Module 02

# Pandas: Indexing and Selecting Data

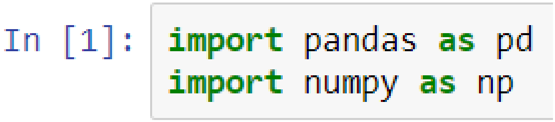
Data Science Developer

Purwadhika
Startup and Coding School

# Outline

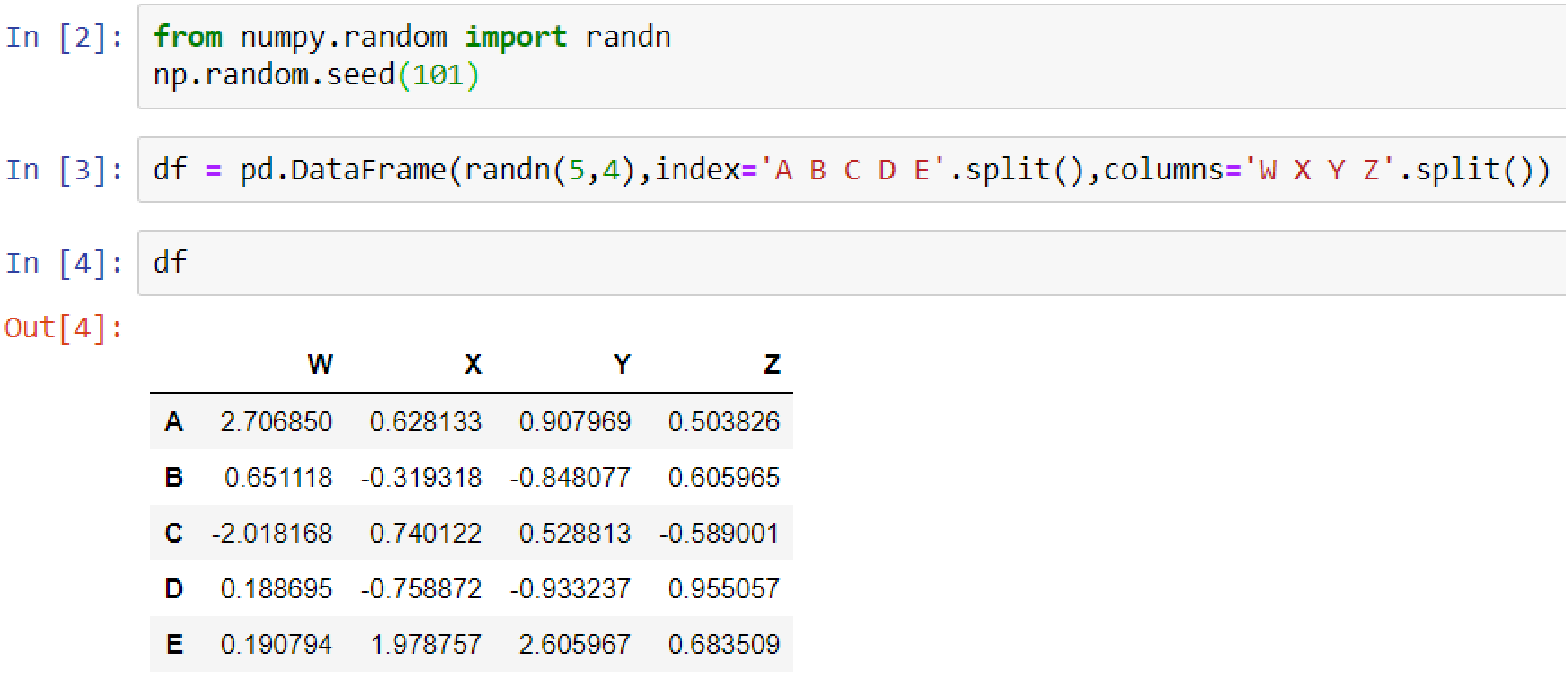
- Data Selection and Indexing
- Conditional Selection

# Using Numpy and Pandas



```
In [1]: import pandas as pd
        import numpy as np
```

# Create a DataFrame



```
In [2]: from numpy.random import randn
        np.random.seed(101)
```

```
In [3]: df = pd.DataFrame(randn(5,4),index='A B C D E'.split(),columns='W X Y Z'.split())
```

```
In [4]: df
```

