

python_函数

1、函数的位置传参

备注：函数按位置进行传参

```
def fun_class(name ,sex):  
  
    print('name:%s sex:%s'%(name,sex))
```

```
fun_class('zhaozhao','nv')
```

```
fun_class('nv','zhaozhao')
```

```
D:\study\python\test\venv\Scripts\python.exe D:/study/python/test/dd.py
```

```
name:zhaozhao sex:nv
```

```
name:nv sex:zhaozhao
```

2、函数默认值传参

备注：默认值传参，如果无传参，传默认的，如果有传参，使用最新的传参

```
def fun_class(name ,sex='nv'):  
  
    print('name:%s sex:%s'%(name,sex))
```

```
fun_class('zhaozhao','nan')
```

```
fun_class('zhaozhao')
```

```
D:\study\python\test\venv\Scripts\python.exe D:/study/python/test/dd.py
```

```
name:zhaozhao sex:nan
```

```
name:zhaozhao sex:nv
```

3、函数可变参数*args

这种传参的特点是：

1、不是必填参数

2、不限制参数的个数

3、传入多个参数的时候，他把参数放到一个元组里面

备注：args只是一个名字，可以随意取名字，但是大家使用的时候，都会写成args

```
def fun_class(*args): #参数个数可变

    print(args)

    print(type(args)) #返回的类型是元组


fun_class() #不传参数

fun_class('zhaozhao') #传一个参数

fun_class('zhaozhao', 'nv') #传2个参数

fun_class('zhaozhao', 'nv', '18') #传3个参数
```

D:\study\python\test\venv\Scripts\python.exe D:/study/python/test/dd.py

```
()

<class 'tuple'>

('zhaozhao',)

<class 'tuple'>

('zhaozhao', 'nv')

<class 'tuple'>

('zhaozhao', 'nv', '18')

<class 'tuple'>
```

4、函数可变参数**args

这种传参的特点：

1、不是必传参

2、必须是key=value这种方式进行调用及传参，不限制参数个数

3、传入多个参数后，他把参数放到了一个字典中

args只是一个名字，可以随意取名字，但是大家使用的时候，都会写成args

```
def fun_class(**args): #参数个数可变

    print(args)

    print(type(args)) #返回的类型是字典


fun_class() #不传参数

fun_class(name='zhaozhao') #传一个参数，且key=value模式传参

fun_class(name='zhaozhao', sex='nv') #传2个参数，且key=value模式传参

fun_class(name='zhaozhao', sex='nv', age='18') #传3个参数，且key=value模式传参
```

D:\study\python\test\venv\Scripts\python.exe D:/study/python/test/dd.py

```
{}
```

```
<class 'dict'>
```

```
{'name': 'zhaozhao'}
```

```
<class 'dict'>
```

```
{'name': 'zhaozhao', 'sex': 'nv'}
```

```
<class 'dict'>
```

```
{'name': 'zhaozhao', 'sex': 'nv', 'age': '18'}
```

```
<class 'dict'>
```

5、函数