

TRƯỜNG ĐẠI HỌC VĂN LANG  
KHOA CÔNG NGHỆ THÔNG TIN



**ĐỒ ÁN CUỐI KỲ**  
**MÔN HỌC LẬP TRÌNH PYTHON NÂNG CAO**

**Bài tập 01:**

**Calculations Basic**

**SVTH: Trần Ngọc Phước – 2274802010699**

**LỚP: 241\_71ITSE31003\_02**

**GVHD: Huỳnh Thái Học**

**TP. Hồ Chí Minh – 10/2024**

# MỤC LỤC

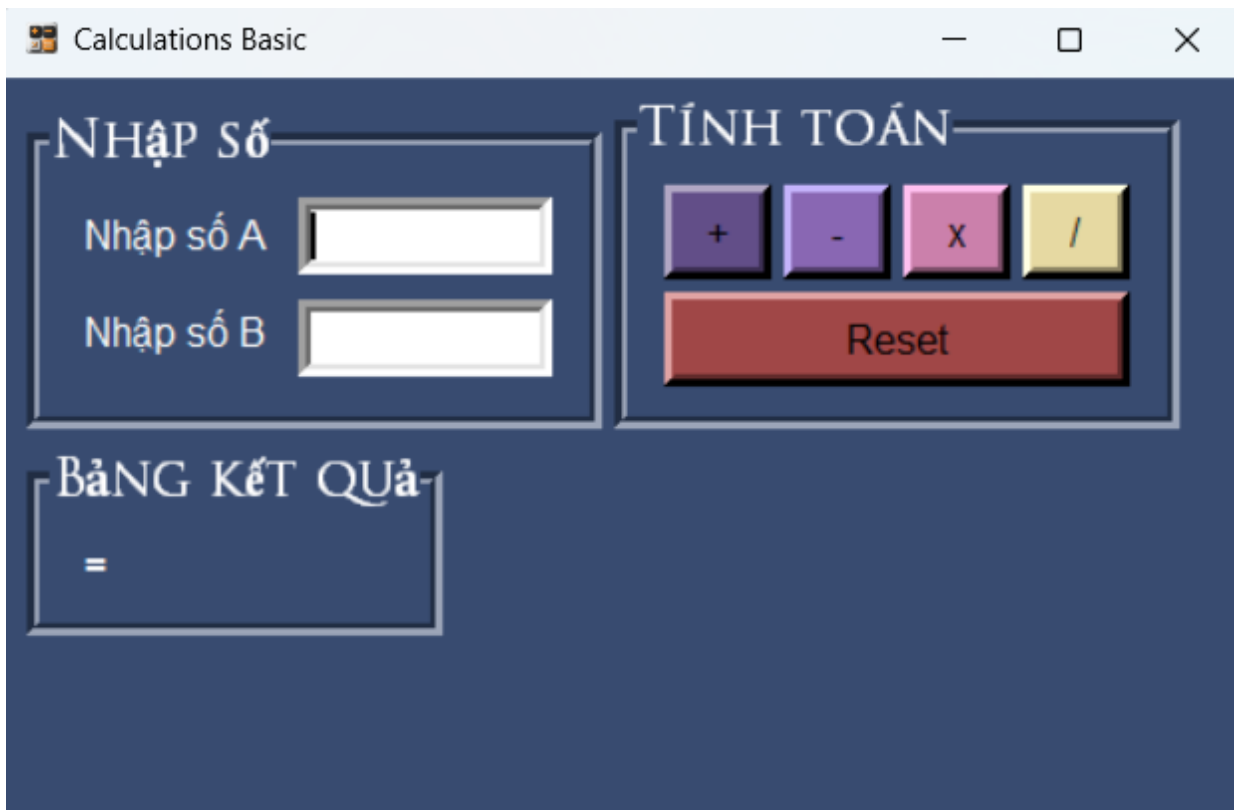
1. Giới thiệu về Tkinter.....	3
2. Giao diện ứng dụng Calculations Basic. ....	4
3. Chức năng.....	5
3.1 Nhập liệu:.....	5
3.2 Thực hiện các phép toán cơ bản: .....	5
3.3 Hiển thị kết quả:.....	5
3.4 Nút Reset: .....	5
4. Mã chương trình. ....	6
5. Đường link github.....	8
--- HẾT--- .....	8

