Pandas (Cont.)

DataFrame

การเลือก df แบบหลายเงื่อนไข

In [63]:
```
df = pd.DataFrame({'col1':[1,2,3,4],
                   'col2':[444,555,666,444],
                   'col3':['abc','def','ghi','xyz']})
df
```

Out[63]:

	col1	col2	col3
0	1	444	abc
1	2	555	def
2	3	666	ghi
3	4	444	xyz

In [64]:
```
(df['col2']>500) & ( df['col1']>=3 )
```

Out[64]:
```
0    False
1    False
2     True
3    False
dtype: bool
```

In [65]:
```
df[(df['col2']>500) & ( df['col1']>=3 ) ]
```

Out[65]:

	col1	col2	col3
2	3	666	ghi

DataFrame

In [22]:
```
df = pd.DataFrame(np.random.randint(1,100,[5,4]),index='a b c d e'.split()
                  ,columns = 'W X Y Z'.split())
df
```

Out[22]:

	W	X	Y	Z
a	49	18	74	12
b	37	9	81	94
c	44	30	64	47
d	75	30	10	31
e	57	53	23	99

DataFrame

การเพิ่มคอลัมน์ใหม่

```
In [23]: df['I'] = [1,2,3,4,5]
         df
```

Out[23]:

	W	X	Y	Z	I
a	49	18	74	12	1
b	37	9	81	94	2
c	44	30	64	47	3
d	75	30	10	31	4
e	57	53	23	99	5

```
In [24]: df['J'] = np.zeros(len(df.index))
         df
```

Out[24]:

	W	X	Y	Z	I	J
a	49	18	74	12	1	0.0
b	37	9	81	94	2	0.0
c	44	30	64	47	3	0.0
d	75	30	10	31	4	0.0
e	57	53	23	99	5	0.0

DataFrame

การเพิ่มคอลัมน์

```
In [25]: df['Sum'] = df['W'] + df['X']
         df
```

Out[25]:

	W	X	Y	Z	I	J	Sum
a	49	18	74	12	1	0.0	67
b	37	9	81	94	2	0.0	46
c	44	30	64	47	3	0.0	74
d	75	30	10	31	4	0.0	105
e	57	53	23	99	5	0.0	110

DataFrame

การแก้ไขข้อมูลทั้งคอลัมน์

```
In [26]: df['J'] = [3,3,3,3,3]
         df
```

Out[26]:

	W	X	Y	Z	I	J	Sum
a	49	18	74	12	1	3	67
b	37	9	81	94	2	3	46
c	44	30	64	47	3	3	74
d	75	30	10	31	4	3	105
e	57	53	23	99	5	3	110

DataFrame

การลบคอลัมน์

```
In [29]: df.drop('I',axis=1,inplace = True)
         df
```

Out[29]:

	W	X	Y	Z	J	Sum
a	49	18	74	12	3	67
b	37	9	81	94	3	46
c	44	30	64	47	3	74
d	75	30	10	31	3	105
e	57	53	23	99	3	110

```
In [30]: df.drop('J',axis=1,inplace = True)
         df.drop('Sum',axis=1,inplace = True)
```

DataFrame

การเพิ่มข้อมูลทีละแถว (สามารถใช้ Series ที่มีเฉพาะบาง index ได้)

```
In [32]: df.loc['f'] = [1,2,3,4]
         df
```

Out[32]:

	W	X	Y	Z
a	49	18	74	12
b	37	9	81	94
c	44	30	64	47
d	75	30	10	31
e	57	53	23	99
f	1	2	3	4

DataFrame

การแก้ไขข้อมูล

```
In [35]: df.loc['f'] = [5,6,7,8]
         df
```

Out[35]:

	W	X	Y	Z
a	49	18	74	12
b	37	9	81	94
c	44	30	64	47
d	75	30	10	31
e	57	53	23	99
f	5	6	7	8

```
In [36]: df.loc['f']['W'] = 9
         df
```

Out[36]:

	W	X	Y	Z
a	49	18	74	12
b	37	9	81	94
c	44	30	64	47
d	75	30	10	31
e	57	53	23	99
f	9	6	7	8

