

## 第2课 自定义函数的创建与调用

## 函数的返回值

函数不是直接显示输出的，它会处理一些数据并返回一个或一组值。在函数中用return语句将值返回到调用函数的代码行，返回值能将程序大部分繁重的工作移交到函数中去完成，从而简化主程序。

下面是一个简单的程序，接收姓氏和名称，返回完整的人名信息

## 函数的返回值

```
def name(first_name, last_name):  
    full_name=first_name+" "+last_name  
    return full_name  
print(name("zhang", "san"))
```

运行结果：  
zhang san

## 函数的返回值

函数可以返回任何类型的值，包括字典、列表这样较复杂的数据结构。还是上面的例子，返回一个表示人的字典：

```
def name(first_name, last_name,):  
    full_name={"first":first_name, "last":last_name}  
    return full_name  
print(name("zhangsan", "lisi"))
```

运行结果为：

```
{'first': 'zhangsan', 'last': 'lisi'}
```

## 函数的返回值

函数传递列表，传递列表在函数中很有用，在列表中包  
含数字、名字甚至更复杂的对象，下面举一个例子：

```
def f(names):  
    for i in names:  
        print("Hello"+" "+i+"!!")  
f(["zhangsan", "lisi", "wangwu"])
```

运行结果：

Hello zhangsan!!

Hello lisi!!

Hello wangwu!!

## 函数的返回值

函数传递列表，传递列表在函数中很有用，在列表中包  
含数字、名字甚至更复杂的对象，下面举一个例子：

```
def f(names):  
    for i in names:  
        print("Hello"+" "+i+"!!")  
f(["zhangsan", "lisi", "wangwu"])
```

运行结果：

Hello zhangsan!!

Hello lisi!!

Hello wangwu!!

## 小试牛刀

带返回值函数

```
def fact(n):  
    factorial=1  
    for counter in range(1,n+1):  
        factorial *=counter  
    return factorial  
n=int(input('calculate n! Enter n=?'))  
print(n,'!=',fact(n))
```

## 小试牛刀

参数带默认值的函数

```
def rt1(a=3):  
    for n in range(a):  
        for m in range(n+1):  
            print('*',end='')  
        print()  
rt1()  
rt1(5)
```