

python 第二部分

流程控制语句及各种表达式

本次分享的主要内容都在下面这段代码当中,大家将就看看哈:p

```
1  #encoding:utf-8
2  def transfer(str):
3      a = 0
4      while True:
5          if str[a].isupper():
6              str = str[0:a]+str[a].lower()+str[a+1:]
7          else:
8              str = str[0:a]+str[a].upper()+str[a+1:]
9          yield str
10         a += 1
11  lam = lambda :raw_input('请输入姓名的拼音:')
12  name = lam()
13  oiter = transfer(name)
14  nlist = [i for i in name]
15  for i in nlist:
16      print oiter.next()
```

- 第2行: 函数声明
- 第4行: while循环
- 第5行: if 语句
- 第9行: 生成器
- 第11行: lambda表达式
- 第13行: 函数调用
- 第14行: 列表推导式
- 第15行: for循环

函数

1. 声明: def 关键字

```
def printer():  
    print '我是一个函数'
```

2. 调用: 函数名+括号

```
def printer():  
    print '我是一个函数'  
  
printer()
```

函数参数

1. 关键字参数

```
def printer(str='anjuke'):
    print '我是'+str

printer()
printer("baidu")
```

2. 不定参数

```
def printer(str='anjuke',*option):
    print '我是'+str
    print option

printer('baidu','moto','apple')
```

```
def printer(str='anjuke',**option):
    print '我是'+str
    print option

printer('baidu',x='moto',y='apple')
```

函数返回值

1. 多个返回值

```
def printer(a,b):  
    a += 1;  
    b += 1;  
    return a,b  
  
print printer(1,2)
```

循环语句

1.while

```
n = 10;  
while n>0:  
    print n  
    n -= 1;
```

2.for

```
for i in range(10):  
    print i  
for i,j in [(1,2),(3,4)]:  
    print i,j
```

流程控制

1.if 语句

```
n = 10
if n>10:
    print n
elif n == 10:
    print n-1
else:
    print n+1
```

列表推导式

```
print [i for i in range(10) if i%2 == 1]
```

#上面这段代码等价于

```
a = []  
for i in range(10):  
    if i%2 == 1:  
        a.append(i)  
print a
```

```
print [(i,j) for i in range(10) for j in range(20) if i%2 == 1 if j%10 == 1]
```

#上面这段代码等价于

```
a = []  
for i in range(10):  
    if i%2 == 1:  
        for j in range(20):  
            if j%10 == 1:  
                a.append((i,j))  
print a
```


lambda

#直接使用

```
g = lambda x:x*x  
print g(2)
```

#和列表结合

```
l = [(lambda x:x+1),(lambda y:y*2)]  
print l[0](1),l[1](2)
```

#作为参数传递

```
print map(lambda x:x+10,range(10))
```

生成器

#无参数

```
def generator():  
    a = 10  
    while a>0:  
        yield a  
        a -= 1  
for i in generator():  
    print i
```

#有参数

```
def generator(data):  
    while data>0:  
        yield data*2  
        data -= 1  
for i in generator(5):  
    print i
```