

常用数据格式学习笔记

CSV, JSON 和 XML 是日常使用中常见的数据格式, 而 HDF5 则非常适用于大数据的存储和传输, 因此对以上几种数据类型进行学习。

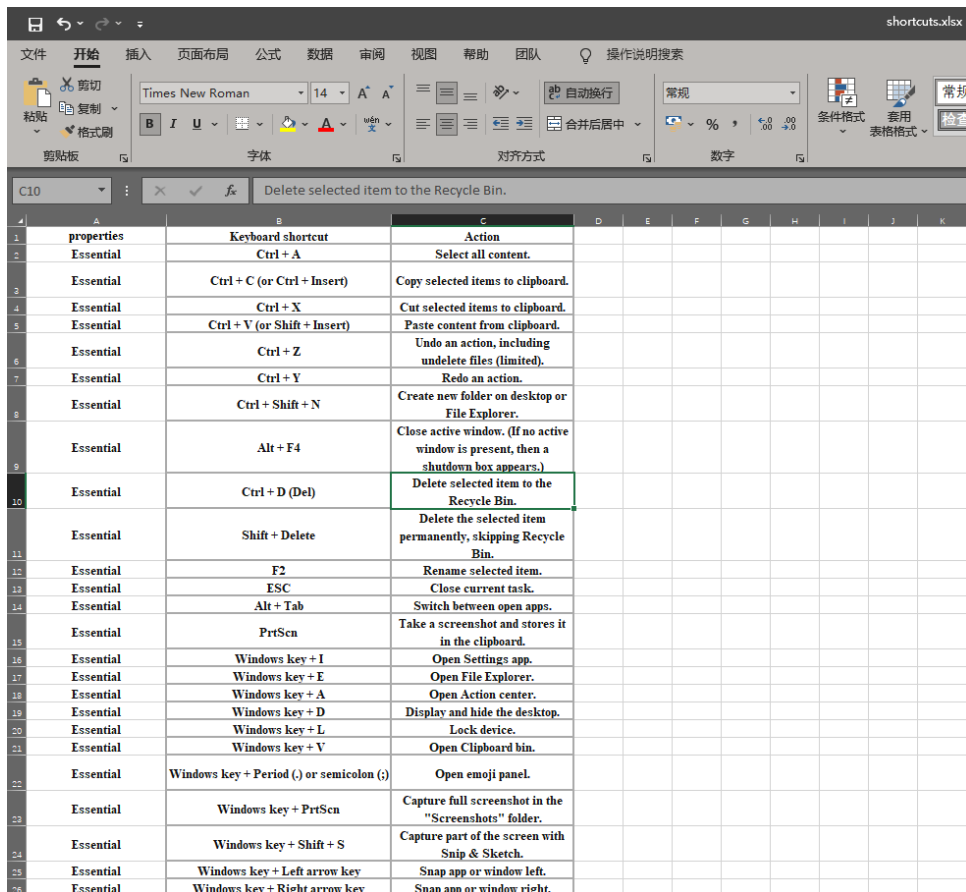
目 录

1. CSV 格式.....	1
1.1 文件写入.....	1
1.2 文件读取.....	4
1.3 CSV 转 JSON 和 XML 格式.....	5
2. JSON 格式.....	6
3. XML 格式.....	7
4. HDF5 格式.....	8

1. CSV 格式

1.1 文件写入

首先随便找了一些 Windows 常用快捷键, 创立了一个 Excel 表, 用于最初数据的读取:

