Python和人工智能基础课程（第二课）

张威， 雷萧萧

今日课程

- 前期回顾
- Python代码运行规则
- 变量
- 数据类型
- 算数运算
- 逻辑流程

前期回顾

- 在Sublime Text里面编写代码
- 保存代码到指定路径（桌面，test.py）
- 打开Anaconda Prompt
- 通过cd命令来切换路径，并切换到存储代码文件的路径（切换到桌面）
- 通过运行‘python test.py’来运行存储的代码
- 实现‘Hello World’和‘ask user for input’两个实例

Python代码运行规则

- 编译器以‘行’为单位来读取代码
  一行，一行，一行

- 每一行的代码分为两种类
  Statement
  print ‘Hello World’
  Assignment
  =
- 通过tab来缩进，来表示包含关系
  逻辑语句

- 如何加注释
  通过#字符来表示注释的开始
  注释是不会被运行的
  注释可以帮助自己或者别人看懂代码

变量

- 变量是数据存储的载体
  变量由它的名称，和它的值两部分组成
- 变量的生成，存储，计算都在内存中完成
- 变量有它的生命周期
  只要程序开启，一直存在
- 不同于硬盘存储
  断电不会消失
- 对变量的引用是对内存中地址的引用

变量使用

首先要给变量起个名，遵循以下原则

- 数字，下划线(_)，英文字母（大小写敏感）
- 数字不能作为开头字符
- 变量中间不能有空格

- 要有描述性

  money？password …
- 要精简
- 可读性强

  myincome2018

变量使用

赋值

=　　称为赋值号，将右边的值赋给左边

```
a = 1, b = 2
a = b
print a          返回 b
b = a
print b          返回?

a = 1
a = a + 1
print a          返回2
```

变量使用

初始化变量

- 在使用变量前一定要声明它，否则计算机不知道你在说谁

  my_income_2017 = 5000

- 声明之后便可以对变量进行引用

  print my_income_2017 返回?

变量的值可以修改

my_income_2018 = 10000 或

my_income_2018 = my_income_2017 + 3000

print my_income_2018 返回?

变量使用

变量删除

- 在使用变量之后，或在某些情景的需要时，可以删除变量

  del my_income_2018

- 变量删除之后变量不复存在

  print my_income_2018 返回?

关键字

- 有些单词不能作为变量名称
  比如：int，self，def，for，while，break，continue

- 这些被电脑保留的单词叫关键字
- 关键字不可以作为变量名称被使用

数据类型

- 整数：int
- 小数：float（和int之间的转换）
- 字符：char
- 字符串：string
- 列表：list
- 数组：array
- 元组：tuple
- 字典：dictionary

整数/整型： int (integer)

- 概念：int类型的变量存储不带小数点的整数
- 整数与整数运算，结果还是整数

  1等不等于1.0？答案是不等于

  1/1.0 或 1.0/1结果分别是什么?

- 举例

  a = 1

  b = 10000

小数/浮点数：float

- 概念：float类型的变量存储带小数点的数
- 浮点数由整数和小数部分组成

  a = 3.14

- 小数点后没内容时，要保留小数点和一位0

  income2018 = 20000.0

- 初始化小数时，小数点后用了2个或以上个0，会自动缩为1个

  a = 2.00

  print a　　返回2.0

算数运算和类型转换

- 理解int与float

1+ 2.0

2 / 3

2.0 / 3.0

- float与float运算 -> 浮点数
- float与int运算 - >浮点数

把float转化为int，会自动转化为float整数部分的值

3.9 -> 3

Boolean，布尔值

Boolean变量只有两个值True，False

True可以由1来表示/代替

False可以由0来表示/代替

print 5 == 10

返回False

Print 1 == 1.0

返回True

True的原因是，Python自动做了转换，都是数字

字符串String

- 使用引号 ‘或“来创建字符串，如a = “resent” b = ‘anomaly’
- 一个字的字符串被称为“字符”，“char” 如 x = ‘R’, y = ‘6’
- z = 6

  print y == z 结果是不等于的
- 两个变量相等：首先类型相等，其次数值相等

字符运算

- 访问子字符串中的元素或子字符串，通过[ ]来索引或范围切片

  print a[0]　　返回r

  print a[0:5]　　返回resen

  print a[-2]　　返回n

  print a[2:]

  print a[:-2]　　会返回什么呢?

