



# Python数据分析——第1周

**【声明】** 本视频和幻灯片为炼数成金网络课程的教学资料，所有资料只能在课程内使用，不得在课程以外范围散播，违者将可能被追究法律和经济责任。

课程详情访问炼数成金培训网站

<http://edu.dataguru.cn>

# 关注炼数成金企业微信



- 提供全面的数据价值资讯，涵盖商业智能与数据分析、大数据、企业信息化、数字化技术等，各种高性价比课程信息，赶紧掏出您的手机关注吧！



- ◆ Python是一种解释型、面向对象、动态数据类型的高级程序设计语言。
- ◆ Python由Guido van Rossum于1989年底发明，第一个公开发行人版发行于1991年。像Perl语言一样, Python 源代码同样遵循 GPL(GNU General Public License)协议。
- ◆ Python 是一个高层次的结合了解释性、编译性、互动性和面向对象的脚本语言。
- ◆ Python 的设计具有很强的可读性，相比其他语言经常使用英文关键字，其他语言的一些标点符号，它具有比其他语言更有特色语法结构。



# Python的简介

- ◆ Python 是一种解释型语言
- ◆ Python 是交互式语言
- ◆ Python 是面向对象语言
- ◆ Python 是初学者的语言
  
- ◆ 网址：<https://www.python.org/>

- ◆ Python 是由 Guido van Rossum 在八十年代末和九十年代初，在荷兰国家数学和计算机科学研究所设计出来的。
- ◆ Python 本身也是由诸多其他语言发展而来的,这包括 ABC、Modula-3、C、C++、Algol-68、SmallTalk、Unix shell 和其他的脚本语言等等。
- ◆ 像 Perl 语言一样，Python 源代码同样遵循 GPL(GNU General Public License)协议。
- ◆ 现在 Python 是由一个核心开发团队在维护，Guido van Rossum 仍然占据着至关重要的作用，指导其进展。

- ◆ 1.易于学习
- ◆ 2.易于阅读
- ◆ 3.易于维护
- ◆ 4.一个广泛的标准库
- ◆ 5.互动模式
- ◆ 6.可移植
- ◆ 7.可扩展
- ◆ 8.数据库
- ◆ 9.GUI编程
- ◆ 10.可嵌入

## ◆ NumPy ( Numerical Python ) , Python科学计算的基础包

- 快速高效的多维数组对象ndarray
- 用于对数组执行元素级计算以及直接对数组执行数学运算的函数
- 用于读写硬盘上基于数组的数据集的工具
- 线性代数运算、傅里叶变换, 以及随机数生成
- 用于将C、C++、Fortran代码集成到Python的工具

## ◆ Pandas

- 快速便捷地处理结构化数据的大量数据结构和函数
- DataFrame

```
>>> frame
   total_bill  tip  sex  smoker  day  timesize
1    16.99    1.01 Female    No   Sun    Dinner    2
2    10.34    1.66  Male    No   Sun    Dinner    3
3    21.01    3.5  Male    No   Sun    Dinner    3
4    23.68    3.31  Male    No   Sun    Dinner    2
5    24.59    3.61 Female    No   Sun    Dinner    4
6    25.29    4.71  Male    No   Sun    Dinner    4
7     8.77     2  Male    No   Sun    Dinner    2
8    26.88    3.12  Male    No   Sun    Dinner    4
9    15.04    1.96  Male    No   Sun    Dinner    2
10   14.78    3.23  Male    No   Sun    Dinner    2
```



## ◆ Matplotlib

- 最流行的绘制数据图表的Python库

## ◆ SciPy

- Scipy.integrate
- Scipy.linalg
- Scipy.optimize
- Scipy.signal
- Scipy.sparse
- Scipy.special
- Scipy.stats
- Scipy.weave

# IPython：一个交互式shell 工具

- ◆ IPython是Python可行计算标准工具集的组成部分，为交互式和探索式计算提供了一个强健而高效的环境
  - 一个类似于Mathematica的HTML笔记本
  - 一个基于Qt框架的GUI控制台
  - 用于交互式并行和分布式计算的基础架构
  
- ◆ 网址：<http://ipython.org/>

◆ 软件下载地址：<https://www.python.org/downloads/>

- 选择合适的系统版本

◆ 版本2.x与3.x的区别

- 性能
- 编码
- 语法
- 字符串和字节串
- 数据类型
- 模块

- ◆ ( 1 ) NumPy
- ◆ <http://sourceforge.net/projects/numpy/files/>
- ◆ ( 2 ) SciPy和Matplotlib
- ◆ [www.enthought.com/products/epd.php](http://www.enthought.com/products/epd.php)。
- ◆ 在安装过程中，你可能需要将一个文件msvcp71.dll放到目录C:\Windows\system32下
- ◆ ( 3 ) IPython
- ◆ <http://ipython.scipy.org/Wiki/IpythonOnWindows>。
- ◆ ( 4 ) Canopy

