

Python编程

& 数据科学入门

第 3 课

数据结构、函数、条
件和循环（2）



解密大数据团队出品

◎ 本课内容



数据结构

- 元组
- 字典

01



循环

- while循环
- 循环控制
- 嵌套循环

02



函数

- 如何自定义函数

03



案例讲解

- 工资计算器函数

04

② 数据结构



◎ 数据结构



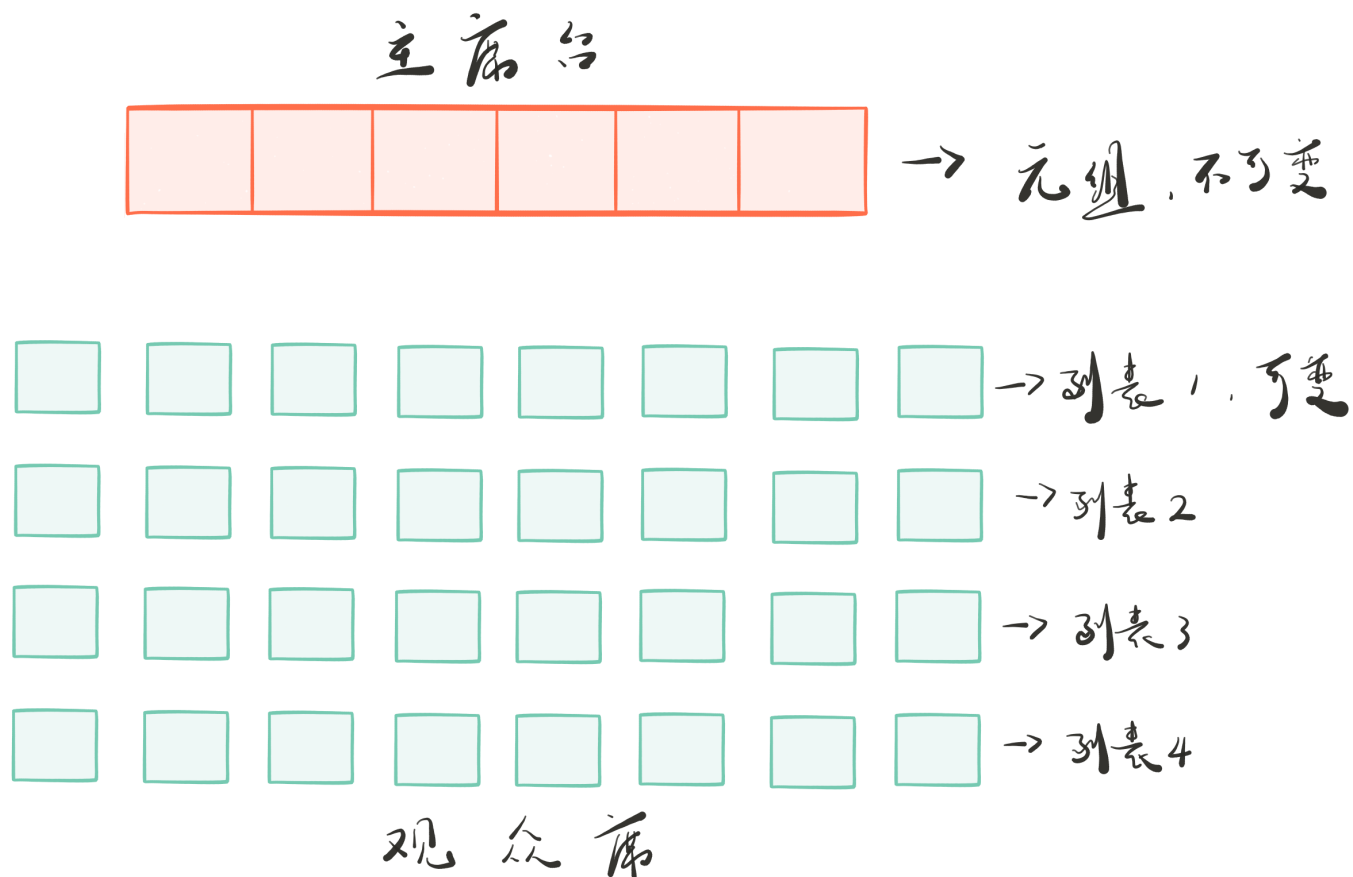
元组 与列表非常相似，是由若干值组成的一个序列。

元组	列表
T = ('A', 'B', 'C', 'D')	L = ['A', 'B', 'C', 'D']
都是序列类型数据	
存储的元素可以是任何数据类型	
都可以使用索引和切片操作	
不可变类型	可变类型
效率比列表更高	/



