Experiment No. 8
Creating GUI with python containing widgets such as labels, textbox, radio, checkboxes and custom dialog boxes
Date of Performance: 21/03/24
Date of Submission: 21/03/24

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'.

Code:

```
from tkinter import *
from tkinter import messagebox


def validate_login():
    username = username_entry.get()
    password = password_entry.get()
    remember_me = remember_me_var.get()


    if username == "admin" and password == "password":
        if remember_me:
            messagebox.showinfo("Login Successful", "Welcome, Admin! Remember Me Checked.")
```

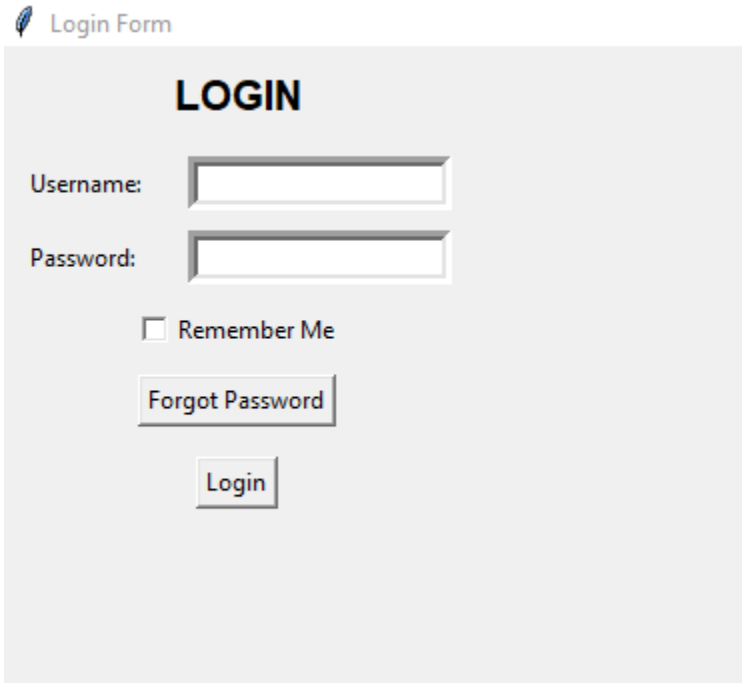
```
        else:
            messagebox.showinfo("Login Successful", "Welcome, Admin!")
    else:
        messagebox.showerror("Login Failed", "Invalid username or password")


def forgot_password():
    def change_password():
        messagebox.showinfo("Change Password", "Please proceed to change your password.")

    # Create a new window for the "Forgot Password" option
    forgot_pw_window = Toplevel(top)
    forgot_pw_window.title("Forgot Password")

    # Label and Button for changing password
    change_pw_label = Label(forgot_pw_window, text="Forgot your password? You can change it here:")
    change_pw_label.pack(pady=5)
```

