Pandas-7

January 13, 2023

[25]:
```
import numpy as np
import pandas as pd
```

[29]:
```
# örnek data frame oluşturma
data = np.random.randint(-20,20,450).reshape(50,9)
kolonlar = ["k1","k2","k3","k4","k5","k6","k7","k8","k9"]

df = pd.DataFrame(data, columns=kolonlar)
df.head(10)
```

[29]:

	k1	k2	k3	k4	k5	k6	k7	k8	k9
0	13	8	-3	4	-9	-20	1	4	-1
1	-8	15	-5	-4	5	-15	15	-17	19
2	-13	-14	5	12	14	10	-8	3	18
3	3	-13	-20	-10	16	-8	-20	3	-14
4	-13	18	-18	12	-15	18	-11	-5	-3
5	3	-16	-17	-6	16	9	-18	-16	13
6	19	12	-9	3	4	1	3	-20	18
7	-14	-19	-19	11	-8	-19	-17	-5	16
8	-10	-6	8	-15	-5	-19	18	-12	7
9	-11	10	10	-11	-7	2	2	-7	17

[32]:
```
# frame içerisindeki 2 ye bölünebilenler
df[df % 2 ==0].head()
```

[32]:

	k1	k2	k3	k4	k5	k6	k7	k8	k9
0	NaN	8.0	NaN	4.0	NaN	-20.0	NaN	4.0	NaN
1	-8.0	NaN	NaN	-4.0	NaN	NaN	NaN	NaN	NaN
2	NaN	-14.0	NaN	12.0	14.0	10.0	-8.0	NaN	18.0
3	NaN	NaN	-20.0	-10.0	16.0	-8.0	-20.0	NaN	-14.0
4	NaN	18.0	-18.0	12.0	NaN	18.0	NaN	NaN	NaN

[16]:
```
# frame içerisindeki 5 ten büyük olanlar
df[df > 5].head()
```

[16]:

	k1	k2	k3	k4	k5	k6	k7	k8	k9
0	NaN	NaN	13.0	NaN	7.0	NaN	NaN	NaN	16.0
1	NaN	NaN	10.0	6.0	NaN	NaN	NaN	NaN	NaN

```
2  19.0 NaN  18.0   NaN  16.0  NaN  NaN NaN  17.0
3   NaN NaN   NaN   NaN   NaN  6.0  NaN NaN  15.0
4   NaN NaN   NaN  15.0   NaN  NaN  8.0 NaN   9.0
```

[17]:
```
# frame içerisindeki 5 ten küçük olanlar
df[df < 5].head()
```

[17]:
```
     k1   k2   k3    k4    k5    k6    k7    k8   k9
0 -16.0  NaN  NaN   NaN   NaN -16.0 -20.0  -6.0  NaN
1  -3.0 -3.0  NaN   NaN  -9.0 -11.0 -14.0 -13.0  1.0
2   NaN  1.0  NaN  -7.0   NaN  -6.0   2.0  -7.0  NaN
3   3.0 -1.0  NaN -12.0 -12.0   NaN  -3.0  -1.0  NaN
4 -17.0  0.0  0.0   NaN -10.0   3.0   NaN -11.0  NaN
```

[36]:
```
# frame içerisindeki k2 sütununda 5 ten küçük olanlar
df[df["k2"] < 5]
```

[36]:
```
    k1  k2  k3  k4  k5  k6  k7  k8  k9
2  -13 -14   5  12  14  10  -8   3  18
3    3 -13 -20 -10  16  -8 -20   3 -14
5    3 -16 -17  -6  16   9 -18 -16  13
7  -14 -19 -19  11  -8 -19 -17  -5  16
8  -10  -6   8 -15  -5 -19  18 -12   7
13  -9   1  18  19   3  17 -18  15   6
14   5 -14  12  -9  -4 -18  -8 -20  12
15 -15   3   7 -10  -6  -8 -20 -18   9
16 -19 -19 -14  14 -14  16  13  13   9
17 -11 -19  -4   0   0   7  10 -13 -15
18  19  -3  -2  10 -19   5  -4   2   8
19 -12 -19 -20  11   7  14  -7 -12  -5
24 -16   4   6  -8 -12   3   0  11  -9
25   8 -10  13  -3 -12  -6  18 -18  -1
26  -1 -13 -17  14   2  16   5  17 -10
27   0 -13   3   4  -4   1 -11 -16 -14
28 -11   2 -10 -16  -1  13 -19 -13 -17
29  10 -20 -16   2 -14  17 -13   9  17
32  -9  -2   6  18  14  16   0  -3 -18
33  -3  -7  -5 -15 -12  -3  16  17  12
34  15 -16   5  -3 -12 -10  -5  18   2
37  -2 -14  19 -14 -17  -9  12  13  10
38  15 -18  -9  -8  11 -13  -5  -6  11
39  18 -10  -7  13  10  12  14  -3  19
40 -10 -16 -20 -20 -11  -3  -5  -1   7
41  -6  -6 -14   3 -16 -11   8   4  16
42  11 -16  -2  -7  -5   5  12  -5   5
45   2 -15  19  18 -17  -3  11   7   4
46 -16  -8  19  -3 -17  -8  14 -14  14
```

```
47   7  -6   4   1   9  -8   0  -4  -7
48 -18 -17   6  -8  12  18  15   1 -19
49  18   4  19 -15  -1 -16 -17 -16  -3
```