DataFrame

การแก้ไขข้อมูล

```
In [37]: df.loc[ ['a','b'] , 'X':'Z' ] = [ [1,2,3] , [4,5,6] ]
         df
```

Out[37]:

	W	X	Y	Z
a	49	1	2	3
b	37	4	5	6
c	44	30	64	47
d	75	30	10	31
e	57	53	23	99
f	9	6	7	8

DataFrame

การลบข้อมูลทีละแถว

```
In [38]: df.drop('f',axis=0,inplace = True)
         df
```

Out[38]:

	W	X	Y	Z
a	49	1	2	3
b	37	4	5	6
c	44	30	64	47
d	75	30	10	31
e	57	53	23	99

DataFrame

การแก้ชื่อคอลัมน์

```
In [39]: df.rename( columns = {'W':'pi','X':'rad'},inplace = True )
         df
```

Out[39]:

	pi	rad	Y	Z
a	49	1	2	3
b	37	4	5	6
c	44	30	64	47
d	75	30	10	31
e	57	53	23	99

DataFrame

การแก้ชื่อ index

```
In [40]: df.rename( index = {'a':'a1','b':'b1'} , inplace = True )
         df
```

Out[40]:

	pi	rad	Y	Z
a1	49	1	2	3
b1	37	4	5	6
c	44	30	64	47
d	75	30	10	31
e	57	53	23	99

DataFrame

การรีเซ็ต index กลับไปเป็นการนับ

```
In [41]: df.reset_index(inplace = True)
         df
```

Out[41]:

	index	pi	rad	Y	Z
0	a1	49	1	2	3
1	b1	37	4	5	6
2	c	44	30	64	47
3	d	75	30	10	31
4	e	57	53	23	99

```
In [43]: df.loc[3]
Out[43]: index     d
         pi       75
         rad      30
         Y        10
         Z        31
         Name: 3, dtype: object
```

```
In [42]: df.loc['d']
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-42-080085fccc7d> in <module>
----> 1 df.loc['d']
```

DataFrame

การเซ็ตคอลัมน์อื่นๆ ให้เป็น index

```
In [44]: df.rename( columns = {'index':'tmp'} , inplace = True )
         df
```

Out[44]:

	tmp	pi	rad	Y	Z
0	a1	49	1	2	3
1	b1	37	4	5	6
2	c	44	30	64	47
3	d	75	30	10	31
4	e	57	53	23	99

```
In [45]: df.set_index('tmp',inplace=True)
         df
```

Out[45]:

	pi	rad	Y	Z
tmp				
a1	49	1	2	3
b1	37	4	5	6
c	44	30	64	47
d	75	30	10	31
e	57	53	23	99

DataFrame

In [56]:
```
df1 = pd.DataFrame({ 'A': ['A0', 'A1', 'A2', 'A3'],
                     'B': ['B0', 'B1', 'B2', 'B3'],
                     'C': ['C0', 'C1', 'C2', 'C3'],
                     'D': ['D0', 'D1', 'D2', 'D3']},
                       index=[0, 1, 2, 3])
df2 = pd.DataFrame({ 'A': ['A4', 'A5', 'A6', 'A7'],
                     'B': ['B4', 'B5', 'B6', 'B7'],
                     'C': ['C4', 'C5', 'C6', 'C7'],
                     'E': ['D4', 'D5', 'D6', 'D7']},
                       index=[2, 3, 4, 5])
df3 = pd.DataFrame({ 'A': ['A8', 'A9', 'A10', 'A11'],
                     'B': ['B8', 'B9', 'B10', 'B11'],
                     'C': ['C8', 'C9', 'C10', 'C11'],
                     'D': ['D8', 'D9', 'D10', 'D11']},
                       index=[6, 7, 8, 9])
```