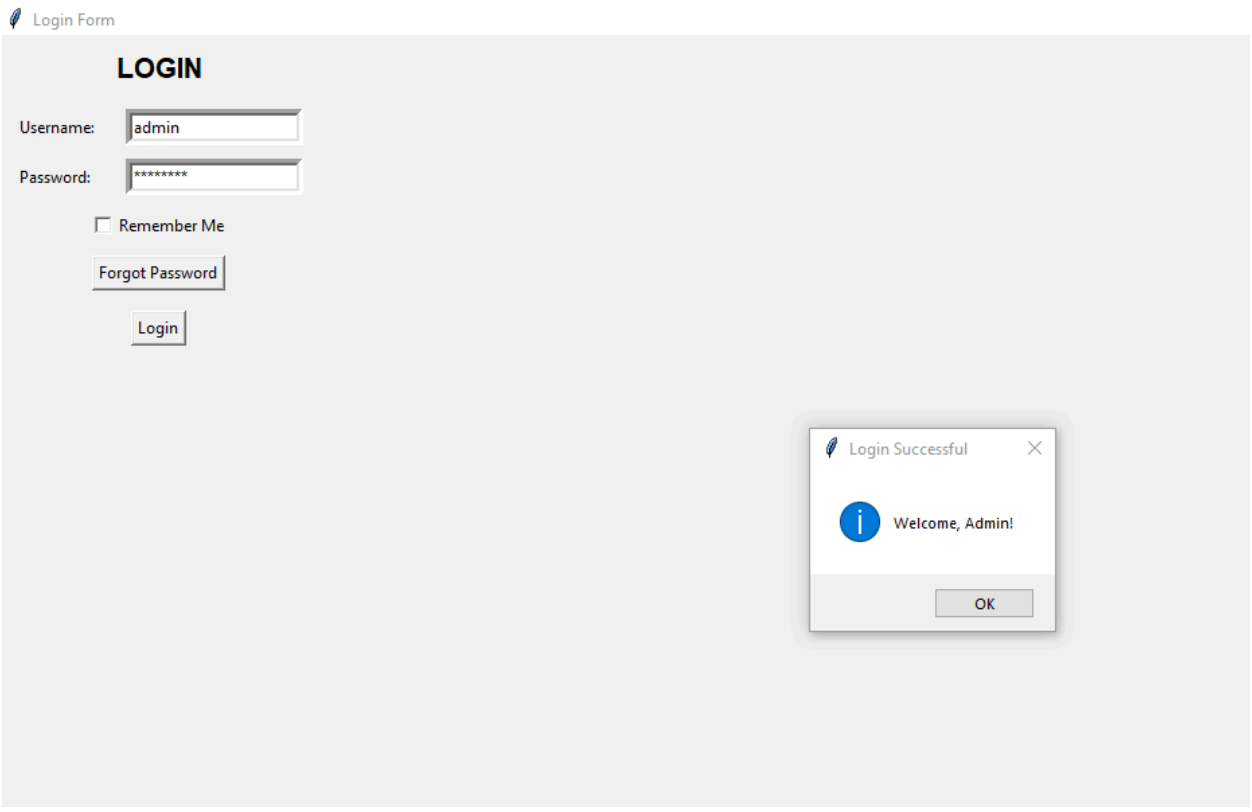
```
    change_pw_button = Button(forgot_pw_window, text="Change Password", command=change_password)
    change_pw_button.pack(pady=5)


top = Tk()
top.title("Login Form")


# Label for LOGIN
login_label = Label(top, text="LOGIN", font=("Helvetica", 16, "bold"))
login_label.grid(row=0, column=0, columnspan=2, pady=10)


# Label and Entry for username
username_label = Label(top, text="Username:")
username_label.grid(row=1, column=0, sticky=W, padx=10, pady=5)


username_entry = Entry(top, bd=5)
username_entry.grid(row=1, column=1, padx=10, pady=5)


# Label and Entry for password
```

