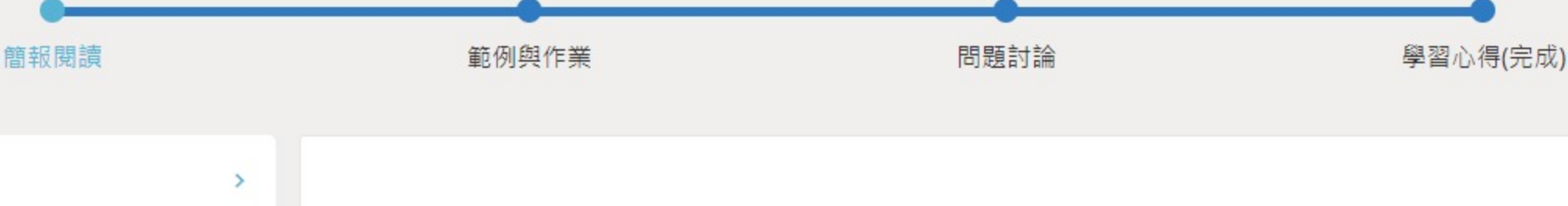


D09 使用 Pandas 讀寫各種常用的檔案格式



>

重要知識點

>

讀寫 csv

>

讀寫 excel

>

讀寫 json

>

讀寫 SQL 資料庫



重要知識點



- 讀寫csv
- 讀寫excel
- 讀寫json
- 讀寫SQL資料庫

讀寫 csv

CSV的資料由很多文本資料組成，資料之間以逗點隔開。

首先我們使用 Pandas 的套件中 `pd.read_csv` 讀取一個 CSV 檔案夾中的 `iris.csv`，讀取之後即為 `DataFrame` 的資料型態，有時候資料太多不想一次讀取這麼多行可以使用 `usecols` 參數指定讀取的行名稱。

```
iris_data = pd.read_csv('iris.csv')
iris_data
```

	sepal length	sepal width	petal length	petal width	target
0	5.1	3.5	1.4	0.2	0
1	4.9	3.0	1.4	0.2	0
2	4.7	3.2	1.3	0.2	0
3	4.6	3.1	1.5	0.2	0
4	5.0	3.6	1.4	0.2	0

有時候資料太多不想一次讀取這麼多行可以使用 `usecols` 參數指定讀取的行名稱。

```
iris_data = pd.read_csv('iris.csv', usecols=['petal length', 'petal width', 'target'])
iris_data
```

	petal length	petal width	target
0	1.4	0.2	0
1	1.4	0.2	0
2	1.3	0.2	0
3	1.5	0.2	0
4	1.4	0.2	0
5	1.7	0.4	0

可以再資料圖取中利用 `names` 參數指定行的名稱，因為有指定行名稱所以必須以 `header=0` 參數跳過檔案裡放置行名稱的列。

```
iris_data = pd.read_csv('iris.csv', header=0, names=['feature1', 'feature2', 'feature3', 'feature4', 'target'])
iris_data
```

	feature1	feature2	feature3	feature4	target
0	5.1	3.5	1.4	0.2	0
1	4.9	3.0	1.4	0.2	0
2	4.7	3.2	1.3	0.2	0
3	4.6	3.1	1.5	0.2	0
4	5.0	3.6	1.4	0.2	0
5	5.4	3.9	1.7	0.4	0

輸出 csv 檔案使用 `.to_csv()` 的方法就可以將資料存到指定路徑下。

```
iris_data.to_csv('my_iris.csv')
```

讀寫 excel

Pandas 利用 `pd.read_excel()` 函數支援讀取 Excel 2003 之後的格式資料，此方法內部是使用 `XLRD` 或 `OpenPyXL` 套件，所以使用前確認至少安裝其中一項。

```
data = pd.read_excel('data.xls')
data
```

	sepal length	sepal width	petal length	petal width	target
0	5.1	3.5	1.4	0.2	0
1	4.9	3.0	1.4	0.2	0
2	4.7	3.2	1.3	0.2	0
3	4.6	3.1	1.5	0.2	0
4	5.0	3.6	1.4	0.2	0

`pd.read_excel()` 只會讀第一個工作簿，如今天 excel 工作簿不只一個，我們可以使用 `sheetname` 參數傳入要讀取頁面的名稱。

```
boston_data = pd.read_excel('data.xls', sheet_name='boston')
boston_data
```

