

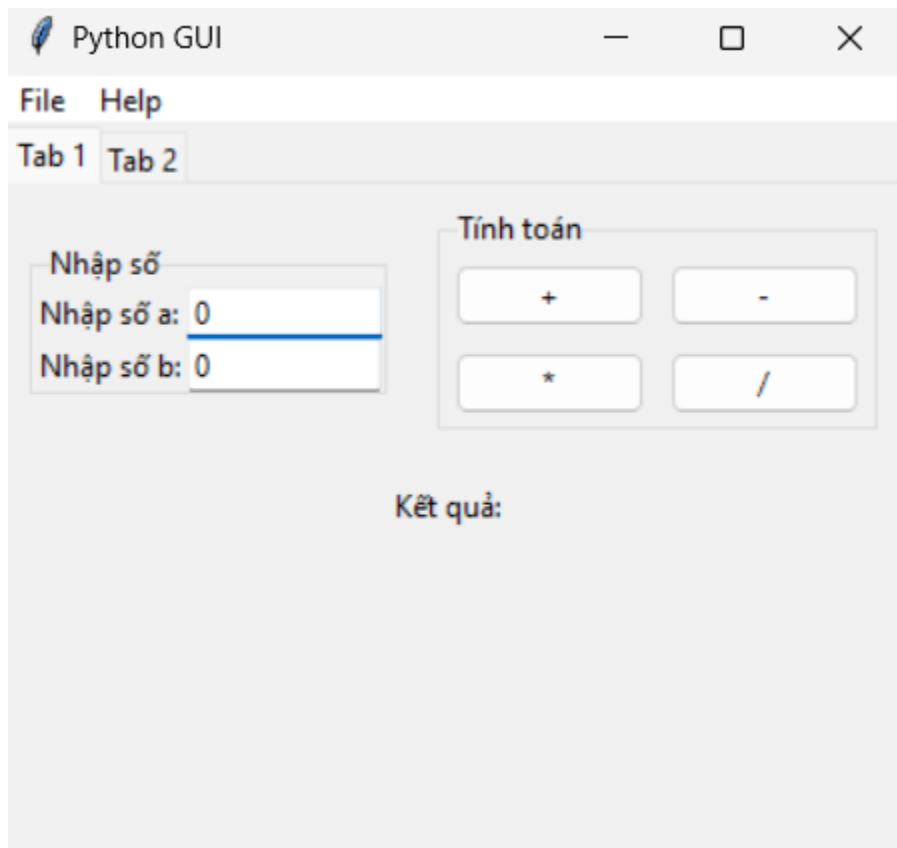
Bài Báo Cáo

TẠO GIAO DIỆN ĐƠN GIẢN

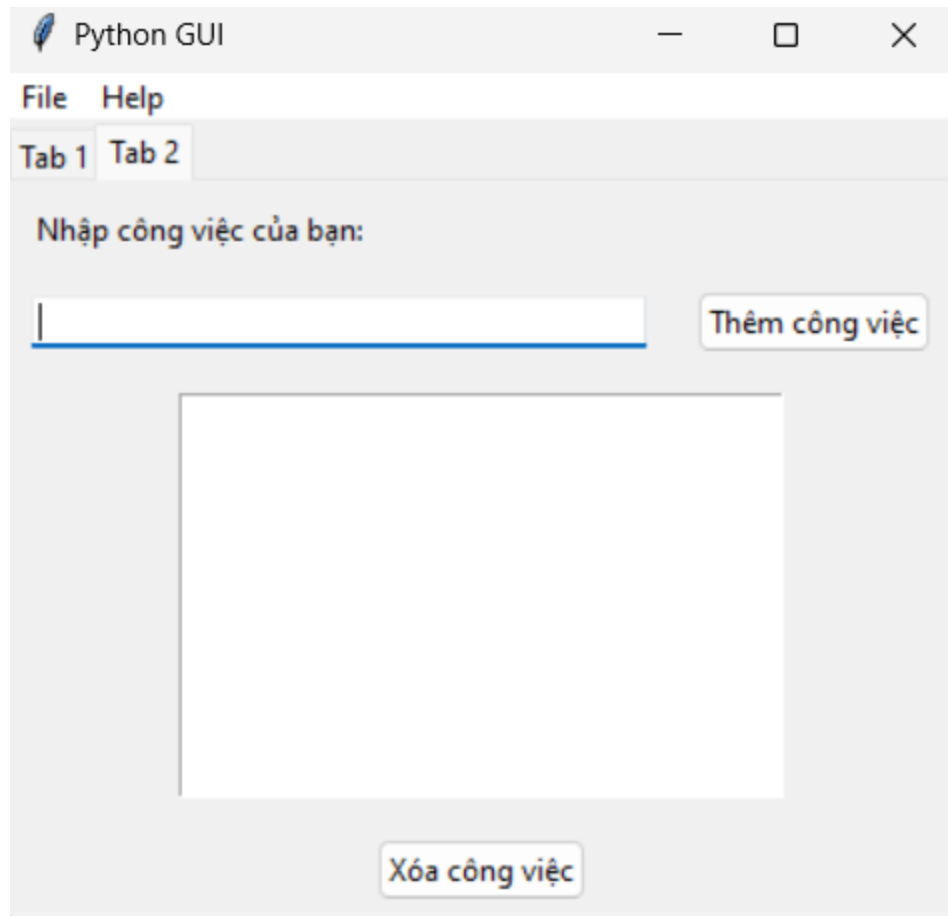
Họ và Tên: Châu Gia Kiệt - MSSV: 2274802010449

