Big Data for Media and Communication

00：主要介绍GitHub

其他同学的项目：<https://github.com/data-projects-archive/>

这门课主要包括：python基础、数据抓取、表操作和数据挖掘

markdowm是一种标记语言，它可以是呈现格式，也可以是源代码格式。常规扩展名是.md,内容是纯文本。

01:

介绍了有关Mac上终端的基本知识，以及如何在终端中导航文件系统，使用shell，创建第一个python脚本并执行它。

终端、命令、文件、ls、cd、pwd

touch意味着创建一个文件

•rm表示删除文件

•mkdir表示创建文件夹

•rmdir表示删除文件夹

•mv表示重命名

在命令行中获取内联帮助

命令返回和命令输出

编辑并执行python文件

02:

在本章中，我们将重点介绍Python基础知识，包括变量、基本数据类型、算术、函数和几个常用模块。

**(1)变量和赋值**

把一个变量看作一个附加到特定对象上的名称。在Python中，不需要预先声明或定义变量。因此，您应该给它一个定义或者给它赋值。等号=用于为变量赋值。

**(2)基本数据类型**

整数：int

浮点（实数）：浮点float

布尔值：bool

字符串：STR

字符串中的转义符

转义字符是“对字符序列中的后续字符调用替代解释的字符”。

Python有自己的规则和语法。就像需要在字符串中使用特殊字符“”一样，应该使用转义字符。因为python将“”视为字符串的符号，所以如果在字符串内部使用“”，它将导致无效的语法错误。通常，我们使用\（反斜杠）作为的转义符。以下是常用的示例。

•\'表示单引号

•“表示双引号

•\\表示反斜杠

•\n表示新行

•\r表示回车

\t表示制表符

•\b表示退格

•\f表示形式提要

•\v表示垂直制表符

**(3)模块、函数和包功能**

函数是Python中的一个基本构建块，它包含一些可以重用的代码。函数以“参数”的形式接受输入并给出相应的输出。例如，print（）是一个内置函数，它接受一系列参数的输入，这些参数表示要打印的值。print（）使用这些参数，将它们转换为str格式，并将输出写到屏幕上。int（）是另一个内置函数，它接受任意值作为输入并输出相应的整数。

注意“函数调用”符号，即（）在函数名后面。当Python解释器看到这个符号时，它将进入函数体并在那里执行代码。当您发现自己在Python中重复执行某些操作时，应该考虑将这些代码包装成一个函数，并在您想要使用它的地方使用带有适当参数的函数调用。在第三章中，我们将进一步讨论如何写作以及如何调用自己的函数。

模块：模块是Python中的一个更高级别的构建块，包含某些功能，可以重用。有些例子像麻木，坐立不安和地缘。这些模块扩展了基本的python函数，这样您就可以轻松地完成复杂的任务，比如计算矩阵乘法和计算两个城市之间的距离。否则，您需要编写数百行代码，才能完成这项工作。

软件包

包是Python模块的集合。它可以是单个.py文件或.py文件的文件夹结构。不用担心内部布局，使用包的方式与使用模块的方式类似。我们将在以后的讨论中交替使用这两个术语。

**（3）如何使用模块步骤**

1）:PIP安装模块

在导入或使用模块之前，您需要做一些准备工作。您需要先安装。您可以从软件包的官方网站安装，也可以使用我们推荐的其他第三方工具，如pip。PIP是python packaging authority（pypa）的一个功能，它是一个工作组，负责维护python packaging中的许多相关项目。如果您使用的是python 2>=2.7.9或python 3>=3.4，则已经安装了pip。您可以键入python--version来签出当前版本。如果您是旧版本，请查看此处以安装和升级PIP。

注意：如果使用python 3，请使用pip3进行安装。

•要安装最新版本的“someproject”：请在终端中键入pip3安装“someproject”。

•安装特定版本：pip3 install'someproject==1.4'

•升级包：pip3安装--升级someproject

•要安装与当前用户隔离的包：pip3 install--user someproject。在学校的计算机实验室里，你应该使用这种方法，因为你没有安装整个计算机的权限，只需要你的帐户，这样计算机就可以保存你的记录。

步骤2：导入模块

Import numpy

提供1。任意齐次项的数组对象2。数组上的快速数学运算3。线性代数、傅立叶变换、随机数生成

**(4)如何找到我们想要的模块和包**

**(5)如何调用函数**

1）.成员引用的符号

模块的成员可以是函数或变量。正如您从前面的内容中已经知道的，模块中有许多函数。我们使用模块名+.+成员名来引用模块中的一个成员。如果不知道此模块中的成员是什么，可以使用帮助（模块）加载帮助文档，或使用dir（模块）签出可用的成员。

2）（）调用函数的符号

在我们引用一个函数之后，通常需要输入参数来调用函数。我们在函数后使用（“参数”）来调用函数，执行此命令后，您可以得到想要的结果。

练习3：计算圆的面积

**第2章scipy&numpy基本功能：数组中应该知道的通用模块和函数**

**（6）**注意：后面的文本不是代码，我们称之为注释。我们可以在代码块中的文本前面使用来注释这些文本，这将向系统提供不需要执行此行的指令。正是这些代码的解释或说明告诉了其他人你在做什么，以便他们能迅速理解。或者，即使在稍后回顾代码时，您也可以快速回忆起您所做的工作。这是一个好习惯，尤其是当你与他人合作时。您可以直接在代码/文本前面添加注释。如果要注释多行。您可以选择这些行并键入command+/。然后再次键入command+/取消注释。

**关于数据类型中的索引**

**字符串函数（str.\*）公共函数**

**python的经典百分号字符串插值**

在引入str.format（）之前，使用%进行简单的字符串插值非常容易。如果参数的顺序不太可能更改，并且您只需要连接很少的元素，那么可以使用它。

**St.格式（）**

str.format（）作为字符串格式化操作执行，它提供了执行复杂变量替换和值格式化的能力，以及对字符串输出的极大灵活性，这种方式比简单的旧连接更容易读取、写入和维护。

**随机随机.random（）**

该模块实现了各种分布的伪随机数生成器。有许多有用且简单的函数，如random.shuffle（）、random.random（）random.sample（）。您可以查看他们的文档以了解详细信息。

随机。randint（a，b）

随机选择（顺序）

随机选择（总体，权重=无，\*，累计权重=无，k=1）

随机。随机播放（列表）

随机抽样（人口，k）

练习和挑战设计和计算简单的媒体业务模型

计算您需要携带多少SD卡

媒体监控的简单模型

马克六号彩票机

参考文献

03:第3周：控制流程

之前，我们学习了Python的基本知识，包括数据类型、算术、函数和几个常用模块，有了这些帮助，您可以构建一个计算器来进行一些简单的案例分析。本周，我们将进一步学习复合数据类型、如何使用它们的功能和方法。同时，我们将触摸基本控制流程，以更好地理解编码工作背后的逻辑。之后，您可以自己创建一些函数，这样您就可以使用Python做更多的工作。

**（1）多用“帮助”自己学习**

* Type 'q' to quit help;
* Type 'j' to scroll down;
* Type 'k' to scroll up.

**（2）bool和比较**

1）逻辑运算符and, or, not

2）比较运算符==、！=、>、<、

3）STR比较

4）int比较

**（3）控制流**

注意：在“控制流”一章中，我们在处理if/for/while/def/class时会遇到很多缩进。在python shell中输入代码既困难又不方便，所以请在文本编辑器中写下代码并将其保存为.py文件，这样您只需执行文件一次即可得到答案。如果您忘记了如何操作，请参阅第2章。

1）有选择地执行代码：if else语句

注意：所有函数定义或条件比较都应以“：”结尾，并且这些函数和条件中的所有内容都需要缩进。只需单击“选项卡”即可缩进。

2）重复类似操作：循环

3）for循环

用于语句获取值

用于循环计算

4）while循环

for循环和while循环之间的区别：While Loops允许您在其中放置一个条件时，比如当i<10时，当条件不再满足时，它将停止（i>=10）。您还可以用布尔值（真/假）替换10以及许多其他类型的变量。

For Loops，您可以多次在循环中运行，您希望它在范围（0100）中运行，这将不断增加i，直到条件返回false（>100），您可以用其他数字和变量替换10，如name\_list中的name，这意味着您希望循环整个name\_list以在问题。一旦条件不再满足，它就会退出。

一般来说，如果您想使用循环进行条件比较，而循环对您有效，如果您想循环整个列表中的每个元素，for更好。

5）中断并继续语句

中断语句

Continue语句

集成示例：for和if

奖金：备选方案和更有效的计算

**（4）函数**

1）功能定义（def）

**函**数是只在调用时才运行的代码块。可以通过向函数传递参数来调用此函数。然后函数可以返回结果数据。

关键字def（表示确定）标记函数头的开始。

唯一标识它的函数名。

参数（参数）是可选的，通过它我们可以向函数传递值。

冒号：标记函数头的结尾。

描述函数的作用。

语句必须具有相同的缩进级别（通常单击键盘上的制表符进行缩进）。

用于从函数返回值的可选返回语句。

当你发现你在一次又一次地使用一个函数时。您可以使用“def语句”来复制逻辑。

键入“tab”将截面向右移动。键入“tab”+“shift”向左移动部分。

加：函数中变量的范围

个函数的内部定义屏蔽了外部的定义。为了使一个函数内部的操作能够操作外部的变量，可以使用全局关键字和非局部关键字。请查看此日志中的更多详细信息。但是，不建议使用全局变量，因为它会使程序在大小变大时难以维护。有很多方法可以解决问题。在上面的示例中，我们可以使用简单的方法通过参数列表将a传递到函数中，并使用返回语句返回更改后的值：

2）错误和异常

Try and except

对于大多数错误，Python解释器将发出异常。实际上，在许多情况下，我们需要控制可能生成异常的代码。在Python中，错误和异常处理允许我们在发生异常时继续我们的程序。

Try语句的工作方式如下。

将执行try和except（关键字）之间的try语句。

如果不发生异常，则跳过except子句，并完成try语句的执行。

如果在执行try子句期间发生异常，并且其类型与except关键字后命名的异常匹配，则将执行except子句，并在try语句之后继续执行。

如果发生与except子句中指定的异常不匹配的异常，则会将其传递给外部try语句；如果找不到处理程序，则是未处理的异常，执行将停止。

Raising errors

raise语句允许您强制发生指定的异常。它可以在下面的try和if中使用。

异常错误类型

注意，“exception”实际上是Python中的类。python有许多内置异常，这些异常分组在一个称为继承/派生的层次结构中。其好处是，代码的最外层可以捕获更广泛的异常，而代码的内层可以同时引发更窄/更具体的异常。有关内置异常的完整列表，请参阅有关异常的官方文档。

通用编码模式

表示数据集

列表/二维数组/矩阵列表

List-of-dict/ list of records/ records

Dict-of-list/ series/ column-first representation

Dict-of-dict

处理重复工作

无限循环（守护进程）

雷普尔

如果..否则；或尝试..除非

多回路

尝试子任务并重新分组成功/失败案例

重复任务的测试和批处理执行

练习和挑战

生成详细的抵押计划

表格格式提示

数学公式提示

财务报表自动编写器

计算贷方和支出分期付款

工具书类

04:第4周：数据结构

复合数据类型（集合）

**1）列表[]**

列表是按顺序排列的项目序列。它是Python中最常用的数据类型之一，非常灵活。列表中的所有项不必是同一类型。

声明列表非常简单。用逗号分隔的项目括在括号[]内。

列表的常见特征

序列中的每个元素都是有序的，并且可以被索引。第一个索引为0，第二个索引为1

列表可以添加和相乘

可以在列表中添加或删除元素

可以检查列表中是否有一个元素

一个人可以把名单分割开来

列表函数

len(list) max(list) min(list)

列表方法

append（）在列表末尾添加元素

count（）返回具有指定值的元素数

extend（）将列表的元素（或任何iterable）添加到当前列表的末尾

index（）返回具有指定值的第一个元素的索引

insert（）在指定位置添加元素

pop（）删除指定位置的元素

remove（）删除具有指定值的第一个项

reverse（）反转列表的顺序

sort（）对列表排序

在列表上的运算符中

列表切片

**2）DITC{}**

字典是一个杂乱、易变、索引的集合。

共同特点

可以通过键访问dict中的值

通过添加/删除/更新键和值更改字典

DICT函数

len(dict) str(dict)

DICT方法

fromkeys（）根据给定的序列创建字典

get（）返回键的值，默认值为none

items（）返回字典（键、值）对的视图

keys（）返回所有键的视图对象

contains(key)通过检查键是否在dict中包含（key）返回bool值

pop（）返回并删除具有给定键的元素

values（）返回字典中所有值的视图

update（）更新字典

听写中的运算符

Tuple（）

元组类似于列表，但不能修改元组的元素。一个元组的函数和方法与list相似，因此我们不再讨论。

奖励：复制集合对象

奖励：类和对象

创建一个类，init（）函数

奖金：阶级继承

练习和挑战

随机划分Hw1组：（病例贡献）

小组分配挑战提示

单字频率

额外奖励：按订单格式打印精美

奖励：使用柜台类

将香港地区名称转换成方便的数据结构

简体中文与繁体中文的转换

05:第05周：序列化-文件，csv，json

以前，我们学习了复合数据类型和基本控制流。本周，我们将学习如何获取数据、转换数据和保存数据。在我们开始从HTML中进行抓取之前，我们将介绍一种更方便的方法——API。从本周开始，您逐渐进入“大数据”的大门，API允许您同时获得更多类型和更大的数据低音。本周之后，您可以使用API方法从开放数据网站和社交媒体平台请求数据。甚至在Twitter上创建了一个自动写作机器人。

文件操作

写文件

读取文件

追加（）

猪瘟病毒

阅读器

作家

写入行

写入行

运动

杰森

JSON函数

json.dumps：将对象序列化为json格式的str

加载：将json str反序列化为python对象

json.load和json转储

练习和挑战

工具书类