

函数返回值

在python中我们怎样返回多个值？

一、多个return？

```
def create_nums():  
    print("----1---")  
    return 1 # 函数中下面的代码不会被执行，因为return除了能够将数据返回之外，还有一个隐藏的功能：结束函数  
    print("----2---")  
    return 2  
    print("----3---")
```

总结1：

- 一个函数中可以有多个return语句，但是只要有一个return语句被执行到，那么这个函数就会结束了，因此后面的return没有什么用处
- 如果程序设计为如下，是可以的因为不同的场景下执行不同的return

```
def create_nums(num):  
    print("----1---")  
    if num == 100:  
        print("----2---")  
        return num+1 # 函数中下面的代码不会被执行，因为return除了能够将数据返回之外，还有一个隐藏的功能：结束函数  
    else:  
        print("----3---")  
        return num+2  
    print("----4---")  
  
result1 = create_nums(100)  
print(result1) # 打印101  
result2 = create_nums(200)  
print(result2) # 打印202
```

二、一个函数返回多个数据的方式

```
def divid(a, b):
    shang = a//b
    yushu = a%b
    return shang, yushu #默认是元组

result = divid(5, 2)
print(result) # 输出(2, 1)
```

总结2:

- return后面可以是元组、列表、字典等，只要是能够存储多个数据的类型，就可以一次性返回多个数据。

```
def function():
    # return [1, 2, 3]
    # return (1, 2, 3)
    return {"num1": 1, "num2": 2, "num3": 3}
```

- 如果return后面有多个数据，那么默认是元组。

对返回的数据直接拆包

```
def get_my_info():
    high = 178
    weight = 100
    age = 18
    return high, weight, age # 函数返回三个数据，会自动打包为元组

# result = get_my_info() # result 接收到一个元组
# print(result)

my_high, my_weight, my_age = get_my_info() # 直接把元组拆分为三个变量来使用，更加方便
print(my_high)
print(my_weight)
print(my_age)
```

总结:

- 拆包时要注意，需要拆的数据的个数要与变量的个数相同，否则程序会异常
- 除了对元组拆包之外，还可以对列表、字典等拆包

```
In [17]: a, b = (11, 22)
```

```
In [18]: a
```

```
Out[18]: 11
```

```
In [19]: b
```

```
Out[19]: 22
```

```
In [20]: a, b = [11, 22]
```

```
In [21]: a
```

```
Out[21]: 11
```

```
In [22]: b
```

```
Out[22]: 22
```

```
In [23]: a, b = {"m":11, "n":22} # 取出来的是key, 而不是键值对
```

```
In [24]: a
```

```
Out[24]: 'm'
```

```
In [25]: b
```

```
Out[25]: 'n'
```

千锋Python人工智能学院