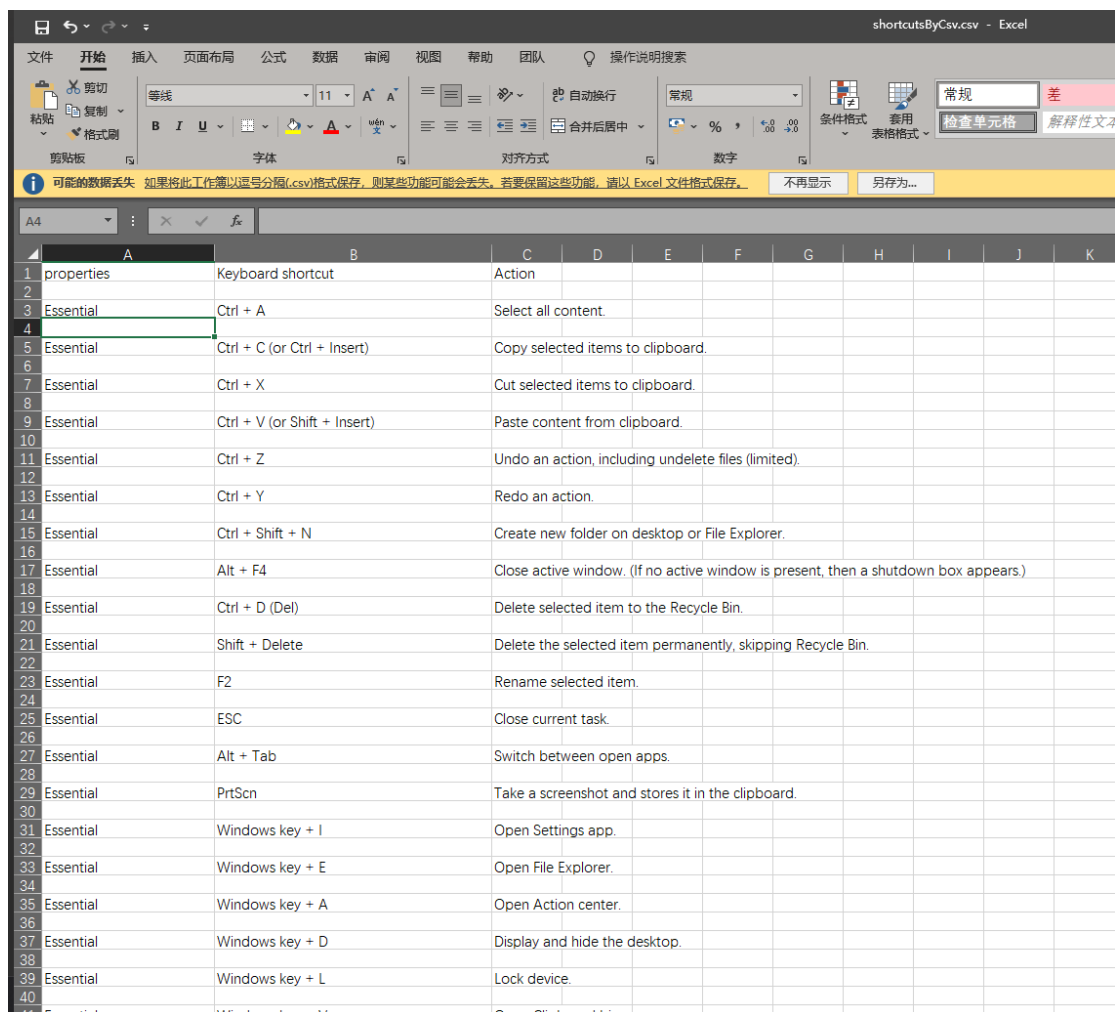


A	B	C
properties	Keyboard shortcut	Action
Essential	Ctrl + A	Select all content.
Essential	Ctrl + C (or Ctrl + Insert)	Copy selected items to clipboard.
Essential	Ctrl + X	Cut selected items to clipboard.
Essential	Ctrl + V (or Shift + Insert)	Paste content from clipboard.
Essential	Ctrl + Z	Undo an action, including undelete files (limited).
Essential	Ctrl + Y	Redo an action.
Essential	Ctrl + Shift + N	Create new folder on desktop or File Explorer.
Essential	Alt + F4	Close active window. (If no active window is present, then a shutdown box appears.)
Essential	Ctrl + D (Del)	Delete selected item to the Recycle Bin.
Essential	Shift + Delete	Delete the selected item permanently, skipping Recycle Bin.
Essential	F2	Rename selected item.
Essential	ESC	Close current task.
Essential	Alt + Tab	Switch between open apps.
Essential	PrtScn	Take a screenshot and stores it in the clipboard.
Essential	Windows key + I	Open Settings app.
Essential	Windows key + E	Open File Explorer.
Essential	Windows key + A	Open Action center.
Essential	Windows key + D	Display and hide the desktop.
Essential	Windows key + L	Lock device.
Essential	Windows key + V	Open Clipboard bin.
Essential	Windows key + Period (.) or semicolon (;)	Open emoji panel.
Essential	Windows key + PrtScn	Capture full screenshot in the "Screenshots" folder.
Essential	Windows key + Shift + S	Capture part of the screen with Snip & Sketch.
Essential	Windows key + Left arrow key	Snap app or window left.
Essential	Windows key + Right arrow key	Snap app or window right.

