

# PYTHON BASICS

## 列表解析式

2020-05-05

STEVEN WANG

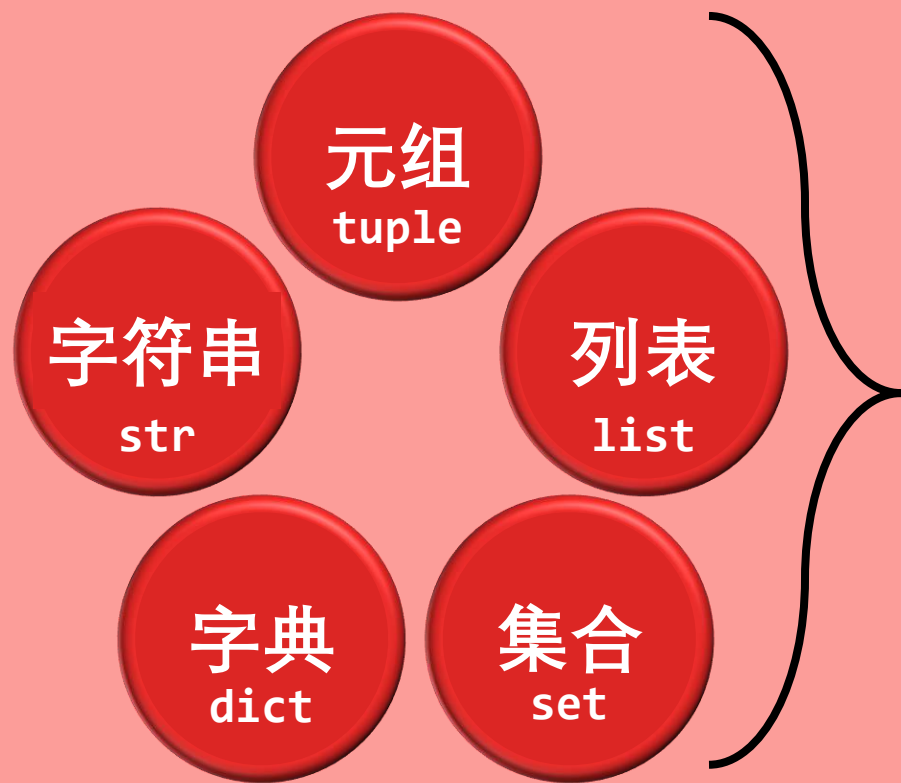




# 上节总结

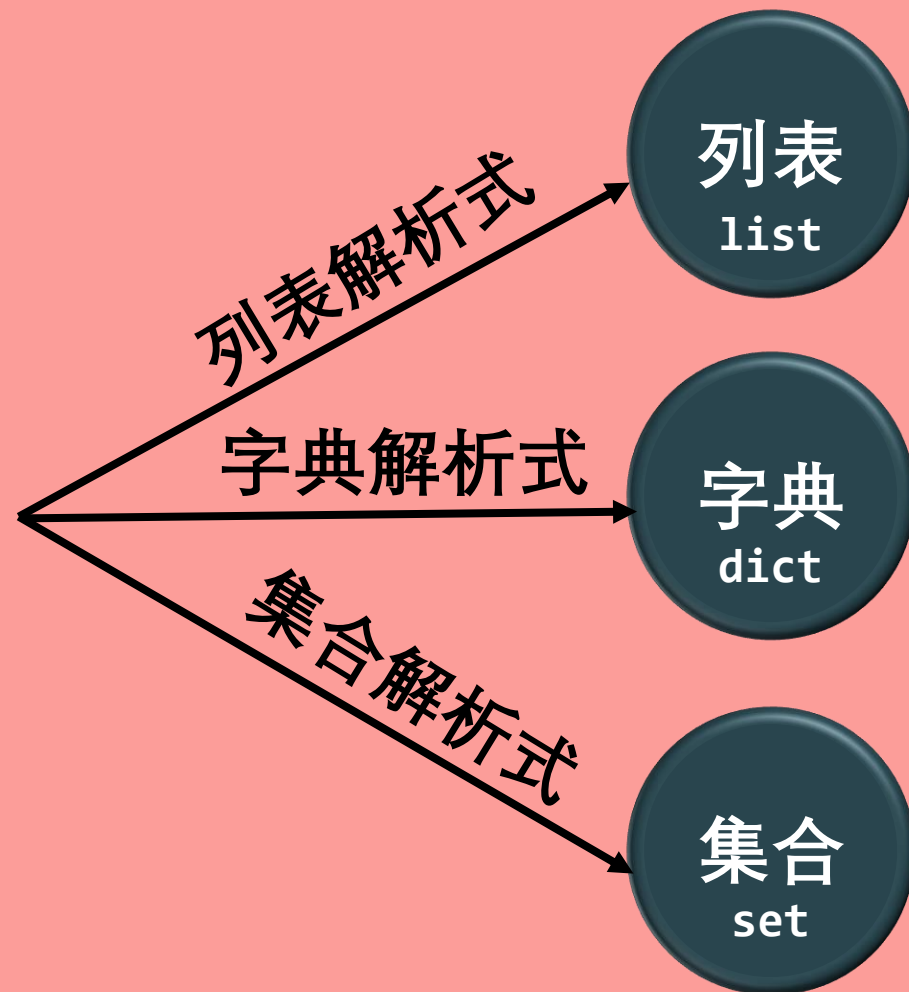
内容	核心知识点	用处
格式化	% 字符, \$ 字符, format 函数, f-string	按 <b>模式</b> 美化
元字符	集合 <b>[]</b> 次数 <b>* + ? {}</b> 并列 <b> </b> 提取 <b>()</b> 转义 <b>. ^ \$ \</b>	创建 <b>模式</b>
正则化	match, search, findall, finditer, split, sub, compile	按 <b>模式</b> 获取

## 可迭代对象

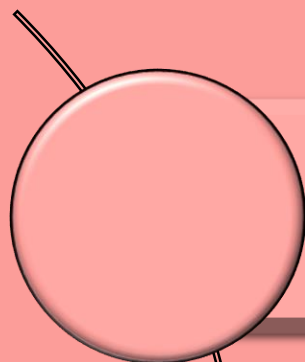


解析

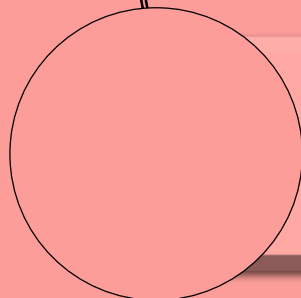
## 可迭代对象



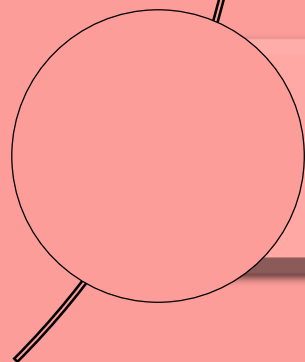
# 解析表达式



列表解析式



字典解析式



集合解析式

```
odds = []  
for n in lst:  
    if n % 2 == 1:  
        odds.append(n*n)
```

For 循环

创建列表

For 循环

If 条件

添加元素

列表解析式

创建列表

```
odds = [n*n for n in lst if n % 2 == 1]
```

添加元素

For 循环

If 条件

```
new = []  
for item in old:  
    if cond(item):  
        new.append(f(item))
```

For 循环

创建列表

For 循环

If 条件

添加元素

列表解析式

创建列表

```
new = [f(item) for item in old if cond(item)]
```

添加元素

For 循环

If 条件

```
flatten = []  
for row in lst:  
    for n in row:  
        flatten.append(n*n)
```

