

基础入门：认识 Python

讲师：萨缪尔 Samuel

课程

亮点

1

Python内容模块及数据分析学习
的顺序

2

常用IDE介绍及Anaconda的
安装

3

Python基础与入门操作（打
印输出、数据类型、运算符）

Python内容模块及 数据分析学习的顺 序





Python主要应用领域

- 数据科学 (Data Science & Visualization)
- 机器学习 (Machine Learning and AI)
- 网络爬虫 (Web Scraping)
- Web开发 (Django)
- 自动化运维
- 游戏开发
- 网络编程
- ...

Python数据分析入门的路径与步骤

数据分析基础

- Numpy库
- Pandas
- Matplotlib

数据可视化

- Seaborn
- Pyecharts
- Ggplot
- plotnine
- Plotly
- Bokeh
- Geoplotlib
- folium

爬虫

- pandas网页表格爬虫
- request+lxml
- BeautifulSoup
- Scrapy

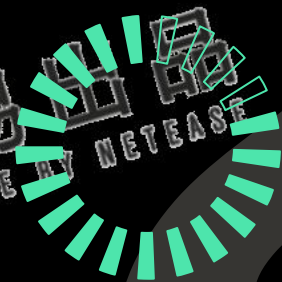
机器学习

- Sklearn
- Tensorflow
- Keras
- PyTorch

高级可视化 高阶爬虫 高阶机器学习

- 高级可视化库:
pyecharts、bokeh
- 高阶爬虫:
Scrapy
- 高阶机器学习:
Keras、Tensorflow、
PyTorch

常用IDE介绍及 Anaconda的安装



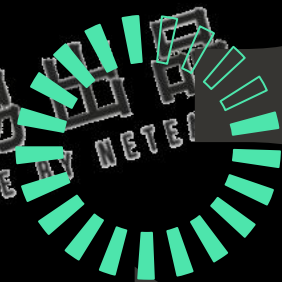
Python常用的IDE

IDE	简介	优点
Jupyter notebook (数据分析推荐使用)	网页端应用，通过pip安装之后，在命令行输入jupyter notebook会在默认浏览器打开	<ul style="list-style-type: none">•可共享•支持40多种编程语言•轻量•交互式•优秀的可视化服务•支持Markdown
Pycharm	专门用于Python开发的IDE	<ul style="list-style-type: none">•代码补全•智能提示•语法检查•集成了版本控制、单元测试、git功能•可以快速创建Django， Flask等Python Web框架
Sublime Text	轻量级的代码编辑器	<ul style="list-style-type: none">•跨平台•支持几十种编程语言•小巧灵活，运行轻快•支持代码高亮•代码补全•语法提示
Vs Code Atom ...		

Anaconda的安装

1. <https://www.anaconda.com/products/individual>, 根据电脑配置选择合适的版本
2. 按照操作步骤安装即可
3. 安装好Anaconda后, 启用Jupyter Notebook扩展插件 (参考网址: <https://zhuanlan.zhihu.com/p/339143873>)
4. 基础案例操作代码

Python的基础与入门操作



3

Python入门操作：输出打印

`print()` 函数的详细语法格式如下：`print (value,...,sep=" ",end="\n",file=sys.stdout,flush=False)`

In [34]:

#方法一：顺序输出

```
user_name = 'Charlie'
```

```
user_age = 8
```

```
print("读者名: ",user_name,"\n年龄: ",user_age)
```

读者名: Charlie

年龄: 8

In [38]:

#方法二：集中输出

```
user_name = 'Charlie'
```

```
user_age = 8
```

```
print("读者名: %s \n年龄:%d"%(user_name,user_age))
```

读者名: Charlie

年龄:8

In [42]:

#方法三：Format格式输出

```
user_name = 'Charlie'
```

```
user_age = 8
```

```
print("读者名: {}\n年龄: {}".format(user_name,user_age))
```

读者名: Charlie

年龄: 8

Python7种标准的数据类型

数据类型	具体类型	示例
Number数值型	int（整型）、float（浮点型）、complex（复数类型）	0, 1, 1.1, 1+2i
Boolean布尔类型	True和False（注意首字母要大写）	
String字符串型	用单引号（' '）或双引号（" "）包裹起来即可得到字符串，同时可用反斜杠“\”表示特殊的转义字符	string = 'Hello World!'
List列表	用方括号（[]）括起来，类似于数组	list= ['Google', 'Runoob', 1997, 2000]
Tuple元组	使用一对圆括号“()”，元素后面添加逗号	tuple = ('Google', 'Runoob', 1997, 2000)
Dictionary字典	由多个“键（key）/值（value）对”构成	dict = {'Name': 'Runoob', 'Age': 7, 'Class': 'First'}
Set集合	所有元素用花括号{}或者（[]）括起来的，元素之间用逗号分隔	set(['[1,2,3]', 'h', 'o', 'n', 'p', 't', 'qiwsir', 'y'])

Python入门操作：7种运算符

- 算术运算符 (+ - * / % ** //)
- 比较运算符(== != > < >= <=)
- 赋值运算符(= += -= *= /= %= **= // =)
- 逻辑运算符(and or not)

- 位运算符(& | ^ ~ << >>)
- 成员运算符(in not in)
- 身份运算符(is is not)

示例：赋值运算符

运算符	描述	实例
=	简单的赋值运算符	c = a + b, 将 a + b 的运算结果赋值给 c
+=	加法赋值运算符	c += a, 等效于 c = c + a
-=	减法赋值运算符	c -= a, 等效于 c = c - a
*=	乘法赋值运算符	c *= a, 等效于 c = c * a
/=	除法赋值运算符	c /= a, 等效于 c = c / a
%=	取模赋值运算符	c %= a, 等效于 c = c % a
**=	幂赋值运算符	c **= a, 等效于 c = c ** a
//=	取整（地板）除赋值运算符	c //= a, 等效于 c = c // a

案例：使用python实现9*9乘法口诀

```
In [31]: ▼ for i in range(1,10):  
          ▼ for j in range(1,i+1):  
              print("%d*%d=%2d"%(i,j,i*j),end=" ")  
          print(" ")
```

```
1*1= 1  
2*1= 22*2= 4  
3*1= 33*2= 63*3= 9  
4*1= 44*2= 84*3=124*4=16  
5*1= 55*2=105*3=155*4=205*5=25  
6*1= 66*2=126*3=186*4=246*5=306*6=36  
7*1= 77*2=147*3=217*4=287*5=357*6=427*7=49  
8*1= 88*2=168*3=248*4=328*5=408*6=488*7=568*8=64  
9*1= 99*2=189*3=279*4=369*5=459*6=549*7=639*8=729*9=81
```

课程总结

1

Python的[应用场景](#)以及Python数据分析的[学习路径](#)

2

介绍了Python常用的编辑器，数据分析领域推荐使用[Jupyter notebook](#)

3

讲解了Python的[3种输出方式](#)

4

讲解了Python的[7种数据类型](#)以及[7种运算符](#)

谢谢观看

