

Python匿名函数

讲师：Wayne

从业十余载，漫漫求知路

匿名函数

- 匿名，即没有名字
- 匿名函数，即没有名字的函数
 - 没有名字如何定义
 - 没有名字如何调用
 - 如果能调用，如何使用

匿名函数

□ Python借助Lambda表达式构建匿名函数

□ 格式

□ lambda 参数列表：表达式

```
lambda x : x ** 2
```

```
(lambda x : x ** 2)(4) # 调用
```

```
foo = lambda x, y : (x + y) ** 2 # 不推荐这么用
```

```
foo(2, 1)
```

```
def foo(x,y): # 建议使用普通函数
```

```
    return (x + y) ** 2
```

```
foo(2, 1)
```

匿名函数

- 使用lambda关键字来定义匿名函数
- 参数列表不需要小括号
- 冒号是用来分割参数列表和表达式的
- 不需要使用return，表达式的值，就是匿名函数返回值
- lambda表达式（匿名函数）**只能写在一行上**，被称为单行函数
- 用途
 - 在高阶函数传参时，使用lambda表达式，往往能简化代码

匿名函数

- ❑ `print((lambda :0)())`
 - ❑ `print((lambda x, y=3: x + y)(5))`
 - ❑ `print((lambda x, y=3: x + y)(5, 6))`
 - ❑ `print((lambda x, *, y=30: x + y)(5))`
 - ❑ `print((lambda x, *, y=30: x + y)(5, y=10))`
 - ❑ `print((lambda *args: (x for x in args))(*range(5)))`
 - ❑ `print((lambda *args: [x+1 for x in args])(*range(5)))`
 - ❑ `print((lambda *args: {x+2 for x in args})(*range(5)))`
-
- ❑ `[x for x in (lambda *args: map(lambda x: x+1, args))(*range(5))]` # 高阶函数
 - ❑ `[x for x in (lambda *args: map(lambda x: (x+1,args), args))(*range(5))]`

谢谢

咨询热线 400-080-6560