元组VS列表

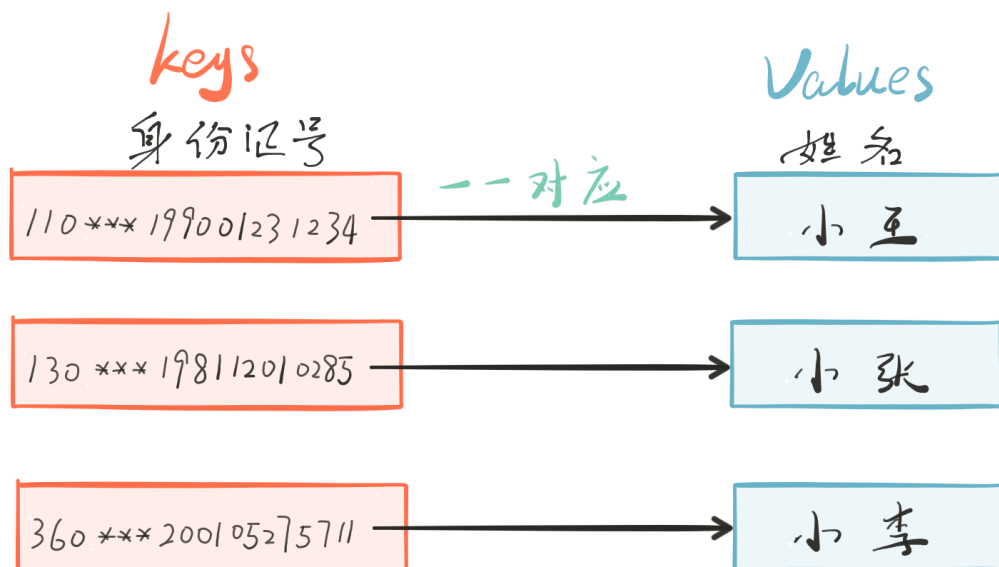
- 列表可变
- 元组不可变



◎ 数据结构



字典可看作是 键(key) 与 值(value) 之间的一系列一一对应关系。



字典：键 - 值对 (key - value pairs)

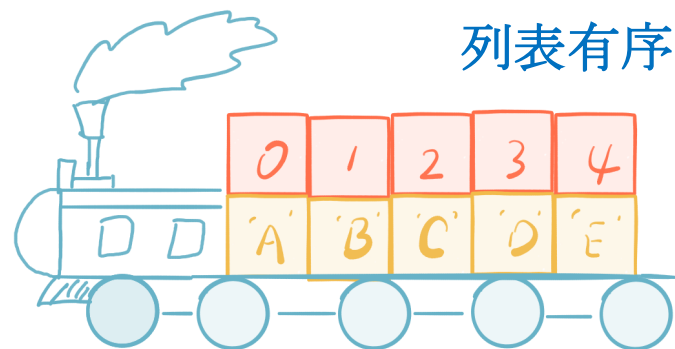
```
{'110***199001231234': '小王',  
'130***198112010285': '小张',  
'360***200105275711': '小李'}
```

◎ 数据结构



字典 VS 列表

- 列表由整数作为索引，字典由key作为索引。
 - 列表中的元素有顺序，按索引0、1、2……这么排列。
 - 字典中的元素没有固定顺序，常用字符串类型做key。
- 列表和字典都是可变类型，可进行增删改操作。
- 都可以通过 [] 获取元素。