利用 pandas 库对 Excel 进行读取（参见 <https://geek-docs.com/pandas/pandas-read-write/pandas-to-read-and-write-excel.html>），分别利用 Python 自带的 csv 库和 pandas 库进行 CSV 的写入（参见 <https://geek-docs.com/pandas/pandas-read-write/csv-pandas-speaking-reading-and-writing.html>），代码如下：

```
01. import csv
02. import pandas as pd
03. # reading data from excel
04. data = pd.read_excel('shortcuts.xlsx')
05.
06. # writing to csv file by pandas tool
07. data.to_csv("shortcutsByPandas.csv", index=False, header=True)
08.
09. # writing to csv file by csv tool
10. filename = "shortcutsByCsv.csv"
11. with open(filename, 'w+', newline='') as csvfile:
12.     csvwriter = csv.writer(csvfile)
13.     csvwriter.writerow(data.head())
14.     csvwriter.writerows(data.values)
```

值得注意的是，在采用 csv 写入的时候，会出现空行：



	A	B	C	D	E	F	G	H	I	J	K
1	properties	Keyboard shortcut	Action								
2											
3	Essential	Ctrl + A	Select all content.								
4											
5	Essential	Ctrl + C (or Ctrl + Insert)	Copy selected items to clipboard.								
6											
7	Essential	Ctrl + X	Cut selected items to clipboard.								
8											
9	Essential	Ctrl + V (or Shift + Insert)	Paste content from clipboard.								
10											
11	Essential	Ctrl + Z	Undo an action, including undelete files (limited).								
12											
13	Essential	Ctrl + Y	Redo an action.								
14											
15	Essential	Ctrl + Shift + N	Create new folder on desktop or File Explorer.								
16											
17	Essential	Alt + F4	Close active window. (If no active window is present, then a shutdown box appears.)								
18											
19	Essential	Ctrl + D (Del)	Delete selected item to the Recycle Bin.								
20											
21	Essential	Shift + Delete	Delete the selected item permanently, skipping Recycle Bin.								
22											
23	Essential	F2	Rename selected item.								
24											
25	Essential	ESC	Close current task.								
26											
27	Essential	Alt + Tab	Switch between open apps.								
28											
29	Essential	PrtScn	Take a screenshot and stores it in the clipboard.								
30											
31	Essential	Windows key + I	Open Settings app.								
32											
33	Essential	Windows key + E	Open File Explorer.								
34											
35	Essential	Windows key + A	Open Action center.								
36											
37	Essential	Windows key + D	Display and hide the desktop.								
38											
39	Essential	Windows key + L	Lock device.								
40											
41	Essential	Windows key + V	Open Clipboard history.								

根据 python 参考手册里的解释，在 windows 这种使用\r\n的系统里，不用newline= ‘ ’的话，会自动在行尾多添加个\r，导致多出一个空行，即行尾为\r\r\n（参见<https://blog.csdn.net/pfm685757/article/details/47806469>）：

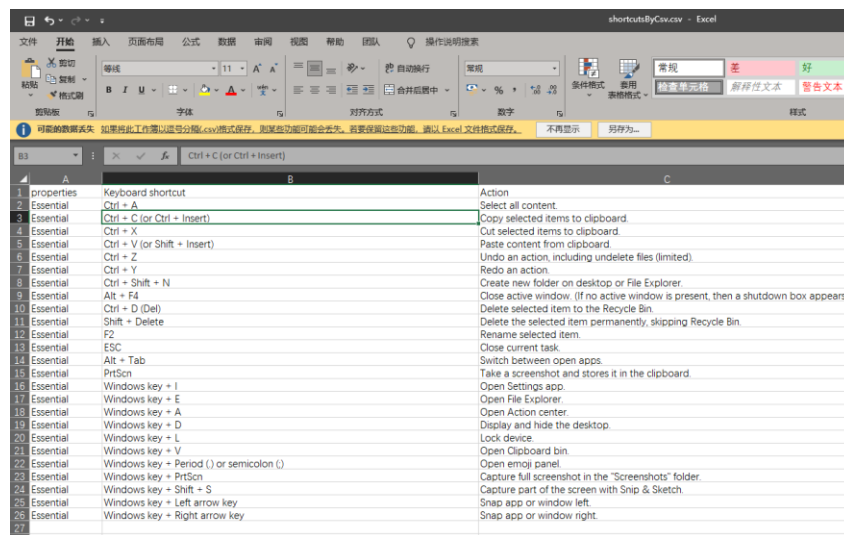
In Python 2.X, it was required to open the csvfile with 'b' because the csv module does its own line termination handling.