## ◆ IDE，集成开发环境

- Eclipse+PyDev插件
- Python Tools for Visual Studio ( 针对Windows用户 )
- PyCharm
- Spyder
- Komodo IDE

- ◆ 交互式编程：不需要创建脚本文件，是通过 Python 解释器的交互模式进来编写代码
- ◆ 脚本式编程：通过脚本参数调用解释器开始执行脚本，直到脚本执行完毕。当脚本执行完成后，解释器不再有效。
- ◆ Python标识符
  - 由字母、数字、下划线组成
  - 不能以数字开头
  - 区分大小写
  - 以下划线开头的标识符是有特殊意义

## ◆ Python保留字符

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
- 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

## ◆ 多行语句

- 一般以新行作为为语句的结束符
- 可以使用斜杠（\）将一行的语句分为多行显示

## ◆ Python 引号

- 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

## ◆ Python注释

- 单行注释采用 # 开头
- 多行注释使用三个单引号('')或三个双引号(''')



## ◆ Python空行

- 函数之间或类的方法之间用空行分隔，表示一段新的代码的开始
- 空行与代码缩进不同，空行并不是Python语法的一部分

## ◆ 同一行显示多条语句

- 可以在同一行中使用多条语句，语句之间使用分号(;)分割

## ◆ 多个语句构成代码组

- 缩进相同的一组语句构成一个代码块，我们称之代码组
- 首行及后面的代码组称为一个子句(clause)

## ◆ 标准数据类型

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

## ◆ 变量赋值

- 变量不需要声明，变量的赋值操作既是变量声明和定义的过程
- 等号 ( = ) 用来给变量赋值
- Python允许你同时为多个变量赋值

## ◆ 数字

- 数字数据类型用于存储数值
- 不可改变的数据类型
- 四种数值型
  - 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 )

## ◆ 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
- 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

## ◆ 列表

- Python 中使用最频繁的数据类型

## ◆ 元组

- 用"()"标识。内部元素用逗号隔开
- 元素不能二次赋值，相当于只读列表

## ◆ 字典

- 无序的对象集合
- 字典当中的元素是通过键来存取的，而不是通过偏移存取
- 用"{}"标识。字典由索引(key)和它对应的值value组成

# Python数据类型转换

函数	描述
<code>int(x [,base])</code>	将x转换为一个整数
<code>long(x [,base] )</code>	将x转换为一个长整数
<code>float(x)</code>	将x转换到一个浮点数
<code>complex(real [,imag])</code>	创建一个复数
<code>str(x)</code>	将对象 x 转换为字符串
<code>repr(x)</code>	将对象 x 转换为表达式字符串
<code>eval(str)</code>	用来计算在字符串中的有效Python表达式,并返回一个对象
<code>tuple(s)</code>	将序列 s 转换为一个元组
<code>list(s)</code>	将序列 s 转换为一个列表
<code>set(s)</code>	转换为可变集合
<code>dict(d)</code>	创建一个字典。d 必须是一个序列 (key,value)元组。

<code>frozenset(s)</code>	转换为不可变集合
<code>chr(x)</code>	将一个整数转换为一个字符
<code>unichr(x)</code>	将一个整数转换为Unicode字符
<code>ord(x)</code>	将一个字符转换为它的整数值
<code>hex(x)</code>	将一个整数转换为一个十六进制字符串
<code>oct(x)</code>	将一个整数转换为一个八进制字符串

## ◆ 算术运算符

运算符	描述	实例
+	加 - 两个对象相加	a + b 输出结果 30
-	减 - 得到负数或是一个数减去另一个数	a - b 输出结果 -10
*	乘 - 两个数相乘或是返回一个被重复若干次的字符串	a * b 输出结果 200
/	除 - x除以y	b / a 输出结果 2
%	取模 - 返回除法的余数	b % a 输出结果 0
**	幂 - 返回x的y次幂	a**b 为10的20次方，输出结果 100000000000000000000
//	取整除 - 返回商的整数部分	9//2 输出结果 4，9.0//2.0 输出结果 4.0

## ◆ 比较运算符

运算符	描述	实例
==	等于 - 比较对象是否相等	(a == b) 返回 False。
!=	不等于 - 比较两个对象是否不相等	(a != b) 返回 true。
<>	不等于 - 比较两个对象是否不相等	(a <> b) 返回 true。这个运算符类似 != 。
>	大于 - 返回x是否大于y	(a > b) 返回 False。
<	小于 - 返回x是否小于y。所有比较运算符返回1表示真，返回0表示假。这分别与特殊的变量True和False等价。注意，这些变量名的大写。	(a < b) 返回 true。
>=	大于等于 - 返回x是否大于等于y。	(a >= b) 返回 False。
<=	小于等于 - 返回x是否小于等于y。	(a <= b) 返回 true。



