

python

The Python logo, consisting of two interlocking snakes, one blue and one yellow, is positioned below the word "python".

```
import turtle
turtle.setup(650,350,200,200)
turtle.penup()
turtle.fd(-250)
turtle.pendown()
turtle.pensize(25)
turtle.pencolor("purple")

for i in range(4):
    turtle.circle(40, 80)
    turtle.circle(-40, 80)
    turtle.circle(40, 80/2)
    turtle.fd(40)
    turtle.circle(16, 180)
    turtle.fd(40 * 2/3)
```

Python语言程序设计

二维数据的格式化和处理



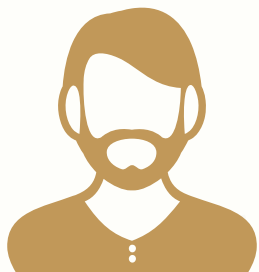
嵩 天
北京理工大学



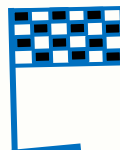


单元开篇

二维数据的格式化和处理



- 二维数据的表示
- CSV数据存储格式
- 二维数据的存储
- 二维数据的处理





二维数据的表示

二维数据的表示

使用列表类型

排名	学校名称	省市	总分	指标得分
				生源质量（新生高考成绩得分） ▾
1	清华大学	北京	94.0	100.0
2	北京大学	北京	81.2	96.1
3	浙江大学	浙江	77.8	87.2
4	上海交通大学	上海	77.5	89.4
5	复旦大学	上海	71.1	91.8
6	中国科学技术大学	安徽	65.9	91.9
7	南京大学	江苏	65.3	87.1
8	华中科技大学	湖北	63.0	80.6
9	中山大学	广东	62.7	81.1
10	哈尔滨工业大学	黑龙江	61.6	76.4

- 列表类型可以表达二维数据

- 使用二维列表

```
[ [3.1398, 3.1349, 3.1376],  
  [3.1413, 3.1404, 3.1401] ]
```

二维数据的表示

使用列表类型

```
[ [3.1398, 3.1349, 3.1376],  
  [3.1413, 3.1404, 3.1401] ]
```

- 使用两层for循环遍历每个元素
- 外层列表中每个元素可以对应一行，也可以对应一列

一二维数据的Python表示

数据维度是数据的组织形式

- 一维数据：列表和集合类型

`[3.1398, 3.1349, 3.1376]` 数据间有序

`{3.1398, 3.1349, 3.1376}` 数据间无序

- 二维数据：列表类型

`[[3.1398, 3.1349, 3.1376],
[3.1413, 3.1404, 3.1401]]`



CSV格式与二维数据存储

CSV数据存储格式

CSV: Comma-Separated Values

- 国际通用的一二维数据存储格式，一般.csv扩展名
- 每行一个一维数据，采用逗号分隔，无空行
- Excel软件可读入输出，一般编辑软件都可以产生

CSV数据存储格式

城市	环比	同比	定基
北京	101.5	120.7	121.4
上海	101.2	127.3	127.8
广州	101.3	119.4	120.0
深圳	102.0	140.0	145.5
沈阳	100.0	101.4	101.6



城市,环比,同比,定基

北京,101.5,120.7,121.4

上海,101.2,127.3,127.8

广州,101.3,119.4,120.0

深圳,102.0,140.0,145.5

沈阳,100.0,101.4,101.6

CSV数据存储格式

CSV: Comma-Separated Values

- 如果某个元素缺失，逗号仍要保留
- 二维数据的表头可以作为数据存储，也可以另行存储
- 逗号为英文半角逗号，逗号与数据之间无额外空格

二维数据的存储

按行存？按列存？

- **按行存或者按列存都可以，具体由程序决定**
- **一般索引习惯：ls[row][column]，先行后列**
- **根据一般习惯，外层列表每个元素是一行，按行存**



二维数据的处理

二维数据的读入处理

从CSV格式的文件中读入数据

```
fo = open(fname)
ls = []
for line in fo:
    line = line.replace("\n", "")
    ls.append(line.split(","))
fo.close()
```

二维数据的写入处理

将数据写入CSV格式的文件

```
ls = [[], [], []] #二维列表
```

```
f = open(fname, 'w')
```

```
for item in ls:
```

```
    f.write(','.join(item) + '\n')
```

```
f.close()
```


二维数据的逐一处理

采用二层循环

```
ls = [[], [], []] #二维列表
for row in ls:
    for column in row:
        print(ls[row][column])
```



单元小结

二维数据的格式化和处理

- 二维数据的表示：列表类型，其中每个元素也是一个列表
- CSV格式：逗号分隔表示一维，按行分隔表示二维
- 二维数据的处理：for循环+`.split()`和`.join()`



