

Trong bài học này, chúng ta sẽ lập trình sao cho khi nhấn nút các số thì số đó được hiển thị lên hộp văn bản.

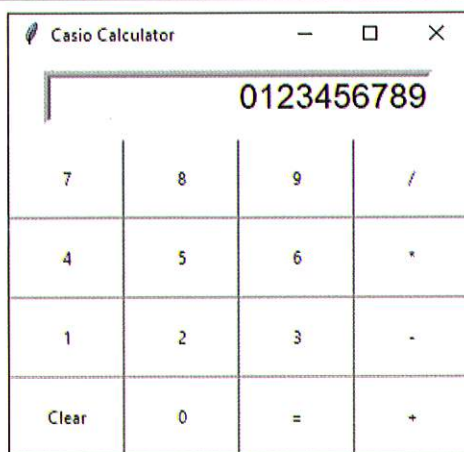
```
def bt_click(number):
    e.insert(END, number)

e = Entry(width=20, borderwidth=5, font="Arial 18",
          justify=RIGHT)

bt_7 = Button(text="7", width=10, height=3,
              command=lambda: bt_click(7))
bt_8 = Button(text="8", width=10, height=3,
              command=lambda: bt_click(8))
bt_9 = Button(text="9", width=10, height=3,
              command=lambda: bt_click(9))

bt_4 = Button(text="4", width=10, height=3,
              command=lambda: bt_click(4))
bt_5 = Button(text="5", width=10, height=3,
              command=lambda: bt_click(5))
bt_6 = Button(text="6", width=10, height=3,
              command=lambda: bt_click(6))

bt_1 = Button(text="1", width=10, height=3,
              command=lambda: bt_click(1))
bt_2 = Button(text="2", width=10, height=3,
              command=lambda: bt_click(2))
bt_3 = Button(text="3", width=10, height=3,
              command=lambda: bt_click(3))
bt_0 = Button(text="0", width=10, height=3,
              command=lambda: bt_click(0))
```



*Kết quả của bài học*



### 3.1. Lập trình hiển thị số trên hộp văn bản khi nhấn các nút chữ số

Khi nhấn các nút số từ 0 đến 9, chữ số đó sẽ được thêm vào hộp văn bản (trong trường hợp không nhấn các nút phép toán). Đầu tiên, chúng ta sẽ lập trình tạo một hàm giúp thêm một ký tự vào hộp văn bản (ví dụ hàm có tên **bt\_click()**).

Hàm là một tập hợp các lệnh, được sử dụng để thực hiện một nhiệm vụ nhất định. Một hàm bao gồm các thành phần: Tên hàm, tham số và thân hàm. Chúng ta đã sử dụng nhiều hàm có sẵn trong python (ví dụ **print()**, **input()**) hay hàm có sẵn trong các module **pygame** (ví dụ **set\_mode()** và **blit()**) và module **tkinter** (ví dụ **grid()**). Để tự tạo một hàm, chúng ta sử dụng từ khóa **def** và định nghĩa hàm theo cú pháp:

```
def <tên hàm>([danh sách tham số]):
    <thân hàm>
```

Trong đó, tên hàm có quy tắc đặt tên giống tên biến, các tham số cách nhau bởi dấu phẩy và không bắt buộc. Dưới đây là định nghĩa hàm **bt\_click()** thực hiện thêm ký tự "0" vào hộp văn bản ở vị trí đầu tiên.

```
def bt_click():
    e.insert(0, "0")
```

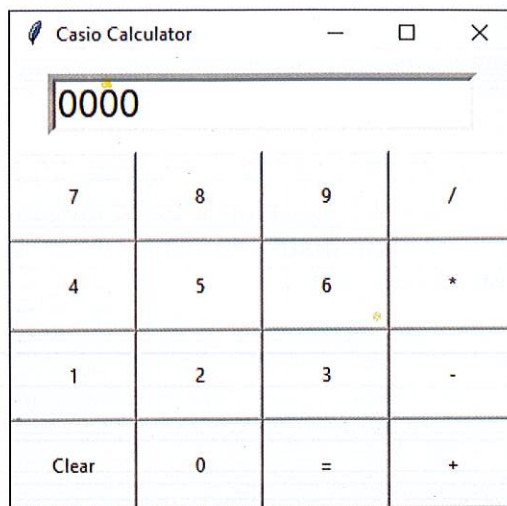
Phương thức **insert()** của lớp Entry **insert(i, s)** có 2 tham số vị trí **i** và chuỗi **s**, thực hiện việc chèn chuỗi **s** vào sau ký tự thứ **i** trong hộp văn bản. Với **i = 0**, chuỗi **s** được chèn lên vị trí đầu tiên; với **i = 1**, chuỗi **s** được chèn sau ký tự thứ 1 và cứ tương tự như vậy với các giá trị **i** khác.

Các bạn lập trình khi nút 0 được nhấn thì thực hiện hàm **bt\_click()** trên bằng cách bổ sung tham số **command** khi khởi tạo nút. Tham số này có giá trị là tên hàm thực hiện khi nút được nhấn. Các bạn lưu ý, lệnh định nghĩa hàm cần được đặt lên trước khi sử dụng. Hình dưới đây mô tả kết quả chạy chương trình khi ta nhấn chuột 4 lần vào nút "0".

```
1 from tkinter import *
2 wd = Tk()
3 wd.title("Casio Calculator")

4 def bt_click():
5     e.insert(0, "0")
...
20 bt_0 = Button(text="0", width=10, height=3,
                command=bt_click)
```





