**[[转]python条件表达式的四种实现方法 和 and、or 的计算规则](https://www.cnblogs.com/OoycyoO/p/9864629.html)**

a, b, c = 1, 2, 3  
   
**# 1.常规**  
if a>b:  
    1

else:

if c>d:

3  
else

4

**# 2.表达式**  
1 if a>b else 3 if c>d else 4   # 先执行中间的if，如果返回True，就是左边，False是右边。

a,b,c = 5,4,3

>>> 1 if b>c else 2 if a>b else 3

If a>b:

c=a

else:

c=b  
**# 3.二维列表**  
c = [b,a][a>b]    #实际是[b,a][False]，因为False被转换为0，所以是[1,2][0]，也就是[1]a=  
 # False返回第一个，True 返回第一个。  
   
**# 4**  
c = (a>b and [a] or [b])[0]  
# 这个比较好玩，False and [1] or [2]，因为and的优先级高于or，先算and  
# False和[1] and之后还是False，和[2]or之后却成了[2]  
# True 和[1] and之后是[1]，[1]和[2]or结果是[1]  
# 也就是False和True在和别人做boolean运算的时候，根据and还是or，F和T在前在后有不一样的数据转换规则。

1. 在纯and语句中，如果每一个表达式都不是假的话，那么返回最后一个，因为需要一直匹配直到最后一个。如果有一个是假，那么返回假  
   2. 在纯or语句中，只要有一个表达式不是假的话，那么就返回这个表达式的值。只有所有都是假，才返回假  
   3. 在or和and语句比较难表达，总而言之，碰到and就往后匹配，碰到or如果or左边的为真，那么就返回or左边的那个值，如果or左边为假，继续匹配or右边的参数。