[43]:
```
# k1 sütununa göre filtreleme
df[df["k1"] % 2 ==0].head(10)
```

[43]:
```
     k1  k2  k3  k4  k5  k6  k7  k8  k9
1    -8  15  -5  -4   5 -15  15 -17  19
7   -14 -19 -19  11  -8 -19 -17  -5  16
8   -10  -6   8 -15  -5 -19  18 -12   7
11  -8  14 -10  -5 -19  12  17   2 -12
19 -12 -19 -20  11   7  14  -7 -12  -5
21  -2  19  12  15  -9   8  -7  -5 -18
23  -4  13   8  -4 -13  -3  19 -12  12
24 -16   4   6  -8 -12   3   0  11  -9
25   8 -10  13  -3 -12  -6  18 -18  -1
27   0 -13   3   4  -4   1 -11 -16 -14
```

[45]:
```
# k1 sütununa göre filtreleme ve sadece k1 sütununu gösterme
df[df["k1"] % 2 == 0]["k1"] #.loc[:,"k1"]
```

[45]:
```
1     -8
7    -14
8    -10
11    -8
19   -12
21    -2
23    -4
24   -16
25     8
27     0
29    10
31    14
35     4
37    -2
39    18
40   -10
41    -6
43    14
44   -20
45     2
46   -16
48   -18
49    18
Name: k1, dtype: int32
```

[59]:
```
# k1 ve k3 sütununa göre filtreleme
df[(df["k1"] > 0) & (df["k3"] < 0) & (df["k7"] % 2 == 0)].loc[:
 ↪,["k1","k3","k7"]]
```

[59]:

	k1	k3	k7
3	3	-20	-20
5	3	-17	-18
18	19	-2	-4
22	17	-2	0
31	14	-16	-20
39	18	-7	14
42	11	-2	12

[22]:
```
# k1 veya k3 sütununa göre filtreleme
df[(df["k1"] < 5) | (df["k3"] < 0)].loc[:,["k1","k3"]]
```

[22]:

	k1	k3
0	-16	13
1	-3	10
3	3	5
4	-17	0
5	10	-18
7	14	-10
8	0	-12
9	3	-8
10	-8	-16
11	17	-14
12	-3	18
13	-1	15
15	-17	-7
16	-13	13
17	-2	2
18	-14	2
19	-2	14
21	-6	7
22	-12	-10
23	-16	11
24	-1	7
25	-12	-17
26	-10	-6
27	-6	17
28	-1	16
29	0	-1
30	-20	15
31	-4	5
32	18	-3
33	2	3

```
34 -18 -15
35 -11   6
36   0 -14
37  -8 -18
38   0 -12
39  10  -3
40 -12  12
41 -20  -6
42   8  -4
43  -6  11
44  -2  13
45  -8 -11
46 -10   3
48  -9 -18
49  -4   2
```

[71]:
```
# k2 ve k5 sütunlarını query ile filtreleme
df.query("k2 % 2 ==0 & k5 %2 == 0").loc[:,["k2","k5"]]
```

[71]:
```
    k2  k5
2  -14  14
5  -16  16
6   12   4
12  12  -8
14 -14  -4
20   8  18
24   4 -12
25 -10 -12
29 -20 -14
32  -2  14
34 -16 -12
39 -10  10
41  -6 -16
44   8  16
```

[72]:
```
# k2 veya k5 sütunlarını query ile filtreleme
df.query("not(k2 % 2 ==0 | k5 %2 == 0)").loc[:,["k2","k5"]]
```

[72]:
```
    k2  k5
1   15   5
10  17  -3
13   1   3
18  -3 -19
19 -19   7
21  19  -9
22   9 -11
23  13 -13
```

```
45 -15 -17
```

[ ]:

[ ]: