**1.anaconda环境配置和解释器**

将anaconda安装目录下的scripts文件的全路径添加到系统变量path中，python的默认解释器就是anaconda了

**2. print和input**

1）print(\*objects, sep=' ', end='\n', file=sys.stdout)，无论什么类型，数值，布尔，列表，元组、字典...都可以直接输出，格式化输出中%字符是标记转换说明符的开始

其中参数：

objects -- 复数，表示可以一次输出多个对象。输出多个对象时，需要用 , 分隔。

sep -- 用来间隔多个对象，默认值是一个空格。

end -- 用来设定以什么结尾。默认值是换行符 \n，我们可以换成其他字符串。

file -- 要写入的文件对象。

python默认的print()是换行输出的，不换行输出可以print(x，end = '' )

2）Python3.x中input()函数接受一个标准输入数据，返回为string类型。当写完a = input()按回车时，光标闪烁即为在等待用户输入内容。此时，你可以输入任何字符串，输入的内容会存放到a变量中

语法格式如下：

**3.python基础讲解**

1）python变量特性+命名规则

变量命名规则：

1.变量名的长度不受限制，但其中的字符必须是字母、数字、或者下划线（\_），而不能使用空格、连字符、标点符号、引号或其他字符。

2.变量名的第一个字符不能是数字，而必须是字母或下划线。

3.Python区分大小写。

4.不能将Python关键字用作变量名。

2）注释方法

单行注释和多行注释：

Python中单行注释以#开头，多行注释用三个单引号 ''' 或者三个双引号 """ 将注释括起来。

3）python中“：”作用

冒号用于定义分片、步长，a[ : n]表示从第0个元素到第n个元素(不包括n)，a[1: ] 表示该列表中的第1个元素到最后一个元素。

4）学会使用dir( )及和help( )

dir()用来查询一个类或者对象所有属性，help()函数帮助我们了解模块、类型、对象、方法、属性的详细信息，如dir(list)，help(list)

5）import使用

通常模块为一个文件，直接使用import来导入，用逗号分割模块名称就可以同时导入多个模块，可以使用 as 关键字来改变模块的引用对象名字；通常包总是一个目录，可以使用import导入包，或者from + import来导入包中的部分模块，使用from语句可以将模块中的对象直接导入到当前的名字空间，import 语句可以在程序的任何位置使用

6）pep8介绍

PEP8-style for python code

1 缩进与换行 每级缩进使用四个空格

2 限制每行的最大长度为79个字符

3 空行

顶层函数和类之间使用两个空行

类的方法之间用一个空行

在函数中使用空行表示不同逻辑段落

4 导入位于文件的顶部

5 避免多余空格

6 注释

注释要保持与时俱进 一句后面两个空格 跟注释

7 命名规范

除了正常的命名规范外

不要使用 大小写的L 大写的O 作为变量名

类名首字母大写 内部类 加上前导下划线

函数名应该小写 增强可读性可以使用下划线分割

8 其他

别用 ‘==‘ 进行布尔值 和 True 或者 False 的比较 应该用 is

4.python数值基本知识

1）python中数值类型，int，float，bool，e记法等

创建 int 值有两种方式：直接赋予变量整数值和使用构造器 int()创建 int类型实例，针对第二种方式，如果没有任何输入参数，那么创建 int实例值为0，int()可选参数 base 表示第一个参数值所属进制，默认为 10，表示输入值为十进制数。在所有的进制中，2-进制，8-进制 和 16-进制 可以通过添加前缀 0b/0B, 0o/0O, or 0x/0X 的方式进行转换。

创建 float 值有两种方式：直接赋予变量整数值和使用构造器 float() 创建 float 类型实例，使用第一种方式，如果该数值没有小数，需要添加后缀 .0，否则，解释器会认为这是 int 类型数值，使用 float() 构造器还可以定义无穷大（Infinity 或者 inf）和无穷小

布尔型仅有两个实例对象 False 和 True，布尔型是 int 类型的子类，False 等同于 0，True 等同于 1，对于构造器 bool() 方法来说，如果输入为空或者为 0，得到 False；否则得到 True

2）算数运算符

+

-

\*两个数相乘或是返回一个被重复若干次的字符串

/

%返回除法的余数

\*\*幂，返回x的y次幂

//取整除，返回商的整数部分（向下取整）

3）逻辑运算符

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。

or：布尔"或" - 如果 x 是非 0，它返回 x 的值，否则它返回 y 的计算值。

not：布尔"非" - 如果 x 为 True，返回 False 。如果 x 为 False，它返回 True。

4）成员运算符

in：如果在指定的序列中找到值返回 True，否则返回 False。

not in：如果在指定的序列中没有找到值返回 True，否则返回 False。

5）身份运算符

用于比较两个对象的存储单元：is和is not，is 是判断两个标识符是不是引用自一个对象，is not 是判断两个标识符是不是引用自不同对象。

is 用于判断两个变量引用对象是否为同一个(同一块内存空间)， == 用于判断引用变量的值是否相等

6）运算符优先级

以下表格列出了从最高到最低优先级的所有运算符：

|  |  |
| --- | --- |
| **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 and or | 逻辑运算符 |