**EXPERIMENT NO:16**

**Aim:** Write a python program to implement GUI Application using Tkinter.

**Program:**

**GUI.py**

from tkinter import \*

from tkinter import messagebox

from subprocess import call

login = Tk()

def open():

call(["python","Quiz.py"])

def get\_data(event=None):

username = usernameEntry.get()

password = PasswordEntry.get()

if username == '' or password == '':

print("Can't be Empty")

messagebox.showerror("Login Error", "Password or Username can't be Empty")

return

elif username == 'RS' or password == 'RS123':

print(username + " " + password)

login.destroy()

open() # Call the open function here

else:

messagebox.showerror("Login Error", "Password or Username is Incorrect")

window\_width = 600 # Set window width

window\_height = 400 # Set window height

login.geometry(f'{window\_width}x{window\_height}+200+10') # Set window size and position

login.title('Login Page')

login.configure(bg='#6667AB') # Change background color

def on\_enter(e):

if usernameEntry.get() == 'Username':

usernameEntry.delete(0, END)

usernameEntry.config(fg='#18237b') # Change text color when typed

def on\_enter1(e):

if PasswordEntry.get() == 'Password':

PasswordEntry.delete(0, END)

PasswordEntry.config(show='\*', fg='#18237b') # Change text color and show password characters when typed

def toggle\_password\_visibility():

if PasswordEntry['show'] == '\*':

PasswordEntry.config(show='')

eye\_button.config(text='Hide')

else:

PasswordEntry.config(show='\*')

eye\_button.config(text='Show')

heading = Label(login, text='USER LOGIN', font=('Georgia', 23, 'bold'), bg='#6667AB', fg='#abeedc')

heading.place(relx=0.5, rely=0.2, anchor="center") # Center align the heading

usernameEntry = Entry(login, width=25, font=('Times', 13), bd=0, fg='#6667AB')

usernameEntry.place(relx=0.5, rely=0.3, anchor="center") # Center align the username entry

usernameEntry.insert(0, 'Username')

usernameEntry.bind("<FocusIn>", on\_enter)

frame1 = Frame(login, width=250, height=2, bg='#abeedc')

frame1.place(relx=0.5, rely=0.35, anchor="center") # Center align the frame

PasswordEntry = Entry(login, width=25, font=('Times', 13), bd=0, fg='#6667AB', show='')

PasswordEntry.place(relx=0.5, rely=0.4, anchor="center") # Center align the password entry

PasswordEntry.insert(0, 'Password')

PasswordEntry.bind("<FocusIn>", on\_enter1)

PasswordEntry.bind("<Return>", get\_data) # Bind the <Return> event to the get\_data function

frame2 = Frame(login, width=250, height=2, bg='#abeedc')

frame2.place(relx=0.5, rely=0.45, anchor="center") # Center align the frame

eye\_button = Button(login, text='Show', bd=0, bg='#6667AB', activebackground='#6667AB', cursor='hand2', fg='#abeedc', command=toggle\_password\_visibility)

eye\_button.place(relx=0.73, rely=0.4, anchor="center") # Position the eye button

forgetbutton = Button(login, text='Forget Password?', bd=0, bg='#6667AB', activebackground='#6667AB', cursor='hand2', fg='#abeedc')

forgetbutton.place(relx=0.5, rely=0.5, anchor="center") # Center align the forget password button and move down

loginbutton = Button(login, text='Login', font=('open sans', 16, 'bold'), fg='white', bg='#abeedc', activeforeground='white', activebackground='green', cursor='hand2', bd=0, width=19, command=get\_data)

loginbutton.place(relx=0.5, rely=0.6, anchor="center") # Center align the login button and move down

loginbutton.bind("<Return>", get\_data)

login.mainloop()

**Quiz.py**

import tkinter as tk

from tkinter import messagebox

class QuizApp(tk.Tk):

def \_\_init\_\_(self, \*args, \*\*kwargs):

super().\_\_init\_\_(\*args, \*\*kwargs)

self.title("Quiz App")

self.geometry("400x300")

self.current\_question = 0

self.score = 0

self.questions = [

{

"question": "What is the capital of France?",

"options": ["Paris", "London", "Berlin", "Rome"],

"answer": "Paris"

},

{

"question": "How many continents are there?",

"options": ["5", "6", "7", "8"],

"answer": "7"

