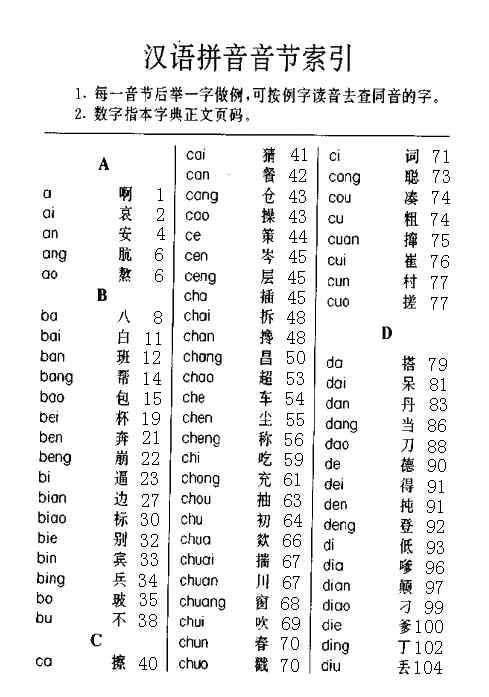
1. **Python数据类型-Dictionary（字典）**

**第一部分**

1. **内容的映入**
2. 字典的索引页



2、如何记载野餐朋友带的食物数量，包括朋友的名字，每个朋友所带食物的名字，每种食物的个数?

dict = {'小明':{'红烧肉':6,'大闸蟹':8,'水煮鱼':1},'小花':{'黄瓜':9,'番茄':2},'小六子':{'啤酒':30,'可乐':40}}

3、比如你要记录一次会议参加的人员的基本信息,大家的信息都不一样时,:dict = {'男性':{'张博':{'年龄':30,'职业':'老师','级别':'高级教师'},'王强':{'手机':'1309999999'}},'女性':{'小花':{'爱好':'化妆','职业':'美容师'}}}

**二、字典的定义和创建**

字典dictionary是Python中另一个非常有用的内置数据类型,是由若干对无序键值对组成，字典的每个键值(key=>value)对用冒号(:)分割，每对之间用逗号(,)分割，整个字典包括在花括号({})中,格式如下所示：

d = {key1 : value1, key2 : value2 }

>>> dict1={"one":"北京","two":"上海"}

需要注意的是，创建空字典用{ }。

**三、字典键的特性**

字典是另一种可变容器模型，且可存储任意类型对象，键必须是唯一的，但值则不必，值可以取任何数据类型，但键必须是不可变的，如字符串，数字或元组。

# 键是唯一的，如果出现相同的键，前一个键所对应的值被后一个覆盖

>>> dict1={"one":"北京","two":"上海","one":"深圳"}

>>> dict1

{'one': '深圳', 'two': '上海'}

# 键是字母

>>> dict1={"one":"北京","two":"上海"}

#字典嵌套

>>>dict3={1:{"one":"北京","two":"上海"},2:{"three":"深圳","four":"珠海"}}

# 键是字母

>>> dict2={1:["ascd","we23"],2:["we23ascd","21we23"]}

# 键是元祖

>>>dict4={("1",):{"one":"北京","two":"上海"},("2",):{"three":"深圳","four":"珠海"}}

#键不可以是list，因为它是可变的

>>> dict4={["a"]:{"one":"北京","two":"上海"}}

Traceback (most recent call last):

File "<pyshell#45>", line 1, in <module>

dict4={["a"]:{"one":"北京","two":"上海"}}

TypeError: unhashable type: 'list'

#键不可以是字典，因为它是可变的

>>> dict4={{"one":"北京","two":"上海"}:"a"}

Traceback (most recent call last):

File "<pyshell#46>", line 1, in <module>

dict4={{"one":"北京","two":"上海"}:"a"}

TypeError: unhashable type: 'dict'

**四、字典操作**

1、访问字典里的值

字典中的值是通过访问字典的键获取的，把相应的键放入熟悉的方括弧，语法如下：

value=字典[key],例如：

>>> dict = {'Name': 'Runoob', 'Age': 7, 'Class': 'First'}

>>> dict['Age']

7

如果去访问一个不存在的键，会报错。

>>> dict['we23']

Traceback (most recent call last):

