

Python Talk 4



这个指引文档在 知识共享 署名-相同方式共享 3.0协议之条款下提供

This guidance is available under the Creative Commons Attribution-ShareAlike License

复习

- ▶ 使用while计算以下等比数列的乘积
 - ▶ 27, 81, 243, 729, 2187, 6561
- ▶ 使用for和range计算以下等差数列的乘积
 - ▶ 10, 17, 24, 31, 38, 45



循环和判断

▶ 循环 (PythonTalk3)

- ▶ `for`

- ▶ `while`

▶ 循环控制 (PythonTalk4)

- ▶ `break`

- ▶ `continue`

▶ 判断 (PythonTalk4)

- ▶ `if`

- ▶ `else`

- ▶ `elif`



if语句

► 语法

```
if Expression :  
    Code  
  
elif Expression :  
    Code  
  
else :  
    Code  
  
Code
```

► 对比C语言

```
if (Expression) {  
    Code;  
}  
else if (Exp) {  
    Code;  
}  
else {  
    Code;  
}  
  
Code;
```



if例子

- ▶ 解释数值a和b的大小关系

```
if a > b :  
    print('Bigger')  
elif a < b :  
    print('Smaller')  
else :  
    print('Equal')
```



if 练习

- ▶ 输出a年是否为闰年
 - ▶ 公元年分除以4不可整除，为平年。
 - ▶ 公元年分除以4可整除但除以100不可整除，为闰年。
 - ▶ 公元年分除以100可整除但除以400不可整除，为平年。
 - ▶ 公元年分除以400可整除，为闰年。
 - ▶ 以上规则来自 维基百科



答案

程序1

```
if a % 4 == 0 :  
    if a % 100 == 0 :  
        if a % 400 == 0 :  
            print(True)  
        else :  
            print(False)  
    else :  
        print(True)  
else :  
    print(False)
```

程序2

```
if a % 400 == 0 :  
    print(True)  
elif a % 100 == 0 :  
    print(False)  
elif a % 4 == 0 :  
    print(True)  
else :  
    print(False)
```



break和continue

- ▶ 作用于最近的**for**或**while**
 - ▶ Code1上的**break**或**continue**作用于“**for j in b :**”
- ▶ **break**停止循环
 - ▶ 若Code1为**break**，则跳到Flag2
- ▶ **continue**跳过这次循环
 - ▶ 若Code1为**continue**，则跳到Flag1

```
for i in a :  
    for j in b :  
        Code1 <<<  
        Code2  
        # Flag1  
    # Flag2  
Code3
```



break练习

- ▶ 输出整数a是否是质数

```
flag = True
for i in range(2, a) :
    if a % i == 0 :
        print(False)
        flag = False
        break
if flag :
```

```
    print(True)
```

- ▶ 思考：如何更改程序，在 $a \leq 1$ 时输出一个错误信息？



质数算法优化1

- 优化1: 不计算的多余偶数

```
if a == 2 :  
    print(True)  
elif a % 2 == 0 :  
    print(False)  
else :  
    for i in range(3, a, 2) :  
        if a % i == 0 :  
            print(False)  
            break  
    print(True)
```



- 提示: 以上代码有误, 请自行更正

import

- ▶ Python程序可以通过**import**调用一个包，来进行复杂计算和操作
- ▶ 例

```
import math
```

```
math.sqrt(33)    # 得到33的平方根
```

- ▶ 对于平方根，可以不使用**math**包：

```
33**0.5
```

但是使用诸如sin, cos的复杂函数时需要**math**包



质数算法优化2

- ▶ 优化2: 计算到 $\lfloor \sqrt{a} \rfloor$, 而不是 $a - 1$
 - ▶ 例如对于49, 判断2, 3, 5, 7, 而不是2, 3, 5, 7, 9, ..., 47

```
if a == 2 :  
    print(True)  
elif a % 2 == 0 :  
    print(False)  
else :  
    import math  
    e = int(math.sqrt(a)) + 1  
    for i in range(3, e, 2) :  
        if a % i == 0 :  
            print(False)  
            break  
    print(True)
```



- ▶ 提示: 以上代码有误, 请自行更正

感谢参加此次活动

