

Pyt基础DAY07

一、数据类型的补充

- int:补充方法:
 - bit_length():查询占的位数
- 字符串:
 - capitalize():首字母大写
 - title():每个单词的首字母大写
 - swapcase():交换大小写,即原来大写变小写,小写变大写
 - index():根据内容查找索引,找不到会报错
 - find():根据内容查找索引,找不到会返回-1
 - center(参数1,参数2):填充.参数1表示总长度为多少,参数2表示如果占不满则用该字符填充
 - join():拼接

```
s = "ABcd ancF"  
a = ["ab", "cd", "ss"]  
print("_".join(s))  
print("_".join(a))
```

- format():格式化,使用方法:

```
s1 = "ads {} dad {} adsa {}"  
print(s1.format("bb", "cc", "dd"))
```

```
s1 = "ads {2} dad {1} adsa {0}"  
print(s1.format("11", "22", "#3"))
```

```
s1 = "ads {b} dad {a} adsa {c}"  
print(s1.format(a="aa", b="bb", c="cc"))
```

- 布尔值
 - 若boo()中的参数为其他类型的值,只要不是空就是True,否则就是False

- 列表

- 反转:reverse:原地修改,若用切片步长设-1的方式则要开辟新空间
- 排序:sort():默认按照升序方式排序,若想降序排列则加参数 reverse=True
- 列表进行乘法时共享元素
- 面试题:将两个列表合并a = [1,2,3,[5]] b = [6,7]
 - 方法一:a = a.extend(b) 原地修改
 - 方法二:a = a + b 新开辟空间
- 面试题:

The screenshot shows the PyCharm IDE interface. On the left, the Project Explorer displays a file structure with 's25homework' as the root, containing 'day06' and 'day07.py'. The main editor window shows the code in 'day07.py' with line numbers 24 to 29. The code is as follows:

```
24
25
26 lst = [1, 2, []]
27 lst1 = lst * 3
28 lst[2].append(2)
29 print(lst1)
```

Below the editor, the Run tool window is open, showing the command used to execute the script: `D:\2019python\Scripts\python.exe C:/Users/W0/PycharmProjects/Python26/day07.py`. The output of the script is displayed as: `[1, 2, [2], 1, 2, [2], 1, 2, [2]]`.

- 元组

```

list_p.py
day05
dict.py
homework.py
s25homework.py
day06
day07.py
External Libraries
Scratches and Console

27
28 tu = (2)
29 print(type(tu))
30 tu1 = (2,)
31 print(type(tu1))
32 tu2 = ()
33 print(type(tu2))

day07 x
D:\2019python\Scripts\python.exe C:/Users/W0/PycharmProjects/Python26/day07
<class 'int'>
<class 'tuple'>
<class 'tuple'>
Process finished with exit code 0

```

- 字典

- 字典的几种定义方式:

```

External Libraries
Scratches and Console

38
39 print(dict(k=1, v=2))
40 print(dict([(1, 2), (3, 4), (5, 6)]))

day07 x
D:\2019python\Scripts\python.exe C:/Users/W0/PycharmProjects/Python26/day07.py
{'k': 1, 'v': 2}
{1: 2, 3: 4, 5: 6}

```

- 字典的fromkeys(参数1,参数2):批量创建键值对,参数1要是可迭代数据类型,迭代添加为键,值都是参数2,要注fromkeys方法使用后要给赋值给一个字典,还有就是值是共享的

```

day07.py
External Libraries
Scratches and Console

34 dic = {}
35 dic.fromkeys("abc", 12)
36 print(dic)

day07 x
D:\2019python\Scripts\python.exe C:/Users/W0/PycharmProjects/Python26/day07.py
{}

```

day00

day07.py

External Libraries

Scratches and Console

33

34

35

36

print(type(tu2))

dic = {}

dic = dic.fromkeys("abc",12)

print(dic)

day07 ×

D:\2019python\Scripts\python.exe C:/Users/W0/PycharmProjects/Python26/

{'a': 12, 'b': 12, 'c': 12}

day07.py

External Libraries

Scratches and Console

34

35

36

37

dic = {}

dic = dic.fromkeys("abc",[1,2])

dic["a"] = [2]

print(dic)

day07 ×

D:\2019python\Scripts\python.exe C:/Users/W0/PycharmProjects/Python26/day07.p

{'a': [2], 'b': [1, 2], 'c': [1, 2]}

day06

day07.py

External Libraries

Scratches and Console

33

34

35

36

37

print(type(tu2))

dic = {}

dic = dic.fromkeys("abc",[1,2])

dic["a"].append(3)

print(dic)

