Chapter 4 - Exercise 2: Hãy thực hiện những yêu cầu liên quan tới Data Frame

In [1]:
```
import numpy as np
import pandas as pd
```

In [2]:
```
# Câu 1: Cho dictionary như sau:
dic_1 = {'X':[78,85,96,80,86], 'Y':[84,94,89,83,86], 'Z':[86,97,96,72,83]}

# Tạo dataframe df1 từ dic_1
df1 = pd.DataFrame(dic_1)

# In nội dung của dataframe df1
df1
```

Out[2]:

	X	Y	Z
0	78	84	86
1	85	94	97
2	96	89	96
3	80	83	72
4	86	86	83

In [3]:
```
# Câu 2:
# Cho dictionary exam_data như sau:
exam_data  = {'name': ['Anastasia', 'Dima', 'Katherine', 'James', 'Emily',
                       'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'],
        'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5,np.nan, 8, 19],
        'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],
        'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no',
                    'no', 'yes']}

# Cho list labels như sau:
labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']
```

In [4]:
```
# Câu 2a: Tạo dataframe df2 từ exam_data, với index của dataframe là labels
df2 = pd.DataFrame(exam_data, index = labels)

# In nội dung của dataframe df2
df2
```

Out[4]:

	name	score	attempts	qualify
a	Anastasia	12.5	1	yes
b	Dima	9.0	3	no
c	Katherine	16.5	2	yes
d	James	NaN	3	no
e	Emily	9.0	2	no
f	Michael	20.0	3	yes
g	Matthew	14.5	1	yes
h	Laura	NaN	1	no
i	Kevin	8.0	2	no
j	Jonas	19.0	1	yes

In [5]:
```
# Câu 2b: Xem thông tin (info()) của dataframe df2
df2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 10 entries, a to j
Data columns (total 4 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   name      10 non-null     object 
 1   score     8 non-null      float64
 2   attempts  10 non-null     int64  
 3   qualify   10 non-null     object 
dtypes: float64(1), int64(1), object(2)
memory usage: 400.0+ bytes
```

In [6]:
```
# Câu 3: Tạo dataframe df3 từ df2, chỉ chứa 2 cột là name và score
df3 = df2[['name', 'score']]
# Xem kiểu dữ liệu (type) và kích thước (shape) của df3
print(type(df3))
print(df3.shape)
# Hiển thị các dòng dữ liệu đầu tiên (head) của df3
df3.head()
```

```
<class 'pandas.core.frame.DataFrame'>
(10, 2)
```

Out[6]:

	name	score
a	Anastasia	12.5
b	Dima	9.0
c	Katherine	16.5
d	James	NaN
e	Emily	9.0

In [7]:
```
# Câu 4: Hiển thị thông tin thống kê chung (describe) của dataframe df2
df2.describe(include = 'all')
```

Out[7]:

	name	score	attempts	qualify
count	10	8.000000	10.000000	10
unique	10	NaN	NaN	2
top	Laura	NaN	NaN	yes
freq	1	NaN	NaN	5
mean	NaN	13.562500	1.900000	NaN
std	NaN	4.693746	0.875595	NaN
min	NaN	8.000000	1.000000	NaN
25%	NaN	9.000000	1.000000	NaN
50%	NaN	13.500000	2.000000	NaN
75%	NaN	17.125000	2.750000	NaN
max	NaN	20.000000	3.000000	NaN

In [8]:
```
# Câu 5: Tạo dataframe df4 từ df2, chỉ chứa 2 cột là name và score,
# và chỉ có các dòng 1, 3, 5, 6
df4 = df2.iloc[[1, 3, 5, 6], [0, 1]]
# In nội dung của dataframe df4
df4
```

Out[8]:

	name	score
b	Dima	9.0
d	James	NaN
f	Michael	20.0
g	Matthew	14.5

In [9]:
```
# Câu 6: Từ dataframe df2, in các dòng có dữ liệu score bị null
df2[df2['score'].isnull()]
```

Out[9]:

	name	score	attempts	qualify
d	James	NaN	3	no
h	Laura	NaN	1	no

In [10]:
```
df2.isnull().sum()
```

Out[10]:
```
name        0
score       2
attempts    0
qualify     0
dtype: int64
```

In [11]:
```
# Câu 7: Từ dataframe df2, in các dòng có score > 15 và <=20
df2[(df2['score'] > 15) & (df2['score'] <= 20)]
```

Out[11]:

	name	score	attempts	qualify
c	Katherine	16.5	2	yes
f	Michael	20.0	3	yes
j	Jonas	19.0	1	yes

In [12]:
```
# Câu 8: Cập nhật điểm (score) ở dòng 'd' thành 18
df2.loc['d', 'score'] = 18
df2.head()
```

Out[12]:

	name	score	attempts	qualify
a	Anastasia	12.5	1	yes
b	Dima	9.0	3	no
c	Katherine	16.5	2	yes
d	James	18.0	3	no
e	Emily	9.0	2	no

In [13]:
```
# Câu 9: Cho biết điểm (score) nào có tần suất xuất hiện nhiều nhất trong df2,
# và in ra những dòng có điểm là tần suất xuất hiện nhiều nhất
mark = df2['score'].mode()
print(mark[0])

score = df2[(df2['score']==mark[0])]
score
```

```
9.0
```

Out[13]:

	name	score	attempts	qualify
b	Dima	9.0	3	no
e	Emily	9.0	2	no

In [14]:
```
df2['score'].value_counts() # cách khác
```

Out[14]:
```
9.0     2
20.0    1
16.5    1
14.5    1
18.0    1
8.0     1
19.0    1
12.5    1
Name: score, dtype: int64
```