Mỗi khi nhấn nút “0”, một ký tự “0” sẽ được thêm vào hộp văn bản. Chúng ta có thể làm tương tự với các nút khác từ 1 đến 9, nhưng như vậy thì cần tạo thêm 9 hàm khác. Thay vì vậy, chúng ta sẽ sử dụng một tham số (ví dụ: **number**) để truyền vào **bt\_click()** một giá trị nhất định, hàm sẽ sử dụng giá trị này để thêm vào hộp văn bản. Như vậy, gọi hàm **bt\_click(0)** sẽ thêm số 0, gọi hàm **bt\_click(1)** sẽ thêm số 1 vào hộp văn bản, tương tự với các giá trị khác. Các bạn lập trình để nút “0” sẽ gọi hàm **bt\_click(0)**, nút “1” sẽ gọi hàm **bt\_click(1)**, các nút còn lại sử dụng tham số tương ứng như vậy.

```
4 def bt_click(number):
5     e.insert(0, number)
...
20 bt_0 = Button(text="0", width=10, height=3,
                command=bt_click(0))
```

Các bạn chạy thử sẽ thấy nút không hoạt động được nữa, lý do vì khi **command** nhận giá trị là những hàm có tham số, chúng ta cần thêm một từ khóa đặc biệt: **lambda**. Kiến thức về **lambda** khá nâng cao và không được giải thích kỹ trong sách, các bạn chỉ cần hiểu ta cần từ khóa **lambda** trong trường hợp này và có thể tự mình tìm hiểu thêm.

```
20 bt_0 = Button(text="0", width=10, height=3,
                command=lambda: bt_click(0))
```

Các bạn bổ sung việc lập trình nhấn nút này với một nút số bất kỳ khác sẽ thấy số mới luôn được chèn vào vị trí đầu tiên. Để số chèn xuống vị trí cuối cùng, tham số vị trí trong hàm **insert()** cần có giá trị là vị trí cuối của chuỗi đang có trong hộp văn bản chứ không phải giá trị 0 như hiện tại. Thư viện **Tkinter** cung cấp sẵn hằng số **END**, có giá trị là vị trí cuối trong chuỗi, chúng ta sẽ truyền giá trị này làm tham số vị trí cho hàm **insert()**.



```
4 def bt_click(number):
5     e.insert(END, number)
```

Lúc này, khi chúng ta nhấn nút, ký tự sẽ được thêm vào cuối xâu. Các bạn lập trình cho toàn bộ các nút chữ số khác tương tự như nút "0". Khi chạy thử chương trình, các bạn nhấn và kiểm tra toàn bộ các nút chữ số để đảm bảo không bị sót. Ngoài ra, chúng ta có thể thiết lập thuộc tính **justify** của Entry **e** giá trị **RIGHT** để chữ trong hộp văn bản được căn chỉnh về bên phải.

```
4 def bt_click(number):
5     e.insert(END, number)
6 e = Entry(width=20, borderwidth=5, font="Arial 18",
7 justify=RIGHT)
8 bt_7 = Button(text="7", width=10, height=3,
9 command=lambda: bt_click(7))
10 bt_8 = Button(text="8", width=10, height=3,
11 command=lambda: bt_click(8))
12 bt_9 = Button(text="9", width=10, height=3,
13 command=lambda: bt_click(9))
...
14 bt_4 = Button(text="4", width=10, height=3,
15 command=lambda: bt_click(4))
16 bt_5 = Button(text="5", width=10, height=3,
17 command=lambda: bt_click(5))
18 bt_6 = Button(text="6", width=10, height=3,
19 command=lambda: bt_click(6))
...
20 bt_1 = Button(text="1", width=10, height=3,
21 command=lambda: bt_click(1))
22 bt_2 = Button(text="2", width=10, height=3,
23 command=lambda: bt_click(2))
24 bt_3 = Button(text="3", width=10, height=3,
25 command=lambda: bt_click(3))
...
26 bt_0 = Button(text="0", width=10, height=3,
27 command=lambda: bt_click(0))
```

Các bạn chạy chương trình và nhấn lần lượt các nút từ 0 đến 9 sẽ được kết quả như hình mô tả kết quả bài học.

### 3.2. Lập trình chức năng xóa hộp văn bản khi nhấn nút “Clear”

Tiếp theo, chúng ta sẽ lập trình cho nút “Clear” thực hiện việc xóa hộp văn bản khi được nhấn. Để xóa hộp văn bản, chúng ta sử dụng phương thức **delete()** của đối tượng Entry **e**: **e.delete(<vị trí đầu>, <vị trí cuối>)**, hàm này thực hiện việc xóa đoạn ký tự từ vị trí đầu đến vị trí cuối của văn bản trong hộp văn bản. Nếu muốn xóa toàn bộ, vị trí đầu cần nhận giá trị 0 và vị trí cuối nhận giá trị END. Các bạn có thể tách công việc này riêng thành một hàm (giống hàm **bt\_click()**) hoặc sử dụng trực tiếp như sau:

```
19 bt_clear = Button(text="Clear", width=10, height=3,
                    command=lambda: e.delete(0, END))
```



#### Tóm tắt lý thuyết và bài tập thực hành

Trong bài học này chúng ta đã nắm được cách thêm một xâu hoặc một ký tự bất kỳ vào hộp văn bản bằng phương thức **insert()** của Entry, cách thêm tham số **command** cho nút bấm, qua đó lập trình được công việc thêm chữ số vào hộp văn bản khi nút được nhấn. Ngoài ra các bạn cũng đã biết cách xóa một hoặc nhiều ký tự trên hộp văn bản bằng phương thức **delete()** của Entry.

**Bài tập 1.** Thay đổi thuộc tính **justify** của Entry **e** thành các giá trị LEFT, CENTER và quan sát kết quả.

**Bài tập 2.** Thêm một nút bất kỳ và lập trình cho nút đó xóa từng ký tự trên hộp văn bản khi được nhấn.