In Python 3.X, the csv module still does its own line termination handling, but still needs to know an encoding for Unicode strings. The correct way to open a csv file for writing is:

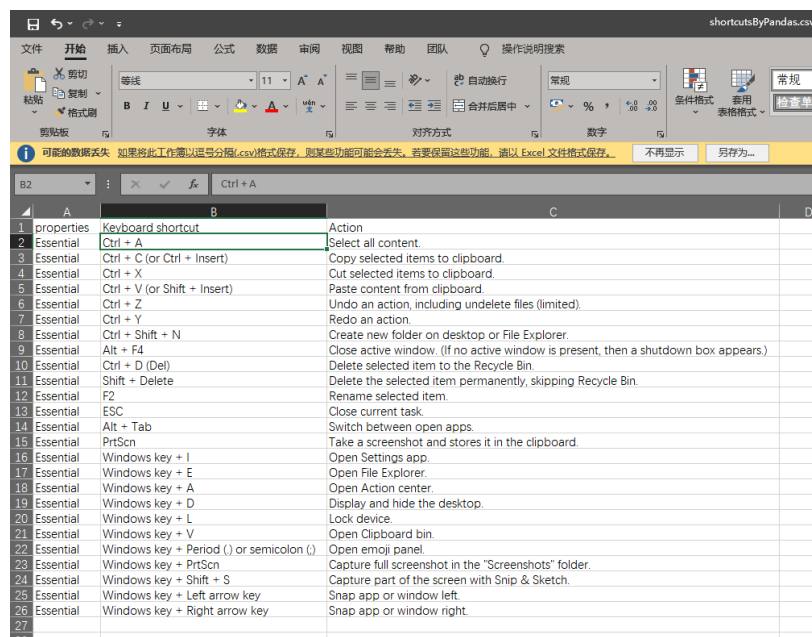
```
outputfile=open("out.csv",'w',encoding='utf8',newline='')
```

encoding can be whatever you require, but newline='' suppresses text mode newline handling. On Windows, failing to do this will write \r\r\n file line endings instead of the correct \r\n. This is mentioned in the 3.X csv.reader documentation only, but csv.writer requires it as well.

因此，需要在打开文件的时候增加 newline=‘ ’，结果如下：



采用 pandas 进行写入，默认索引和列名称连同数据一起写入，可以设置 index 和 header 选项的参数，按照自己理想模式进行写入，最终结果如下：



1.2 文件读取

分别采用 csv 和 pandas 库对 CSV 格式文件进行读取，代码如下：

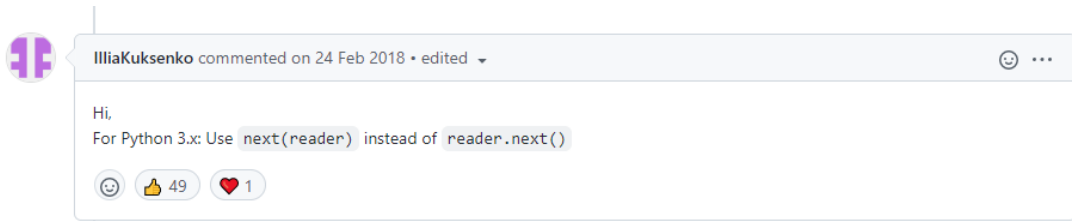
```
01. import csv
02. import pandas as pd
03. filename = "shortcuts.csv"
04.
05. # reading data from csv by pandas
06. data = pd.read_csv(filename)
07. print('Reading data from csv by pandas:')
08. print(data)
09. print('#####')
10.
11. # reading data from csv by csv
12. fields = []
13. rows = []
14. with open(filename, 'r', newline='') as csvfile:
15.     csvreader = csv.reader(csvfile)
16.     fields = next(csvreader)
17.     for row in csvreader:
18.         rows.append(row)
19. print('Reading data from csv by csv:')
20. for row in rows:
21.     print(row)
```