# 1. Giới thiệu về Tkinter.

Tkinter là một thư viện giao diện đồ họa (GUI) mặc định của Python. Nó được tích hợp sẵn trong Python, vì vậy không cần cài đặt thêm, ưu điểm là dễ dàng sử dụng cho cả những người mới bắt đầu. Tkinter cho phép bạn xây dựng các cửa sổ giao diện với nhiều thành phần như nút bấm (Button), hộp văn bản (Entry), nhãn (Label), hộp thông báo (Messagebox), và nhiều hơn thế nữa. Bạn có thể quản lý vị trí các thành phần thông qua các phương pháp như `grid()`, `pack()`, và `place()`, giúp dễ dàng sắp xếp và điều chỉnh bố cục giao diện theo ý mình.

Tkinter là sự lựa chọn phổ biến khi cần tạo ra các ứng dụng nhỏ nhanh chóng mà không yêu cầu các thư viện GUI phức tạp hơn. Mặc dù không mạnh mẽ như các framework hiện đại khác, nhưng Tkinter có ưu điểm là đơn giản, nhẹ nhàng và tích hợp sẵn trong mọi phiên bản Python. Điều này làm cho nó trở thành một công cụ lý tưởng để tạo ra các ứng dụng desktop nhỏ hoặc các dự án thử nghiệm GUI.

