



VIETCODE ACADEMY INTRODUCTION TO PROGRAMMING

BUỔI 5: Hàm (Function)





THÔNG TIN GIÁO VIÊN

GIÁO VIÊN CHÍNH



TRỢ GIẢNG





Hàm là gì và tại sao cần ?

Chúng ta đã tối ưu các chương trình code bằng hàm for, thay vì viết 1000 dòng print thì chúng ta chỉ cần 2 đến 3 dòng

Vậy hãy thử ví dụ bài toán in ra các số lẻ từ 1 đến n tuy nhiên lúc này số lượng n ta muốn chương trình chạy là 10

Vậy thì nếu viết 10 for và 10 if để in ra các số lẻ thì rất vô nghĩa và tốn sức

Vậy thì hàm đã giúp ta giải quyết việc này chỉ bằng cách định nghĩa và gọi tên hàm 10 lần một cách rất ngắn gọn

=> Hàm là một chương trình con có thể sử dụng lại dùng để thực hiện một số thao tác hoặc trả lại 1 giá trị thông qua tên của nó





Cách sử dụng hàm

Vậy thì để xử lý bài toán in các số lẻ từ 1 đến n bằng cách sử dụng hàm thì ta phải lập trình như thế nào



```
def print_odd_numbers(n):  
    for i in range(1,n+1):  
        if(i%2!=0):  
            print(i)  
  
print_odd_numbers(5)  
print_odd_numbers(100)
```

Ở đoạn code trên ta đã định nghĩa hàm `print_odd_numbers` có chức năng in các số lẻ từ 1 đến n, và sau đó ta **gọi** hàm với 2 số n là 5 và n là 100. Nên lúc này ở terminal sẽ hiện ra các số lẻ từ 1 đến 5 và sau đó là các số lẻ từ 1 đến 100.



Cấu trúc hàm

Như đã thấy ở đoạn code trên, hàm có cấu trúc cơ bản như sau:

```
def function_name (parameter1, parameter2, parameter3, ...):  
    code
```

Đầu tiên cần đặt tên hàm và tên các tham số sẽ được sử dụng (số lượng tham số tùy thuộc vào mục đích của hàm). Sau đó trong code sẽ thực thi một tập hợp những câu lệnh sử dụng những tham số ta đã khai báo ở bên cạnh `function_name`. Các tham số ở đây có thể là string, int, bool hoặc thậm chí là list.

Lưu ý rằng đây mới là định nghĩa hàm còn để những đoạn code thực thi thì ta cần gọi tên hàm và những tham số **đã có giá trị tương ứng với những gì ta đã định nghĩa**. Ví dụ: `function_name(1,2,3,..)`



Bài tập làm quen

Định nghĩa hàm tên `print_sum` in ra tổng của hai tham số `a` và `b`.

Định nghĩa hàm tên `print_sum_of_odd_numbers` in ra tổng tất cả các số lẻ trong tham số list `a`

=> Tham số nhận vào có thể có số lượng tham số hoặc các dạng khác nhau tùy vào mục đích sử dụng. Tuy nhiên lúc gọi hàm chúng ta cần phải cho các giá trị **tương ứng**

```
def print_sum(a,b):  
    result=a+b  
    print(result)
```

```
print_sum(3,6)
```

```
def print_sum_of_odd_numbers(a):  
    result=0  
    for i in range(len(a)):  
        if(a[i]%2!=0):  
            result+=a[i]  
    print(result)
```

```
print_sum_of_odd_numbers([1,3,4,5,7,12])
```





Trả giá trị cho hàm và tại sao cần

Ngoài tác dụng in ra các giá trị thì hàm còn có thể nhận các giá trị. Vì trong lập trình ta không chỉ in ra mà ta cần phải sử dụng giá trị để thực hiện các thao tác khác.

Ví dụ sử dụng hàm để nhận giá trị tổng của hai tham số a và b , sau đó in ra bình phương giá trị đó thì ta lập trình như nào

```
def sum_of_two_number(a,b):  
    result=a+b  
    return result  
  
result=sum_of_two_number(2,3)  
result=result*result  
print(result)
```





Cấu trúc trả giá trị cho hàm

Như đã thấy ở ví dụ trên, để trả giá trị cho hàm có cấu trúc như sau:

```
def function_name (parameter1, parameter2, parameter3, ...):
```

```
    code
```

```
    return value
```

Lúc này function_name sẽ nhận giá trị value. Value có thể là string, int, bool, hay thậm chí là list tùy vào mục đích của hàm

Lưu ý return cũng giống như break một khi đã chạy thì tất cả những đoạn code dưới trong hàm sẽ không được chạy và thoát ra trả giá trị value cho hàm



Câu hỏi luyện tập

```
def mysterious_function(a):  
    result=[]  
    for i in range(1,a+1):  
        if(i%2==0):  
            result.append(i)  
    return result
```

Hàm mysterious_function sẽ nhận có giá trị là một list chứa các số chẵn từ 1 đến tham số a



```
def mysterious_function(a):  
    if(a%2==0):  
        return True  
    else:  
        return False
```

Hàm mysterious_function sẽ kiểm tra xem số a có phải số chẵn không. Nếu có hàm sẽ có giá trị là True còn nếu không hàm sẽ có giá trị là False



THỰC HÀNH VỚI FILE INCLASS5