其结果如下：

```
PS E:\learning and training\dataFormatConversion> python -u "e:\learning and training\dataFormatConversion\readCSV.py"
Reading data from csv by pandas:
properties Keyboard shortcut Action
0 Essential Ctrl + A Select all content.
1 Essential Ctrl + C (or Ctrl + Insert) Copy selected items to clipboard.
2 Essential Ctrl + X Cut selected items to clipboard.
3 Essential Ctrl + V (or Shift + Insert) Paste content from clipboard.
4 Essential Ctrl + Z Undo an action, including undelete files (limited).
5 Essential Ctrl + Y Redo an action.
6 Essential Ctrl + Shift + N Create new folder on desktop or File Explorer.
7 Essential Alt + F4 Close active window. (If no active window is present, then a shutdown box appears.)
8 Essential Ctrl + D (Del) Delete selected item to the Recycle Bin.
9 Essential Shift + Delete Delete the selected item permanently, skipping Recycle Bin.
10 Essential F2 Rename selected item.
11 Essential ESC Close current task.
12 Essential Alt + Tab Switch between open apps.
13 Essential PrtScn Take a screenshot and stores it in the clipboard.
14 Essential Windows key + I Open Settings app.
15 Essential Windows key + E Open File Explorer.
16 Essential Windows key + A Open Action center.
17 Essential Windows key + D Display and hide the desktop.
18 Essential Windows key + L Lock device.
19 Essential Windows key + V Open Clipboard bin.
20 Essential Windows key + Period (.) or semicolon (;) Open emoji panel.
21 Essential Windows key + PrtScn Capture full screenshot in the "Screenshots" folder.
22 Essential Windows key + Shift + S Capture part of the screen with Snip & Sketch.
23 Essential Windows key + Left arrow key Snap app or window left.
24 Essential Windows key + Right arrow key Snap app or window right.
#####
Reading data from csv by csv:
['Essential', 'Ctrl + A', 'Select all content.']
['Essential', 'Ctrl + C (or Ctrl + Insert)', 'Copy selected items to clipboard.']
['Essential', 'Ctrl + X', 'Cut selected items to clipboard.']
['Essential', 'Ctrl + V (or Shift + Insert)', 'Paste content from clipboard.']
['Essential', 'Ctrl + Z', 'Undo an action, including undelete files (limited).']
['Essential', 'Ctrl + Y', 'Redo an action.']
['Essential', 'Ctrl + Shift + N', 'Create new folder on desktop or File Explorer.']
['Essential', 'Alt + F4', 'Close active window. (If no active window is present, then a shutdown box appears.)']
['Essential', 'Ctrl + D (Del)', 'Delete selected item to the Recycle Bin.']
['Essential', 'Shift + Delete', 'Delete the selected item permanently, skipping Recycle Bin.']
['Essential', 'F2', 'Rename selected item.']
['Essential', 'ESC', 'Close current task.']
['Essential', 'Alt + Tab', 'Switch between open apps.']
['Essential', 'PrtScn', 'Take a screenshot and stores it in the clipboard.']
['Essential', 'Windows key + I', 'Open Settings app.']
['Essential', 'Windows key + E', 'Open File Explorer.']
['Essential', 'Windows key + A', 'Open Action center.']
['Essential', 'Windows key + D', 'Display and hide the desktop.']
['Essential', 'Windows key + L', 'Lock device.']
['Essential', 'Windows key + V', 'Open Clipboard bin.']
['Essential', 'Windows key + Period (.) or semicolon (;)', 'Open emoji panel.']
['Essential', 'Windows key + PrtScn', 'Capture full screenshot in the "Screenshots" folder.']
['Essential', 'Windows key + Shift + S', 'Capture part of the screen with Snip & Sketch.']
['Essential', 'Windows key + Left arrow key', 'Snap app or window left.']
['Essential', 'Windows key + Right arrow key', 'Snap app or window right.']
PS E:\learning and training\dataFormatConversion>
```

可以看到，直接采用 pandas 读取的是以列表的形式展现的，而 csv 则是以 list 形式展示的。

需要注意的是，教程中的 `csvreader.next()` 是 Python2 中的用法，在这里应该使用 `next(csvreader)` 来替代（参见 <https://github.com/GalvanizeDataScience/building-spark-applications-live-lessons/issues/2>）



