



Chapter 4 - Exercise 2: Hãy thực hiện những yêu cầu liên quan tới Data Frame, và đối chiếu với kết quả cho trước

```
[ ] import numpy as np
import pandas as pd
```

```
[ ] # 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, columns=['X', 'Y', 'Z'])
# In nội dung của dataframe df1
df1
```

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

```
[ ] # 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']
```

```
[ ] # Câu 2a: Tạo dataframe df2 từ exam_data, với index của dataframe là labels
df2 = pd.DataFrame(exam_data, index=labels, columns=['name', 'score', 'attempts', 'qualify'])
# In nội dung của dataframe df2
df2
```

	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

```
[ ] # 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
```

► Nhấn vào đây để xem kết quả!

```
[ ] # Câu 3: Tạo dataframe df3 từ df2, chỉ chứa 2 cột là name và score
df3 = pd.DataFrame(df2, columns=['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)
```

	name	score
--	------	-------

a	Anastasia	12.5
---	-----------	------

b	Dima	9.0
---	------	-----

c	Katherine	16.5
---	-----------	------

d	James	NaN
---	-------	-----

e	Emily	9.0
---	-------	-----

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

	name	score	attempts	qualify
count	10	8.000000	10.000000	10
unique	10	NaN	NaN	2
top	James	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

```
[ ] # Câu 5: Tạo dataframe df4 từ df2, chỉ chứa 2 cột là name và score, và chỉ có các dòng 1, 3, 5, 6
LstDong = df2.iloc[[1,3,5,6],:0]
print(LstDong)
print(type(LstDong))
df4 = pd.DataFrame(df2,index=['b', 'd', 'f', 'g'],columns=['name','score'])
df4
```

Empty DataFrame

Columns: []

Index: [b, d, f, g]

<class 'pandas.core.frame.DataFrame'>

name score

b Dima 9.0

d James NaN

f Michael 20.0

g Matthew 14.5

```
[ ] # Câu 6: Từ dataframe df2, in các dòng có dữ liệu score bị null
print('---ban dau---')
print(df2)
print('---sau---')
df2[df2['score'].isnull()]
```

---ban dau---

	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

---sau---

	name	score	attempts	qualify
--	------	-------	----------	---------

d	James	NaN	3	no
----------	-------	-----	---	----

h	Laura	NaN	1	no
----------	-------	-----	---	----


```
[ ] # Câu 7: Từ dataframe df2, in các dòng có score > 15 và <=20
print('---ban dau---')
print(df2)
print('---sau---')
df2.loc[(df2['score']>15) & (df2['score']<=20)]
```

---ban dau---

	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

---sau---

	name	score	attempts	qualify
--	------	-------	----------	---------

c	Katherine	16.5	2	yes
f	Michael	20.0	3	yes
j	Jonas	19.0	1	yes

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

---ban dau---

	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

► Nhấn vào đây để xem kết quả!

```
# 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
d = df2['score'].mode()
print(d.iloc[0])
df2.loc[df2['score']==d.iloc[0]]
# Cho biết có bao nhiêu dòng thỏa điều kiện?
```

9.0

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

Câu 10: Thêm dòng k có nội dung như sau: ['Suresh', 15.5, 1, 'yes'] vào df2
print(df2)
df_2 = pd.DataFrame([nd], columns=['name', 'score', 'attempts', 'qualify'])
df2.loc['k'] = ['Suresh', 15.5, 1, 'yes']
Hiển thị 5 dòng cuối cùng (tail) của df2.
df2.tail()

↗

	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
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

	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

```
[ ] # Câu 11a: Thêm dòng 1 có nội dung như sau: ['Janny', 12.5, 2, 'yes'] vào df2.  
df2.loc['1'] = ['Janny', 12.5, 2, 'yes']  
# Hiển thị 5 dòng cuối cùng (tail) của df2.  
df2.tail()
```

	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

```
[ ] # Câu 11b: Xóa bỏ dòng 1 của df2.  
df2 = df2.drop(['1'])  
# Hiện thị lại 5 dòng cuối cùng (tail) của df2.  
df2.tail()
```

	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

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

	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

```
[ ] # 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
```

	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

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

	name	score	attempts	qualify	result
i	Kevin	8.0	2	no	0
b	Dima	9.0	3	no	0
e	Samantha	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


```
# Câu 15: Duyệt df2, in name, score, result: nếu nếu giá trị cột result = 1 thì in 'Pass', ngược lại thì in 'Fail'
df2['result'] = df2['result'].map(lambda x : 'Pass' if x == 1 else 'Fail')
print(df2)
```

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