Chapter 4 - Ex4: Combining Data

Câu 1: Cho dữ liệu employees1.csv và employees2.csv

- Đọc dữ liệu từ 2 tập tin trên
- Kết hợp 2 dữ liệu trên thành 1 DataFrame

Câu 2: Cho dữ liệu department.csv

- Đọc dữ liệu từ tập tin trên
- Kết hợp dữ liệu này với dữ liệu kết quả từ câu 1

Câu 3: Cho dữ liệu skills.csv

- Đọc dữ liệu từ tập tin trên
- Kết hợp dữ liệu này với dữ liệu kết quả từ câu 2

Câu 4: Cho dữ liệu salary.csv

- Đọc dữ liệu từ tập tin trên
- Kết hợp dữ liệu này với dữ liệu ở câu 1 (gợi ý: dùng right_on & left_on khi merge vì trong salay có cột name, còn df ở câu 1 lại có cột employee trùng nội dung, bỏ cột name)

Câu 1+: Cho dữ liệu employees1.csv và employees2.csv

- Đọc dữ liệu từ 2 tập tin trên vào 2 DataFrame với index của các DataFrame là 'employee'
- Kết hợp 2 DataFrame trên thành 1 DataFrame dùng chung 1 index là 'employee' (gợi ý: dùng left_index hoặc/và right_index hoặc dùng dataframe1.join(dataframe2))

Câu 5: Cho dữ liệu như sau:

1. df6 = pd.DataFrame({'name': ['Peter', 'Paul', 'Mary'],

'food': ['fish', 'beans', 'bread']},
columns=['name', 'food'])

2. df7 = pd.DataFrame({'name': ['Mary', 'Joseph'],

'drink': ['wine', 'beer']},
columns=['name', 'drink'])

- Kết hợp 2 bộ dữ liệu này với tham số how='inner', how='outer', how='left', how='right'. Quan sát kết quả trong từng trường hợp.

Câu 6: Cho dữ liệu như sau:

1. df8 = pd.DataFrame({'name': ['Bob', 'Jake', 'Lisa', 'Sue'],

'rank': [1, 2, 3, 4]})

2. df9 = pd.DataFrame({'name': ['Bob', 'Jake', 'Lisa', 'Sue'],

'rank': [3, 1, 4, 2]})

- Kết hợp 2 bộ dữ liệu này với tham số on='tên_cột_trùng_lắp' và/hoặc suffixes=["_L", "_R"]

Câu 1: Gợi ý

In [1]:

```
import pandas as pd
```

In [2]:

```
df1 = pd.DataFrame({'employee': ['Bob', 'Jake',
                                 'Lisa', 'Sue', 'John', 'Billy'],
                    'group': ['Accounting', 'Engineering',
                              'Engineering', 'HR', 'IT', 'HR']})
df2 = pd.DataFrame({'employee': ['Lisa', 'Bob',
                                 'Jake', 'Sue', 'John', 'Billy'],
                    'hire_date': [2004, 2008, 2012, 2014, 2010, 2015]})
```

In [3]:

```
df1.to_csv("employees1.csv")
df2.to_csv("employees2.csv")
```

In [4]:

```
e1 = pd.read_csv("employees1.csv", index_col=0)
e2 = pd.read_csv("employees2.csv", index_col=0)
```

In [5]:

```
e1
```

Out[5]:

	employee	group
0	Bob	Accounting
1	Jake	Engineering
2	Lisa	Engineering
3	Sue	HR
4	John	IT
5	Billy	HR

In [6]:

```
e2
```

Out[6]:

	employee	hire_date
0	Lisa	2004
1	Bob	2008
2	Jake	2012
3	Sue	2014
4	John	2010
5	Billy	2015

In [7]:

```
df = pd.merge(df1, df2)
df
```

Out[7]:

	employee	group	hire_date
0	Bob	Accounting	2008
1	Jake	Engineering	2012
2	Lisa	Engineering	2004
3	Sue	HR	2014
4	John	IT	2010
5	Billy	HR	2015

