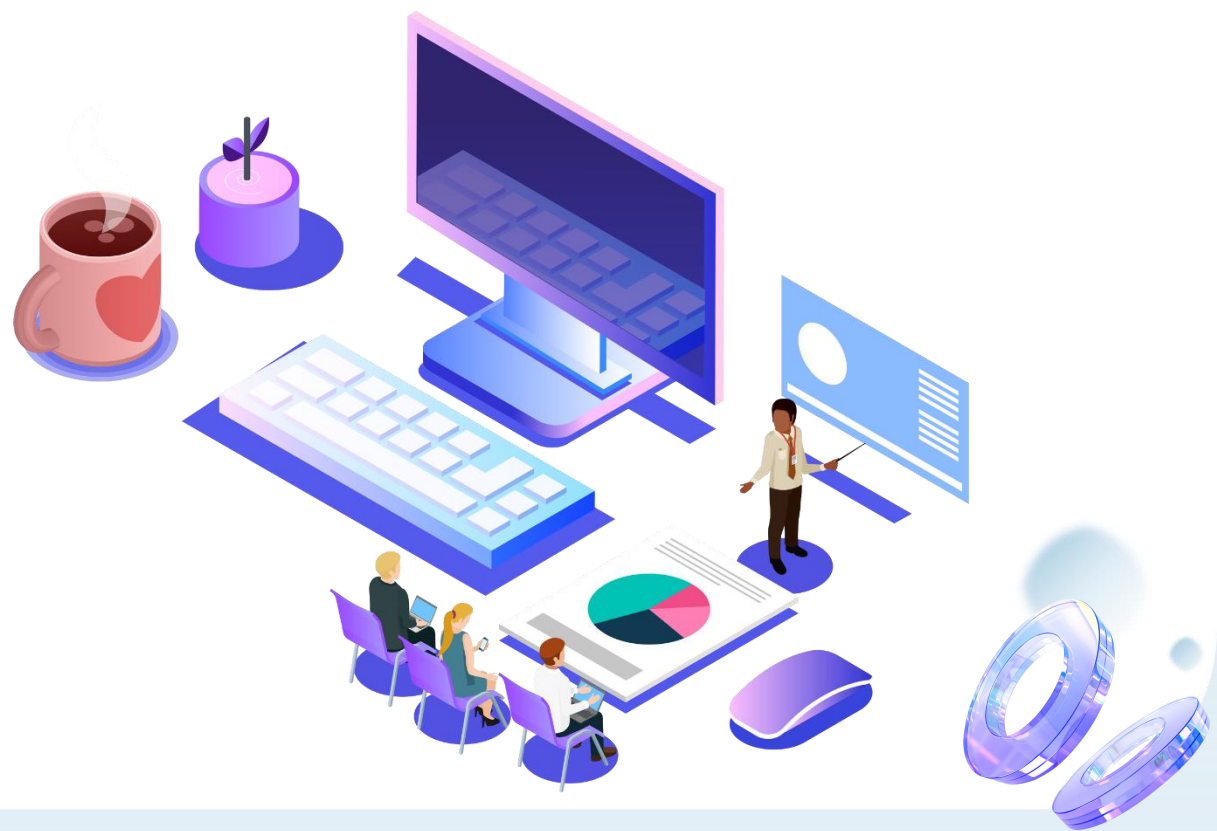


Python数据类型:

□ 标准数据类型: Python 定义了一些标准类型, 用于存储各种类型的数据。

Python有五个标准的数据类型:

- Numbers (数字)
- String (字符串)
- List (列表)
- Tuple (元组)
- Dictionary (字典)



Python数据类型:

□ Python数字: 数字数据类型用于存储数值。他们是不可改变的数据类型, 这意味着改变数字数据类型会分配一个新的对象。当你指定一个值时, Number对象就会被创建。可以通过使用del语句删除单个或多个对象的引用。例如: Python支持四种不同的数字类型:

- int (有符号整型)
- long (长整型[也可以代表八进制和十六进制])
- float (浮点型)
- complex (复数)



Python数据类型:

- Python字符串: 字符串或串(String)是由数字、字母、下划线组成的一串字符。python的字串列表有2种取值顺序:从左到右索引默认0开始的, 最大范围是字符串长度少1。从右到左索引默认-1开始的, 最大范围是字符串开头。如果你要实现从字符串中获取一段子字符串的话, 可以使用变量 [头下标: 尾下标], 就可以截取相应的字符串, 其中下标是从 0 开始算起, 可以是正数或负数, 下标可以为空表示取到头或尾。



Python数据类型:

- Python列表: List (列表) 是 Python 中使用最频繁的数据类型。列表中值的切割也可以用到变量 [头下标:尾下标] , 就可以截取相应的列表, 从左到右索引默认 0 开始, 从右到左索引默认 -1 开始, 下标可以为空表示取到头或尾。加号 + 是列表连接运算符, 星号 * 是重复操作。如下实例:

```
#!/usr/bin/python
# -*- coding: UTF-8 -*-

list = [ 'runoob', 786 , 2.23, 'john', 70.2 ]
tinylist = [123, 'john']

print list           # 输出完整列表
print list[0]        # 输出列表的第一个元素
print list[1:3]       # 输出第二个至第三个的元素
print list[2:]        # 输出从第三个开始至列表末尾的所有元素
print tinylist * 2     # 输出列表两次
print list + tinylist  # 打印组合的列表
```



Python数据类型:

- Python 字典：字典(dictionary)是除列表以外python之中最灵活的内置数据结构类型。列表是有序的对象结合，字典是无序的对象集合。两者之间的区别在于：字典当中的元素是通过键来存取的，而不是通过偏移存取。字典用"{}"标识。字典由索引(key)和它对应的值value组成。

```
#!/usr/bin/python
# -*- coding: UTF-8 -*-

dict = {}
dict['one'] = "This is one"
dict[2] = "This is two"

tinydict = {'name': 'john', 'code': 6734, 'dept': 'sales'}


print dict['one']      # 输出键为 'one' 的值
print dict[2]          # 输出键为 2 的值
print tinydict         # 输出完整的字典
print tinydict.keys()  # 输出所有键
print tinydict.values() # 输出所有值
```



Python数据类型：

□ Python数据类型转换

函数	描述
<u>int(x [,base])</u>	将x转换为一个整数
<u>long(x [,base])</u>	将x转换为一个长整数
<u>float(x)</u>	将x转换到一个浮点数
<u>complex(real [,imag])</u>	创建一个复数
<u>dict(d)</u>	创建一个字典。d 必须是一个序列 (key,value)元组。
<u>frozenset(s)</u>	转换为不可变集合
<u>chr(x)</u>	将一个整数转换为一个字符
<u>unichr(x)</u>	将一个整数转换为Unicode字符
<u>ord(x)</u>	将一个字符转换为它的整数值
<u>hex(x)</u>	将一个整数转换为一个十六进制字符串
<u>oct(x)</u>	将一个整数转换为一个八进制字符串

