1. **Python数据类型-Tuple（元组）**
2. **问题的引入**

1、如何定义一个坐标 (x,y)

2、如何定义一个包含小时，分钟，秒的数值

(hh:ss:mm)

**二、创建元组**

元组（tuple）也是Python中的数据类型，元组（tuple）与列表类似，不同之处在于元组的元素不能修改,第二不同点是元组写在小括号(())里,元组是写在方括号( )之间、用逗号分隔开的。

举例：

>>> tuple\_a=(12,34,56,78,90)

>>> tuple\_a

(12, 34, 56, 78, 90)

>>> tuple\_b=('w2','ew23','e32','r45')

>>> tuple\_b

('w2', 'ew23', 'e32', 'r45')

元组中元素的类型可以不相同，它支持数字,字符串甚至可以包含列表,及其包含另外一个元组,实现元组的嵌套。

>>> tuple\_c=('we23',34,'we23',1234)

>>> tuple\_c

('we23', 34, 'we23', 1234)

元组中包含列表：

>>> list\_a=[23,'a','be2']

>>> tuple\_d=(23,45,'sw23',list\_a)

>>> tuple\_d

(23, 45, 'sw23', [23, 'a', 'be2'])

元组的嵌套：

>>> tuple\_a=('a',23,'sdwe34')

>>> tuple\_b=(23,45,tuple\_a)

>>> tuple\_b

(23, 45, ('a', 23, 'sdwe34'))

创建元祖时，需要的几点：  
1、创建空元组的代码：

>>> tupNull=()

>>> tupNull

()

2、创建包含一个元素的元祖   
元组中只包含一个元素时，需要在元素后面添加逗号来消除歧义

>>> tupOne=(1) # Python以为你只是声明一个整数

>>> tupOne

1

>>> tupOne=(1,)

>>> tupOne

(1,)

解释：

比如数学运算，就会用到小括号，比如(7+8),如果这么写，python会如何处理呢?直接输出15，加个“,”，就是明确的告诉python，这是一个tuple。  
 但是，要定义一个只有1个元素的tuple，如果你这么定义：

>>> t = (1)  
>>> t  
1

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，按小括号进行计算，计算结果自然是1。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：

>>> 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。

**三、访问元组**

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。元组可以使用下标索引来访问元组中的值，语法：元祖[下标值]，

如果下标值为正数，从头部开始，下标值从0开始。

>>> tuple\_a=('a',23,'sdwe34')

>>> tuple\_a(1)

23

如果下标值为负数，从尾部开始，下标值从1开始。

>>> tuple\_a[-1]

'sdwe34'

>>> tuple\_a[-2]

23  
**四、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:

# 以下修改元组元素操作是非法的

>>> tuple\_a=('a',23,'sdwe34')

>>> tuple\_a[1]='23we'

Traceback (most recent call last):

File "<pyshell#15>", line 1, in <module>

tuple\_a[1]='23we'

TypeError: 'tuple' object does not support item assignment

>>>

**五、元组的连接**

使用+可以实现元组进行连接组合。  
>>> tup1=('s2',34,102)

>>> tup2=(123,'e34',23)

>>> tup1+tup2

('s2', 34, 102, 123, 'e34', 23)

使用\*可以实现重复元祖

>>> tup1\*5

('s2', 34, 102, 's2', 34, 102, 's2', 34, 102, 's2', 34, 102, 's2', 34, 102)

**六、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:

>>> tup1=('s2',34,102)

>>> del tup1

>>> tup1

Traceback (most recent call last):

File "<pyshell#21>", line 1, in <module>

tup1

NameError: name 'tup1' is not defined  
#以上实例元组被删除后，输出变量会有异常信息，'tup1’没有定义。

**七、元组的截取**  
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，得到一个子元组,语法如下：

元祖[头下标:尾下标]

头下标和:尾下标之间用“：”隔开，下标从0开始数

如果尾下标尾负数，则从尾部开始，从右边往左边数。

>>> tup=('qw23',123,45,'dre34',2345)

>>> tup[0:3] #截取第1个元素到第3个元素

('qw23', 123, 45)

>>> tup[2:-1] #截取第3个元素到倒数第2个元素

(45, 'dre34')

>>> tup[1:-2] #截取第2个元素到倒数第3个元素

(123, 45) **八、元组内置函数**  
Python元组包含了以下内置函数  
1、len(tuple)：计算元组元素个数。  
2、max(tuple)：返回元组中元素最大值。  
3、min(tuple)：返回元组中元素最小值。  
4、tuple(seq)：将列表转换为元组。

**九、元组的使用场景**

tuple描述的是一个事物，元组的每个元素是这个事物的一个特征,比较具体的，比如定义一个point=(x,y)，定义一个timer=(hour,minute,second)。

当需要使用一组数据且数据不用修改时，保证你的对象不能被改变，比如调用别人的，或者开源的第三方接口，又不知道会不会改变你的对象，就用元组。