

3 : 'jing'

'a' : 6

"joe" : "smith"

4 : ['X', 'Y']

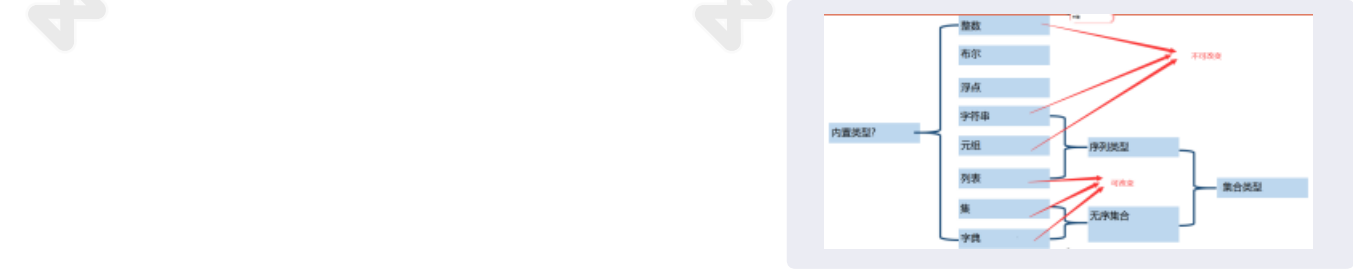
键

key

值

value

- 一个字典的每一个元素都是一个pair，键值对 (key: value)
- 值 (value)：可以是任何对象 (mutable/immutable)，比如数值、字符串、列表、元组，甚至又是一个dictionary。
- 键 (key)：只能是immutable对象，如数值、字符串、元组，而不能是list，或dict。
- 字典是无序的。无论是在print (dict)，或是创建pairs的时候，字典都是**无序的**。



字符串列表元组字典

字符串

基本

换行 str="123 /n 345"

储存：以整数形式

chr () 数字--字符

ord() 字符--数字

切片str[s:e:step]

函数

子主题

str.split()

输出一个列表

str.join()

join() 方法用于将序列中的元素以指定的字符连接生成一个新的字符串

```
str = "-";
seq = ("a", "b", "c"); # 字符串序列
print str.join( seq );    a-b-c
```

下标: find/index

find(要找,始, 终) 输出下标, 不存在-1

index 输出下标, 不存在报错

替换: str.replace(要替换, 被替换)

{:.nf}.formate(a)

排序: 大到小 sorted(str)

返回出一个排序了的列表，但是字符串 元组本身不发生改变

比较

比较 ascii 码

单个字符 直接比较

字符串 对应位置上一 一相比，出现第一个不一样的比较两个字符的 ascii 码

若长度不同：若前几个字符都相同，长的大，若不同，比较第一个不同

Str1="1234" str2="133" str3='123' str2>str1 str1>str3

列表

比较

下标: l.index ()

排序

l.sort()--从大到小

l.reserved()---反序

添加

l.append()

末尾添加一个元素

l.extend()

对象为列表，将列表中所有元素添加

删除

l.insert(位置, 元素)-----插入

l.pop(位置)

l.remove(元素) 删第一个

深浅拷贝

针对到列表里的复杂对象（列表中有元素是列表）

浅拷贝只拷贝地址改变原列表中复杂对象新列表跟着一起改变，深拷贝连着数值一起拷贝

元组

创建（元素，）逗号是关键

例：（1，）

函数

类型函数

tuple(list) 元组--list

tuple((1:2,3:4)) #针对字典 会返回字典的key组成的tuple

list(tumple) list---元组

sorted ()

slicing and indexing
in/not in operators
len() function
iterable(可迭代的)
+ and *