In [57]:
```
display(df1,df2,df3)
```

	A	B	C	D
0	A0	B0	C0	D0
1	A1	B1	C1	D1
2	A2	B2	C2	D2
3	A3	B3	C3	D3

	A	B	C	E
2	A4	B4	C4	D4
3	A5	B5	C5	D5
4	A6	B6	C6	D6
5	A7	B7	C7	D7

	A	B	C	D
6	A8	B8	C8	D8
7	A9	B9	C9	D9
8	A10	B10	C10	D10
9	A11	B11	C11	D11

DataFrame

การนำแถวของ df มาต่อกัน

```
In [58]: df = pd.concat( [df1,df2] )
         df
```

Out[58]:

	A	B	C	D	E
0	A0	B0	C0	D0	NaN
1	A1	B1	C1	D1	NaN
2	A2	B2	C2	D2	NaN
3	A3	B3	C3	D3	NaN
2	A4	B4	C4	NaN	D4
3	A5	B5	C5	NaN	D5
4	A6	B6	C6	NaN	D6
5	A7	B7	C7	NaN	D7

DataFrame

การนำหลักของ df มาต่อกัน

```
In [61]: df = pd.concat( [df1,df2,df3] ,axis = 1 )
         df
```

Out[61]:

	A	B	C	D	A	B	C	E	A	B	C	D
0	A0	B0	C0	D0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	A1	B1	C1	D1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	A2	B2	C2	D2	A4	B4	C4	D4	NaN	NaN	NaN	NaN
3	A3	B3	C3	D3	A5	B5	C5	D5	NaN	NaN	NaN	NaN
4	NaN	NaN	NaN	NaN	A6	B6	C6	D6	NaN	NaN	NaN	NaN
5	NaN	NaN	NaN	NaN	A7	B7	C7	D7	NaN	NaN	NaN	NaN
6	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	A8	B8	C8	D8
7	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	A9	B9	C9	D9
8	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	A10	B10	C10	D10
9	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	A11	B11	C11	D11

DataFrame

In [71]:
```
d = {
'A': [1,2,np.nan],
'B': [5,np.nan,np.nan],
'C': [1,2,3]
}
df = pd.DataFrame(d)
df
```

Out[71]:

	A	B	C
0	1.0	5.0	1
1	2.0	NaN	2
2	NaN	NaN	3

DataFrame

การนำแถวที่เป็น Null ออก

```
In [72]: df.dropna(inplace = False)
```

Out[72]:

	A	B	C
0	1.0	5.0	1

```
In [74]: df.dropna(thresh = 2)
```

Out[74]:

	A	B	C
0	1.0	5.0	1
1	2.0	NaN	2

DataFrame

การนำคอมลัมน์ที่เป็น Null ออก

```
In [75]: df.dropna(axis = 1 )
```

Out[75]:

	C
0	1
1	2
2	3

DataFrame

การแก้ไขค่าที่เป็น Null

```
In [76]: df.fillna(value = 10)
```

Out[76]:

	A	B	C
0	1.0	5.0	1
1	2.0	10.0	2
2	10.0	10.0	3

```
In [78]: df['B'].fillna(value=10)
```

```
Out[78]: 0     5.0
         1    10.0
         2    10.0
         Name: B, dtype: float64
```

```
In [80]: df['A'].fillna(value= df['A'].mean() ,inplace = True )
         df
```

Out[80]:

	A	B	C
0	1.0	5.0	1
1	2.0	NaN	2
2	1.5	NaN	3

DataFrame

การตั้งค่าให้เป็น null

```
In [125]: df.loc[2,'A'] = None
          df
```

Out[125]:

	A	B	C
0	1.0	5.0	1
1	2.0	NaN	2
2	NaN	NaN	3

DataFrame

การเช็ค่าคอลัมน์ไหนมี null อยู่บ้าง

```
In [96]: df['A'].isnull()

Out[96]: 0    False
         1    False
         2     True
         Name: A, dtype: bool

In [97]: df[df['A'].isnull()]
```

Out[97]:

	A	B	C
2	NaN	NaN	3

DataFrame

การเช็ค่าคอลัมน์ไหนไม่มี null อยู่บ้าง

```
In [126]: df['A'].notnull()

Out[126]: 0     True
          1     True
          2    False
          Name: A, dtype: bool

In [127]: df[df['A'].notnull()]

Out[127]:
```

	A	B	C
0	1.0	5.0	1
1	2.0	NaN	2