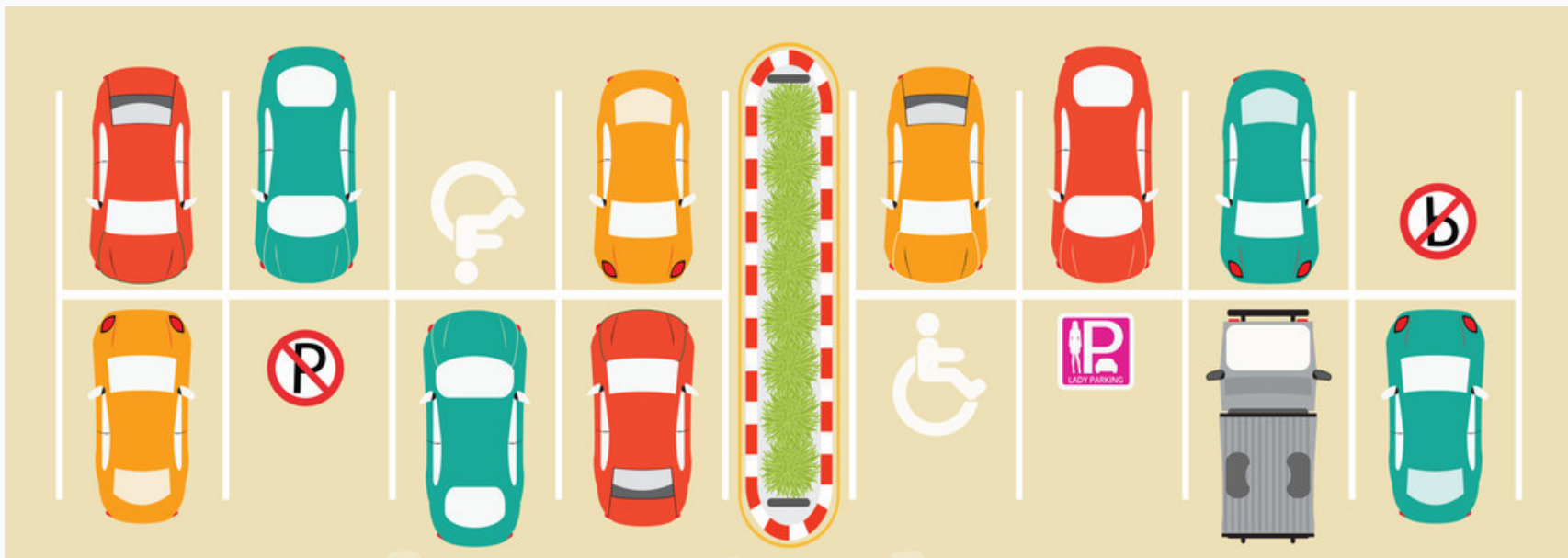
◎ 循环



while 循环

在停车场**找停车位**，假设车位从1开始编号，一直到30号。

待停的车辆从1号开始，逐个查看每个车位是否为空，重复这样的动作，直到找到空车位停车，或者查看完所有的车位后驶出停车场。



◎ 循环



while 循环

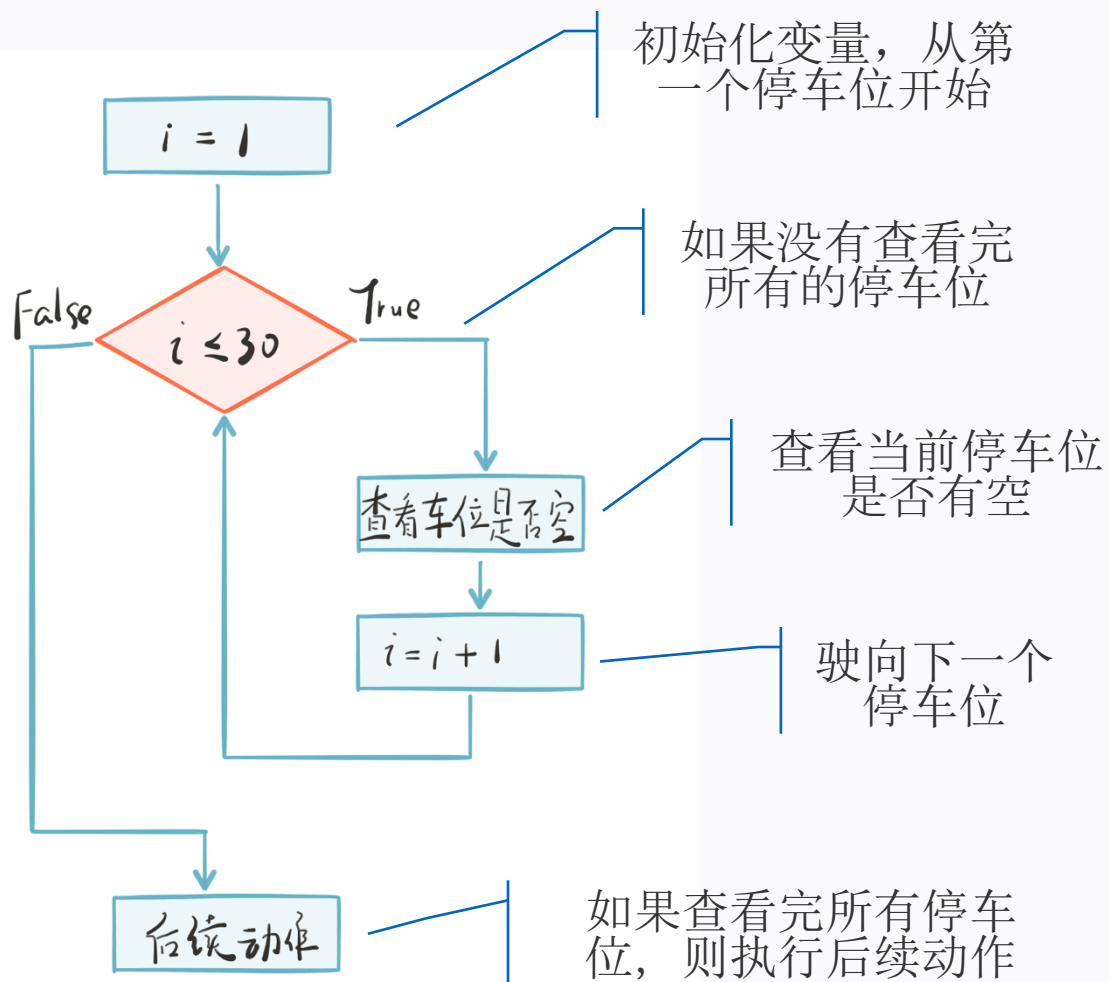
```
i = 1
```

```
while i <= 30 :
```

```
    查看该车位是否为空
