

Python3 元组(tuple)

Python 的元组与列表类似，不同之处在于元组的元素不能修改。

元组使用小括号，列表使用方括号。

元组创建很简单，只需要在括号中添加元素，并使用逗号隔开即可。

实例

```
>>>tup1 = ('Google', 'Unigress', 1997, 2000);
>>> tup2 = (1, 2, 3, 4, 5 );
>>> tup3 = "a", "b", "c", "d";    # 不需要括号也可以
>>> type(tup3)
<class 'tuple'>
```

创建空元组

```
tup1 = ();
```

元组中只包含一个元素时，需要在元素后面添加逗号，否则括号会被当作运算符使用：

实例

```
>>>tup1 = (50)
>>> type(tup1)    # 不加逗号，类型为整型
<class 'int'>

>>> tup1 = (50,)
>>> type(tup1)    # 加上逗号，类型为元组
<class 'tuple'>
```

元组与字符串类似，下标索引从0开始，可以进行截取，组合等。

访问元组

元组可以使用下标索引来访问元组中的值，如下实例:

实例(Python 3.0+)

```
#!/usr/bin/python3

tup1 = ('Google', 'Unigress', 1997, 2000)
tup2 = (1, 2, 3, 4, 5, 6, 7 )

print ("tup1[0]: ", tup1[1])
print ("tup2[1:5]: ", tup2[1:5])
```

以上实例输出结果：

```
tup1[0]: Unigress
tup2[1:5]: (2, 3, 4, 5)
```

修改元组

元组中的元素值是不允许修改的，但我们可以对元组进行连接组合，如下实例：

实例(Python 3.0+)

```
#!/usr/bin/python3

tup1 = (12, 34.56)
tup2 = ('abc', 'xyz')

# 以下修改元组元素操作是非法的。
# tup1[0] = 100

# 创建一个新的元组
tup3 = tup1 + tup2
print(tup3)
```

以上实例输出结果：

```
(12, 34.56, 'abc', 'xyz')
```

删除元组

元组中的元素值是不允许删除的，但我们可以使用del语句来删除整个元组，如下实例：

```
#!/usr/bin/python3

tup = ('Google', 'Unigress', 1997, 2000)

print (tup)
del tup;
print ("删除后的元组 tup : ")
print (tup)
```

以上实例元组被删除后，输出变量会有异常信息，输出如下所示：

```
删除后的元组 tup :
Traceback (most recent call last):
  File "test.py", line 8, in <module>
    print (tup)
NameError: name 'tup' is not defined
```

元组运算符

与字符串一样，元组之间可以使用 + 号和 * 号进行运算。这就意味着他们可以组合和复制，运算后会生成一个新的元组。

Python 表达式	结果	描述
len((1, 2, 3))	3	计算元素个数
(1, 2, 3) + (4, 5, 6)	(1, 2, 3, 4, 5, 6)	连接
('Hi!') * 4	('Hi!', 'Hi!', 'Hi!', 'Hi!')	复制
3 in (1, 2, 3)	True	元素是否存在
for x in (1, 2, 3): print(x,)	1 2 3	迭代

元组索引，截取

因为元组也是一个序列，所以我们可以访问元组中的指定位置的元素，也可以截取索引中的一段元素，如下所示：
元组：

```
L = ('Google', 'Taobao', 'Unigress')
```

Python 表达式	结果	描述
L[2]	'Unigress'	读取第三个元素
L[-2]	'Taobao'	反向读取；读取倒数第二个元素
L[1:]	('Taobao', 'Runoob')	截取元素，从第二个开始后的所有元素。

运行实例如下：

```
>>> L = ('Google', 'Taobao', 'Unigress')
>>> L[2]
'Unigress'
>>> L[-2]
'Taobao'
>>> L[1:]
('Taobao', 'Unigress')
```

元组内置函数

Python元组包含了以下内置函数

序号	方法及描述	实例
1	len(tuple) 计算元组元素个数。	<pre>>>> tuple1 = ('Google', 'Unigress', 'Taobao') >>> len(tuple1) 3 >>></pre>
2	max(tuple) 返回元组中元素最大值。	<pre>>>> tuple2 = ('5', '4', '8') >>> max(tuple2) '8' >>></pre>
3	min(tuple) 返回元组中元素最小值。	<pre>>>> tuple2 = ('5', '4', '8') >>> min(tuple2) '4' >>></pre>
4	tuple(seq) 将列表转换为元组。	<pre>>>> list1= ['Google', 'Taobao', 'Unigress', 'Baidu'] >>> tuple1=tuple(list1) >>> tuple1 ('Google', 'Taobao', 'Unigress', 'Baidu')</pre>