

# 電子商務技術 HW1

sales.xlsx 為公司的產品銷售紀錄，此資料集目前有欄位錯置的問題(e.g.A欄跑到B欄)。請將此資料集整理成正確的格式

---

## 已知資訊

- (1) 欄位的命名方式都是大寫字母開頭
  - (2) 公司共有以下 5 種 Product line：「Camping Equipment、Golf Equipment、Mountaineering Equipment、Outdoor Protection、Personal Accessories」
  - (3) 後續的分析作業只對「Climbing Accessories、Cooking Gear、First Aid、Golf Accessories、Insect Repellents、Sleeping Bags」這 6 種產品類型(Product type)感興趣，因此最後產出的資料集請過濾掉不相干的產品類型的銷售紀錄。
  - (4) 公司共提供以下 7 種下單方式(Order method type)：「E-mail、Fax、Mail、Sales visit、Special、Telephone、Web」
  - (5) 總共有13個欄位，參考第13題答案
- 

## 題目 (總分 100)

\* 第 8~11 題可以不用完全按照題目順序，但請標明提號！

1. 使用 pandas 將資料集載入為 DataFrame 格式。(2%)
2. 列出目前資料集所有欄位名稱及資料型態。(2%)
3. 依已知資訊(1)，請列出所有正確的欄位名稱。(2%)
4. 請檢查第 1 個欄位 Year 是否有資料錯置問題。(2%)
5. 請檢查第 2 個欄位 Product line 的資料錯置情況，並將此欄位整理成正確的形式。(2%)
6. 請依已知資訊(3)過濾掉不相干的 Product type 的銷售紀錄 (移除一整列)，請保留每一筆資料原本的 index (不需重置 index)。(10%)
7. 請檢查第 3 個欄位 Product type 的資料錯置情況，並將此欄位整理成正確的形式。(5%)
8. 請檢查第 4 個欄位 Product 的資料錯置情況，並將此欄位整理成正確的形式。(30%)
9. 請檢查第 5 個欄位 Order method type 的資料錯置情況，並將此欄位整理成正確的形式。(15%)
10. 請檢查第 6 個欄位Retailer country 的資料錯置情況，並將此欄位整理成正確的形式。(15%)
11. 請將剩下的欄位整理成正確的形式。(10%)

12. 請將整理完的資料集以 **index** 排序並匯出成 **csv** 檔(需保留 **index**)，檔名為 **HW1\_學號.csv**。(最後輸出的資料集須移除不需要的欄位) **(3%)**
13. 檢查處理完後的資料集是否能產出如下的結果。 **(2%)**

```
sales_cleaned.groupby('Product type').count()
```

	Year	Product line	Product	Order method type	Retailer country	Revenue	Planned revenue	Product cost	Quantity	Unit cost	Unit price	Gross profit	Unit sale price
Product type													
Climbing Accessories	3087	3087	3087	3087	3087	729	729	729	729	729	729	729	729
Cooking Gear	5880	5880	5880	5880	5880	2059	2059	2059	2059	2059	2059	2059	2059
First Aid	2205	2205	2205	2205	2205	812	812	812	812	812	812	812	812
Golf Accessories	1764	1764	1764	1764	1764	619	619	619	619	619	619	619	619
Insect Repellents	2205	2205	2205	2205	2205	795	795	795	795	795	795	795	795
Sleeping Bags	3087	3087	3087	3087	3087	1189	1189	1189	1189	1189	1189	1189	1189

## 作業繳交說明

- 繳交期限：3/1 00:00
- 請繳交 **.ipynb** 檔和處理完的資料集，檔名分別為 **HW1\_學號.ipynb** 和 **HW1\_學號.csv**  
 程式中請以註解或文字方塊標示題號及適當說明  
 沒標示好會斟酌扣分!
- 上傳至 **ee-class** 作業區，遲交一天扣該次作業得分 5%

## 作業提示

### (I) 檢查是否有欄位錯置 → 利用 **groupby()** 及 **size()**

- 以第 1 個欄位(**Year**)為例

由以下兩圖可看到第 2 個欄位(**Product**)的值都與年份無關，可推論 **Year** 欄位沒有位置錯誤問題。(沒錯置到下一欄位，就不會錯置到更後面的欄位)

sales.groupby(['Year']).size()		sales.groupby(['Year', 'Product']).size()	
Year		Year	Product
2004	21168	2004	Camping 6027
2005	21168		Golf 2205
2006	21168		Mountaineering 3087
2007	2031		Outdoor 2205
dtype: int64			Personal 7644
		2005	Camping 6027
			Golf 2205
			Mountaineering 3087
			Outdoor 2205
			Personal 7644
		2006	Camping 6027
			Golf 2205
			Mountaineering 3087
			Outdoor 2205
			Personal 7644
		2007	Camping 2031
		dtype: int64	