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
0	0.00632	18.0	2.31	0	0.538	6.575	65.2	4.0900	1	296	15.3	396.90	4.98
1	0.02731	0.0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.90	9.14
2	0.02729	0.0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03
3	0.03237	0.0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94
4	0.06905	0.0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18.7	396.90	5.33

`pd.read_excel` 和 `pd.read_csv` 一樣有 `usecols`、`header`、`names` 可以使用

```
boston_data = pd.read_excel('data.xls', sheet_name='boston', header=0 \
                             , usecols=['TAX', 'PTRATIO', 'B', 'LSTAT'])
boston_data
```

	TAX	PTRATIO	B	LSTAT
0	296	15.3	396.90	4.98
1	242	17.8	396.90	9.14
2	242	17.8	392.83	4.03
3	222	18.7	394.63	2.94
4	222	18.7	396.90	5.33

輸出 excel 檔案使用 `.to_excel()` 的方法就可以將資料存到指定路徑下，也可以使用 `sheet_name` 更改工作簿名稱。

```
boston_data.to_excel('my_boston.xlsx', sheet_name='boston')
```

讀寫 json

利用 `pd.read_json()` 函數支援讀取 JavaScript 物件表示法 (JSON) 格式資料，json 具有跨平台與程式語言的可攜性。

```
boston_data = pd.read_json('boston.json')
boston_data
```

	TAX	PTRATIO	B	LSTAT
0	296	15.3	396.90	4.98
1	242	17.8	396.90	9.14
2	242	17.8	392.83	4.03
3	222	18.7	394.63	2.94
4	222	18.7	396.90	5.33

輸出 json 檔案使用 `.to_json()` 的方法就可以將資料存到指定路徑下。

```
boston_data.to_json('boston.json')
```

讀寫 SQL 資料庫

任何 SQL 資料庫如果支援遵守 Python DB-API 都可以被 Pandas 讀取，以下先用 `boston.csv` 的資料寫入 `SQLite3` 資料庫中並且命名 `boston`，由 `if_exists` 參數判斷是否存在資料庫，如果檔案不存在會立即被建立，如果存在 `if_exists='replace'` 將會取代掉原本資料，`if_exists='append'` 將會繼續寫在原有資料下。

```
import sqlite3

boston_data = pd.read_excel('data.xls', sheet_name='boston', header=0 \
                             , usecols=['TAX', 'PTRATIO', 'B', 'LSTAT'])
connection = sqlite3.connect('sql_db.sqlite')
boston_data.to_sql('boston', connection, if_exists='replace')
connection.commit()
connection.close()
```

讀取 `SQLite3` 資料庫可以使用 `pd.io.sql.read_sql`，可以直接下 SQL 指令對 `sql_db` 中的 `boston` 做搜尋。

```
connection = sqlite3.connect('sql_db.sqlite')
boston_data_sql = pd.io.sql.read_sql("select * from boston", connection)
connection.close()
boston_data_sql
```

	index	TAX	PTRATIO	B	LSTAT
0	0	296	15.3	396.90	4.98
1	1	242	17.8	396.90	9.14
2	2	242	17.8	392.83	4.03
3	3	222	18.7	394.63	2.94
4	4	222	18.7	396.90	5.33

知識點回顧

- 資料型態多元，我們可以使用 Pandas 針對不同資料型態讀取，讀寫 `csv(read_csv、to_csv)`、讀寫 `excel(read_excel、to_excel)`、讀寫 `json(read_json、to_json)`、讀寫 `SQL 資料庫(io.sql.read_sql、to_sql)`，讀取進來都是 `DataFrame` 的型態，方便之後進行資料的分析，也可以藉由 Pandas 做資料型態的轉換。