Câu 2: Gợi ý

In [8]:

```
df4 = pd.DataFrame({'group': ['Accounting', 'Engineering',
                              'HR', 'IT'],
                    'supervisor': ['Carly', 'Guido',
                                   'Steve', 'Jame']})
```

In [9]:

```
df4.to_csv("department.csv")
```

In [10]:

```
d = pd.read_csv("department.csv", index_col=0)
```

In [11]:

```
d
```

Out[11]:

	group	supervisor
0	Accounting	Carly
1	Engineering	Guido
2	HR	Steve
3	IT	Jame

In [12]:

```
df_with_dept = pd.merge(df, d)
```

In [13]:

```
df_with_dept
```

Out[13]:

	employee	group	hire_date	supervisor
0	Bob	Accounting	2008	Carly
1	Jake	Engineering	2012	Guido
2	Lisa	Engineering	2004	Guido
3	Sue	HR	2014	Steve
4	Billy	HR	2015	Steve
5	John	IT	2010	Jame

Câu 3: Gợi ý

In [14]:

```
df5 = pd.DataFrame({'group': ['Accounting', 'Accounting',
                              'Engineering', 'Engineering',
                              'HR', 'HR', 'IT', 'IT'],
                    'skills': ['math', 'spreadsheets',
                               'coding', 'linux',
                               'spreadsheets', 'organization',
                               'coding', 'math']})
```

In [15]:

```
df5.to_csv("skills.csv")
```

In [16]:

```
skills = pd.read_csv("skills.csv", index_col=0)
```

In [17]:

```
skills
```

Out[17]:

	group	skills
0	Accounting	math
1	Accounting	spreadsheets
2	Engineering	coding
3	Engineering	linux
4	HR	spreadsheets
5	HR	organization
6	IT	coding
7	IT	math

In [18]:

```
df_with_dept_skills = pd.merge(df_with_dept, skills)
```

In [19]:

```
df_with_dept_skills
```

Out[19]:

	employee	group	hire_date	supervisor	skills
0	Bob	Accounting	2008	Carly	math
1	Bob	Accounting	2008	Carly	spreadsheets
2	Jake	Engineering	2012	Guido	coding
3	Jake	Engineering	2012	Guido	linux
4	Lisa	Engineering	2004	Guido	coding
5	Lisa	Engineering	2004	Guido	linux
6	Sue	HR	2014	Steve	spreadsheets
7	Sue	HR	2014	Steve	organization
8	Billy	HR	2015	Steve	spreadsheets
9	Billy	HR	2015	Steve	organization
10	John	IT	2010	Jame	coding
11	John	IT	2010	Jame	math

Câu 4: Gợi ý

In [20]:

```
df3 = pd.DataFrame({'name': ['Bob', 'Jake', 'Lisa', 'Sue', 'John', 'Billy'],
                    'salary': [70000, 80000, 120000, 90000, 125000, 92000]})
```

In [21]:

```
df3.to_csv("salary.csv")
```

In [22]:

```
salary = pd.read_csv("salary.csv", index_col=0)
salary
```

Out[22]:

	name	salary
0	Bob	70000
1	Jake	80000
2	Lisa	120000
3	Sue	90000
4	John	125000
5	Billy	92000

In [23]:

```
df_em_salary = pd.merge(df, salary,
                        left_on="employee",
                        right_on="name").drop('name', axis=1)
```

In [24]:

```
df_em_salary
```

Out[24]:

	employee	group	hire_date	salary
0	Bob	Accounting	2008	70000
1	Jake	Engineering	2012	80000
2	Lisa	Engineering	2004	120000
3	Sue	HR	2014	90000
4	John	IT	2010	125000
5	Billy	HR	2015	92000

Câu 1+: Gợi ý

In [25]:

```
e1a = pd.read_csv("employees1.csv", index_col=0)
e2a = pd.read_csv("employees2.csv", index_col=0)
e1a = e1a.set_index('employee')
e2a = e2a.set_index('employee')
```

In [26]:

```
display(e1a, e2a)
```

employee	group
Bob	Accounting
Jake	Engineering
Lisa	Engineering
Sue	HR
John	IT
Billy	HR

employee	hire_date
Lisa	2004
Bob	2008
Jake	2012
Sue	2014
John	2010
Billy	2015

In [27]:

```
df_merge = pd.merge(e1a, e2a, left_index=True, right_index=True)
df_merge
```

Out[27]:

employee	group	hire_date
Bob	Accounting	2008
Jake	Engineering	2012
Lisa	Engineering	2004
Sue	HR	2014
John	IT	2010
Billy	HR	2015

In [28]:

```
df_join = e1a.join(e2a)
df_join
```

Out[28]:

employee	group	hire_date
Bob	Accounting	2008
Jake	Engineering	2012
Lisa	Engineering	2004
Sue	HR	2014
John	IT	2010
Billy	HR	2015

Câu 5: Gợi ý

In [29]:

```
df6 = pd.DataFrame({'name': ['Peter', 'Paul', 'Mary'],
                    'food': ['fish', 'beans', 'bread']},
                   columns=['name', 'food'])
df7 = pd.DataFrame({'name': ['Mary', 'Joseph'],
                    'drink': ['wine', 'beer']},
                   columns=['name', 'drink'])
```

In [30]:

```
display(df6, df7)
```

	name	food
0	Peter	fish
1	Paul	beans
2	Mary	bread

	name	drink
0	Mary	wine
1	Joseph	beer

In [31]:

```
df67_merge = pd.merge(df6, df7)
df67_merge
```

Out[31]:

	name	food	drink
0	Mary	bread	wine

In [32]:

```
df67_inner = pd.merge(df6, df7, how='inner')
df67_inner
```

Out[32]:

	name	food	drink
0	Mary	bread	wine

In [33]:

```
df67_outer = pd.merge(df6, df7, how='outer')
df67_outer
```

Out[33]:

	name	food	drink
0	Peter	fish	NaN
1	Paul	beans	NaN
2	Mary	bread	wine
3	Joseph	NaN	beer

In [34]:

```
df67_left = pd.merge(df6, df7, how='left')
df67_left
```

Out[34]:

	name	food	drink
0	Peter	fish	NaN
1	Paul	beans	NaN
2	Mary	bread	wine

In [35]:

```
df67_right = pd.merge(df6, df7, how='right')
df67_right
```

Out[35]:

	name	food	drink
0	Mary	bread	wine
1	Joseph	NaN	beer

Câu 6: Gợi ý

In [36]:

```
df8 = pd.DataFrame({'name': ['Bob', 'Jake', 'Lisa', 'Sue'],
                    'rank': [1, 2, 3, 4]})
df9 = pd.DataFrame({'name': ['Bob', 'Jake', 'Lisa', 'Sue'],
                    'rank': [3, 1, 4, 2]})
```

In [37]:

```
display(df8, df9)
```

	name	rank
0	Bob	1
1	Jake	2
2	Lisa	3
3	Sue	4

	name	rank
0	Bob	3
1	Jake	1
2	Lisa	4
3	Sue	2

In [38]:

```
df89_on = pd.merge(df8, df9, on='name')
df89_on
```

Out[38]:

	name	rank_x	rank_y
0	Bob	1	3
1	Jake	2	1
2	Lisa	3	4
3	Sue	4	2

In [39]:

```
df89_on_suff = pd.merge(df8, df9, on='name', suffixes=['_L', '_R'])
df89_on_suff
```

Out[39]:

	name	rank_L	rank_R
0	Bob	1	3
1	Jake	2	1
2	Lisa	3	4
3	Sue	4	2

In [ ]:

```

```