



科学编程基础

3. 列表和字典

余恒

北师大天文系

列表（list）

- 成员对象灵活

- `a = [99, "bottles of beer", ["on", "the", "wall"]]`

- 运算符与字符串相同

- `a+b`, `a*3`, `a[0]`, `a[-1]`, `a[1:]`, `len(a)`

- 内容修改

- `a[0] = 98`

- `a[1:2] = ["bottles", "of", "beer"]`

- `#=> [98, "bottles", "of", "beer", ["on", "the", "wall"]]`

- `del a[-1]` `# => [98, "bottles", "of", "beer"]`

列表方法

```
>>> a = range(5)           # [0,1,2,3,4]
>>> a.append(5)             # [0,1,2,3,4,5]
>>> a.pop()                 # [0,1,2,3,4]
5
>>> a.insert(0, 42)         # [42,0,1,2,3,4]
>>> a.pop(0)                # [0,1,2,3,4]
42
>>> a.sort()                # [0,1,2,3,4]
>>> a.reverse()             # [4,3,2,1,0]
```

- 字符串排序使用字典序,逐位比较字母

列表函数

- `range()` 生成数组列表
 - `>>> range(10)`
 - `[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]`
 - `>>> range(2, 30, 5)`
 - `[2, 7, 12, 17, 22, 27]`
- `sum()` 求和
- `min()` 返回列表中最小元素
- `max()` 返回列表中最大元素

字典 (dictionary)

- 字典是一个用大括号括起来的键值对，字典元素分为两部份，键(key)和值(value)。
 - `d = {"duck": "animal", "water": "element"}`
- 查找：
 - `d["duck"] -> "animal"`
 - `d["back"] # raises KeyError exception`
- 删除、插入、修改：
 - `del d["water"] # {"duck": "animal "}`
 - `d["back"] = "rug" # {"duck": "animal ", "back": "rug"}`
 - `d["duck"] = "duik" # {"duck": "duik ", "back": "rug"}`

字典构造

- `sub = {'zhao':1, 'li':2, 'qian':3}`
- `print(sub)`
- `sub = dict(zhao = 1, li = 2, qian = 3)`
- `print(sub)`
- `keys = ['zhao', 'li', 'qian', 'sun']`
- `vals = [1,2,3,4]`
- `sub = dict(zip(keys,vals))`
- `print(sub)`

字典方法

- Keys, values, items:
 - `d.keys()` -> `["duck", "back"]`
 - `d.values()` -> `["duik", "rug"]`
 - `d.items()` -> `[("duck", "duik"), ("back", "rug")]`
- Presence check:
 - `d.has_key("duck")` -> `1`; `d.has_key("spam")` -> `0`
- Values of any type; keys almost any
 - `{"name": "Guido", "age": 43, ("hello", "world"): 1, 42: "yes", "flag": ["red", "white", "blue"]}`

字典遍历

```
d = dict(a=12, b="abc", c=15)
print(d)
for item in d.items():
    print(item)
for key in d:
    print(key, d[key])
for value in d.values():
    print(value)
```


其他类型

- 元组：内容不可变的列表
 - `T=('spam', 3.0, [11, 22, 33])`
 - `print(T[1])`
 - `print(T[2][1])`
 - `T[2]=5`
- 集合：无序不重复元素集合
 - `x=set('abcdeab')`
 - `y=set('cdfxyc')`
 - `print(x|y)`
 - `print(x-y)`
 - `print(x&y)`

随堂练习

- 已知 `S = "s,pa,m,@,bnu,dot,edu,dot,cn"`
- 将其转换为邮件地址
- 提取出用户名和邮件域名
- `n = S.replace(',', '').replace('dot', '.')`
- `name = n[:4]`
- `domain = n[5:]`
- `new = n.split('@')`

作业3： 文本解密

- 完成网站 <http://202.112.85.96/wiki> 中 python练习1:
- 已知 $b \rightarrow e$, $k \rightarrow n$, 试破译下面的句子。
- Vwxglhv vhuyh iru gholjkw, iru ruqdphqw, dqg iru delolwb. Wkhlu fklhi xvh iru gholjkw, lv lq sulydwhqhvv dqg uhwlulqj; iru ruqdphqw, lv lq glvfrxuvh; dqg iru delolwb, lv lq wkh mxgjphqw dqg glvsrvlwlrq ri exvlqhvv.
- 函数提示: chr() ord()

ASCII Code Chart

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL

ASCII（American Standard Code for Information Interchange，美国信息互换标准代码）是基于拉丁字母的一套电脑编码系统，现今最通用的单字节编码系统。它是由美国国家标准学会(American National Standard Institute , ANSI)于1967年制定的。0~31及127(共33个)是控制字符或通信专用字符，其余为可显示字符。

交互读取input

```
>>> raw_input_A = raw_input("RInput: ")
```

```
RInput: abc
```

```
>>> input_A = input("Input: ")
```

```
Input: "abc"
```

```
>>> raw_input_B = raw_input("RInput: ")
```

```
RInput: 123
```

```
>>> type(raw_input_B)
```

```
<type 'str'>
```

```
>>> input_B = input("Input: ")
```

```
Input: 123
```

```
>>> type(input_B)
```

```
<type 'int'>
```

输出print

```
>>> a = 3.1415926
```

```
>>> print("%d"%a)          #整数
```

```
3
```

```
>>> print("%4d"%a)        #指定长度
```

```
3
```

```
>>> print("%f,%3.2f" %(a,a)) #小数
```

```
3.141593, 3.14
```

```
>>> print("%.9f,%e" %(a,a)) #更多小数位
```

```
3.141592600, 3.141593e+00
```

文本读取

- `file`对象是和`string`、`float`一样的内置对象类型，可由`open`命令创建
- `f = open("filename", "r")`
- `r` 只读、`w` 写、`rw` 读写，`rb`二进制读
- 文件对象提供了三个"读"方法：
 - `.read()`读取整个文件，将内容存入字符串变量
 - `.readline()` 每次只读取一行
 - `.readlines()`将文件内容分析成一个行的列表

文本读取2

- 完整版：
 - `f=open('filename','r')`
 - `text=f.read()`
 - `f.close()`
- 简略版：
 - `text=open('filename','r').readlines()`
 - `text=open('filename','r').read().split('\n')`

文本写入

- `mydata = ['Hello World!', 'BNU']`
- `myfile = open('test.txt', 'w')`
- `for line in mydata:`
- `myfile.write(line + '\n')`
- `myfile.close()`
- `myfile = open("test.txt")`
- `myfile.read()`

随堂练习

- 将100以内的奇数保存为文本文件

```
fw = open('nums.txt', 'w')
for i in range(1, 100):
    if i%2 != 0:
        fw.write(str(i) + '\n')
fw.close()
```

从这个文本文件中读取数据，计算这些数的平方，保存为第二列

```
lines = open('nums.txt').readlines()
fn = open('num2.txt', 'w')
for line in lines:
    num = int(line.strip())
    fn.write(line.strip()+" "+str(num**2)+"\n")
fn.close()
```

作业：输出15行杨辉三角

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

.....

- 又称贾宪三角形，帕斯卡三角形，是二项式系数的一种空间排列形式。