In [15]:
```
# Câu 10: Thêm dòng k có nội dung như sau: ['Suresh', 15.5, 1, 'yes'] vào df2
df2.loc['k'] = ['Suresh', 15.5, 1, 'yes']
# Hiển thị 5 dòng cuối cùng (tail) của df2.
df2.tail()
```

Out[15]:

	name	score	attempts	qualify
g	Matthew	14.5	1	yes
h	Laura	NaN	1	no
i	Kevin	8.0	2	no
j	Jonas	19.0	1	yes
k	Suresh	15.5	1	yes

In [16]:
```
# Câu 11a: Thêm dòng l có nội dung như sau: ['Janny', 12.5, 2, 'yes'] vào df2.
df2.loc['l'] = ['Janny', 12.5, 2, 'yes']
# Hiển thị 5 dòng cuối cùng (tail) của df2.
df2.tail()
```

Out[16]:

	name	score	attempts	qualify
h	Laura	NaN	1	no
i	Kevin	8.0	2	no
j	Jonas	19.0	1	yes
k	Suresh	15.5	1	yes
l	Janny	12.5	2	yes

In [17]:
```
# Câu 11b: Xóa bỏ dòng l của df2.
df2 = df2.drop(['l'])
df3.tail()
# Hiển thị lại 5 dòng cuối cùng (tail) của df2.
df2.tail()
```

Out[17]:

	name	score	attempts	qualify
g	Matthew	14.5	1	yes
h	Laura	NaN	1	no
i	Kevin	8.0	2	no
j	Jonas	19.0	1	yes
k	Suresh	15.5	1	yes

In [18]:
```
# Câu 12: Sắp xếp df2 tăng dần theo điểm (score)
df2 = df2.sort_values(by='score')
df2
```

Out[18]:

	name	score	attempts	qualify
i	Kevin	8.0	2	no
b	Dima	9.0	3	no
e	Emily	9.0	2	no
a	Anastasia	12.5	1	yes
g	Matthew	14.5	1	yes
k	Suresh	15.5	1	yes
c	Katherine	16.5	2	yes
d	James	18.0	3	no
j	Jonas	19.0	1	yes
f	Michael	20.0	3	yes
h	Laura	NaN	1	no

In [19]:
```
# Câu 13: Thêm cột result vào df2,
# dựa vào dữ liệu của cột'score',
# nếu dòng nào có điểm >=10 thì giá trị của cột result = 1, ngược lại = 0
df2['result'] = df2['score'].map(lambda x: 1 if x >= 10 else 0)
# In nội dung của dataframe df2
df2
```

Out[19]:

	name	score	attempts	qualify	result
i	Kevin	8.0	2	no	0
b	Dima	9.0	3	no	0
e	Emily	9.0	2	no	0
a	Anastasia	12.5	1	yes	1
g	Matthew	14.5	1	yes	1
k	Suresh	15.5	1	yes	1
c	Katherine	16.5	2	yes	1
d	James	18.0	3	no	1
j	Jonas	19.0	1	yes	1
f	Michael	20.0	3	yes	1
h	Laura	NaN	1	no	0

In [20]:
```
df2['result_'] = np.where(df2['score']>=10, 1, 0)
df2
```

Out[20]:

	name	score	attempts	qualify	result	result_
i	Kevin	8.0	2	no	0	0
b	Dima	9.0	3	no	0	0
e	Emily	9.0	2	no	0	0
a	Anastasia	12.5	1	yes	1	1
g	Matthew	14.5	1	yes	1	1
k	Suresh	15.5	1	yes	1	1
c	Katherine	16.5	2	yes	1	1
d	James	18.0	3	no	1	1
j	Jonas	19.0	1	yes	1	1
f	Michael	20.0	3	yes	1	1
h	Laura	NaN	1	no	0	0

In [21]:
```
# Câu 14: Trong df2, thay tên 'Emily' thành 'Samantha'
df2['name'] = df2['name'].replace('Emily', 'Samantha')
# In lại nội dung của dataframe df2
df2
```

Out[21]:

	name	score	attempts	qualify	result	result_
i	Kevin	8.0	2	no	0	0
b	Dima	9.0	3	no	0	0
e	Samantha	9.0	2	no	0	0
a	Anastasia	12.5	1	yes	1	1
g	Matthew	14.5	1	yes	1	1
k	Suresh	15.5	1	yes	1	1
c	Katherine	16.5	2	yes	1	1
d	James	18.0	3	no	1	1
j	Jonas	19.0	1	yes	1	1
f	Michael	20.0	3	yes	1	1
h	Laura	NaN	1	no	0	0

In [22]:
```
# Câu 15: Duyệt df2, in name, score, result:
# nếu giá trị cột result = 1 thì in 'Pass', ngược lại thì in 'Fail'
for index, row in df2.iterrows():
    print(row['name'],' - Score:', row['score'],
          ' - Result:', "Pass" if row['result'] == 1 else "Fail")
```

```
Kevin  - Score: 8.0  - Result: Fail
Dima  - Score: 9.0  - Result: Fail
Samantha  - Score: 9.0  - Result: Fail
Anastasia  - Score: 12.5  - Result: Pass
Matthew  - Score: 14.5  - Result: Pass
Suresh  - Score: 15.5  - Result: Pass
Katherine  - Score: 16.5  - Result: Pass
James  - Score: 18.0  - Result: Pass
Jonas  - Score: 19.0  - Result: Pass
Michael  - Score: 20.0  - Result: Pass
Laura  - Score: nan  - Result: Fail
```

In [ ]: