Bài tập Tiền xử lý dữ liệu 1 - Mẫu

Họ và tên: Lê Hoàng Vũ

Mã sinh viên: 23A4040156

Khai báo thư viện

In [29]:
```
import pandas as pd # Thư viện xử lý dữ liệu
import matplotlib.pyplot as plt # Thư viện trực quan hóa dữ liệu
import numpy as np # Thư viện thực hiện các mảng với dữ liệu
```

Đọc dữ liệu từ tệp CSV với Pandas

> Trong thư viện Pandas, DataFrame là một cấu trúc dữ liệu 2 chiều, giống mảng 2 chiều, gồm hàng và cột. Dữ liệu được lấy từ CSV sẽ được trả về dạng DataFrame. Sử dụng phương thức loc[...] để lấy ra giá trị hàng muốn lấy

In [30]:
```
df1_1 = pd.read_csv("bank-data-1.1-6thuoctinh.csv")
df1_2 = pd.read_csv("bank-data-1.2-7thuoctinh.csv")
df2 = pd.read_csv("bank-data-2-12thuoctinh.csv")
```

Kiểm tra kích thước tệp dữ liệu

> Thuộc tính shape đưa ra thông tin về hàng và cột

In [31]:
```
print("Kích thước của tập dữ liệu 1.1 ", df1_1.shape)
print("Kích thước của tập dữ liệu 1.2 ", df1_2.shape)
print("Kích thước của tập dữ liệu 2 ", df2.shape)
```

```
Kích thước của tập dữ liệu 1.1  (300, 6)
Kích thước của tập dữ liệu 1.2  (300, 7)
Kích thước của tập dữ liệu 2  (300, 12)
```

Đọc 10 hàng dữ liệu đầu tiên của tập dữ liệu 1.1

In [32]:
```
df1_1.head(10)
```

Out[32]:

	ID	age	sex	region	income	married
0	1	48	FEMALE	INNER_CITY	17546.0	NO
1	2	40	MALE	TOWN	30085.1	YES
2	3	51	FEMALE	INNER_CITY	16575.4	YES
3	4	23	FEMALE	TOWN	20375.4	YES
4	5	57	FEMALE	RURAL	50576.3	YES
5	6	57	FEMALE	TOWN	37869.6	YES
6	7	22	NaN	RURAL	NaN	NO
7	8	58	MALE	TOWN	24946.6	YES
8	9	37	FEMALE	SUBURBAN	25304.3	YES
9	10	200	MALE	TOWN	24212.1	YES

Kiểm tra các giá trị khuyết thiếu

Thư viện Pandas cung cấp một số cách thức để kiểm tra:

- Phương thức isnull()
- Phương thức notnull()
- Phương thức value_counts(): Trả về số dòng các giá trị khác nhau (frequency of unique value)
- Phương thức count(): Trả về số dòng/cột không null

In [46]:

```
print("Thông tin số lượng các giá trị không rỗng từng thuộc tính")
print(df1_1.count(), "\n")

print("Thông tin tần suất từng giá trị giới tính")
print(df1_1['sex'].value_counts(dropna=False), "\n")
# Tham số dropna mặc định là True, không bao gồm các số dòng có chứa giá trị null
# Tệp dữ liệu trong DataFrame sử dụng ['column_name'] để chọn cột

print("Thông tin số lượng các giá trị không rỗng thuộc tính sex")
print("Số lượng: " , df1_1['sex'].count(), "\n")

print("Kiểm tra dữ liệu nào bị thiếu thông tin thuộc tính sex")
print(df1_1['sex'].isnull(), "\n")
# Nên dùng vòng lặp để lấy ra những hàng bị thiếu thông tin

print("Kiểm tra dữ liệu nào bị thiếu thông tin từng thuộc tính")
print(df1_1.isnull(), "\n")
```

```
Thông tin số lượng các giá trị không rỗng từng thuộc tính
ID         300
age        300
sex        297
region     300
income     298
married    300
dtype: int64

Thông tin tần suất từng giá trị giới tính
MALE      154
FEMALE    143
NaN         3
Name: sex, dtype: int64

Thông tin số lượng các giá trị không rỗng thuộc tính sex
Số lượng:  297

Kiểm tra dữ liệu nào bị thiếu thông tin thuộc tính sex
0      False
1      False
2      False
3      False
4      False
       ...
295    False
296    False
297    False
298    False
299    False
Name: sex, Length: 300, dtype: bool

Kiểm tra dữ liệu nào bị thiếu thông tin từng thuộc tính
        ID    age    sex  region  income  married
0    False  False  False   False   False    False
1    False  False  False   False   False    False
2    False  False  False   False   False    False
3    False  False  False   False   False    False
4    False  False  False   False   False    False
..     ...    ...    ...     ...     ...      ...
295  False  False  False   False   False    False
296  False  False  False   False   False    False
297  False  False  False   False   False    False
298  False  False  False   False   False    False
299  False  False  False   False   False    False

[300 rows x 6 columns]
```

