

# 第九章 有益的探索

---

# 本章导读

## 有益的探索

### 数据类型的底层实现

列表：地址引用

字典：稀疏数组

字符串：紧凑数组

讨论 类型是否可变

小例子

### 简洁的语法

解析语法 [expr for value in iterable if condition]

条件表达式 expr1 if condition else expr2

### 三大神器

生成器

生产表达式

生成器函数 yield

迭代器

可迭代对象

哪些是迭代器

装饰器

简单的装饰器

装饰有参函数

带参数的装饰器



# 目录

---

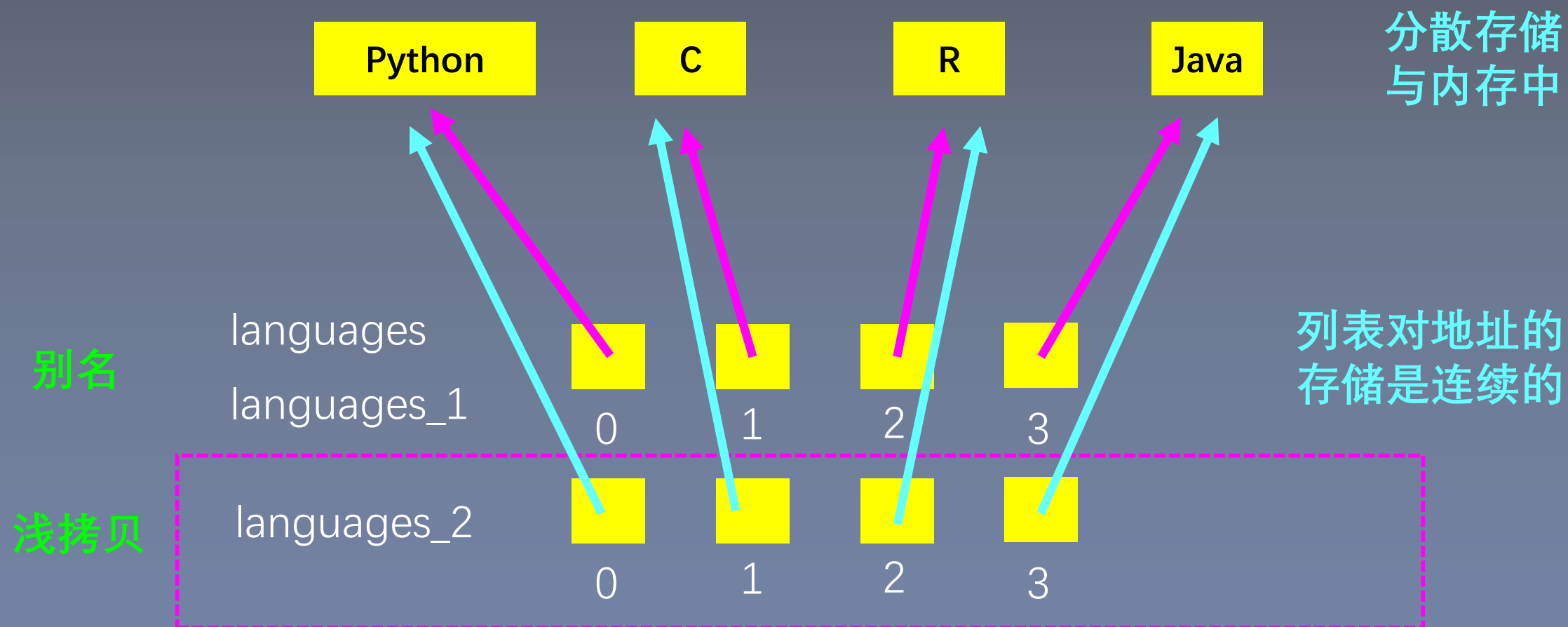
1/ 数据类型底层实现 ( 25 min )

2/ 更简洁的语法( 10 min )

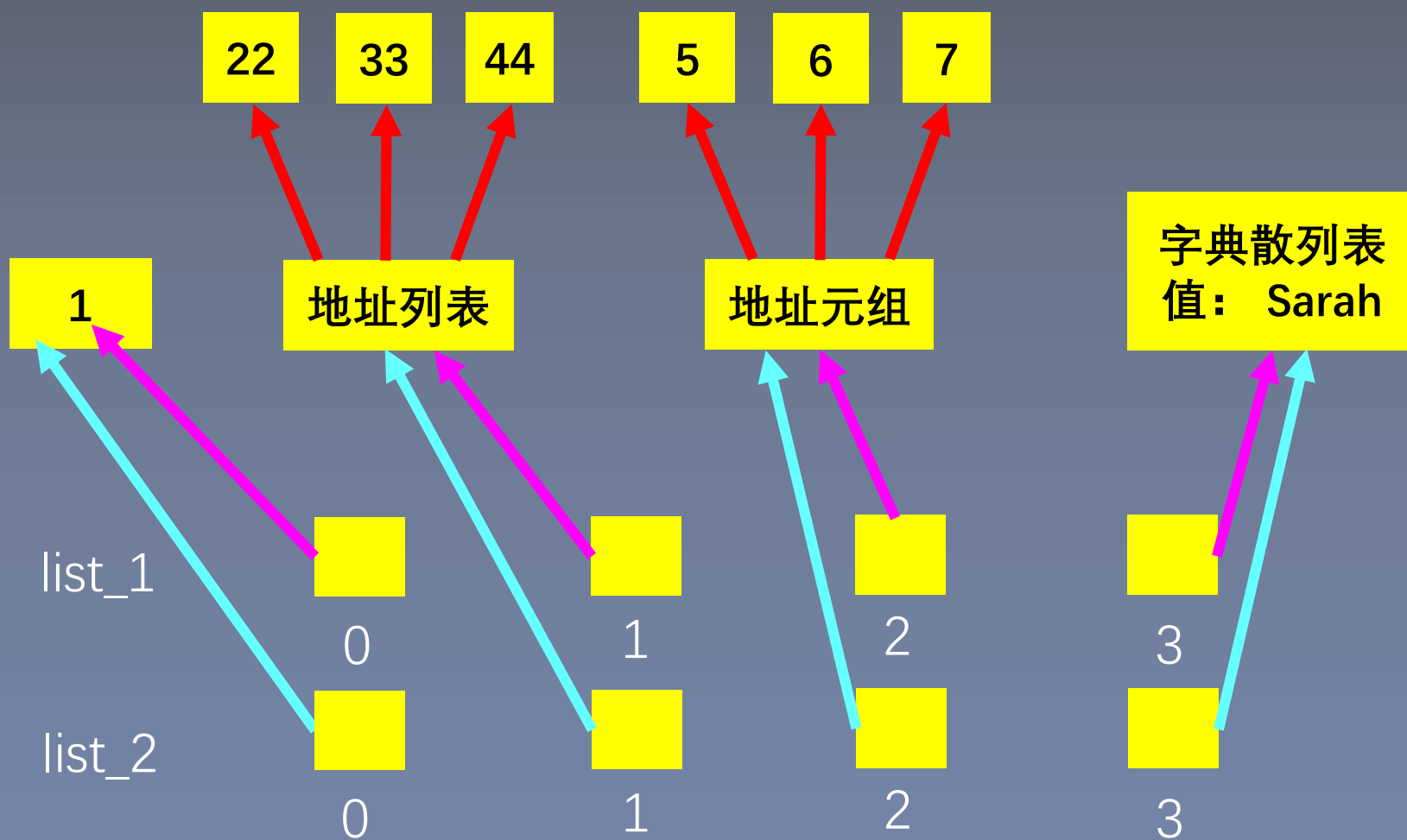
3/ 三大神器 ( 25 min )