1.3 CSV 转 JSON 和 XML 格式

利用 Pandas，我们可以很轻松地将 csv 转换为字典列表，从而利用 dicttoxml 库将其转换为 XML 格式，利用 json 库将其保存为 JSON 格式。

```
01. import pandas as pd
02. from dicttoxml import dicttoxml
03. import json
04. filename = "shortcuts.csv"
05.
06. # reading data from csv by pandas
07. data = pd.read_csv(filename)
08.
09. # converting the dataframe to a dictionary
10. data_dict = data.to_dict(orient="records")
11. with open('csv2json.json', "w+") as json_file:
12.     json.dump(data_dict, json_file, indent=4)
13.
14. # converting the dataframe to XML
15. xml_data = dicttoxml(data_dict).decode()
16. with open("csv2xml.xml", "w+") as xml_file:
17.     xml_file.write(xml_data)
```

转换后的 JSON 格式数据如下：

```
{ } csv2json.json > ...
1  [
2  ....{
3  ....  "properties": "Essential",
4  ....  "Keyboard shortcut": "Ctrl + A",
5  ....  "Action": "Select all content."
6  ....},
7  ....{
8  ....  "properties": "Essential",
9  ....  "Keyboard shortcut": "Ctrl + C (or Ctrl + Insert)",
10 ....  "Action": "Copy selected items to clipboard."
11 ....},
12 ]
```

转换后的 XML 内容如下：

```
<?xml>
1  <essential</properties><Keyboard_shortcut type="str">Ctrl + A</Keyboard_shortcut><Action
```

2. JSON 格式

分别利用 pandas 和 json 库进行 JSON 格式文件的读写（参见 <https://geek-docs.com/pandas/pandas-read-write/pandas-reading-and-writing-json.html>），代码如下：

```
01. import json
02. import pandas as pd
03.
04. # read and write json by pandas
05. data = pd.read_json('csv2json.json', orient='records')
06. export = data.to_json('shortcutsByPandas.json', orient='records')
07.
08. # read and write json by json
09. with open('csv2json.json') as file:
10.     data = json.load(file)
11.     with open('shortcutsByJson.json', 'w+') as json_file:
12.         json.dump(data, json_file, indent=4, sort_keys=True)
```

其中，在利用 json 库中的 dump 函数进行写入时，可以通过自定义参数，修改最终的格式（参见 <https://www.jianshu.com/p/cfbcd9f8691c>），例如此处用到的 indent 参数代表了缩进的空格式，而 sort_keys 是告诉编码器按照字典 key 排序(a 到 z)输出。

通过 pandas 库所生成的 JSON 文件内容如下：

```
{ } shortcutsByPandas.json > ...
1  [{"Action": "Select all content.", "Keyboard shortcut": "Ctrl + A", "properties": "Essential"}
```

而通过 json 库生成的文件内容如下：

```
{ } shortcutsByJson.json > ...
1  [
2  .....{
3  .....    "Action": "Select all content.",
4  .....    "Keyboard shortcut": "Ctrl + A",
5  .....    "properties": "Essential"
6  .....  },
7  .....{
8  .....    "Action": "Copy selected items to clipboard.",
9  .....    "Keyboard shortcut": "Ctrl + C (or Ctrl + Insert)",
10 .....    "properties": "Essential"
11 .....  },
12 .....{
13 .....    "Action": "Cut selected items to clipboard.",
14 .....    "Keyboard shortcut": "Ctrl + X",
15 .....    "properties": "Essential"
```

可以看到，即使读取的是同一个文件，不同的库再次写入的格式大不相同，json 中的 dump 更为美观。

3. XML 格式

利用 Python 的内置 xml 模块和子模块 ElementTree，并依靠 xmltodict 和 dicttoxml，可以实现 XML 数据格式的读写，以及将其转换成 JSON 等格式（参见 <https://mp.weixin.qq.com/s/A7HOW5JXZwtRrZtic7Gdhw>），具体代码如下：

```
01. import xml.etree.ElementTree as ET
02. import xmltodict
03. # from dicttoxml import dicttoxml
04. import dicttoxml
05. import json
06.
07. tree = ET.parse('csv2xml.xml')
08. xml_data = tree.getroot()
09. xmlstr = ET.tostring(xml_data, encoding='utf8', method='xml')
10. data_dict = dict(xmltodict.parse(xmlstr))
11.
12. # write to xml
13. # xml_data = dicttoxml(data_dict).decode()
14. xml_data = dicttoxml.dicttoxml(data_dict).decode()
15. with open("shortcutsByXml.xml", "w+") as xml_file:
16.     xml_file.write(xml_data)
17.
18. # xml to json
19. with open('xml2Json.json', 'w+') as json_file:
20.     json.dump(data_dict, json_file, indent=4, sort_keys=True)
```