Xóa dữ liệu khuyết thiếu với phương thức dropna()

> Phương thức dropna() trả về kết quả một cấu trúc DataFrame mới, là bản sao chép của tập dữ liệu đang được xử lý xóa những dữ liệu có thuộc tính khuyết thiếu

In [47]:
```
drop_df1_1 = df1_1.dropna()
print("Thông tin tập dữ liệu sau khi xóa các dữ liệu có chứa thuộc tính khuyết thiế
print(drop_df1_1)
```

```
Thông tin tập dữ liệu sau khi xóa các dữ liệu có chứa thuộc tính khuyết thiếu
      ID  age     sex      region    income married
0      1   48  FEMALE  INNER_CITY  17546.00      NO
1      2   40    MALE        TOWN  30085.10     YES
2      3   51  FEMALE  INNER_CITY  16575.40     YES
3      4   23  FEMALE        TOWN  20375.40     YES
4      5   57  FEMALE       RURAL  50576.30     YES
..   ...  ...     ...         ...       ...     ...
295  296   34    MALE        TOWN  32548.90     YES
296  297   54  FEMALE       RURAL  24583.40      NO
297  298   18    MALE       RURAL   8639.24     YES
298  299   47  FEMALE  INNER_CITY  17139.50      NO
299  300   24  FEMALE  INNER_CITY  13667.70     YES

[296 rows x 6 columns]
```

Điền thay thế giá trị khuyết thiếu thuộc tính sex với phương thức fillna()

In [49]:
```
df1_1['sex'] = df1_1['sex'].fillna('LGBT')
# Hoặc cũng có thể truyền tham số inplace = True vào fillna()
# để tác động tới dữ liệu trên DataFrame gốc

df1_1.head(10)
```

Out[49]:

	ID	age	sex	region	income	married
0	1	48	FEMALE	INNER_CITY	17546.0	NO
1	2	40	MALE	TOWN	30085.1	YES
2	3	51	FEMALE	INNER_CITY	16575.4	YES
3	4	23	FEMALE	TOWN	20375.4	YES
4	5	57	FEMALE	RURAL	50576.3	YES
5	6	57	FEMALE	TOWN	37869.6	YES
6	7	22	LGBT	RURAL	NaN	NO
7	8	58	MALE	TOWN	24946.6	YES
8	9	37	FEMALE	SUBURBAN	25304.3	YES
9	10	200	MALE	TOWN	24212.1	YES

Điền thay thế các giá trị khuyết thiếu thuộc tính income bằng trung bình cộng

In [55]:
```
income_avg = df1_1['income'].mean()
df1_1['income'] = df1_1['income'].fillna(income_avg)
```

```
df1_1.head(10)
```

Out[55]:

	ID	age	sex	region	income	married
0	1	48	FEMALE	INNER_CITY	17546.000000	NO
1	2	40	MALE	TOWN	30085.100000	YES
2	3	51	FEMALE	INNER_CITY	16575.400000	YES
3	4	23	FEMALE	TOWN	20375.400000	YES
4	5	57	FEMALE	RURAL	50576.300000	YES
5	6	57	FEMALE	TOWN	37869.600000	YES
6	7	22	LGBT	RURAL	27350.725336	NO
7	8	58	MALE	TOWN	24946.600000	YES
8	9	37	FEMALE	SUBURBAN	25304.300000	YES
9	10	200	MALE	TOWN	24212.100000	YES

Ghép các tệp dữ liệu - Mở rộng thuộc tính (Merge)

In [62]:

```
print("Thông tin tệp dữ liệu 1-1 \n")
print(df1_1, "\n\n")

print("Thông tin tệp dữ liệu 1-2 \n")
print(df1_2, "\n\n")

print("Sau khi hợp 2 tệp dữ liệu với nhau \n")
df1 = pd.merge(df1_1, df1_2, on = 'ID')
df1
```

Thông tin tệp dữ liệu 1-1

	ID	age	sex	region	income	married
0	1	48	FEMALE	INNER_CITY	17546.00	NO
1	2	40	MALE	TOWN	30085.10	YES
2	3	51	FEMALE	INNER_CITY	16575.40	YES
3	4	23	FEMALE	TOWN	20375.40	YES
4	5	57	FEMALE	RURAL	50576.30	YES
..	...	...	...	...	...	...
295	296	34	MALE	TOWN	32548.90	YES
296	297	54	FEMALE	RURAL	24583.40	NO
297	298	18	MALE	RURAL	8639.24	YES
298	299	47	FEMALE	INNER_CITY	17139.50	NO
299	300	24	FEMALE	INNER_CITY	13667.70	YES

[300 rows x 6 columns]

Thông tin tệp dữ liệu 1-2

	ID	children	car	save_act	current_act	mortgage	pep
0	1	1	NO	NO	NO	NO	YES
1	2	3	YES	NO	YES	YES	NO
2	3	0	YES	YES	YES	NO	NO
3	4	3	NO	NO	YES	NO	NO
4	5	0	NO	YES	NO	NO	NO
..	...	...	...	...	...	...	...
295	296	0	YES	YES	YES	YES	NO
296	297	2	YES	YES	YES	YES	NO
297	298	2	NO	NO	NO	NO	NO
298	299	2	YES	NO	YES	NO	NO
299	300	0	NO	YES	YES	NO	NO

[300 rows x 7 columns]

Sau khi hợp 2 tệp dữ liệu với nhau

Out[62]:

	ID	age	sex	region	income	married	children	car	save_act	current_ac
0	1	48	FEMALE	INNER_CITY	17546.00	NO	1	NO	NO	NO
1	2	40	MALE	TOWN	30085.10	YES	3	YES	NO	YE
2	3	51	FEMALE	INNER_CITY	16575.40	YES	0	YES	YES	YE
3	4	23	FEMALE	TOWN	20375.40	YES	3	NO	NO	YE
4	5	57	FEMALE	RURAL	50576.30	YES	0	NO	YES	NO
...	...	...	...	...	...	...	...	...	...	.
295	296	34	MALE	TOWN	32548.90	YES	0	YES	YES	YE
296	297	54	FEMALE	RURAL	24583.40	NO	2	YES	YES	YE
297	298	18	MALE	RURAL	8639.24	YES	2	NO	NO	NO
298	299	47	FEMALE	INNER_CITY	17139.50	NO	2	YES	NO	YE
299	300	24	FEMALE	INNER_CITY	13667.70	YES	0	NO	YES	YE

300 rows × 12 columns

Ghép các tệp dữ liệu - Mở rộng mẫu (Concat)

In [63]:
```
df = pd.concat([df1, df2])
df
```

Out[63]:

	ID	age	sex	region	income	married	children	car	save_act	current_ac
0	1	48	FEMALE	INNER_CITY	17546.00	NO	1	NO	NO	NO
1	2	40	MALE	TOWN	30085.10	YES	3	YES	NO	YE
2	3	51	FEMALE	INNER_CITY	16575.40	YES	0	YES	YES	YE
3	4	23	FEMALE	TOWN	20375.40	YES	3	NO	NO	YE
4	5	57	FEMALE	RURAL	50576.30	YES	0	NO	YES	NO
...	...	...	...	...	...	...	...	...	...	.
295	296	61	NU	INNER_CITY	47025.00	NO	2	YES	YES	YE
296	297	30	NU	INNER_CITY	9672.25	YES	0	YES	YES	YE
297	298	31	NU	TOWN	15976.30	YES	0	YES	YES	NO
298	299	29	NAM	INNER_CITY	14711.80	YES	0	NO	YES	NO
299	300	38	NAM	TOWN	26671.60	NO	0	YES	NO	YE

600 rows × 12 columns

BTVN: Chuẩn hóa dữ liệu thuộc tính sex (Normalize)

In [66]:
```
print(df1_1.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 300 entries, 0 to 299
Data columns (total 6 columns):
 #   Column   Non-Null Count  Dtype  
---  ------   --------------  -----  
 0   ID       300 non-null    int64  
 1   age      300 non-null    int64  
 2   sex      300 non-null    object 
 3   region   300 non-null    object 
 4   income   300 non-null    float64
 5   married  300 non-null    object 
dtypes: float64(1), int64(2), object(3)
memory usage: 14.2+ KB
None
```