

- 1. Develop a simple login system with a username and password field. Implement user authentication, and show a success message if the login is successful, or an error message if the login fails.**

```
import tkinter as tk
from tkinter import messagebox

def login():
    if username.get() == "admin" and password.get() == "123":
        messagebox.showinfo("Login", "Login Successful!")
    else:
        messagebox.showerror("Login", "Login Failed!")

root = tk.Tk()
root.title("Login System")

tk.Label(root, text="Username").pack()
username = tk.Entry(root)
username.pack()

tk.Label(root, text="Password").pack()
password = tk.Entry(root, show='*')
password.pack()

tk.Button(root, text="Login", command=login).pack()
root.mainloop()
```

- 2. Build a currency converter application that converts between different currencies. The user should be able to enter an amount, select the input currency, select the output currency, and see the converted amount.**

```
import tkinter as tk
def convert():
    rate = {"USD": 82, "EUR": 89, "INR": 1}
    amount = float(entry.get())
    result = amount * rate[to_currency.get()] / rate[from_currency.get()]
    label_result.config(text=f"Converted Amount: {result:.2f}")

root = tk.Tk()
root.title("Currency Converter")
```

```

entry = tk.Entry(root)
entry.pack()

from_currency = tk.StringVar(value="USD")
to_currency = tk.StringVar(value="INR")

tk.OptionMenu(root, from_currency, "USD", "EUR", "INR").pack()
tk.OptionMenu(root, to_currency, "USD", "EUR", "INR").pack()

tk.Button(root, text="Convert", command=convert).pack()
label_result = tk.Label(root, text="")
label_result.pack()
root.mainloop()

```

3. Design a basic calculator to perform +,-,/,*

```

import tkinter as tk

def click(b):
    current = entry.get()
    entry.delete(0, tk.END)
    if b == "=":
        entry.insert(0, eval(current))
    elif b == "C":
        entry.delete(0, tk.END)
    else:
        entry.insert(0, current + b)

root = tk.Tk()
root.title("Calculator")

entry = tk.Entry(root, width=20, font=("Arial", 18))
entry.grid(row=0, column=0, columnspan=4)

buttons = ["7", "8", "9", "/",
            "4", "5", "6", "*",
            "1", "2", "3", "-",
            "C", "0", "=", "+"]

```

```

row, col = 1, 0
for b in buttons:
    tk.Button(root, text=b, width=5, height=2, command=lambda b=b:
click(b)).grid(row=row, column=col)
    col += 1
    if col > 3:
        col = 0
        row += 1
root.mainloop()

```

4. Quiz Game: Create an interactive quiz game with multiple-choice questions. Display questions one at a time and allow the user to select an answer. Provide feedback on whether the selected answer is correct or incorrect.

```

import tkinter as tk
from tkinter import messagebox

questions = [("Capital of India?", "Delhi", ["Mumbai", "Delhi", "Kolkata"]),
             ("2 + 2 = ?", "4", ["3", "4", "5"])]
q_index = 0

def check_answer(ans):
    global q_index
    if ans == questions[q_index][1]:
        messagebox.showinfo("Quiz", "Correct!")
    else:
        messagebox.showerror("Quiz", "Incorrect!")
    q_index += 1
    if q_index < len(questions):
        show_question()
    else:
        messagebox.showinfo("Quiz", "Quiz Finished")
        root.quit()

```

```
def show_question():
    q, _, options = questions[q_index]
    question_label.config(text=q)
    for i, opt in enumerate(options):
        buttons[i].config(text=opt, command=lambda opt=opt: check_answer(opt))

root = tk.Tk()
root.title("Quiz Game")

question_label = tk.Label(root, text="", font=("Arial", 16))
question_label.pack()

buttons = [tk.Button(root, text="", width=15) for _ in range(3)]
for b in buttons:
    b.pack()

show_question()
root.mainloop()
```