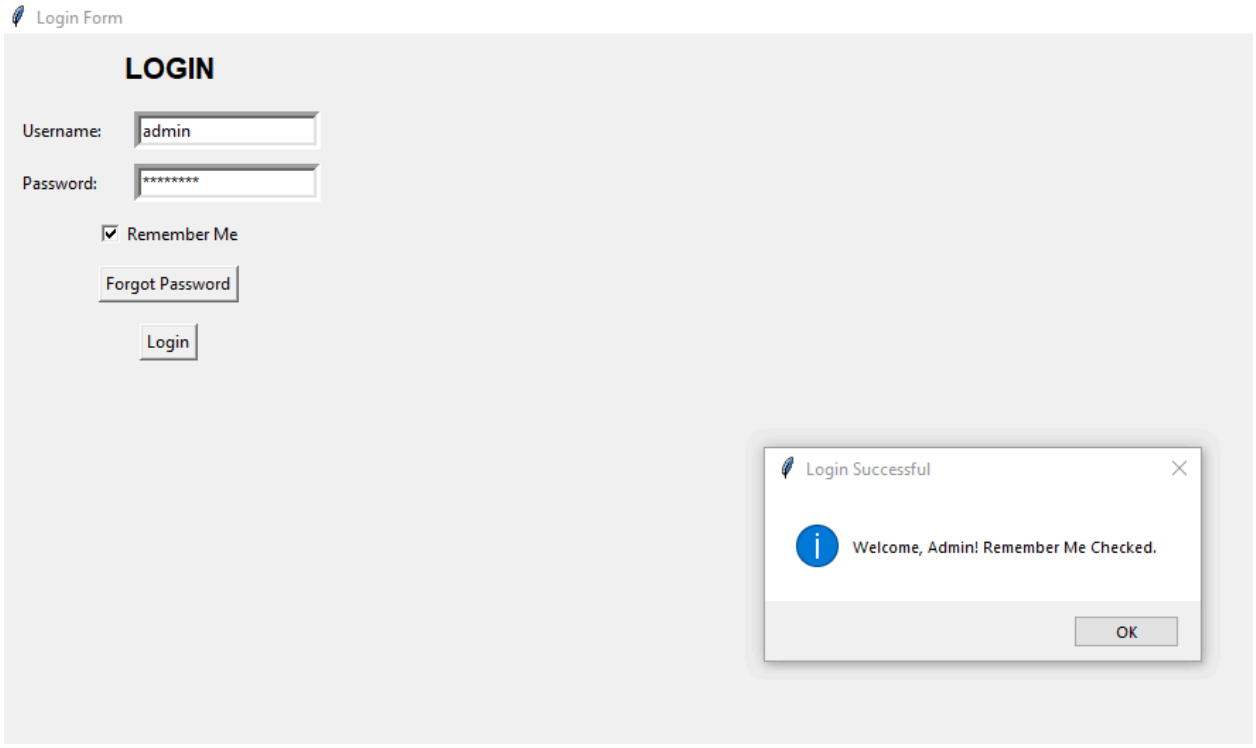
```
password_label = Label(top, text="Password:")

password_label.grid(row=2, column=0, sticky=W, padx=10, pady=5)


password_entry = Entry(top, bd=5, show="*")

password_entry.grid(row=2, column=1, padx=10, pady=5)


# Remember Me checkbox

remember_me_var = IntVar()

remember_me_checkbox = Checkbutton(top, text="Remember Me", variable=remember_me_var)

remember_me_checkbox.grid(row=3, columnspan=2, pady=5)


# Forgot Password button

forgot_password_button = Button(top, text="Forgot Password", command=forgot_password)

forgot_password_button.grid(row=4, columnspan=2, pady=5)


# Login button

login_button = Button(top, text="Login", command=validate_login)
```

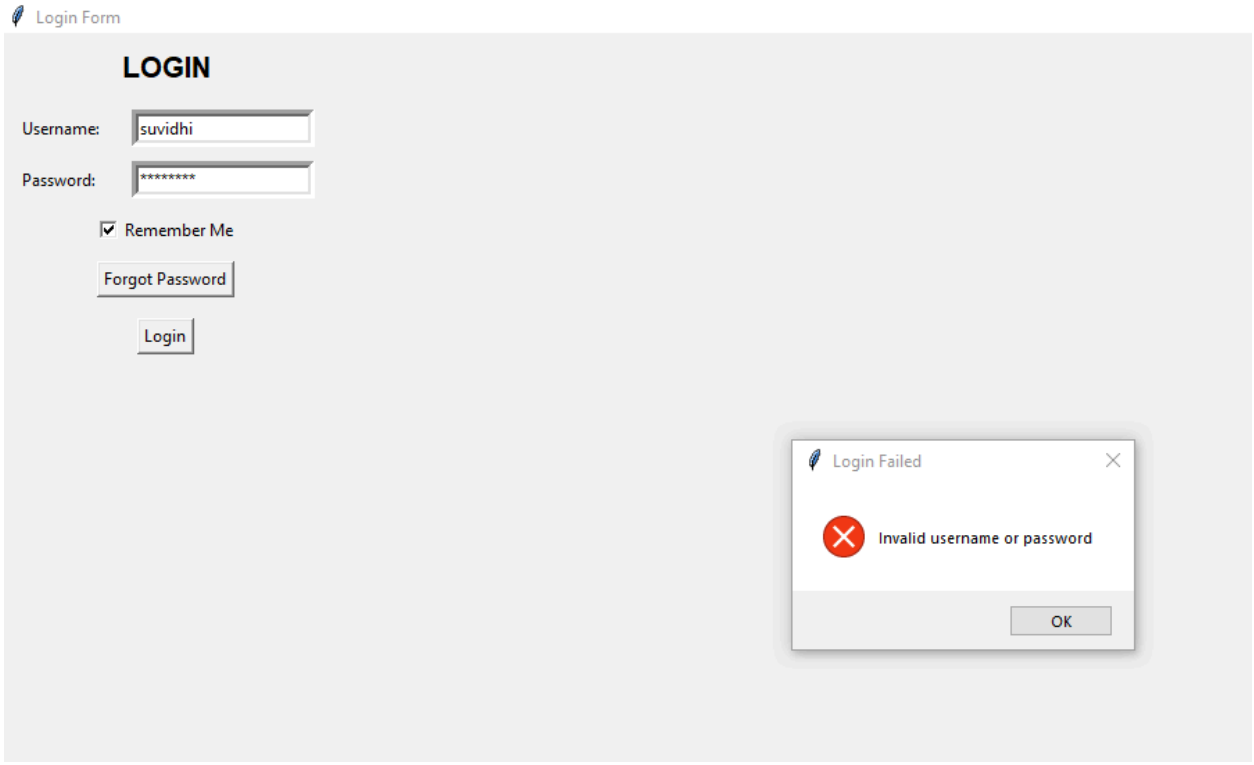
```
login_button.grid(row=5, columnspan=2, pady=10)


top.mainloop()
```