File "<pyshell#51>", line 1, in <module>

dict['we23']

KeyError: 'we23'

2、在字典中添加一对键值对

语法：字典[key]=value

其中一定要注意,key不能和已经存在的键相同，否则就变成修改数据了。

>>> dict4={"key1":"value1","key2":"value2","key3":"value3","key4":"value4"}

>>> dict4

{'key1': 'value1', 'key2': 'value2', 'key3': 'value3', 'key4': 'value4'}

>>> dict4["key5"]="value5"

>>> dict4

{'key1': 'value1', 'key2': 'value2', 'key3': 'value3', 'key4': 'value4', 'key5': 'value5'}

3、在字典中修改值

语法：字典[key]=value

其中一定要注意，key在字典中存在，这个key对应的值重新被赋值。

>>> dict4={"key1":"value1","key2":"value2","key3":"value3","key4":"value4"}

>>> dict4["key1"]="value5";

>>> dict4

{'key1': 'value5', 'key2': 'value2', 'key3': 'value3', 'key4': 'value4'}

**第二部分**

4、删除字典元素

1、删除字典中某对键值对

语法：del 字典[key]

>>> dict={"one":1,"two":2,"three":3}

>>> del dict["one"]

>>> dict

{'two': 2, 'three': 3}

方法二：字典.pop(key)

>>> dict1.pop("one")

'深圳'

>>> dict1

{'two': '上海'}

>>> dict1.pop("three")

Traceback (most recent call last):

File "<pyshell#29>", line 1, in <module>

dict1.pop("three")

KeyError: 'three'

方法三：[popitem()](http://www.runoob.com/python3/python3-att-dictionary-popitem.html)：随机返回并删除字典中的一对键和值(一般删除末尾对)。

>>> dict1={"one":"北京","two":"上海","three":"深圳"}

>>> dict1

{'one': '北京', 'two': '上海', 'three': '深圳'}

>>> dict1.popitem()

('three', '深圳')

>>> dict1

{'one': '北京', 'two': '上海'}

>>> dict1.popitem()

('two', '上海')

2、清空字典

语法：字典对象.clear()

>>> dict={"one":1,"two":2,"three":3}

>>> dict.clear()

>>> dict

{}

3、删除字典

语法：del 字典

>>> dict={"one":1,"two":2,"three":3}

>>> del dict

>>> dict

<class 'dict'>

>>> dict['one']

Traceback (most recent call last):

File "<pyshell#8>", line 1, in <module>

dict['one']

TypeError: 'type' object is not subscriptable

4、字典合并

把字典dict2合并到字典dict1中：

>>> dict1.update(dict2)

>>> dict1

{'one': '北京', 'two': '上海', 'three': '深圳', 'four': '珠海'}

**五、字典内置函数**

1、len()：计算字典中的键值对的个数

>>> dict={"one":1,"two":2,"three":3}

>>> len(dict)

3

2、str(dict)：把字典转化为字符串

>>> dict1={"one":"北京","two":"上海","one":"深圳"}

>>> str(dict1)

"{'one': '深圳', 'two': '上海'}"

3、[copy()](http://www.runoob.com/python3/python3-att-dictionary-copy.html)：拷贝一个字典

>>> dict2=dict1.copy()

>>> dict2

{'one': '深圳', 'two': '上海'}

4、[get(key, default=None)](http://www.runoob.com/python3/python3-att-dictionary-get.html):返回指定键的对应的值，如果值不在字典中返回default值

>>> dict1.get("one")

'深圳'

5、[key in dict](http://www.runoob.com/python3/python3-att-dictionary-in.html):如果键在字典dict里返回true，否则返回false

>>> "one" in dict1

True

>>> "three" in dict1

False

6、[keys()](http://www.runoob.com/python3/python3-att-dictionary-keys.html)：以列表返回一个字典所有的键

>>> len(dict1.keys())

2

>>> "one" in dict1.keys()

True

7、values[()](http://www.runoob.com/python3/python3-att-dictionary-keys.html)：以列表返回一个字典所有的值

>>> len(dict1.values())

2

>>> "北京" in dict1.values()

False

>>> "上海" in dict1.values()

True

>>> dict1.values()

dict\_values(['深圳', '上海'])