字符串String

字符串相加

c = a + b

print c 返回resentanomaly

c = a + “ “ + b

print c 返回resent anomaly

member = “yzs”, score = “80”

想输出yzs的成绩：80

print member + “的成绩：” + score

数组Array

- 数组是由任意数量的数字组成的数据结构
- 只能存储数字
- 通常是做数学运算的媒介
- 常用领域包括：线性代数，统计
- 举例：
  - A =numpy.array([1,2,3,4,5]) #这是一维的数组
  - B = numpy.array([[1,2,3],[4,5,6]]) #这是二维的数组

列表list

- 列表由0至N个元素有限元素组成
- 整体由方括号[ ]包括，每个元素之间由逗号，隔开

  list0 = [ ]

  list1 = [1, 2, 3, 4, 5]

  list2 = [‘a’, ‘b’, ‘c’, ‘d’, ‘e’]
- 列表中的元素可以是相同类型，也可以是不同类型，如数字，字符串

  list3 = [‘a’, ‘b’, ‘c’, ‘d’, ‘e’, 1, 2, 3, 4, 5]
- 列表可以嵌套

  listx = [‘a’, ‘b’, 1, [1, 2, 3], 4]

列表list

- 列表中的每个元素都有它们的值和索引 index
- 索引是从0开始计算的，第一个元素是第0个元素
- 列表的索引从0开始，以自然数顺序递进，0,1,2,3,…

  list1 = [1, 2, 3, 4, 5]
- 索引为0的元素是第一项，索引为1的元素是第二项…

- 可以通过索引来访问第某个元素，获得它的值

  list1[0] = 1　　对吗?

  print1[0]　　返回1

列表list

- 索引可以是单个数，也可以是一个列表范围的子集，获得子列表

```
list1 = [1, 2, 3, 4, 5]
print list1[1:3]                返回[2, 3, 4]
```

- 切片时可以用[x:]，表示包括索引为x及以后的子列表

```
print list1[3:]                 返回[4, 5]
print list[:3]                  返回[1, 2, 3]
```

- 可以用负数来切片，表示倒数第几个，注意-1表示倒数第一个

```
print list1[-2]                 返回4
print[:-2]                      返回[1, 2, 3]
print[-2:]                      返回[4, 5]
```

列表list

更新列表中的元素

- 单个更新

```
list1[3] = 10
list1[-3] = -5
print list1                    返回[1, 2, -5, 10, 5]
```

- 使用切片部分更新

```
list1[1:2] = [11, 12]
print list1                    返回[1, 11, 12, 4, 5]
```

列表list

删除列表中的元素

- 删除整个列表

  del list1

- 删除列表中的某个元素，删除以后的列表会重新调整，删除元素之后的索引会提前

  del list1[3]

  print list1　　返回[1, 2, 3, 5]

  del list1[-3]

- 删除列表中的子集，用切片的方式

  del list1[2:3]

  del list1[2:]　　del list1[:3]

  del list1[-3:]　　del list1[:-2]

列表list

列表的常用方法

- 添加元素append( )

  print list1.append(5)　　返回[1, 2, 3, 4, 5, 5]

  print list1. append(11, 7)　　返回[1, 2, 3, 4, 5, 5, 11, 7]

- 获取元素长度len( )

  print len(list1)　　返回8

- 获取最大元素max( )

  print max(a)　　返回11

- 对列表进行排序sort( )

  print sort(a)　　返回[1, 2, 3, 4, 5, 5, 7, 11]

元组Tuple

- 元组与列表非常相似
- 由0至N个元素有限元素组成
- 整体由小括号( )包括，每个元素之间由逗号，隔开

  tup0 = ( )

  tup1 = (1, 2, 3, 4, 5)

  tup2 = (‘a’, ‘b’, ‘c’, ‘d’, ‘e’)
- 元组中的元素不能够修改，这是它与列表最大的不同
- 元组中的元素可以是相同类型，也可以是不同类型，如数字，字符串

  tup3 = (‘a’, ‘b’, ‘c’, ‘d’, ‘e’, 1, 2, 3, 4, 5)
- 元组可以嵌套

元组Tuple

- 像列表一样，元组也可以通过索引，切片来获取元素

  print tup1[1]　　返回2

  print tup1[1:2]　　返回(2, 3)

