```
□ Python教程
关于作者
```

>>> classmates.pop() 'Adam' >>> classmates ['Michael', 'Jack', 'Bob', 'Tracy'] 要删除指定位置的元素,用 pop(i) 方法, 其中 i 是索引位置: >>> classmates.pop(1) 'Jack' >>> classmates ['Michael', 'Bob', 'Tracy'] 要把某个元素替换成别的元素,可以直接赋值给对应的索引位置: >>> classmates[1] = 'Sarah' >>> classmates ['Michael', 'Sarah', 'Tracy'] list里面的元素的数据类型也可以不同,比如: >>> L = ['Apple', 123, True] list元素也可以是另一个list, 比如: >>> s = ['python', 'java', ['asp', 'php'], 'scheme'] >>> len(s) 4 要注意。只有4个元素,其中。[2]又是一个list,如果拆开写就更容易理解了: >>> p = ['asp', 'php'] >>> s = ['python', 'java', p, 'scheme'] 要拿到 'php' 可以写 p[1] 或者 s[2][1] ,因此 s 可以看成是一个二维数组,类似的还有三维、四维……数组,不过很少用到。 如果一个list中一个元素也没有,就是一个空的list,它的长度为0:

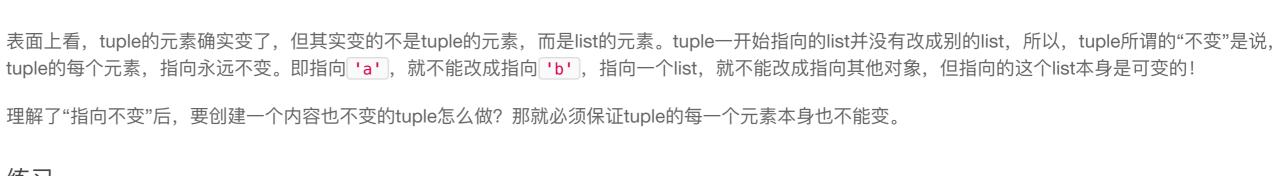
```
>>> classmates = ('Michael', 'Bob', 'Tracy')
现在,classmates这个tuple不能变了,它也没有append(),insert()这样的方法。其他获取元素的方法和list是一样的,你可以正常地使
用 classmates [0] , classmates [-1] ,但不能赋值成另外的元素。
不可变的tuple有什么意义?因为tuple不可变,所以代码更安全。如果可能,能用tuple代替list就尽量用tuple。
tuple的陷阱: 当你定义一个tuple时,在定义的时候,tuple的元素就必须被确定下来,比如:
 >>> t = (1, 2)
 >>> t
 (1, 2)
如果要定义一个空的tuple,可以写成():
 >>> t = ()
```

另一种有序列表叫元组:tuple。tuple和list非常类似,但是tuple一旦初始化就不能修改,比如同样是列出同学的名字:

```
>>> t = (1,)
 >>> t
 (1,)
Python在显示只有1个元素的tuple时,也会加一个逗号,,,以免你误解成数学计算意义上的括号。
最后来看一个"可变的"tuple:
```

定义的不是tuple,是 1 这个数!这是因为括号 () 既可以表示tuple,又可以表示数学公式中的小括号,这就产生了歧义,因此,Python规定,这种情况下,

这个tuple定义的时候有3个元素,分别是 'a', 'b' 和一个list。不是说tuple—旦定义后就不可变了吗?怎么后来又变了?



>>> L = [] >>> len(L)

tuple

>>> t ()

>>> t = (1)

>>> t

1

但是,要定义一个只有1个元素的tuple,如果你这么定义:

所以,只有1个元素的tuple定义时必须加一个逗号,,来消除歧义:

按小括号进行计算, 计算结果自然是 1。

>>> t = ('a', 'b', ['A', 'B'])

别急,我们先看看定义的时候tuple包含的3个元素:

>>> t[2][0] = 'X' >>> t[2][1] = 'Y'

('a', 'b', ['X', 'Y'])

>>> t

请用索引取出下面list的指定元素:

当我们把list的元素 'A' 和 'B' 修改为 'X' 和 'Y' 后, tuple变为:

```
练习
# -*- coding: utf-8 -*-
L = [
     ['Apple', 'Google', 'Microsoft'],
    ['Java', 'Python', 'Ruby', 'PHP'],
    ['Adam', 'Bart', 'Lisa']
```

打印Apple:

打印Python:

print(?)

print(?)

 \Box a = ()

 \Box b = (1)

打印Lisa:

]

```
print(?)
Run
```

```
\Box c = [2]
\Box d = (3,)
\Box e = (4,5,6)
```

② 请问以下变量哪些是tuple类型:

list和tuple是Python内置的有序集合,一个可变,一个不可变。根据需要来选择使用它们。 参考源码 the_list.py

the_tuple.py

Submit

小结

读后有收获可以支付宝请作者喝咖啡, 读后有疑问请加微信群讨论:



Make a comment

Sign in to make a comment



友情链接: 中华诗词 - 阿里云 - SICP - 4clojure

```
本网站运行在阿里云上并使用阿里云CDN加速。
```

廖雪峰的官方网站©2019

Powered by iTranswarp

Next Page >