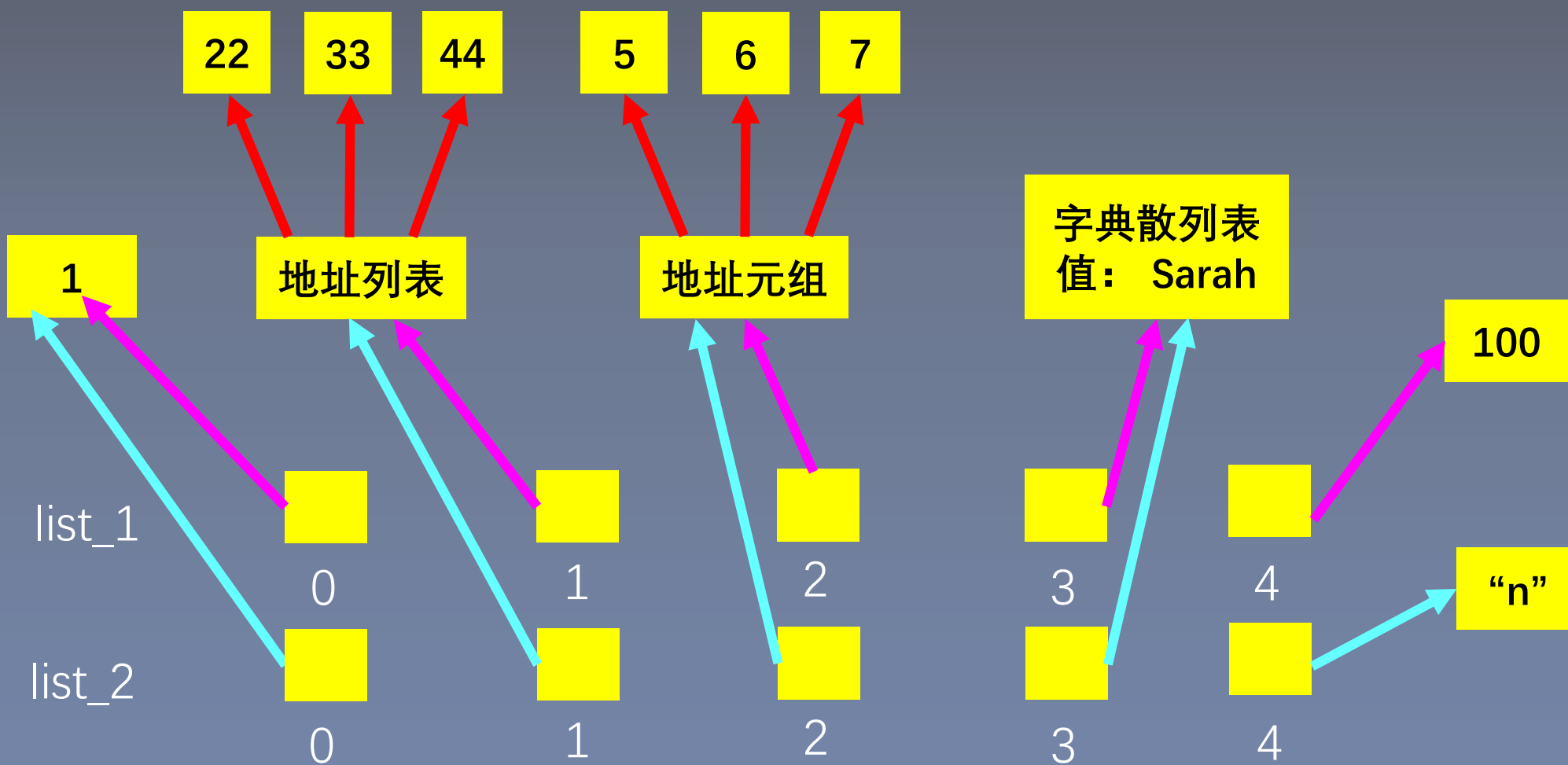
# 引用数组



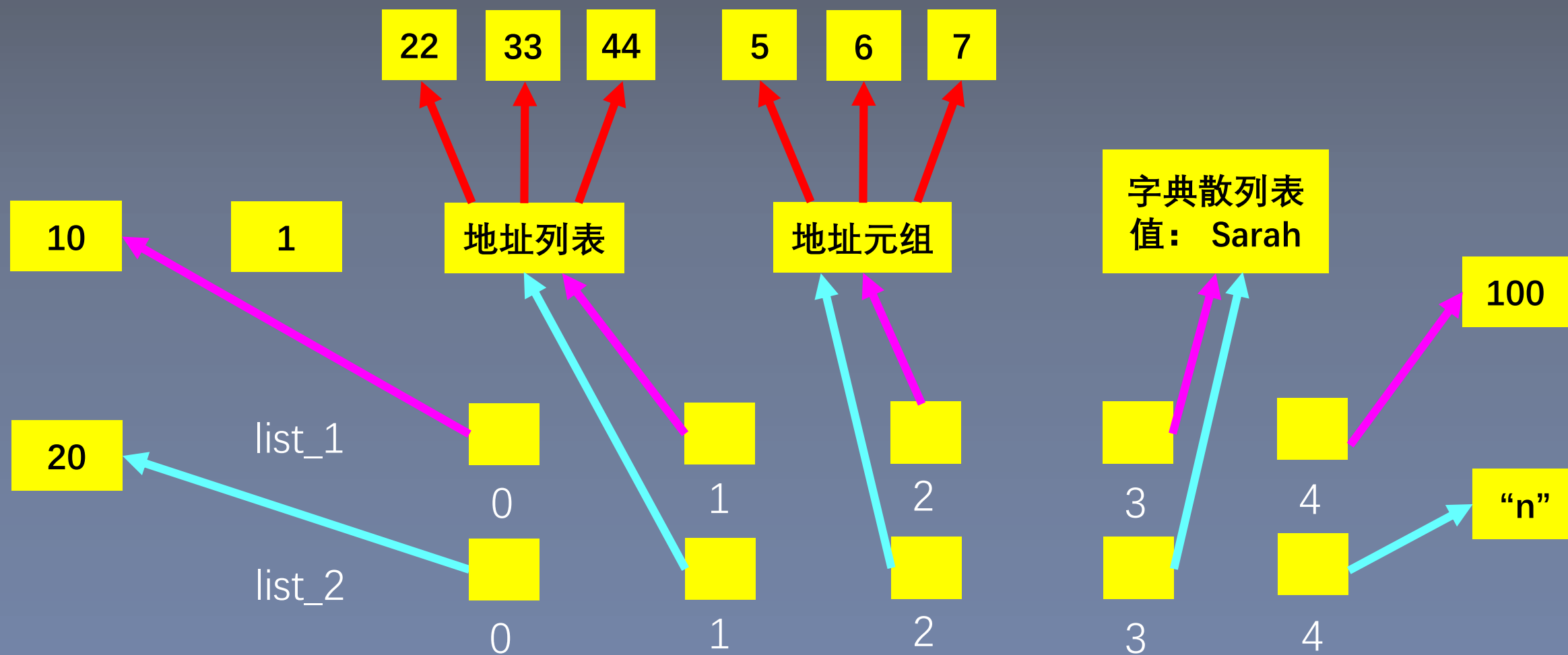
# 奇怪列表的背后