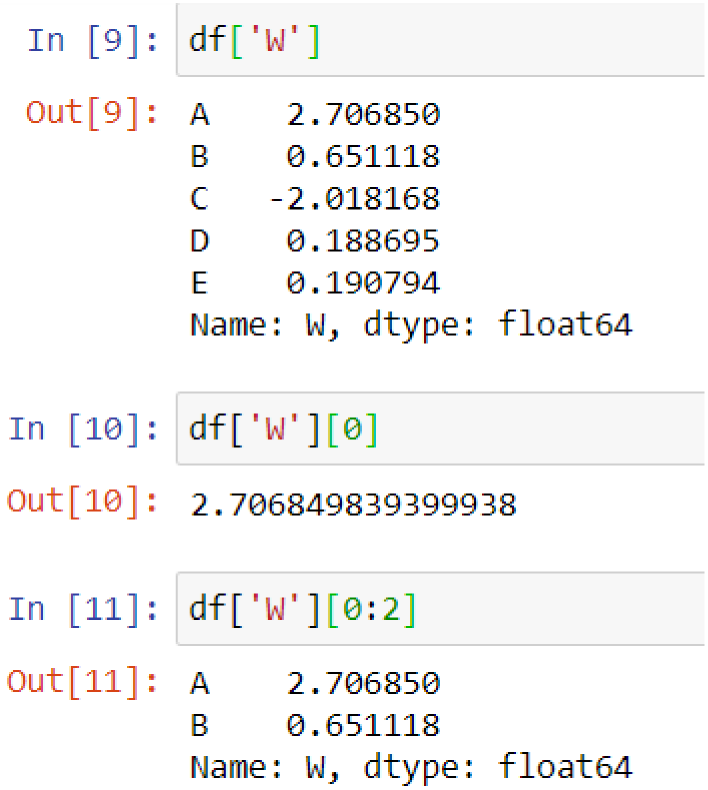
```
Out[4]:
           W          X          Y          Z
A   2.706850   0.628133   0.907969   0.503826
B   0.651118  -0.319318  -0.848077   0.605965
C  -2.018168   0.740122   0.528813  -0.589001
D   0.188695  -0.758872  -0.933237   0.955057
E   0.190794   1.978757   2.605967   0.683509
```

Purwadhika
Startup and Coding School

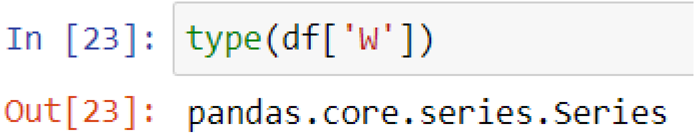
# Data Selection and Indexing

Purwadhika
Startup and Coding School

# Selection and Indexing



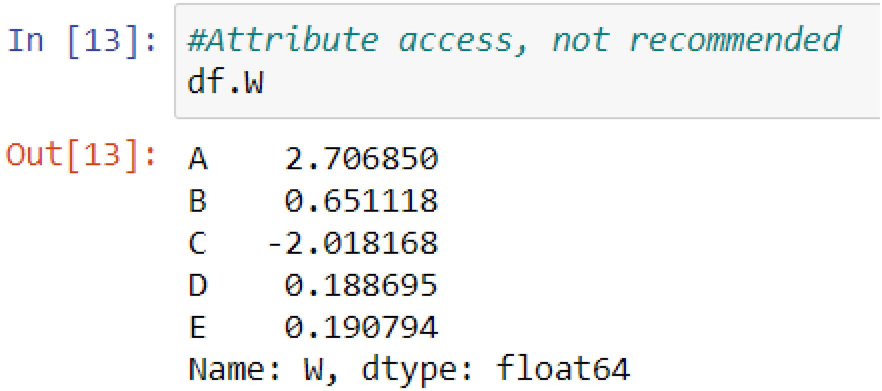
```
In [9]: df['W']

Out[9]: A    2.706850
        B    0.651118
        C   -2.018168
        D    0.188695
        E    0.190794
        Name: W, dtype: float64

In [10]: df['W'][0]

Out[10]: 2.706849839399938

In [11]: df['W'][0:2]

Out[11]: A    2.706850
         B    0.651118
         Name: W, dtype: float64
```



```
In [23]: type(df['W'])

Out[23]: pandas.core.series.Series
```