},

{

"question": "What is the square root of 25?",

"options": ["3", "4", "5", "6"],

"answer": "5"

}

]

self.create\_widgets()

def create\_widgets(self):

self.question\_frame = tk.Frame(self)

self.question\_frame.pack(pady=10)

self.question\_label = tk.Label(self.question\_frame, text="")

self.question\_label.pack()

self.answer\_frame = tk.Frame(self)

self.answer\_frame.pack(pady=10)

self.answer\_var = tk.StringVar()

self.answer\_var.set(None)

self.options\_radio\_buttons = []

for i in range(4):

rb = tk.Radiobutton(self.answer\_frame, text="", variable=self.answer\_var, value="")

rb.pack(anchor=tk.W)

self.options\_radio\_buttons.append(rb)

self.submit\_button = tk.Button(self, text="Submit", command=self.submit\_answer)

self.submit\_button.pack(pady=10)

self.next\_question()

def next\_question(self):

if self.current\_question < len(self.questions):

question\_data = self.questions[self.current\_question]

self.question\_label.config(text=question\_data["question"])

if "options" in question\_data:

for i, option in enumerate(question\_data["options"]):

self.options\_radio\_buttons[i].config(text=option, value=option)

self.options\_radio\_buttons[i].config(state=tk.NORMAL)

for j in range(len(question\_data["options"]), 4):

self.options\_radio\_buttons[j].config(text="", value="", state=tk.DISABLED)

else:

for rb in self.options\_radio\_buttons:

rb.config(text="", value="", state=tk.DISABLED)

self.current\_question += 1

else:

self.show\_results()

def submit\_answer(self):

selected\_answer = self.answer\_var.get()

correct\_answer = self.questions[self.current\_question - 1]["answer"]

if selected\_answer == correct\_answer:

self.score += 1

self.next\_question()

def show\_results(self):

messagebox.showinfo("Quiz Results", f"You scored {self.score}/{len(self.questions)}")

print(f"Quiz Results: You scored {self.score}/{len(self.questions)}")

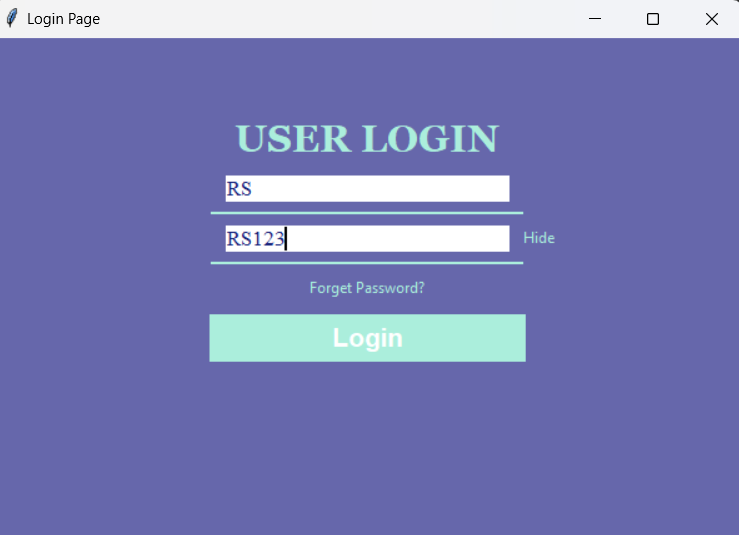
self.destroy()

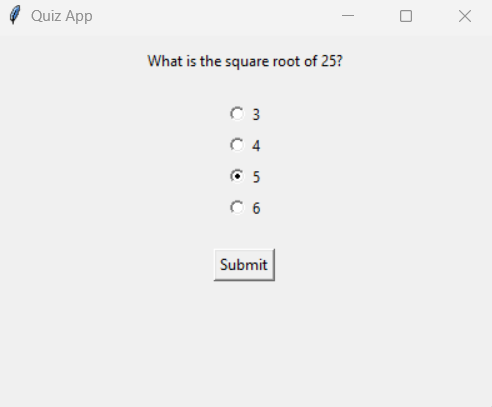
if \_\_name\_\_ == "\_\_main\_\_":

app = QuizApp()

app.mainloop()

**OUTPUT:**



****

**Conclusion:** Hence, we successfully implemented GUI Application using Tkinter.