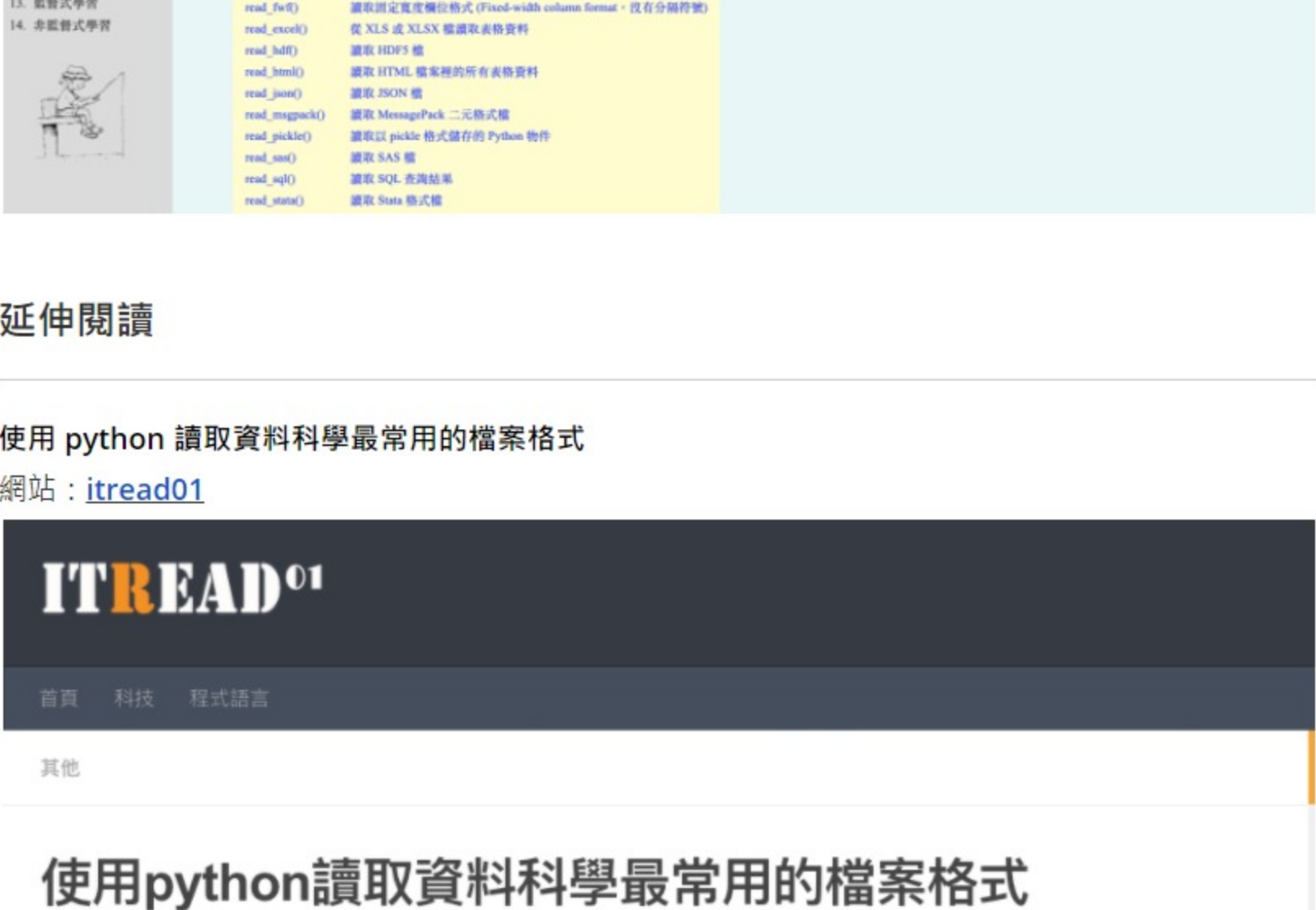
補充教材

python pandas 讀取檔案

網站：[itread01](#)



網站：[yltang](#)



延伸閱讀

使用 python 讀取資料科學最常用的檔案格式

網站：[itread01](#)



Python 數據分析之 Pandas 讀寫外部數據文件

網站：[kknews](#)

Python數據分析之Pandas讀寫外部數據文件

2019-09-07 由 編程python新視野 發表于程式開發

1 引言

數據分析、數據挖掘、可視化是Python的眾多強項之一，但無論是這幾項中的哪一項都必須以數據作為基礎，數據通常都存儲在外部文件中，例如txt、csv、excel、資料庫。本篇中，我們將來一探Python中那些外部數據文件讀取、寫入的常用方法。

undefined

[下一步：閱讀範例與完成作業](#)