```

```
    i = i + 1
```

```
    .....
```



◎ 循环



循环控制: **break**

```
i = 1
```

```
while i <= 30 :
```

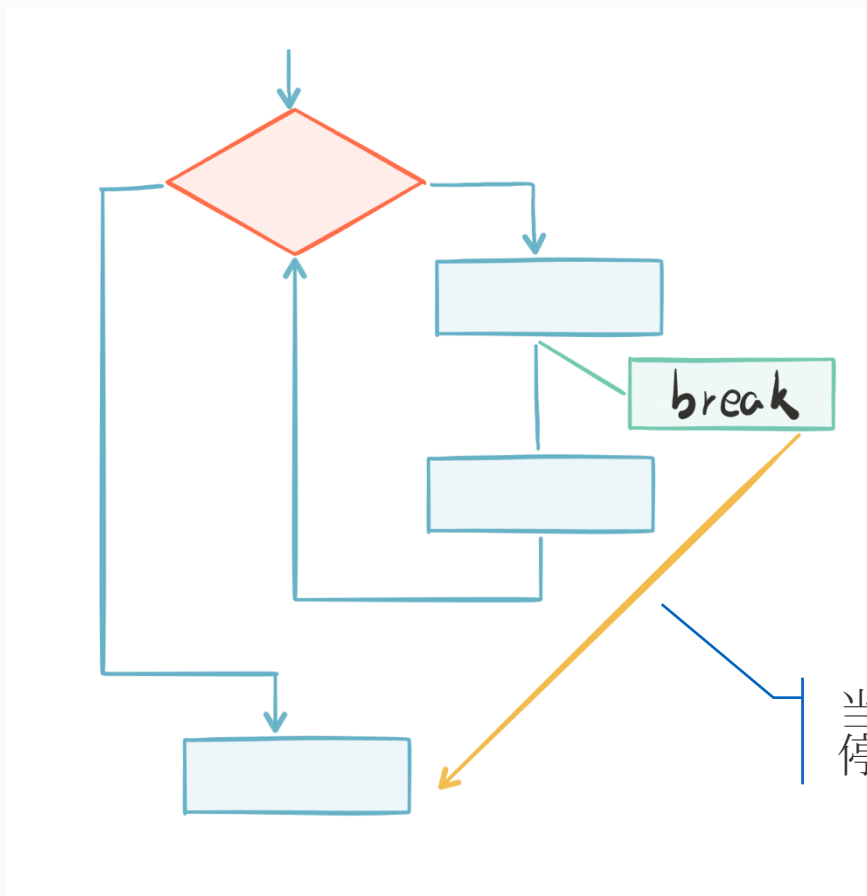
```
    if 车位为空 :
```

```
        停车
```

```
        break
```

```
    i = i + 1
```

```
.....
```



