

## Bài 8 : Hàm

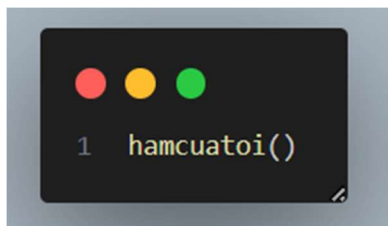
Hàm giống như một nút có các chức năng trong đó, khi mình bấm nút đó thì các câu lệnh mình cho sẵn sẽ được kích hoạt. Hàm trong Python được kí hiệu là **def**. Để tạo một hàm trong Python, ta chỉ cần ghi **def** xong rồi ghi tên của nó, 2 ngoặc tròn và dấu hai chấm. VD :



```
1 def hamcuatoi():  
2     print('Hello World!')
```

Đây là 2 dòng mã để tạo được một hàm tên là “hamcuatoi”, có chức năng là xuất ra chữ “Hello World!” sử dụng hàm **print**.

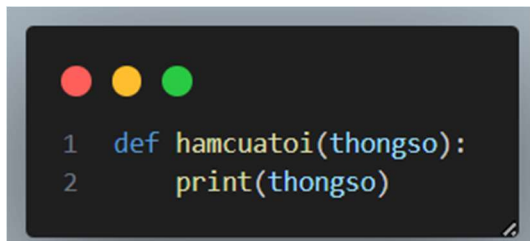
Để gọi hàm vừa tạo, ta có thể viết tên hàm đó và viết thêm 2 dấu ngoặc tròn đằng sau. VD :



```
1 hamcuatoi()
```

Đây là dòng mã để gọi hàm tên là “hamcuatoi”.

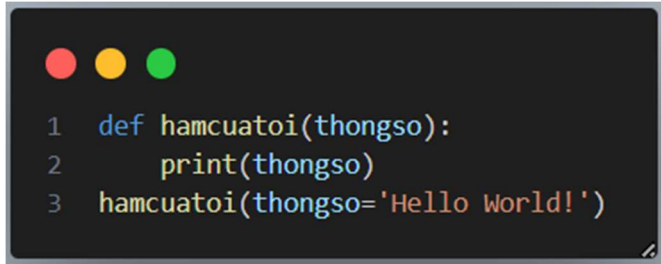
Thông số cũng là một thứ quan trọng của một hàm, khi ta gọi thì ta phải điền các thông số hàm đó yêu cầu (nếu có thông số). Cách tạo thông số :



```
1 def hamcuatoi(thongso):  
2     print(thongso)
```

Đây là một hàm bao gồm cả thông số từ ngoài vào, khi người dùng gọi hàm, nếu không điền thông số thì nó sẽ báo lỗi còn không thì nó sẽ tiếp tục.

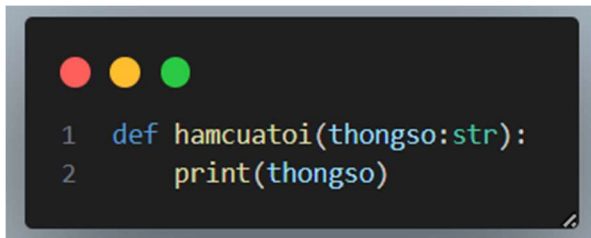
Để điền một thông số thì ta sẽ viết thông số trong đó. VD:



```
1 def hamcuatoi(thongso):  
2     print(thongso)  
3 hamcuatoi(thongso='Hello World!')
```

Đoạn mã này sẽ in ra “Hello World!”. Bạn có thể thay đổi `thongso='Hello World'`.

Làm như thế này thì cái `thongso` có thể là nhiều kiểu dữ liệu khác nhau, để cố định một kiểu dữ liệu thì ta sẽ làm như sau :



```
1 def hamcuatoi(thongso:str):  
2     print(thongso)
```

Khi thêm `:str` vào thông số thì hàm sẽ chỉ chấp nhận `thongso` là string, sẽ báo lỗi khi điền kiểu dữ liệu khác.