

用Python做些事

03-做点小事的语句表达式



做点小事的语句表达式

- ☐ 赋值
- ☐ 输入输出
- ☐ If/else
- ☐ For , while
- ☐ 列表解析
- ☐ 异常处理
- ☐ 猜数字

赋值

多重赋值

```
a, b = 1, 2  
a, b = "bj", "sh"  
a, b = "bj"  
a = b = "bj"  
a, *b = "beijing "  
a, b = (1, 2)  
a, b = [1, 2]  
+=
```

做点小事的语句表达式

☒ 赋值

- ☐ 输入输出
- ☐ If/else
- ☐ For , while
- ☐ 列表解析
- ☐ 异常处理
- ☐ 猜数字

输入

input

raw_input()

input()

黑板客版权所有
www.heibanke.com

输出

3.x

函数

```
print([obj, ...][, sep= ' '][,end= '\n'][,file=sys.stdout])
```

2.x

语句

```
print a, b
```

```
print >>file, a, b
```

```
print '%d,%d,%s'%(a,b,c)
```

```
print "{0} like {1} ".format('we','python')
```

```
print "{a} like {b} ".format(a='we',b='python')
```

做点小事的语句表达式

- ☒ 赋值
- ☒ 输入输出
- ☐ If/else
- ☐ For , while
- ☐ 列表解析
- ☐ 异常处理
- ☐ 猜数字



IF

语法

```
If xxx: statements1  
elif xxx: statements2  
else: statements3
```

```
x,y,z=[1,2,-2]  
if x>0:  
    a=x  
elif x==0:  
    a=y  
else:  
    a=z
```

其他用法

逻辑表达式 not / and / or
三元表达式 a=y if x>0 else z

```
x>5 and 0  
x and y  
x and z  
0 or z  
3>y>1  
1<y<3  
x=3; y=4; z=5
```

避免混用Tab和空格

Pass

分号

换行：括号匹配和反斜线,

做点小事的语句表达式

- ☒ 赋值
- ☒ 输入输出
- ☒ If/else
- ☐ For , while
- ☐ 列表解析
- ☐ 异常处理
- ☐ 猜数字

for/while

while/else

```
While xxx:  
    statements1  
    if xxx: break/continue  
else:  
    statements2
```

小于等于10的偶数

判断是否是质数

for/else

```
For x in objects:  
    statements1  
    if xxx: break/continue  
Else:  
    statements2
```

例子：
列表，字符串，元组，字典，文件

做点小事的语句表达式

- ☒ 赋值
- ☒ 输入输出
- ☒ If/else
- ☒ For, while
- ☐ 列表解析
- ☐ 异常处理
- ☐ 猜数字

列表解析

用法

去除列表中重复元素

```
l1 = ['b','c','d','b','c','a','a']
```

```
l2 = []
```

```
[l2.append(i) for i in l1 if not i in l2]
```

```
print l2
```

举例

```
>>> iter=(x**2 for x in range(10) if x%2==0)
```

```
>>> iter
```

```
<generator object <genexpr> at 0x02419CB0>
```

```
>>> for el in iter:
```

```
    print el
```

```
print [(x,y) for x in (1,2,3,4) for y in (10,15,3,22) if x*y > 25]
```

做点小事的语句表达式

- ☒ 赋值
- ☒ 输入输出
- ☒ If/else
- ☒ For, while
- ☒ 列表解析
- ☐ 异常处理
- ☐ 猜数字

异常处理

用法

try/except , try后语句有异常发生后执行except
try/finally , 无论异常是否发生都执行finally
raise error(message) 触发异常
Assert condition , message
条件触发错误, 触发后打印信息, 并终止程序

举例

```
a[1]          #NameError  
a=2; a[1]     #TypeError  
a=[2]; a[1]   #IndexError  
a={}; a[1]    #KeyError
```

```
Raise IndexError  
Assert False, u" error occur, please check program"
```

做点小事的语句表达式

- ☒ 赋值
- ☒ 输入输出
- ☒ If/else
- ☒ For, while
- ☒ 列表解析
- ☒ 异常处理
- ☐ 猜数字

猜数字

1. 随机产生要猜的数字
2. 输入，用于接收用户输入的数字
3. 循环，如果没猜对则循环接收输入, 并打出提示信息
4. 猜到数字或猜测次数达到一定次数后(6次)打印失败并退出

THANKS

?



作业

- 3-1 改进猜数字游戏. 防作弊, 错误输入判断.
- 3-2 利用上次用户密码作业, 请模拟注册过程: 用户输入用户名后进行检测用户名是否在文件中的过程。并返回合理的错误提示。如果不在则再输入密码, 成功则增加用户信息到文件中, 密码进行md5加密处理
- 3-3 增加用户名, 密码的合法化判断和错误提示
用户名: 字母, 数字, 下划线和横线的组合, 且首字符应是字母, 长度不小于4
密码: 字母, 数字, 下划线和横线的组合, 且长度不小于6,

作业

□ 3-4 继续上一课的公交系统作业

利用循环语句将所有线路的linenum和stations保存到一个字典对象

执行后提示输入公交站名字，在所有公交线路的stations里查询该名字，并将包含有该名字的公交线路存到一个字典进行返回