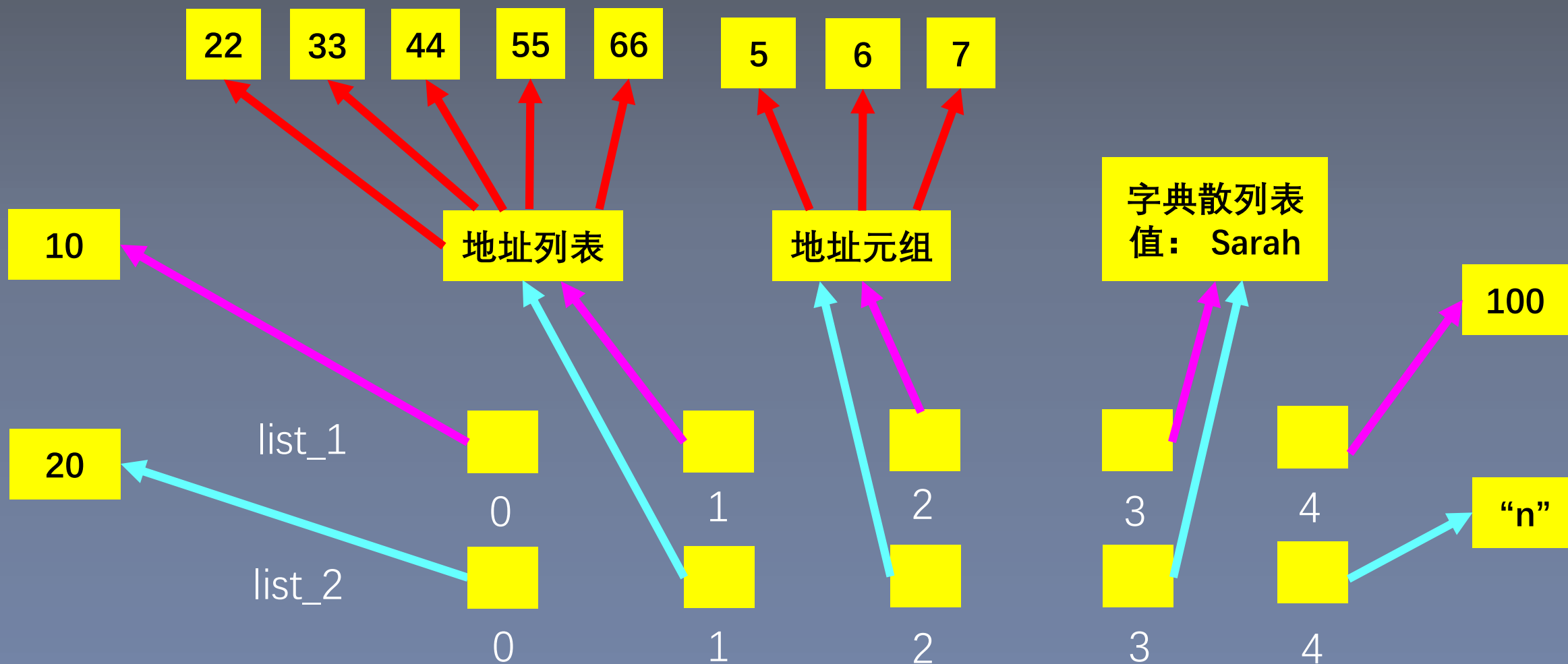
# 奇怪列表的背后



# 奇怪列表的背后

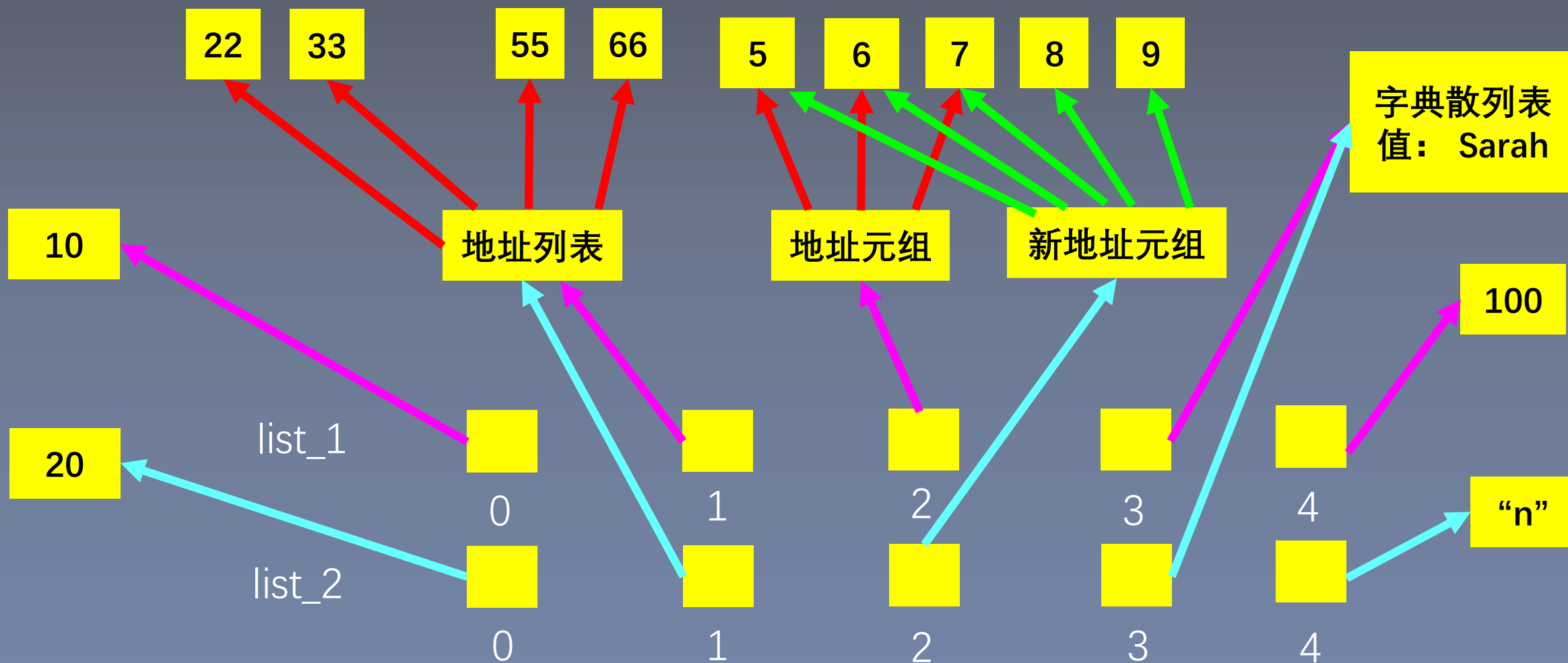