day07 ×

D:\2019python\Scripts\python.exe C:/Users/W0/PycharmProjects/Python26/day07.py

{'a': [1, 2, 3], 'b': [1, 2, 3], 'c': [1, 2, 3]}

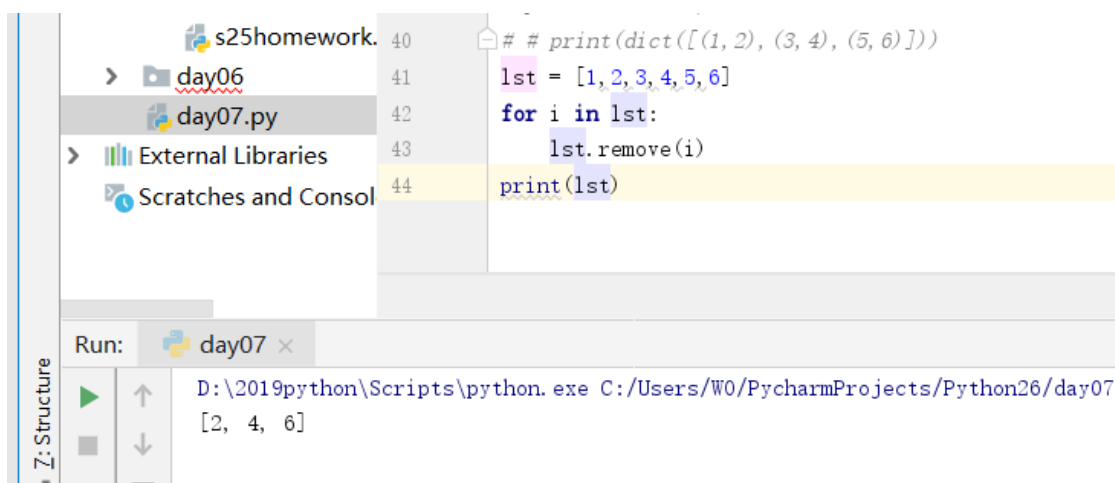
Process finished with exit code 0

- 类型的转换
 - 元组转列表:tuple(list)
 - 列表转元组:list(tuple)
 - 集合转列表:list(set)
 - 列表转集合:set(list)
 - 元组转集合:tuple(set)
 - 集合转元组:set(tuple)
 - 其他类型转成集合类型会出现无序的情况
- 基础类型的总结

	是否有序	是否可变	是否可迭代	取值方式
int	是	否	否	直接取
str	是	否	是	直接取
bool		否	否	直接取
list	是	是	是	通过索引
tuple	是	否	是	通过索引
dict	否	是	是	通过键
set	否	是	是	直接取

二、循环删除的坑

- 由于列表删除后会自动补位,所以如果在for循环中删除列表元素会出现删除不干净的情况,例如下图,第一次循环会删除索引为0的元素,此时原本索引为1的元素会补位,索引变为0,但是下一次的for循环会删除索引为1的元素,所以这个元素就不会被删除



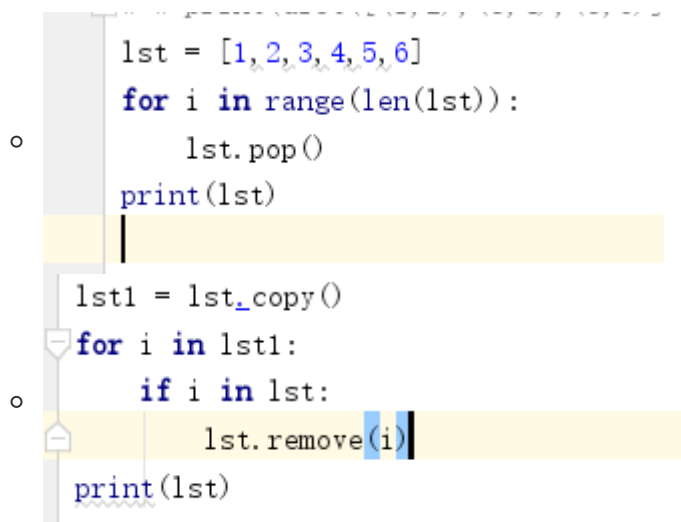
```

s25homework. 40 # # print(dict([(1, 2), (3, 4), (5, 6)]))
> day06 41 lst = [1, 2, 3, 4, 5, 6]
day07.py 42 for i in lst:
43     lst.remove(i)
44     print(lst)

Run: day07 x
D:\2019python\Scripts\python.exe C:/Users/W0/PycharmProjects/Python26/day07.
[2, 4, 6]

```

- 想要通过for删除元素有以下两种方法



```

lst = [1, 2, 3, 4, 5, 6]
for i in range(len(lst)):
    lst.pop()
print(lst)

lst1 = lst.copy()
for i in lst1:
    if i in lst:
        lst.remove(i)
print(lst)

```

- 字典和集合不能循环删除,会报错,要想删除可采取与列表类似的方式
- 字典的大小指字典的长度,即键值对的个数
- 集合的大小指集合的长度,即元素个数

三、二次编码

- encode():编码
- decode():解码
- 编码和解码的集要一直
- python中内存的编码方式是unicode
- 硬盘存储时可选择编码:gbk,utf-8
- 用处:
 - 传输-网络编程
 - 存储-文件操作