(row=5, column=0, padx=10, pady=5, sticky="e")
# Entry Widgets
entry name = tk.Entry(root)
entry name.grid(row=0, column=1, padx=10, pady=5)
entry_email = tk.Entry(root)
entry email.grid(row=1, column=1, padx=10, pady=5)
entry age = tk.Entry(root)
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.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", "Traveling", "Music", "Sports"]
hobbies vars = \{\}
for i, hobby in enumerate(hobbies list):
  var = tk.BooleanVar()
  checkbox = tk.Checkbutton(root, text=hobby, variable=var)
  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)
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)
stop button.grid(row=8+len(hobbies list), column=0, columnspan=2, padx=10, pady=10)
root.mainloop()
```



Output:



Conclusion:

Through Experiment No. 8, the process of creating GUI applications using Python's tkinter library was explored comprehensively. Various widgets such as labels, textboxes, radio buttons, checkboxes, and custom dialog boxes were utilized to develop an interactive form. This experiment served as an effective introduction to GUI development in Python, showcasing its versatility and ease of use. Overall, the hands-on experience provided valuable insights into the capabilities of tkinter and its practical applications in building graphical interfaces for Python programs.