## 2. Giao diện ứng dụng Calculations Basic.



Calculations Basic

**NHẬP SỐ**

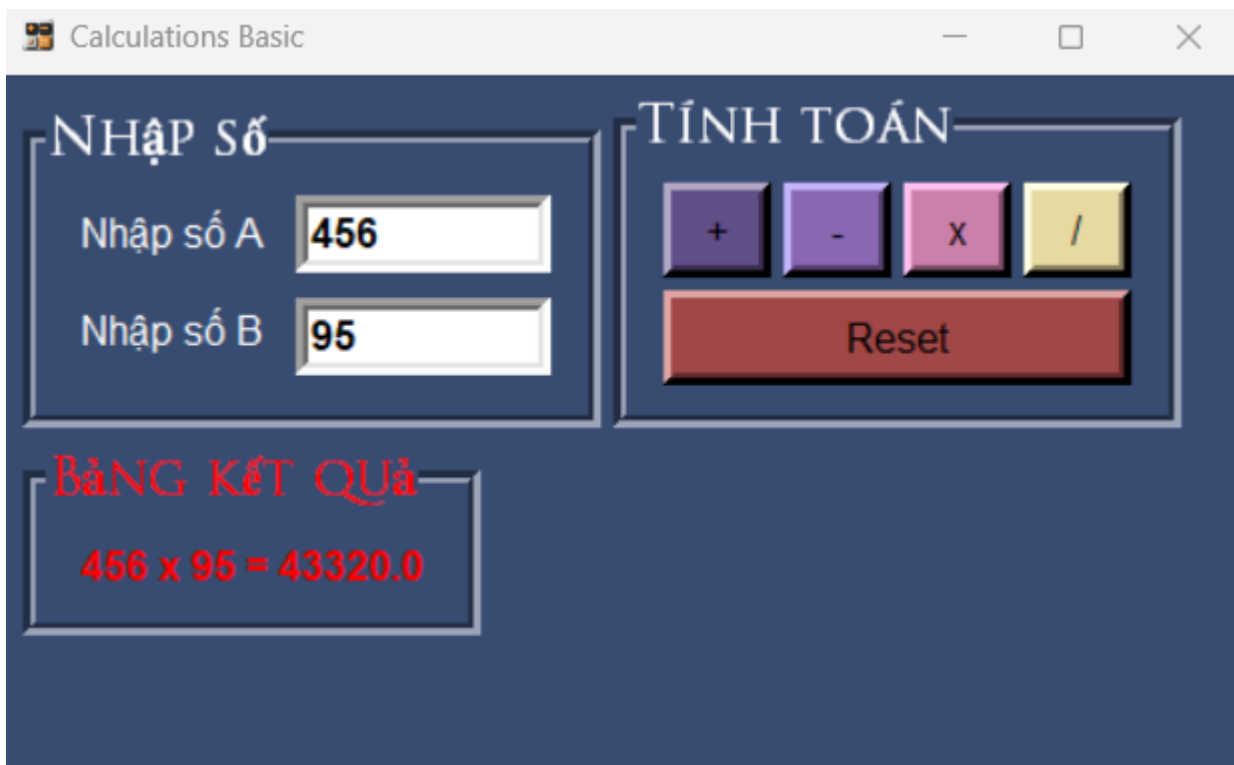
Nhập số A

Nhập số B

**TÍNH TOÁN**

**BẢNG KẾT QUẢ**

=



Calculations Basic

**NHẬP SỐ**

Nhập số A

Nhập số B

**TÍNH TOÁN**

**BẢNG KẾT QUẢ**

**456 x 95 = 43320.0**

### 3. Chức năng.

Ứng dụng Calculations Basic được xây dựng bằng Python và thư viện Tkinter có các chức năng chính sau:

#### 3.1 Nhập liệu:

- Người dùng có thể nhập hai số A và B vào hai ô nhập liệu tương ứng.
- Phần này giúp người dùng cung cấp dữ liệu đầu vào để thực hiện các phép tính.

#### 3.2 Thực hiện các phép toán cơ bản:

- **Cộng (+):** Nút này cho phép thực hiện phép cộng hai số A và B đã nhập. Khi người dùng nhấn vào, kết quả của phép cộng sẽ hiển thị ở phần kết quả.
- **Trừ (-):** Thực hiện phép trừ giữa số A và số B, với kết quả cũng hiển thị ngay lập tức.
- **Nhân (x):** Tính tích của hai số A và B.
- **Chia (/):** Tính thương của số A chia cho số B. Nếu người dùng nhập số B bằng 0, thì ứng dụng hiển thị thông báo lỗi, "Không thể chia cho 0"

#### 3.3 Hiển thị kết quả:

- Sau khi người dùng thực hiện bất kỳ phép tính nào, kết quả sẽ hiển thị ngay bên dưới ở phần "Bảng kết quả". Điều này giúp người dùng dễ dàng theo dõi và kiểm tra kết quả của phép toán vừa thực hiện.

#### 3.4 Nút Reset:

- Chức năng này xóa toàn bộ dữ liệu đã nhập và kết quả đang hiển thị, giúp người dùng làm mới ứng dụng và bắt đầu tính toán lại từ đầu.

## 4. Mã chương trình.

### Về giao diện:

- Toàn bộ giao diện được sắp xếp gọn gàng, sử dụng các phương pháp quản lý bố cục grid() và LabelFrame để tạo các khối nội dung phân tách rõ ràng, gồm ba phần chính: Nhập số, Tính toán, và Bảng kết quả.

```
# create LabelFrame 1 nhập số
label_Frame1 = tk.LabelFrame(win, text='Nhập số', font=font_title, bg=bg_Color, fg=color_text, bd=5, padx=15, pady=10)
label_Frame1.grid(row=0, column=0, padx=(10,5), pady=(10,5) )
labelA = tk.Label(label_Frame1, text='Nhập số A', font=font_text, bg=bg_Color, fg=color_text)
labelA.grid(row=0, column=0, padx=(0,10))
entryA = tk.Entry(label_Frame1, font=font_bold, width=10, bd=5 )
entryA.grid (row=0, column=1)
entryA.focus()
labelB = tk.Label(label_Frame1, text='Nhập số B', font=font_text, bg=bg_Color, fg=color_text)
labelB.grid(row=1, column=0, padx=(0,10))
entryB = tk.Entry(label_Frame1, font=font_bold, width=10, bd=5 )
entryB.grid (row=1, column=1, pady=(10,6))
```

```
# create LabelFrame 2 tính toán
label_Frame2 = tk.LabelFrame(win, text='Tính toán', font=font_title, bg=bg_Color, fg=color_text, bd=5, padx=15, pady=10)
label_Frame2.grid(row=0, column=1)
btn_cong = tk.Button(label_Frame2, text='+', font=font_text, bd=5, width=3, bg=color_btn_cong, command=lambda:cong())
btn_cong.grid(row=0, column=0)
btn_tru = tk.Button(label_Frame2, text='-', font=font_text, bd=5, width=3, bg=color_btn_tru, command=lambda:tru())
btn_tru.grid(row=0, column=1, padx=5)
btn_nhan = tk.Button(label_Frame2, text='x', font=font_text, bd=5, width=3, bg=color_btn_nhan, command=lambda:nhan())
btn_nhan.grid(row=0, column=2, padx=(0,5))
btn_chia = tk.Button(label_Frame2, text='/', font=font_text, bd=5, width=3, bg=color_btn_chia, command=lambda:chia())
btn_chia.grid(row=0, column=3)
btn_reset = tk.Button(label_Frame2, text='Reset', font=font_text, bd=5, width=19, bg=color_btn_reset, command=lambda:reset())
btn_reset.grid(row=1, column=0, columnspan=4, pady=(5,2))
```

```
# create LabelFrame 3 kết quả
label_Frame3 = tk.LabelFrame(win, text='Bảng kết quả', font=font_title, bg=bg_Color, fg=color_text, bd=5, padx=15, pady=10)
label_Frame3.grid(row=1, column=0, padx=10, sticky=tk.W, columnspan=2)
label_kq = tk.Label(label_Frame3, text='', font=font_bold, bg=bg_Color, fg=color_text )
label_kq.grid(row=0, column=0)
```

