

1, `math`包主要处理数学相关的运算

2, `random`模块的作用是产生随机数, 这个模块实现了伪随机数产生器

3, 输入一个数

```
inp = input("请输入(0表示剪刀, 1表示石头, 2表示布)")
```

```
pay = x(inp) 其中x表示数据类型
```

4, 在2.x版本想要使用`end= ' '`, 则需要在代码的首行添加如下语句:

```
from __future__ import print_function
```

一, `if`语句

1, `if`语句:

格式: `if` 判断条件: 执行语句

满足条件才会执行指定的代码, 否则就不执行

注意: ①每个`if`条件后要使用冒号(`:`), 表示接下来是满足条件后要执行的语句

②使用缩进来划分语句块, 相同的缩进数的语句在一起组成一个语句块

③在python中没有`switch-case`语句

2, `if-else`语句

格式: `if` 判断条件成功: 执行语句

`else`: 条件判断不成功执行语句

3, `if-elif`语句

格式: `if` 判断条件1满足: 执行语句1

`elif`判断条件2满足: 执行语句2

`elif`判断语句3满足: 执行语句3

.....

`else`: 执行语句n

注意: `elif`必须和`if`语句一起使用, 否则程序会出错

`if`语句可以进行嵌套, 在`if`或者`if-elif`语句里面包含`if`或者`if-elif`语句

二, 循环语句

1, `while`循环语句

格式: `while` 条件表达式:

条件满足, 执行循环代码

注意: 在`while`循环中, 同样需要注意冒号和缩进, 另外, python中没有`do-while`循环

## 2, for循环语句

格式: for 变量 in 序列

循环语句

注意: python提供了一个内置的range函数, 它可以生成一个个数字序列

格式: for 变量 in range (start, end)

执行循环语句

## 3, while嵌套循环

while嵌套指的是while里面还包含while

格式: while 条件表达式:

循环语句1

while 条件表达式1:

循环语句2

.....

## 三, python的其他语句

1, break语句: break语句用于结束整个循环(当前循环体)

2, continue语句: continue语句的作用是用来结束本次循环, , 紧接着下一次循环

**注意: ①break/continue语句只能用在循环语句中, 除此之外不能单独使用**

**②break/continue在嵌套循环中, 只对最近一层循环起作用**

3, pass语句: python中的pass语句是空语句, 它的出现是为了保持程序结构的完整性, pass

(程序会忽略该语句) 不做任何事情, 一般用做占位语句

4, else语句块:

除了判断语句, python语句中的while和for循环也可以使用else语句, 在循环中使用时, else语句只在循环完成后执行, 也就是说, break语句也会跳过else语句块

**例: 打印直角三角形:**

**方式1:** for i in range(10):

print('\* '\*i)

**方式2:** while i < 8:

j = 0

while j < i:

print("\* ", end='')

j += 1

print ("\n")

```
i+=1
```

例：打印九九乘法表

```
i = 1
```

```
while i < 10:
```

```
    j = 1
```

```
    while j <= i:
```

```
        print("%d*%d=%-5d"%(i, j, i*j), end='')
```

```
        j += 1
```

```
    print("\n")
```

```
    i+= 1
```