

Python 有哪些数据类型？

- 整型
- 列表
- 元组
- 字典
- 字符串
- 集合
- 布尔

有哪些标准数据类型？

- 整型
- 列表
- 元组
- 字典
- 字符串
- 集合

is 和 == 的区别？

- 比较的是两个对象的 id 值是否相等，id 就是对象的内存地址
- == 是比较两个对象的内容是否相等，默认会调用对象的 `__eq__()` 方法

`n=256, n1=256, n is n1` 是 false 还是 true?

true

`n=257, n1=257, n is n1` 是 false 还是 true?

- 如果是不同代码块内容执行就是 false
- 如果是同代码块就是 true

Python 代码，1-100怎么求和，一行代码怎么写？

```
sum(range(1,101))
```

不可变对象和可变对象的区别？

- 可变对象：对象存放在地址中的值不会被改变
- 不可变对象：对象存放在地址中的值会直接改变

不可变对象和可变对象从内存出发说下有什么区别？

不可变对象

- Python 中的变量存放的是对象引用
- 不可变对象是指对象本身不可变
- 变的只是创建了新对象，然后变量改变了对象引用，指向了新对象，就对象会被垃圾回收

可变对象

变的是原来对象的内容，不会创建新对象，而变量也是指向原对象

字符串是可变对象还是不可变对象？

不可变对象

有哪些可变对象，哪些不可变对象？

- 不可变对象：字符串、元组、整形
- 可变对象：数组、字典、集合

怎么判断这个字符串是否为字典里面的键？还有其它方法吗？

简单的方法

```
"key" in dicts
```

```
"key" in dicts.keys()
```

复杂的方法一

```
for i in dicts.keys(): if i == "key": print(True)
```

复杂的方法二

```
if dicts.get("key"): print(True)
```

个位十位百位千位都有1的数，怎么筛选出来？

简单粗暴

```
for i in range(1, 1001): if str(i).find("1") == 0: print(i)
```

range、xrange 的区别？

- xrange 是 python2 的东西，在 python3 里已弃用，range 在 python2、3 都有
- xrange 是一个类，返回的是一个 xrange 对象
- range 是一个函数，返回的是可迭代对象（range 对象），并不是一个列表【这里之前写的是列表！是错的！】

Python 有用过哪些库？

标准库：

- os：提供多种操作系统功能接口的模块
- sys：提供 Python 运行环境的变量、函数的模块
- random：随机数
- math：数学
- datetime：基本的日期、时间类型
- time：时间的访问、转换
- pprint：美观打印
- hashlib：哈希库
- pathlib：文件系统路径库
- logging：日志工具
- json：JSON 解码、编码
- re：正则表达式

第三方库：

- requests：HTTP 请求
- pandas：分析结构化数据的库，比如 excel
- numpy：数学计算库
- pytest：单元测试框架库
- allure：测试报告库
- selenium：ui 自动化测试库
- appium：APP 自动化测试库

Python 怎么把字典转成 json 字符串？

`json.dumps()`

Python 读取一个文本文件要怎么实现？

- `open(file)`
- `with open(file) as`

Python 里面什么叫模块？

任何一个 .py 结尾的 python 文件

from.. import 和 import 有什么区别？

- import 导入的最小单位是模块，不能是变量、函数名、类名

- from.. import 导入的最小单位可以是变量、函数名、类名

Python 解析 XML 有哪些方式？

- SAX
- DOM
- ElementTree

Python 常见的异常？你遇到过得到异常有哪些？

- BaseException：所有异常的基类
- ValueError：无效参数值
- TypeError：无效参数类型
- SyntaxError：语法错误
- KeyError：找不到此键
- IndexError：找不到索引
- AttributeError：找不到属性
- ImportError：导入错误
- ZeroDivisionError：除0错误

Python 2.0 和 3.0 有什么区别？大方面的总结

- 表达式
- print
- range
- 不等于操作
- 编码
- 中文字符字节数

Python 的装饰器是什么？

- 本质上是一个 Python 函数
- 为已存在的方法提供额外的功能
- 可以将大量与方法本身提供的功能无关的代码提取出来，方便重用

Python 的装饰器一般是用来干嘛？

可以直接用框架提供的装饰器，也可以自己写装饰器
一般会用到 pytest、allure 的装饰器，自己写的有

- 异常捕捉：会给自己封装的每个方法加上这个异常捕捉装饰器，如果调用的封装方法报错了，就会进入这个装饰器，捕捉到指定异常后，我会刷新页面，再次执行刚刚报错的封装方法，然后会记录一次失败日志
- 日志：一般自己封装的方法都希望有日志，那如果每个封装的方法里单独调用日志类就会显得很臃肿重复，所以可以用一个日志装饰器代替
- 前置操作：比如多个方法执行前都需要调用同一个方法，那可以将依赖方法写在装饰器中
- 后置操作：比如每次执行方法后都需要还原数据集，可以将清理操作写在装饰器中
- 权限校验：执行方法前先进行权限校验，校验通过才会允许执行方法

Python 的迭代器和生成器的区别？

- 生成器是生成元素的，迭代器是访问集合元素的一种方式
- 迭代器输出生成器的内容
- 迭代器是一种支持 `next()` 操作的对象

深拷贝和浅拷贝的区别？

深拷贝和浅拷贝的实际场景？

"，