Phần 1. Giao Diện

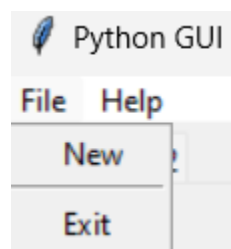
- Giao diện ở Tab 1



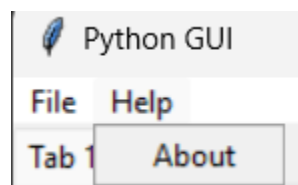
- Giao diện Tab 2



- Giao diện trong nút File



- Giao diện trong nút Help

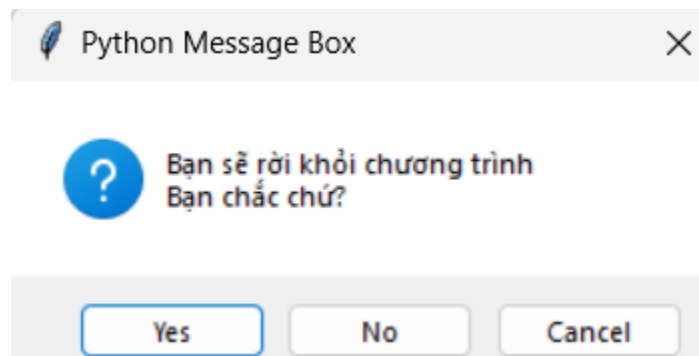


Phần 2. Chức Năng

- Chức năng trong Menu File:
 - Khi ấn New trong menu File sẽ tạo ra một tab khác



- Khi ấn Exit trong menu File sẽ xuất hiện hộp thoại “Bạn sẽ rời khỏi chương trình”. Yêu cầu nhấn “Yes/No/Cancel”



- Chức năng trong Tab 1: Khi đã nhập số vào frame Nhập số bên trái, sau đó chọn phép tính mong muốn. Khi nhấn vào phép tính thì



- Chức năng trong Tab 2:
 - Giao diện trong tab này là ghi lại những việc cần làm, chỉ cần nhập việc muốn làm vào ô và nhấn “Thêm công việc” thì những gì đã nhập sẽ xuất hiện ở dưới


Nhập công việc của bạn:

Ăn trưa |

- Ngoài ra có thể xóa những gì chúng ta đã nhập vào, cần nhấn chọn những gì sẽ xóa. Nếu chưa nhấn chọn hộp thoại sẽ cảnh báo

Nhập công việc của bạn:

Ăn trưa

 Cảnh báo



Vui lòng chọn công việc để xóa

Phần 3. Mã Chương Trình

Link GitHub:

https://github.com/dustin314/Python_Programming/tree/main/Projects/BaiTap1

```

1 # Thon các thư viện cần thiết
2 import tkinter as tk
3 from tkinter import ttk
4 from tkinter import Menu
5 from tkinter import messagebox as msg # Hộp thoại cảnh báo
6
7 class OOP:
8     def __init__(self):
9         # Tạo cửa sổ
10         self.win = tk.Tk()
11         # Title
12         self.win.title("Python GUI")
13
14         # Tạo menu bar
15         self.create_menu_bar()
16         # Tạo các Tab
17         self.create_widgets()
18
19     def create_menu_bar(self):
20         # Tạo menu bar
21         menu_bar = Menu(self.win)
22         self.win.config(menu=menu_bar)
23
24         # File menu
25         file_menu = Menu(menu_bar, tearoff=0)
26         menu_bar.add_cascade(label="File", menu=file_menu) # Tạo ra menu thả xuống
27         # Các hàm con
28         file_menu.add_command(label="New", command=self.create_new_tab) # Tạo thêm tab mới
29         file_menu.add_separator() # Đặt thêm một icon
30         file_menu.add_command(label="Exit", command=self.msgExit) # Thoát khỏi chương trình
31
32         # Help menu
33         help_menu = Menu(menu_bar, tearoff=0)
34         menu_bar.add_cascade(label="Help", menu=help_menu)
35         # Các hàm con
36         help_menu.add_command(label="About")
37
38     # Hộp thoại cảnh báo cho nút exit
39     def msgExit(self):
40         response = msg.askyesnocancel("Python Message Box", "Bấm sẽ rời khỏi chương trình. Bạn chắc chứ?")
41         if response == msg.NO:
42             self.win.quit() # Rời khỏi chương trình
43
44     def create_widgets(self):
45         # Tạo TabControl
46         self.tabControl = ttk.Notebook(self.win)
47
48         # Tab 1
49         tab1 = tk.Frame(self.tabControl)
50         self.tabControl.add(tab1, text="Tab 1")
51
52         # Tab 2 - Tab có List
53         tab2 = tk.Frame(self.tabControl)
54         self.tabControl.add(tab2, text="Tab 2")
55
56         self.tabControl.pack(expand=1, fill="both")
57
58         # Tạo frame nhập số trong Tab 1
59         nhapso_frame = tk.LabelFrame(tab1, text="Nhập số")
60         nhapso_frame.grid(column=0, row=0, padx=10, pady=10)
61
62         tk.Label(nhapso_frame, text="Nhập số a:").grid(column=0, row=0, sticky=tk.W)
63         tk.Label(nhapso_frame, text="Nhập số b:").grid(column=0, row=1, sticky=tk.W)
64
65         # Các biến Integer để lưu trữ giá trị của a và b
66         self.a = tk.IntVar()
67         self.b = tk.IntVar()
68
69         # Entry nhập số a
70         a_entry = tk.Entry(nhapso_frame, width=12, textvariable=self.a)
71         a_entry.grid(column=1, row=0)
72         a_entry.focus()
73
74         # Entry nhập số b
75         b_entry = tk.Entry(nhapso_frame, width=12, textvariable=self.b)
76         b_entry.grid(column=1, row=1)
77
78         # Frame cho các nút tính toán
79         tinhtoan_frame = tk.LabelFrame(tab1, text="Tính toán")
80         tinhtoan_frame.grid(column=1, row=0, padx=10, pady=10)
81
82         # Các nút tính toán
83         tk.Button(tinhtoan_frame, text="+", command=self.cong).grid(column=0, row=0, padx=5, pady=5)
84         tk.Button(tinhtoan_frame, text="-", command=self.tru).grid(column=1, row=0, padx=5, pady=5)
85         tk.Button(tinhtoan_frame, text="*", command=self.nhan).grid(column=2, row=0, padx=5, pady=5)
86         tk.Button(tinhtoan_frame, text="/", command=self.chia).grid(column=3, row=0, padx=5, pady=5)
87
88         # Label để hiển thị kết quả
89         self.ketqua_label = tk.Label(tab1, text="Kết quả: ")
90         self.ketqua_label.grid(column=0, row=1, columnspan=2, padx=10, pady=10)
91
92         # Tab có List cho Tab 2
93         tk.Label(tab2, text="Nhập công việc của bạn:").grid(column=0, row=0, padx=10, pady=10, sticky=tk.W)
94
95         # Entry để nhập công việc
96         self.task_entry = tk.Entry(tab2, width=40)
97         self.task_entry.grid(column=0, row=1, padx=10, pady=5, sticky=tk.W)
98
99         # Nút thêm công việc
100         tk.Button(tab2, text="Thêm công việc", command=self.add_task).grid(column=1, row=1, padx=10, pady=5)
101
102         # Listbox để hiển thị công việc
103         self.tasks_listbox = tk.Listbox(tab2, width=40, height=10)
104         self.tasks_listbox.grid(column=0, row=2, columnspan=2, padx=10, pady=10)
105
106         # Nút xóa công việc
107         tk.Button(tab2, text="Xóa công việc", command=self.remove_task).grid(column=1, row=3, columnspan=2, padx=10, pady=5)
108
109         # Hàm xử lý các phép tính
110         def cong(self):
111             result = self.a.get() + self.b.get()
112             self.ketqua_label.config(text="Kết quả: (result)")
113
114         def tru(self):
115             result = self.a.get() - self.b.get()
116             self.ketqua_label.config(text="Kết quả: (result)")
117
118         def nhan(self):
119             result = self.a.get() * self.b.get()
120             self.ketqua_label.config(text="Kết quả: (result)")
121
122         def chia(self):
123             if self.b.get() != 0:
124                 result = self.a.get() / self.b.get()
125                 self.ketqua_label.config(text="Kết quả: (result)")
126             else:
127                 self.ketqua_label.config(text="Lỗi: không thể chia cho 0")
128
129         # Các hàm cho Tab có List
130         def add_task(self):
131             task = self.task_entry.get()
132             if task:
133                 self.tasks_listbox.insert(tk.END, task)
134                 self.task_entry.delete(0, tk.END)
135             else:
136                 msg.showwarning("Cảnh báo", "Vui lòng nhập công việc")
137
138         def remove_task(self):
139             try:
140                 index = self.tasks_listbox.curselection()[0]
141                 self.tasks_listbox.delete(index)
142             except IndexError:
143                 msg.showwarning("Cảnh báo", "Vui lòng chọn công việc để xóa")
144
145         # Tạo một tab mới không có nội dung
146         def create_new_tab(self):
147             new_tab = tk.Frame(self.tabControl)
148             tab_name = f"Tab (màn hình.tabControl.tabs() + 1)"
149             self.tabControl.add(new_tab, text=tab_name)
150
151     # Chạy chương trình
152 if __name__ == "__main__":
153     app = OOP()
154     app.win.mainloop()

```