For 循环

创建列表

外 for 循环

内 for 循环

添加元素

列表解析式

创建列表

```
flatten = [n*n for row in lst for n in row]
```

添加元素

外 for 循环

内 for 循环



```
flatten = [n for row in lst for n in row]
```

↑  
外 for 循环

↑  
内 for 循环



```
flatten = [n for n in row for row in lst]
```

↑  
内 for 循环

↑  
外 for 循环





列表解析式

创建列表

```
new = [f(item) for item in old if cond(item)]
```

添加元素

For 循环

If 条件

Map / Filter

映射函数

筛选函数 旧列表

```
new = list(map(map_fun, filter(filter_fun, old)))
```

创建列表



**语法糖** (syntactic sugar) 用更简练的方式来表达含义，相当于汉语里的成语。

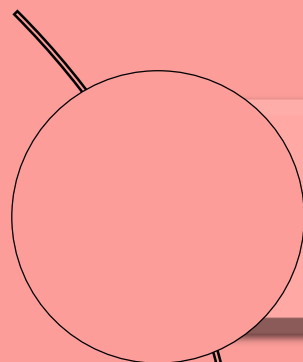


**语法盐** (syntactic salt) 是通过反人性的语法来避免犯错，但降低了编程效率。

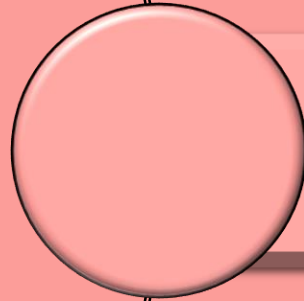


**语法糖浆** (syntactic syrup) 是失败的语法糖，无意义的舔砖加瓦。

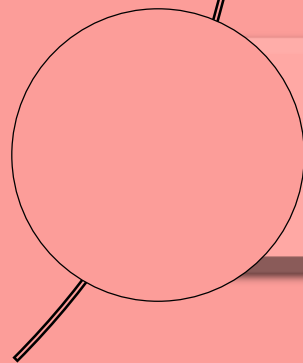
# 解析表达式



列表解析式



字典解析式



集合解析式

```
new = {}
for key, val in items:
    if cond(key, val):
        new[key] = f(val)
```

For 循环

创建字典

For 循环

If 条件

添加元素

字典解析式

创建字典

```
new = {key:f(val) for key,val in items if cond(key,val)}
```

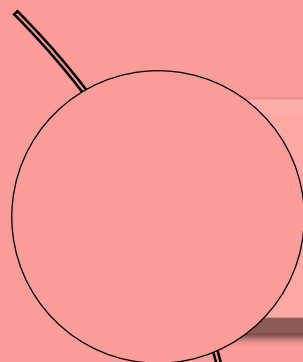
添加元素

For 循环

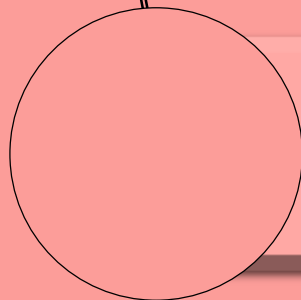
If 条件



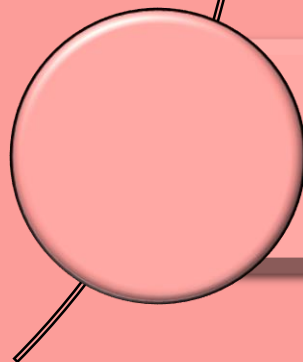
# 解析表达式



列表解析式



字典解析式



集合解析式

```
new = []  
for item in old:  
    if cond(item):  
        new.add(f(item))
```

For 循环

创建集合

For 循环

If 条件

添加元素

集合解析式

创建集合

```
new = {f(item) for item in old if cond(item)}
```

添加元素

For 循环

If 条件

# 总结

类型	含义
列表解析式	<code>[ expr_item for item in collection if cond ]</code>
字典解析式	<code>{key: expr_val for key, val in collection if cond}</code>
集合解析式	<code>{ expr_item for item in collection if cond }</code>
元组解析式	<code>( expr_item for item in collection if cond )</code>



↑  
生成器

## 下节预告：迭代器和生成器



# 终身学习 快乐学习

王圣元      Steven Wang  
微信公众号：王的机器