- 由已知資訊可知 **Product line** 皆由兩個單字組成，可先猜 **Product line** 的每一筆資料都會佔據兩個欄位。以下兩圖使用 `groupby()` 及 `size()` 確認了 **Product line** 佔「**Product**」及「**line**」兩個欄位。

sales.groupby(['Product', 'line', 'Product.1']).size()			
Product	line	Product.1	
Camping	Equipment	Cooking	5880
		Lanterns	5292
		Packs	2646
		Sleeping	3087
		Tents	3207
Golf	Equipment	Golf	1764
		Irons	1764
		Putters	1323
		Woods	1764
Mountaineering	Equipment	Climbing	3087
		Rope	1764
		Safety	1764
		Tools	2646
Outdoor	Protection	First	2205
		Insect	2205
		Sunscreen	2205
Personal	Accessories	Binoculars	2646
		Eyewear	7056
		Knives	3087
		Navigation	4410
		Watches	5733

dtype: int64

- (II) 確認好錯置情況後，將欄位整理成正確的形式

- 將原本的「**Product**」及「**line**」欄位合併，並將合併後的值指定給新增的欄位 **Product line**。
- 使用 **drop()** 移除不需要的欄位。

[illegible]

### (III) 當遇到每一筆資料所佔據的欄位數量不一致時 → 針對每種長度個別處理

#### 以第 4 個欄位(Product)為例

- 由以下兩圖可發現 Product 欄位的值有些只占 1 個欄位，有些占 2 個、3 個或更多。
- 可將原本的 DataFrame 依照 Product 的單字數量分成不同的小 DataFrame，個別處理後再合併成最終的 DataFrame。可參考(IV)(V)做分割與合併。

```
sales.groupby(['Product.2', 'Order']).size()
```

Product.2	Order	
Aloe	Relief	441
BugShield	Extreme	441
	Lotion	882
	Natural	441
	Spray	441
Calamine	Relief	441
Compact	Relief	441
Course	Pro	1764
Deluxe	Family	441
Firefly	Charger	441
	Climbing	441
	Rechargeable	441
Granite	Belay	441
	Carabiner	441
	Chalk	441
	Pulley	441
Hibernator	Camp	441
	E-mail	63
	Extreme	441

```
sales.groupby(['Product.2', 'Order', 'method']).size()
```

Product.2	Order	method	
Aloe	Relief	E-mail	63
		Fax	63
		Mail	63
		Sales	63
		Special	63
		...	
TrailChef	Utensils	Sales	84
		Special	84
		Telephone	84
		Web	84
	Water	Bag	588
Length: 265, dtype: int64			

### (IV) Isin()與 copy(deep=True)

#### 以第 4 個欄位(Product)為例

- 使用 isin()挑選資料列
- p1 代表的是所有 Product 名稱只占 1 個欄位的紀錄(row)。

```
p1 = sales[sales['Order'].isin(['E-mail', 'Fax', 'Mail', 'Sales', 'Special', 'Telephone', 'Web'])].copy(deep=True)
p1.groupby(['Product.2', 'Order']).size()
```

Product.2	Order	
Hibernator	E-mail	63
	Fax	63
	Mail	63
	Sales	63
	Special	63
	Telephone	63
	Web	63

dtype: int64

- 使用「~」否定 isin()的條件。n1 代表的是所有 Product 名稱占超過 1 個欄位的紀錄。

```
n1 = sales[~sales['Order'].isin(['E-mail', 'Fax', 'Mail', 'Sales', 'Special', 'Telephone', 'Web'])].copy(deep=True)
n1.groupby(['Product.2', 'Order']).size()
```

Product.2	Order	
Aloe	Relief	441
BugShield	Extreme	441
	Lotion	882
	Natural	441
	Spray	441
Calamine	Relief	441
Compact	Relief	441

- 可使用如 p1 的方式從小到大依序找出所有長度的 Product 的 DataFrame，也可用如 n1 的方式從大到小來找。
- **注意：在分割 DataFrame 時使用 deep copy 可避免後續的操作出現 Warning 或影響到其他資料。(詳細可查看 shallow copy / deep copy 的相關資料)**

#### (V) pandas.concat()

- 分別處理完每個小 DataFrame 後，最後可使用 concat 合併
- 例如

```
import pandas as pd  
final = pd.concat([p1, p2, p3, p4])
```

#### (VI) 不只有 Product 欄位要分開處理，可能會需要繼續往下分割，如下圖。

