

Data camp 笔记 1

Python list

第一位的 index 是 0

最后一位可以表示为-1

list[1:n] 提取包括 index 为 1 末尾到 n-1 的元素

已有 list 名为 area

```
area_copy = area
```

```
area_copy[5] = 1
```

也会改变 area 中原来的取值

正确赋值是 `area_copy = area[:]`

`areas.append("x")` 在 area 的 list 后面加上 x

`areas.count("x")` 计算 areas 的 list 里面有几个 x

`areas.count ("x ")` 给出 x 在 area 里的有几个

`areas.index("x")` 给出 x 在 area 里面的 index

| | type | examples of methods |
|--------|-------|---|
| Object | str | <code>capitalize()</code> <code>replace()</code> |
| Object | float | <code>bit_length()</code> <code>conjugate()</code> |
| Object | list | <code>index()</code> <code>count()</code> |

调用 numpy 的 package

这样的话使用 numpy 里面的 function array 用法是 `numpy.array`

```
import numpy as np
```

调用 numpy 的 package 名字可以命名为 np

这样的话使用 numpy 里面的 function array 用法是 `np.array`

也可以只调用 array

```
from numpy import array
```

然后直接 `array ()`

create a new array

```
np.array([1, 2,3])
```

注意一定要加括号

注意调用 package 以后里面的 dataset 是直接使用的 不用再加包的名字

例如在 numpy 里面的 dataset height_in 可以直接使用 type(height_in)

多 dimension 的 array 有点像 R 里面 matrix 的用法

如果是提取某一行某一列的 直接是 x[1, 3] 如果 x 是 array

而 x 如果是 list 则是 x[1][3]

想知道 x 的 dimension

命令是 x.shape 前提已经调用了 numpy package