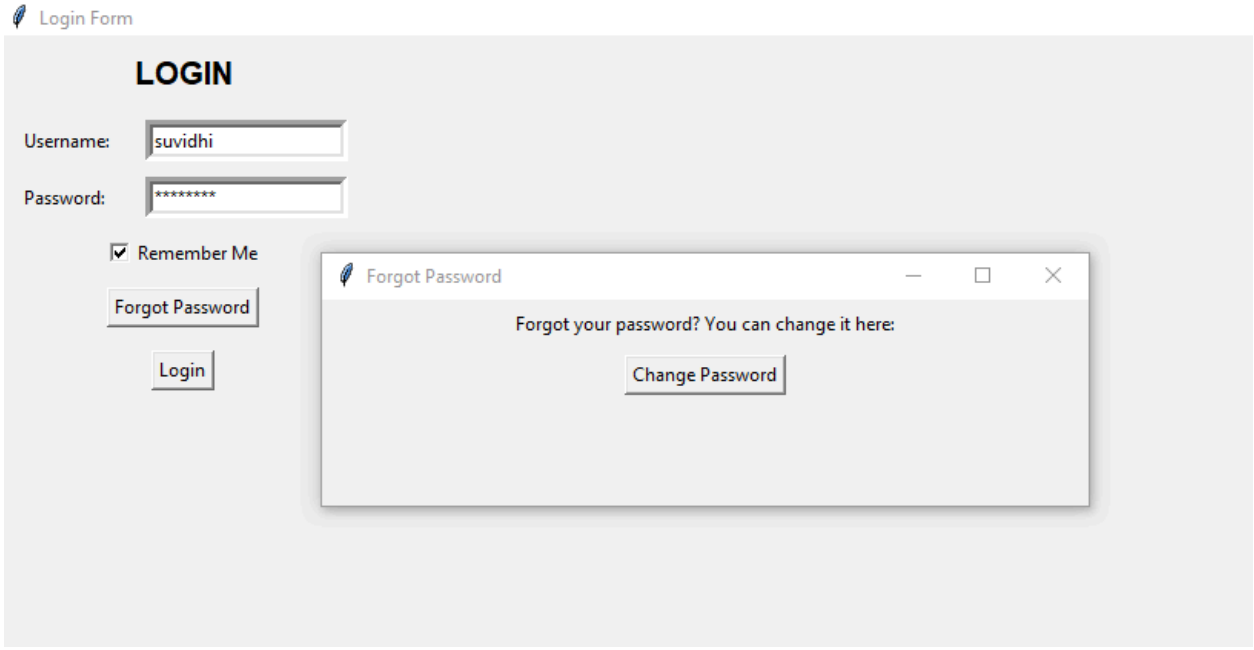
Output:

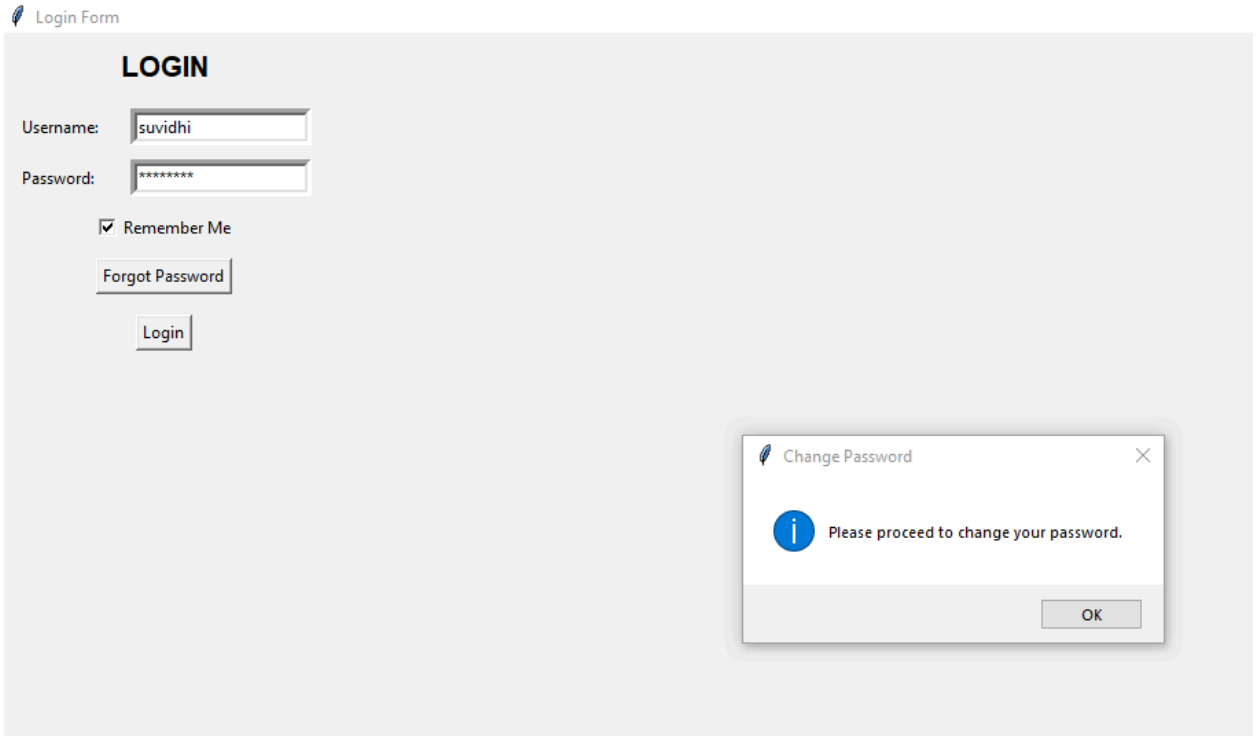



Login Form
LOGIN
Username:
admin
Password:
****
Remember Me
Forgot Password
Login
Login Successful
Welcome, Admin!
OK


Login Form
LOGIN
Username:
admin
Password:
****
Remember Me
Forgot Password
Login
Login Successful
Welcome, Admin! Remember Me Checked.
OK


Login Form
LOGIN
Username:
suvidhi
Password:
****
Remember Me
Forgot Password
Login
Login Failed
Invalid username or password
OK


Login Form
LOGIN
Username:
suvidhi
Password:
****
Remember Me
Forgot Password
Login
Forgot Password
Forgot your password? You can change it here:
Change Password



Conclusion:

GUI package TKinter has been studied and implemented.

In summary, harnessing Python's GUI capabilities with widgets like labels, text boxes, radio buttons, checkboxes, and custom dialog boxes enhances user interaction and empowers developers to create intuitive and functional applications for diverse purposes. Through libraries such as Tkinter, PyQt, or Kivy, Python offers a versatile platform for crafting rich graphical user interfaces tailored to specific application needs.