数据类型

# Python基础

## 标识符

* 在python里，标识符有字母、数字、下划线组成。
* 在python中，所有标识符可以包括英文、数字以及下划线（\_），但不能以数字开头。
* python中的标识符是区分大小写的。
* 以下划线开头的标识符是有特殊意义的。以单下划线开头（\_foo）的代表不能直接访问的类属性，需通过类提供的接口进行访问，不能用"from xxx import \*"而导入；
* 以双下划线开头的（\_\_foo）代表类的私有成员；以双下划线开头和结尾的（\_\_foo\_\_）代表python里特殊方法专用的标识，如\_\_init\_\_（）代表类的构造函数。

## 关键字

保留字不能用作常数或变数，或任何其他标识符名称。所有Python的关键字只包含小写字母。

|  |  |  |
| --- | --- | --- |
| and | exec | not |
| assert | finally | or |
| break | for | pass |
| class | from | print |
| continue | global | raise |
| def | if | return |
| del | import | try |
| elif | in | while |
| else | is | with |
| except | lambda | yield |

## 行和缩进

Python的代码块不使用大括号（{}）来控制类，函数以及其他逻辑判断。python最具特色的就是用缩进来写模块。缩进的空白数量是可变的，但是所有代码块语句必须包含相同的缩进空白数量，这个必须严格执行。

IndentationError: unexpected indent 错误是：文件里格式不对了，可能是tab和空格没对齐的问题"，所有python对格式要求非常严格。

IndentationError: unindent does not match any outer indentation level错误：表明，你使用的缩进方式不一致，有的是 tab 键缩进，有的是空格缩进，改为一致即可。因此，在Python的代码块中必须使用相同数目的行首缩进空格数。

建议：每个缩进层次使用 单个制表符 或 两个空格 或 四个空格 , 切记不能混用

## 多行语句

Python可以在同一行中使用多条语句，语句之间使用分号(;)分割，以下是一个简单的实例：

|  |
| --- |
| import sys; x = 'foo'; sys.stdout.write(x + '\n') |

## Python 引号

Python 接收单引号(' )，双引号(" )，三引号(''' """) 来表示字符串，引号的开始与结束必须的相同类型的。其中三引号可以由多行组成，编写多行文本的快捷语法，常用语文档字符串，在文件的特定地点，被当做注释。

|  |
| --- |
| word = 'word'  sentence = "这是一个句子。"  paragraph = """这是一个段落。  包含了多个语句""" |

## Python注释

python中单行注释采用 # 开头。

|  |
| --- |
| # 第一个注释  print("Hello, Python!") ; # 第二个注释 |

python 中多行注释使用三个单引号(''')或三个双引号(""")。

|  |
| --- |
| '''  这是多行注释，使用单引号。  这是多行注释，使用单引号。  这是多行注释，使用单引号。  '''  """  这是多行注释，使用双引号。  这是多行注释，使用双引号。  这是多行注释，使用双引号。  """ |

# Integral类型

## 整数

整数的大小只限制于内存的大小。默认整数是采用十进制的表达方式，但是也可以使用其他进制。二进制用0b引导，八进制用0o引导，十六进制可以用0x引导，其中的字母也可以使用大写字母。

## 布尔型

# 浮点类型

# 字符串