# 奇怪列表的背后

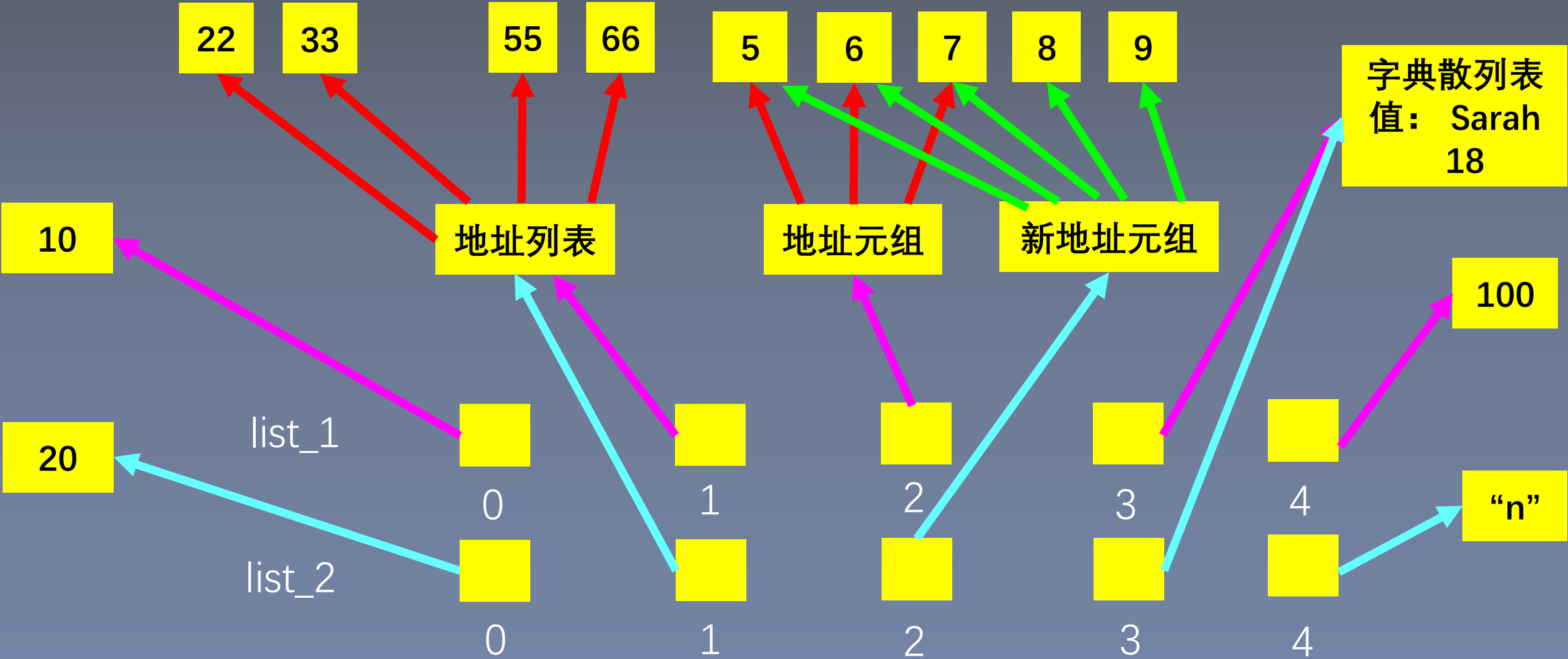




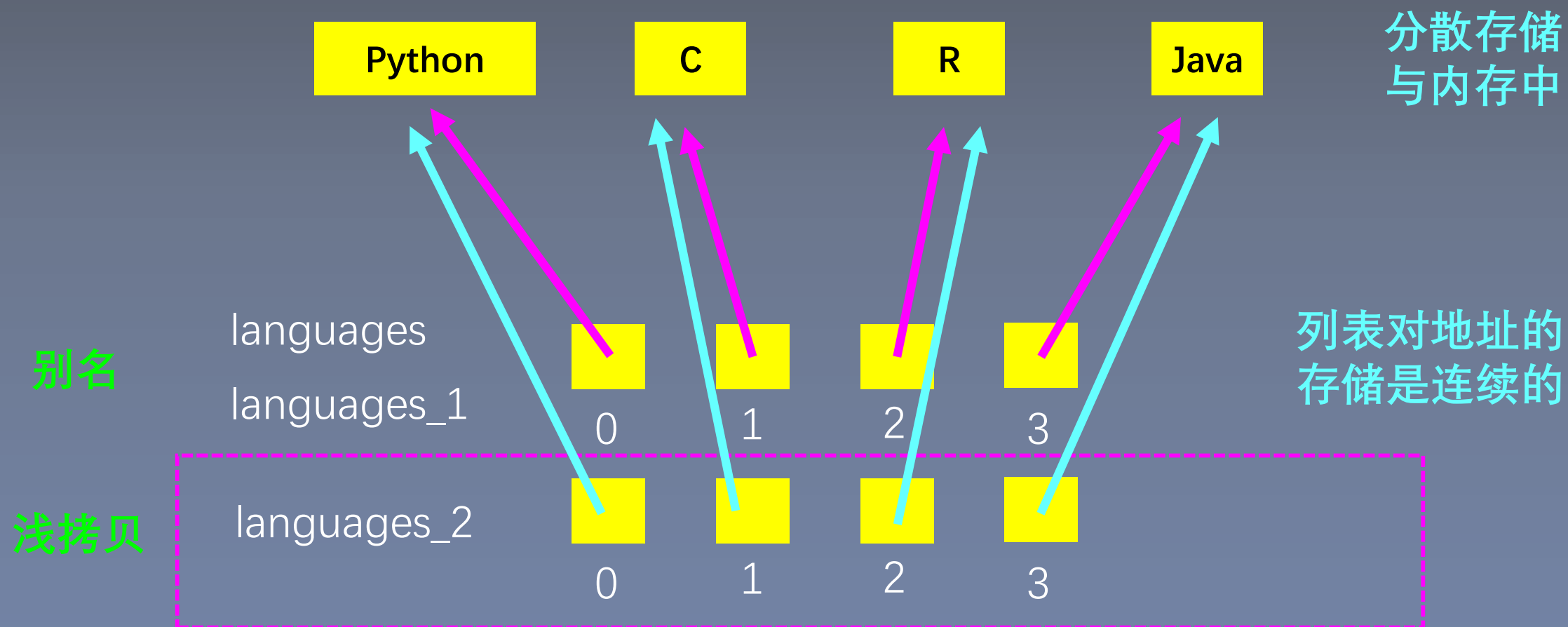
# 奇怪列表的背后



# 奇怪列表的背后



# 