

隨堂練習4

題目：

某位資料科學家想將各個人的基本資料做統計，但他不想用之後每次都用手寫的或是用excel慢慢打，於是想要使用程式來建立一個個人資料表來做後續的資料分析，避免以後資料一多就很難快速分析了。其中他要將張乃云, 蔡芸琤, 許家瑋這3位的資料先記錄下來並做示範。以下是他們的職業、居住城市以及畢業高中的資料表。

姓名	職業	城市	畢業高中
張乃云	工程師	台北	新竹女中
蔡芸琤	教授	台東	台東女中
許家瑋	學生	新北	新莊高中

image.png

這位科學家發現，他還需要增加興趣、性別這2個欄位，但是職業的部分就不需要了。於是他想要這樣實現：

姓名	城市	畢業高中	興趣	性別
張乃云	台北	新竹女中	探尋美食	女
蔡芸琤	台東	台東女中	寫程式	女
許家瑋	新北	新莊高中	睡覺	男

image.png

請根據以上需求，將第一張與第二張的變換過程之程式碼寫出。

目標

將張乃云、蔡芸琤、許家瑋3位的城市、畢業高中、興趣、性別等欄位對應的字串吐出，並將職業一欄刪除。

輸入格式

本題無輸入格式

輸出格式

	姓名	城市	畢業高中	興趣	性別
0	張乃云	台北	新竹女中	探尋美食	女
1	蔡芸琤	台東	台東女中	寫程式	女
2	許家瑋	新北	新莊高中	睡覺	男

Hint

```
df = pd.DataFrame(data)
df['興趣'] = ['探尋美食', '寫程式', '睡覺']
df['性別'] = ['女', '女', '男']
df = df.drop('職業', axis=1) # 刪除 "職業" 列
print(df)
```

步驟分解:

步驟 1: 匯入 **Pandas** 函式庫

```
import pandas as pd
```

步驟 2: 建立原始數據字典

```
data = {
    '姓名': ['張乃云', '蔡芸琤', '許家瑋'],
    '職業': ['工程師', '教授', '學生'],
    '城市': ['台北', '台東', '新北'],
    '畢業高中': ['新竹女中', '台東女中', '新莊高中']
}
```

步驟 3: 將字典轉換為 **DataFrame**

```
1df = pd.DataFrame(data)
```

解釋:

- 使用 `pd.DataFrame(data)` 將字典 `data` 轉換為 Pandas DataFrame, 形成一個表格結構, 方便後續操作。

補充:

`DataFrame` 是 **Pandas** 函式庫中的核心資料結構, 用來儲存和處理表格數據。它類似於 **Excel** 表格 或 **SQL** 資料表, 由 列(**columns**) 和 行(**rows**) 組成。(想像你有一個 **Excel** 表格, 裡面有很多行和很多列, 這張表格就是 **DataFrame** ! 它就像一個「會寫程式的 Excel」, 可以讓你很輕鬆地讀取、修改、分析資料, 而前面的2張圖片就是一種 `DataFrame`。)

DataFrame 的特點

- 結構化表格數據: 如同 Excel 一樣, 每列代表一個變數, 每行代表一個值。
- 支援多種數據類型: 不同欄位可以存放不同的數據類型(數字、字串、布林值等)。
- 方便的資料操作: 可進行篩選、排序、統計、轉換等操作。

DataFrame 的組成

- 索引(**Index**): 用來標識行(類似於 Excel 的行號)。
- 欄位(**Columns**): 表格的標題, 每個欄位代表一種變數。
- 值(**Values**): 儲存的數據內容。

步驟 4: 新增「興趣」、「性別」欄位

```
df['興趣'] = ['探尋美食', '寫程式', '睡覺']  
df['性別'] = ['女', '女', '男']
```

解釋:

- 程式碼在 `df` 中新增欄位, 並為三位人物分別填入對應的興趣、性別。

步驟 5: 刪除「職業」欄位

```
df = df.drop('職業', axis=1) # 刪除 "職業" 列
```

解釋：

- `df.drop('職業', axis=1)` 用來刪除 DataFrame 中的 職業 欄位。
- `axis=1` 代表對「欄」進行操作(`axis=0` 代表刪除「列」)。
- 由於 `drop()` 預設不會直接修改 `df`, 所以需要重新指派 `df = df.drop('職業', axis=1)`, 讓變數 `df` 存儲刪除後的新 DataFrame。

步驟 6: 輸出最終的 DataFrame

```
print(df)
```