

024 str(), [], replace(), 内存分析

6.6 数字转换成字符串型: str()

```
str(3.33)
str(True)
```

6.7 使用[]提取字符

字符串的本质是**字符序列**，用[]可以像序列那样提取单个的成员。（类似于c中的数组，但是操作方法更加多样）

正向搜索：**最左侧**的第一个成员的偏移量是0，到len(str) - 1为止。

反向搜索：**最右侧**的第一个成员的偏移量是-1，到-len(str)为止。

序列	a	b	c
正向编号	0	1	2
反向编号	-3	-2	-1

下面演示了正反编号的使用方法

```
a = 'abcdefghijklmnopqrstuvwxyz'
print("输出a的值\n" + a)
print("a[0] " + a[0])
print("a[3] " + a[3])
length = len(a)
print("a[len(a) - 1] " + a[length - 1])
print("a[-1] " + a[-1])
print("a[-len(a)] " + a[-length])
```

'''

输出a的值

abcdefghijklmnopqrstuvwxyz

a[0] a

a[3] d

a[len(a) - 1] z

a[-1] z

a[-len(a)] a

'''

6.8 replace()实现字符串替换

字符串是不可变的。当试图通过[]方法修改字符串中的字符时发现报错了。但是我们有时确实需要修改字符串中的某些值。这时候就用到了.replace(被替换的字符串, 用来替换的字符串)进行修改。

```
a = 'cbadefghijklmnopqrstuvwxyz'
print(a)
```

```
b = a.replace('cba', 'abc')    # 返回一个新的字符串，原字符串不改变
print(b)
```

```
'''
```

```
cbadefghijklmnopqrstuvwxyz
```

```
abcdefghijklmnopqrstuvwxyz
```

```
'''
```

解释：

a -> 'cbadefghijklmnopqrstuvwxyz'

a. replace('cba', 'abc') 创建了新的对象 'abcdefghijklmnopqrstuvwxyz'

b = a. replace('cba', 'abc')

所以b -> 'abcdefghijklmnopqrstuvwxyz'