  …

- 元组也有len( ), max( )等函数

  print max(tup1)　返回5

  print len(tup1)　返回5

字典dictionary

- 字典由0至N个‘键-值’对（key-value pair）组成
  字典中包括一对儿一对儿的key-value
- 整体由花括号{ }包括，键和它的值中间是冒号：每个元素之间由逗号，隔开

```
dict0 = { }
dict1 = {‘yzs’ : ’20’, ‘zw’ : ’30’}     //‘yzs’ : ’20’ 就是一对儿键-值
```

算数运算

a = 2, b = 3
print a+b　　返回5

-	减	a - b　结果 -1
*	乘	a * b　结果 6
/	除	b / a　结果 1.5
%	取模 - 返回除法的余数	b % a　结果　1
**	幂 - 返回x的y次幂	a**b　结果　8
//	取整除 - 返回商的整数部分	b // a 输出 1

算数运算

列表与列表可以相加，元组也一样

```
a = [1, 2, 3], b = [4, 5], c = (1, 2, 3), d = (6, 7)
print a + b                    返回[1, 2, 3, 4, 5]
print c + d                    返回(1, 2, 3, 6, 7)
```

列表*一个整数n，等于赋值n次这个列表；元组也同样

```
print a*3                      返回[1, 2, 3, 1, 2, 3, 1, 2, 3]
print d*2                      返回[6, 7, 6, 7]
```

列表与元组没有其他算数运算操作

比较运算

a = 10, b = 20

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。

==　　等于 - 比较对象是否相等

print a == b

返回 False

!=　　不等于 - 比较两个对象是否不相等

print a != b

返回 true.

比较运算

<>	不等于 - 比较两个对象是否不相等 print a <> b 返回 true　这个运算符类似 !=
>	大于 - 返回x是否大于y print a > b 返回 False
<	小于 - 返回x是否小于y print a < b 返回 true
>=	大于等于 - 返回x是否大于等于y print a >= b 返回 False
<=	小于等于 - 返回x是否小于等于y print a <= b 返回 true

赋值运算

=	赋值号：将右边的值赋给左边的（一个）变量 c = a + b 将 a + b 的运算结果赋值为 c
+=	c += a 等效于 c = c + a

以此类推

-=	c -= a 等效于 c = c - a
*=	c *= a 等效于 c = c * a
/=	c /= a 等效于 c = c / a
%=	c %= a 等效于 c = c % a
**=	c **= a 等效于 c = c ** a
//=	c //= a 等效于 c = c // a

逻辑运算

任何不为0的整数，浮点数，及其他类型的变量都为True
什么为False?
0，False关键字本身，NA???

a = 0, b = 1, c = 5, d = “love”, e = [1, 2], f = (5, 6)
print a and b 返回False
print b and c 返回True
print c and d 返回True
print d and e 返回True
print e and f 返回True

逻辑运算

and　布尔“与”
x and y的结果：如果x,y全部为真，结果为True或1
如果有一项为假，结果为False或0
a = 1, b = 0　print a and b 返回0
a = 1, b = 1　print a and b 返回1

or　布尔“或“　x or y, 两项其中有一项为真，结果就位真
a = 1, b = 0　print a or b 返回True
not　布尔"非“ not x, x 为 True，not x返回 False
a = 1　print not a 返回False

逻辑运算

逻辑运算可以嵌套
a = 0, b = 1, c = 1
print a or (b and c) 返回True

print a and (b or c) 结果是?
print b and (a and c) 结果是?
print b or (a and c) 结果是?
print not c and (a or b) 结果是?

练习

- 1. 问使用者，圆的直径，算圆的面积
- 2. 月供计算，询问用户房子总价，贷款比例，还款年限，贷款利息，计算月供
- 3. 创建一个list，每回在升序排列中insert一个数字，最多10个