当有停车位时，
停车，跳出循环

◎ 循环



循环控制: **continue**

`i = 1`

while `i <= 30` :

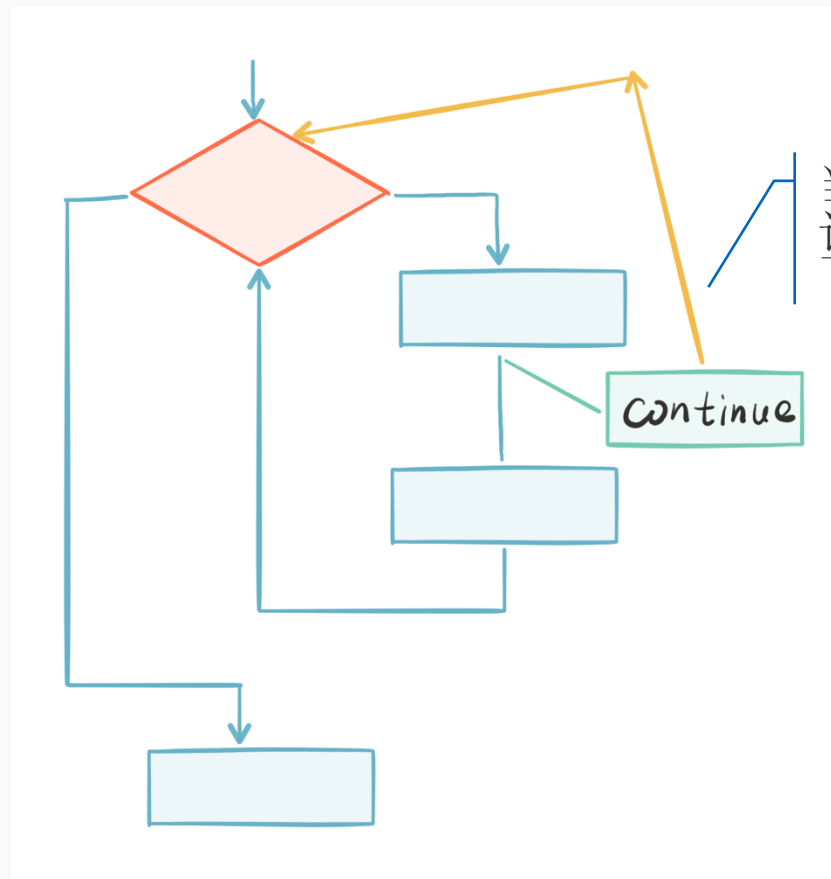
if 车位有🚫标记 :

`i = i + 1`

continue

查看车位是否为空

`i = i + 1`



当有禁停标记时，执行下一轮循环

◎ 循环



嵌套循环



如果是有三层的停车楼，每一层的查找方式同前，只不过需要从第一层开始，逐层查找。



变量 j 代表楼层

$j = 1$

外循环

while $j \leq 3$:

$i = 1$

内循环

while $i \leq 30$:

查看该车位是否为空

$i = i + 1$

$j = j + 1$

.....