在第一次跑程序的过程中，遇到了一个报错：

```
>
xml_data = dicttoxml(data_dict).decode()
TypeError: 'module' object is not callable
```

通过查阅资料，发现代码里我直接 import dicttoxml，底下也直接调用了它。应该写成 from dicttoxml import dicttoxml 再直接调用 dicttoxml 函数，或是底下写成 dicttoxml.dicttoxml 的形式。这里涉及到 import 和 from import 的区别（参见 <https://cloud.tencent.com/developer/article/1579422>）。简而言之，from import 引入的是包或者说是类，而直接 import 引入的是函数或者说是方法，二者是包含关系。那么为什么不都只使用 import 呢？一来是方便起见，如果命名没有冲突且用到的包内函数较少的时候，from import 更加方便；其次直接 import 可能会出现无法直接使用包内的子包和模块（参见 <https://www.pythonf.cn/read/106215>），以后还是要多关注一下这些细节问题。

最终生成的 XML 文件如下：


```

<?xml version="1.0" encoding="UTF-8" ?><root><root type="dict"><item
type="list"><item type="dict"><key name="@type" type="str">dict</key><properties
type="dict"><key name="@type" type="str">str</key><key name="#text"
type="str">Essential</key></properties><Keyboard_shortcut type="dict"><key
name="@type" type="str">str</key><key name="#text" type="str">Ctrl + A</key></
Keyboard_shortcut><Action type="dict"><key name="@type" type="str">str</key><key
name="#text" type="str">Select all content.</key></Action></item><item
type="dict"><key name="@type" type="str">dict</key><properties type="dict"><key
name="@type" type="str">str</key><key name="#text" type="str">Essential</key></
properties><Keyboard_shortcut type="dict"><key name="@type" type="str">str</
key><key name="#text" type="str">Ctrl + C (or Ctrl + Insert)</key></
Keyboard_shortcut><Action type="dict"><key name="@type" type="str">str</key><key
name="#text" type="str">Copy selected items to clipboard.</key></Action></
item><item type="dict"><key name="@type" type="str">dict</key><properties
type="dict"><key name="@type" type="str">str</key><key name="#text"
type="str">Essential</key></properties><Keyboard_shortcut type="dict"><key
name="@type" type="str">str</key><key name="#text" type="str">Ctrl + X</key></

```

可以看到，经过 xml 库调整过的数据比之前转换完直接 dump 进去的美观很多。

4. HDF5 格式

HDF5 是一种层次化的格式（hierarchical format），经常用于存储复杂的科学数据。与其他方式对比，其速度快、压缩效率高，适合大数据的存储。

HDF5 文件结构中有两个关键对象，一个是 Groups，另一个是 Datasets。Groups 就类似于文件夹，每个 HDF5 文件其实就是根目录；而 Datasets 类似于 NumPy 中的数组 array。每个 Dataset 可以分成两部分：原始数据 (raw) data values 和元数据（参见 <https://zhuanlan.zhihu.com/p/104145585>）。

简单生成一个 H5 文件，并读取打印其 key 和 value，代码如下：

```

01. import h5py
02. import numpy as np
03.
04. # writing to h5
05. imgData = np.zeros((30, 3, 128, 256))
06. f = h5py.File('testH5.h5', 'w')
07. f['data'] = imgData
08. f['labels'] = range(100)
09. f.close()
10.
11. # reading from h5
12. f = h5py.File('testH5.h5', 'r')
13. print('key:')
14. print(f.keys())
15. print('#####')
16. a = f['data'][:]
17. print('value:')
18. print(a)
19. f.close()

```

其输出如下：


```

key:
KeysView(<HDF5 file "testH5.h5" (mode r)>)
#####
value:
[[[0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  ...
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]]

[[[0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  ...
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]]

[[[0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  ...
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]]

[[[0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  ...
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]]

[[[0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  ...
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]]

[[[0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  ...
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]]

[[[0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  ...
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]
  [0. 0. 0. ... 0. 0. 0.]]

```