# 21. Ordered merges

## (1) pd.merge_ordered()

Trước tiên sẽ tải hai files ("feb-sales-Hardware.csv" và "feb-sales-Software.csv") và tạo ra hai dataframe tương ứng là (soft và hard).

```
>>> soft=pd.read_csv("feb-sales-Software.csv",parse_dates=['Date']).sort_values('Date')
>>> hard=pd.read_csv("feb-sales-Hardware.csv",parse_dates=['Date']).sort_values('Date')
>>> soft
                 Date          Company   Product  Units
2 2015-02-02 08:33:01           Hooli  Software      3
1 2015-02-03 14:14:18         Initech  Software     13
7 2015-02-04 15:36:29       Streeplex  Software     13
3 2015-02-05 01:53:06  Acme Coporation  Software     19
5 2015-02-09 13:09:55       Mediacore  Software      7
4 2015-02-11 20:03:08         Initech  Software      7
6 2015-02-11 22:50:44           Hooli  Software      4
0 2015-02-16 12:09:19           Hooli  Software     10
8 2015-02-21 05:01:26       Mediacore  Software      3
>>> hard
                 Date          Company   Product  Units
3 2015-02-02 20:54:49       Mediacore  Hardware      9
0 2015-02-04 21:52:45  Acme Coporation  Hardware     14
1 2015-02-07 22:58:10  Acme Coporation  Hardware      1
2 2015-02-19 10:59:33       Mediacore  Hardware     16
4 2015-02-21 20:41:47           Hooli  Hardware      3
>>>
```

Ta có nhận xét: cả soft and hard có tên các cột giống nhau và đều được sắp xếp theo thứ tự thời gian (cột 'Date') nhưng indexes của cả hai bảng rất lộn xộn.

```
>>> pd.merge(soft,hard,how='outer').sort_values('Date')
                 Date          Company   Product  Units
0  2015-02-02 08:33:01           Hooli  Software      3
9  2015-02-02 20:54:49       Mediacore  Hardware      9
1  2015-02-03 14:14:18         Initech  Software     13
2  2015-02-04 15:36:29       Streeplex  Software     13
10 2015-02-04 21:52:45  Acme Coporation  Hardware     14
3  2015-02-05 01:53:06  Acme Coporation  Software     19
11 2015-02-07 22:58:10  Acme Coporation  Hardware      1
4  2015-02-09 13:09:55       Mediacore  Software      7
5  2015-02-11 20:03:08         Initech  Software      7
6  2015-02-11 22:50:44           Hooli  Software      4
7  2015-02-16 12:09:19           Hooli  Software     10
12 2015-02-19 10:59:33       Mediacore  Hardware     16
8  2015-02-21 05:01:26       Mediacore  Software      3
13 2015-02-21 20:41:47           Hooli  Hardware      3
>>>
```

Ngoài ra pandas còn cung cấp một hàm khác tiện lợi hơn đó là pd.merge_ordered() hoạt động tương tự pd.merge() kết hợp với .sort_values(). Nếu pd.merge

```
>>> pd.merge_ordered(soft,hard)
                 Date          Company   Product  Units
0  2015-02-02 08:33:01           Hooli  Software      3
1  2015-02-02 20:54:49       Mediacore  Hardware      9
2  2015-02-03 14:14:18         Initech  Software     13
3  2015-02-04 15:36:29       Streeplex  Software     13
4  2015-02-04 21:52:45  Acme Coporation  Hardware     14
5  2015-02-05 01:53:06  Acme Coporation  Software     19
6  2015-02-07 22:58:10  Acme Coporation  Hardware      1
7  2015-02-09 13:09:55       Mediacore  Software      7
8  2015-02-11 20:03:08         Initech  Software      7
9  2015-02-11 22:50:44           Hooli  Software      4
10 2015-02-16 12:09:19           Hooli  Software     10
11 2015-02-19 10:59:33       Mediacore  Hardware     16
12 2015-02-21 05:01:26       Mediacore  Software      3
13 2015-02-21 20:41:47           Hooli  Hardware      3
>>>
```

So với pd.merge() thì pd.merge_ordered() vẫn nhận các đối số tương tự tuy nhiên có thêm một số lựa chọn khác như fill_method, nếu fill_method = 'ffill' thì tr

```
>>> pd.merge_ordered(soft,hard,on=['Company','Product','Date'],suffixes = ['_soft','_hard'],fill_method='ffill')
                 Date          Company   Product  Units_soft  Units_hard
0  2015-02-04 21:52:45  Acme Coporation  Hardware         NaN          14
1  2015-02-07 22:58:10  Acme Coporation  Hardware         NaN           1
2  2015-02-05 01:53:06  Acme Coporation  Software        19.0           1
3  2015-02-21 20:41:47           Hooli  Hardware        19.0           3
4  2015-02-02 08:33:01           Hooli  Software         3.0           3
5  2015-02-11 22:50:44           Hooli  Software         4.0           3
6  2015-02-16 12:09:19           Hooli  Software        10.0           3
7  2015-02-03 14:14:18         Initech  Software        13.0           3
8  2015-02-11 20:03:08         Initech  Software         7.0           3
9  2015-02-02 20:54:49       Mediacore  Hardware         7.0           9
10 2015-02-19 10:59:33       Mediacore  Hardware         7.0          16
11 2015-02-09 13:09:55       Mediacore  Software         7.0          16
12 2015-02-21 05:01:26       Mediacore  Software         3.0          16
13 2015-02-04 15:36:29       Streeplex  Software        13.0          16
>>>
```

Ở ví dụ trên, thứ tự sắp xếp ưu tiên xét cho cột 'Company' trước, sau đó mới đến 'Product' và cuối cùng là 'Date' theo dữ liệu được truyền vào cho đối số 'on

## (2) pd.merge_asof()

Tương tự như pd.merge_ordered(), chức năng pd.merge_asof() cũng sẽ merging các giá trị theo thứ tự bằng cách sử dụng từ khóa 'on'. Nó hoạt động tương

Chú ý: cột trong 'on' cần phải được sắp xếp trước khi sử dụng pd.merge_**asof**().