引用数组



# 字典

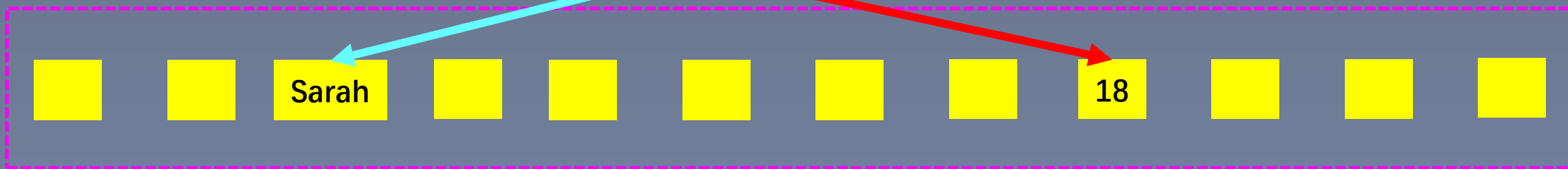
---

`d = {}`

`d["age"] = 18`      `d["name"] = "Sarah"`

`hash("age")`

`hash("name")`



散列表  
稀疏数组



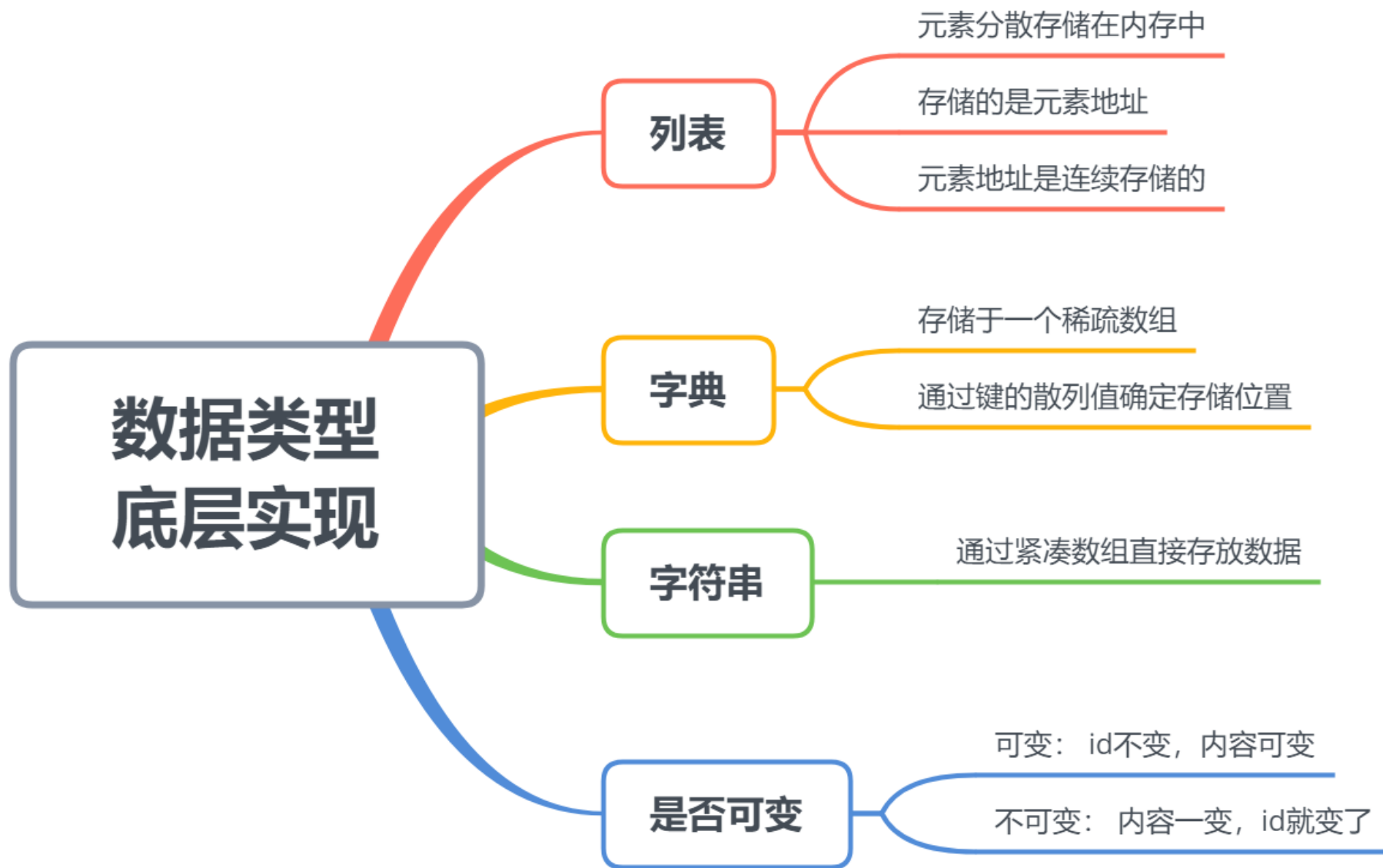
# 字符串

---

s = "Python"

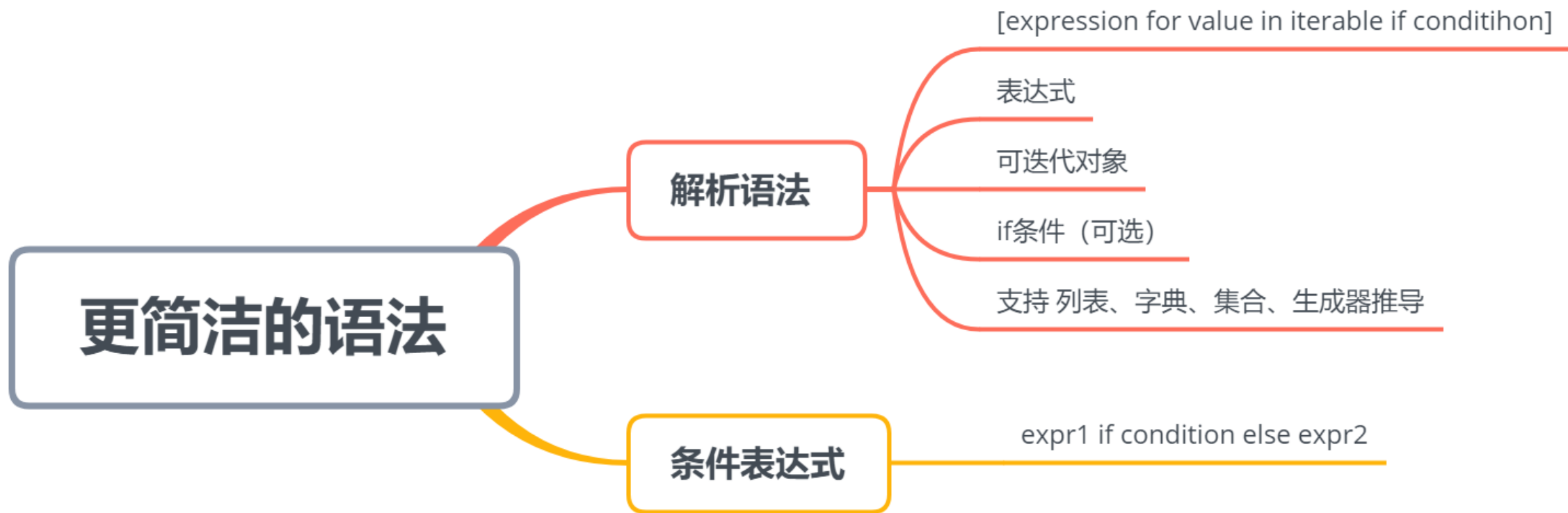
P	y	t	h	o	n
0	1	2	3	4	5

紧凑数组



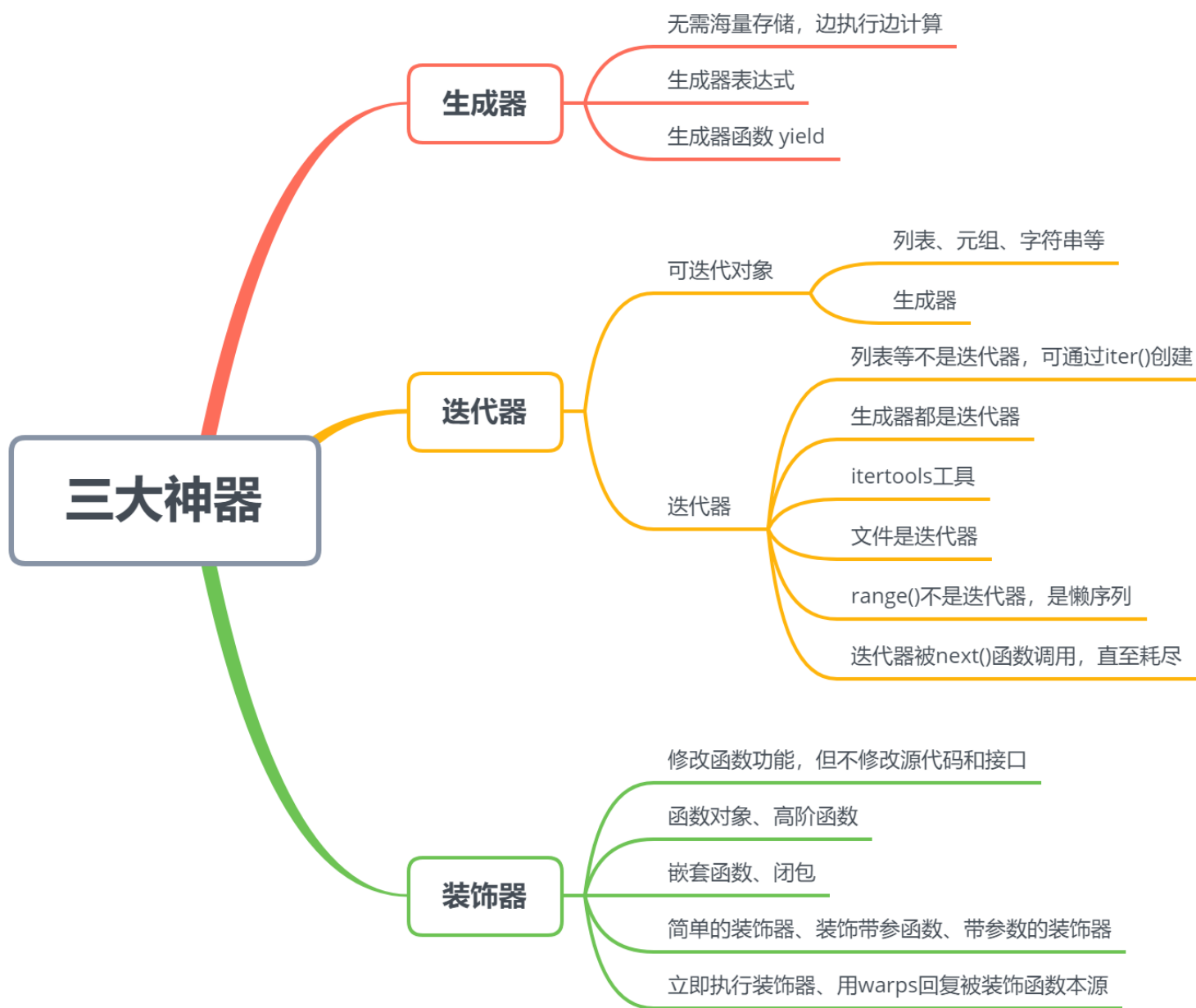
# 重难点回顾

The key to review



# 重难点

The key to rev





# —— 结 语 ——

在这次课程中，我们了解了  
数据类型背后的故事  
以及更加简洁的语法和实用的器具  
那么在下次课程中，我们将深入的聊一聊

**Python自带的标准库**