自定义函数

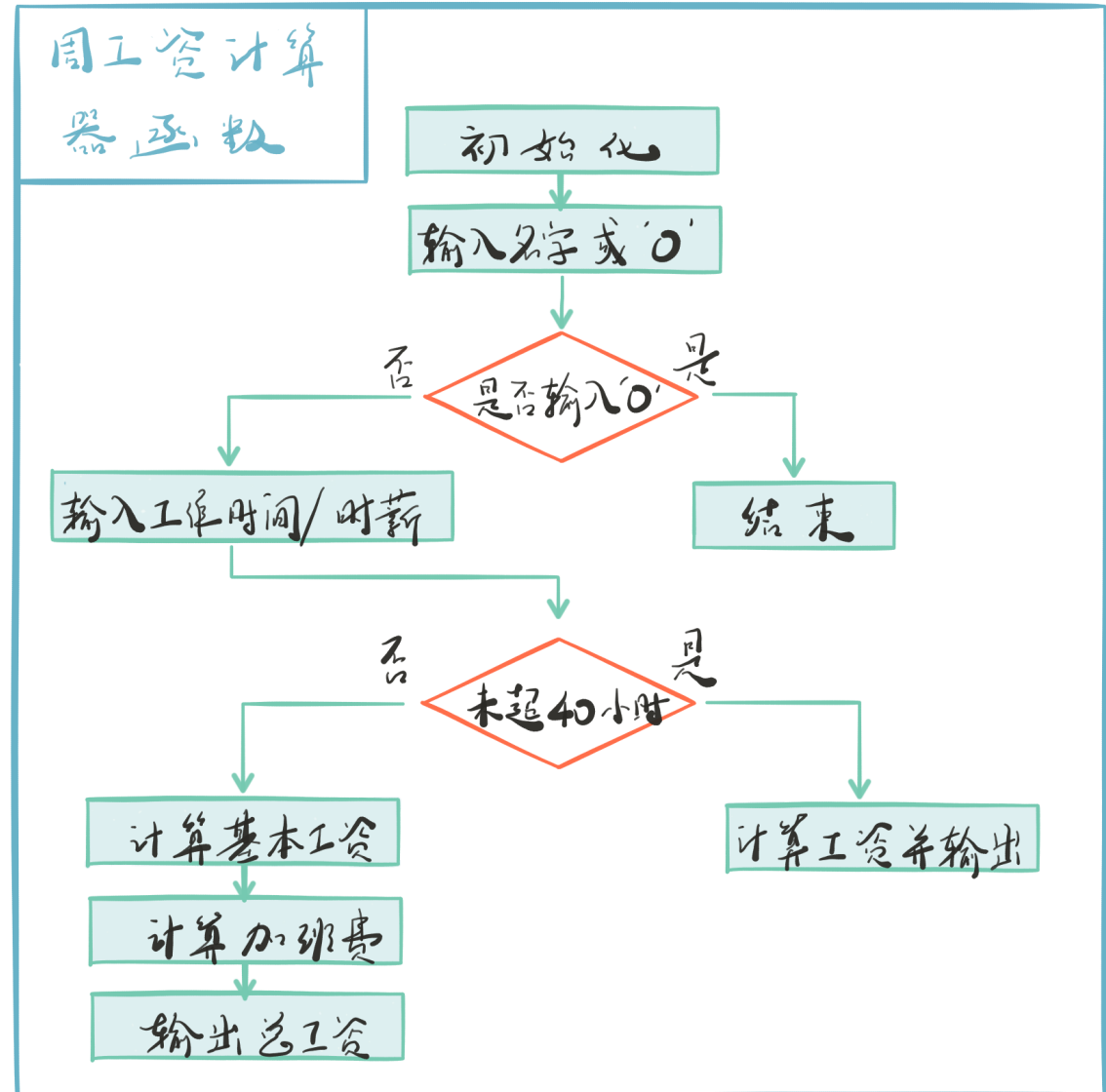




案例讲解

工资计算器函数

- 输入你的姓名，工作时间和单位工资
- 如果一周工作时间超过40小时，加班时间按照常规工资1.5倍计算



答疑时间