## ◆ 赋值运算符

运算符	描述	实例
=	简单的赋值运算符	<code>c = a + b</code> 将 <code>a + b</code> 的运算结果赋值为 <code>c</code>
+=	加法赋值运算符	<code>c += a</code> 等效于 <code>c = c + a</code>
-=	减法赋值运算符	<code>c -= a</code> 等效于 <code>c = c - a</code>
*=	乘法赋值运算符	<code>c *= a</code> 等效于 <code>c = c * a</code>
/=	除法赋值运算符	<code>c /= a</code> 等效于 <code>c = c / a</code>
%=	取模赋值运算符	<code>c %= a</code> 等效于 <code>c = c % a</code>
**=	幂赋值运算符	<code>c **= a</code> 等效于 <code>c = c ** a</code>
//=	取整除赋值运算符	<code>c //= a</code> 等效于 <code>c = c // a</code>

## ◆ 位运算符

运算符	描述	实例
&	按位与运算符：参与运算的两个值,如果两个相应位都为1,则该位的结果为1,否则为0	(a & b) 输出结果 12 ，二进制解释： 0000 1100
	按位或运算符：只要对应的二个二进位有一个为1时，结果位就为1。	(a   b) 输出结果 61 ，二进制解释： 0011 1101
^	按位异或运算符：当两对应的二进位相异时，结果为1	(a ^ b) 输出结果 49 ，二进制解释： 0011 0001
~	按位取反运算符：对数据的每个二进制位取反,即把1变为0,把0变为1	(~a) 输出结果 -61 ，二进制解释： 1100 0011， 在一个有符号二进制数的补码形式。
<<	左移动运算符：运算数的各二进位全部左移若干位，由"<<"右边的数指定移动的位数，高位丢弃，低位补0。	a << 2 输出结果 240 ，二进制解释： 1111 0000
>>	右移动运算符：把">>"左边的运算数的各二进位全部右移若干位，">>"右边的数指定移动的位数	a >> 2 输出结果 15 ，二进制解释： 0000 1111

## ◆ 逻辑运算符

运算符	描述	实例
and	布尔"与" - 如果x为False，x and y返回False，否则它返回y的计算值。	(a and b) 返回 true。
or	布尔"或" - 如果x是True，它返回True，否则它返回y的计算值。	(a or b) 返回 true。
not	布尔"非" - 如果x为True，返回False。如果x为False，它返回True。	not(a and b) 返回 false。

## ◆ 成员运算符

运算符	描述	实例
in	如果在指定的序列中找到值返回True，否则返回False。	x 在 y序列中，如果x在y序列中返回True。
not in	如果在指定的序列中没有找到值返回True，否则返回False。	x 不在 y序列中，如果x不在y序列中返回True。

## ◆ 身份运算符

运算符	描述	实例
is	is是判断两个标识符是不是引用自一个对象	x is y, 如果 id(x) 等于 id(y), <b>is</b> 返回结果 1
is not	is not是判断两个标识符是不是引用自不同对象	x is not y, 如果 id(x) 不等于 id(y). <b>is not</b> 返回结果 1

## ◆ 运算符优先级

运算符	描述
**	指数 (最高优先级)
~ + -	按位翻转, 一元加号和减号 (最后两个的方法名为 +@ 和 -@)
* / % //	乘, 除, 取模和取整除
+ -	加法减法
>> <<	右移, 左移运算符
&	位 'AND'
^	位运算符
<= < > >=	比较运算符
<> == !=	等于运算符
= %= /= //= -= += *= **=	赋值运算符
is is not	身份运算符
in not in	成员运算符
not or and	逻辑运算符

- ◆ Dataguru（炼数成金）是专业数据分析网站，提供教育，媒体，内容，社区，出版，数据分析业务等服务。我们的课程采用新兴的互联网教育形式，独创地发展了逆向收费式网络培训课程模式。既继承传统教育重学习氛围，重竞争压力的特点，同时又发挥互联网的威力打破时空限制，把天南地北志同道合的朋友组织在一起交流学习，使到原先孤立的学习个体组合成有组织的探索力量。并且把原先动辄成千上万的学习成本，直线下降至百元范围，造福大众。我们的目标是：低成本传播高价值知识，构架中国第一的网上知识流转阵地。
- ◆ 关于逆向收费式网络的详情，请看我们的培训网站 <http://edu.dataguru.cn>



# Thanks

## FAQ时间