### Về các chức năng chính:

- Phép toán:** Bốn nút ( + , - , x , / ) thực hiện các phép tính cộng, trừ, nhân, chia tương ứng.

```
# create function cong
def cong():
    try:
        kq = float(entryA.get()) + float(entryB.get())
        label_kq.config(text= entryA.get() + ' + ' + entryB.get() + ' = ' + str(kq), fg=color_text_kq)
        label_Frame3.config(fg=color_text_kq)
    except:
        messagebox.showerror("Lỗi", "Vui lòng nhập số hợp lệ!")
        entryA.delete(0, tk.END)
        entryB.delete(0, tk.END)
```

```
# create function tru
def tru():
    try:
        kq = float(entryA.get()) - float(entryB.get())
        label_kq.config(text=entryA.get() + ' - ' + entryB.get() + ' = ' + str(kq), fg=color_text_kq)
        label_Frame3.config(fg=color_text_kq)
    except:
        messagebox.showerror("Lỗi", "Vui lòng nhập số hợp lệ!")
        entryA.delete(0, tk.END)
        entryB.delete(0, tk.END)
```

```
# create function nhan
def nhan():
    try:
        kq = float(entryA.get()) * float(entryB.get())
        label_kq.config(text=entryA.get() + ' x ' + entryB.get() + ' = ' + str(kq), fg=color_text_kq)
        label_Frame3.config(fg=color_text_kq)
    except:
        messagebox.showerror("Lỗi", "Vui lòng nhập số hợp lệ!")
        entryA.delete(0, tk.END)
        entryB.delete(0, tk.END)
```

```
# create function chia
def chia():
    try:
        b = float(entryB.get())
        if b==0:
            messagebox.showerror('Lỗi', "Không thể chia cho 0")
        else:
            kq = float(entryA.get()) / b
            label_kq.config(text=entryA.get() + ' / ' + entryB.get() + ' = ' + str(kq), fg=color_text_kq)
            label_Frame3.config(fg=color_text_kq)
    except:
        messagebox.showerror("Lỗi", "Vui lòng nhập số hợp lệ!")
        entryA.delete(0, tk.END)
        entryB.delete(0, tk.END)
```

- **Nút Reset:** Nút này sẽ xóa nội dung trong ô nhập liệu và đặt lại phần kết quả về mặc định.

```
# create function reset
def reset():
    entryA.delete(0, tk.END)
    entryB.delete(0, tk.END)
    label_kq.config(text='=', fg=color_text)
    label_Frame3.config(fg=color_text)
```

## 5.Đường link github.

<https://github.com/phuocsasc/PythonNC/tree/master/Baitap01>

--- HẾT---