Chức năng này có thể được sử dụng để căn chỉnh sự xuất hiện datetime khác nhau mà không cần phải tiến hành resample(câu hỏi tại sao lại vậy? bài tập của các bạn!). Để làm rõ ta đi qua ví dụ sau:

Hãy tải hai files dữ liệu là (trades.csv và quotes.csv) và tạo ra hai dataframe tương ứng là (trades và quotes).

```
>>> quotes=pd.read_csv("quotes.csv",parse_dates=['time'])
>>> trades = pd.read_csv("trades.csv",parse_dates=['time'])
>>> quotes
                 time ticker     bid     ask
0 2016-05-25 13:30:00.023   GOOG  720.50  720.93
1 2016-05-25 13:30:00.023   MSFT   51.95   51.96
2 2016-05-25 13:30:00.030   MSFT   51.97   51.98
3 2016-05-25 13:30:00.041   MSFT   51.99   52.00
4 2016-05-25 13:30:00.048   GOOG  720.50  720.93
5 2016-05-25 13:30:00.049   AAPL   97.99   98.01
6 2016-05-25 13:30:00.072   GOOG  720.50  720.88
7 2016-05-25 13:30:00.075   MSFT   52.01   52.03
>>> trades
                 time ticker   price  quantity
0 2016-05-25 13:30:00.023   MSFT   51.95        75
1 2016-05-25 13:30:00.038   MSFT   51.95       155
2 2016-05-25 13:30:00.048   GOOG  720.77       100
3 2016-05-25 13:30:00.048   GOOG  720.92       100
4 2016-05-25 13:30:00.048   AAPL   98.00        10
>>>
```

Sử dụng pd.merge_asof(trades,quotes,on='time',by='ticker') mặc định với direction='backward'

```
>>> pd.merge_asof(trades,quotes,on='time',by='ticker')

                 time ticker   price  quantity     bid     ask

0 2016-05-25 13:30:00.023   MSFT   51.95        75   51.95   51.96

1 2016-05-25 13:30:00.038   MSFT   51.95       155   51.97   51.98

2 2016-05-25 13:30:00.048   GOOG  720.77       100  720.50  720.93

3 2016-05-25 13:30:00.048   GOOG  720.92       100  720.50  720.93

4 2016-05-25 13:30:00.048   AAPL   98.00        10     NaN     NaN

>>>
```

Khi direction='forward'.

```
>>> pd.merge_asof(trades,quotes,on='time',by='ticker',direction='forward')

                 time ticker   price  quantity     bid     ask

0 2016-05-25 13:30:00.023   MSFT   51.95        75   51.95   51.96

1 2016-05-25 13:30:00.038   MSFT   51.95       155   51.99   52.00

2 2016-05-25 13:30:00.048   GOOG  720.77       100  720.50  720.93

3 2016-05-25 13:30:00.048   GOOG  720.92       100  720.50  720.93

4 2016-05-25 13:30:00.048   AAPL   98.00        10   97.99   98.01
```

Khi direction='nearest'

```
>>> pd.merge_asof(trades,quotes,on='time',by='ticker',direction='nearest')

                 time ticker   price  quantity     bid     ask

0 2016-05-25 13:30:00.023   MSFT   51.95        75   51.95   51.96

1 2016-05-25 13:30:00.038   MSFT   51.95       155   51.99   52.00

2 2016-05-25 13:30:00.048   GOOG  720.77       100  720.50  720.93

3 2016-05-25 13:30:00.048   GOOG  720.92       100  720.50  720.93

4 2016-05-25 13:30:00.048   AAPL   98.00        10   97.99   98.01
```

Ngoài ra pd.merge_ asof() còn cung cấp từ khóa tolerance để chỉ chấp nhận khoảng khác biệt khi matching giá trị trong một khoảng nhất định. Cùng xem ví dụ sau khi chỉ chấp nhận khoảng khác biệt là 1ms.

```
>>> pd.merge_asof(trades,quotes,on='time',by='ticker',direction='nearest', tolerance=pd.Timedelta('1ms'))

                 time ticker   price  quantity     bid     ask

0 2016-05-25 13:30:00.023   MSFT   51.95        75   51.95   51.96

1 2016-05-25 13:30:00.038   MSFT   51.95       155     NaN     NaN

2 2016-05-25 13:30:00.048   GOOG  720.77       100  720.50  720.93

3 2016-05-25 13:30:00.048   GOOG  720.92       100  720.50  720.93

4 2016-05-25 13:30:00.048   AAPL   98.00        10   97.99   98.01

>>>
```

## Kết luận

Qua bài học này, bạn đã tìm hiểu về merges khi xem xét đến thứ tự. Điều này rất hữu ích khi bạn muốn hợp nhất DataFrame với các cột có thứ tự tự nhiên như theo thời gian. **pd.merge_ordered()** và **pd.merge_asof() là hai phương thức quan trọng để giải quyết bài toán này.**