

Python 中函数的参数传递

1、形参与实参

在函数名中定义的参数为形参，在函数调用的时候，给函数名中的形参传递的参数为实参

```
def user(name): 形参
    print(name)

user('小明') 实参
```

2、必须参数

在定义函数时，没有默认值且必须在函数执行时传递的参数，实参的数量和形参的数量保持一致

```
def fun1(a, b, c):
    print(a, b, c)

fun1(True, 10, "123")
```

3、默认参数

```
def fun3(name, age=10):
    print("I am", name, "I am", age)

fun3("tom")
fun3("bob", 30)
```

4、不定长参数【可变参数】

能处理比当初声明函数更多的参数

```
def fun4(name, *args, **kwargs):
    # 注意: *的不定长参数被当做元组处理,**的不定长参数被当做字典处理
    # ** 的实参: 需要通过key = value的形式传参
    print(name)      # tom
    print(args)      # ('games', 'dance')
    print(kwargs)    # {'x': 1, 'y': 2}

fun4("tom", "games", "dance", x=1, y=2)
```

5、传入元组及列表类型的数据

当调用函数时参数是元组或列表类型的数据时, 可以在变量名前添加*进行解包

```
def func(a, b, c):
    print(a, b, c)

lst = [1, 2, 3]
func(*lst)    # 1 2 3
```

6、传入字典类型的数据

当调用函数时参数是字典类型的数据时, 可以在变量名前添加**进行解包

```
def func(a, b, c):
    print(a, b, c)

dic = {"a": 1, "b": 2, "c": 3}
func(**dic)    # 1 2 3
```

7、关键字参数

注意: 关键字和形参能够匹配上,并且实参的顺序可以任意

```
def fun(name, age):  
    print("I am", name, "I am", age)
```

```
fun("jack", 20)
```

```
fun(20, "jack")
```

```
# 使用关键字参数
```

```
fun(name="jack", age=18)
```

```
fun(age=18, name="jack")
```