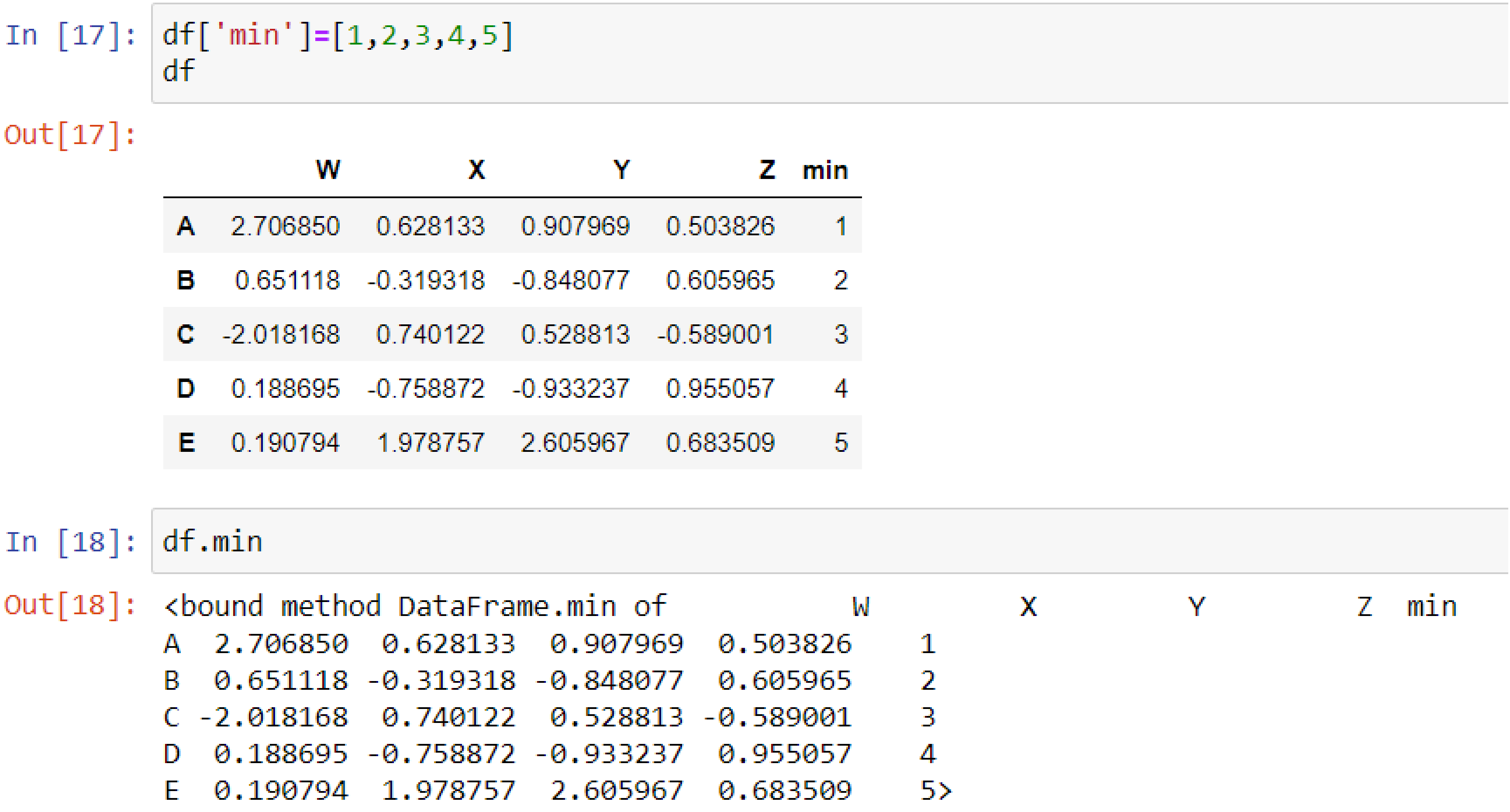
Purwadhika
Startup and Coding School

# Selection and Indexing



```
In [13]: #Attribute access, not recommended
         df.W

Out[13]: A    2.706850
         B    0.651118
         C   -2.018168
         D    0.188695
         E    0.190794
         Name: W, dtype: float64
```

# Selection and Indexing



```
In [17]: df['min']=[1,2,3,4,5]
         df
```

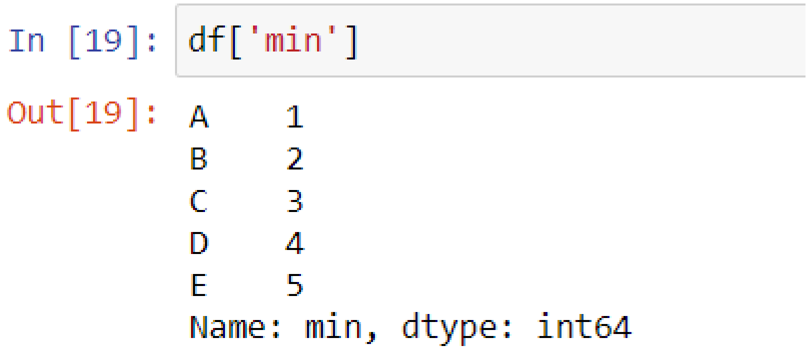
Out[17]:

| | W | X | Y | Z | min |
|---|---|---|---|---|---|
| **A** | 2.706850 | 0.628133 | 0.907969 | 0.503826 | 1 |
| **B** | 0.651118 | -0.319318 | -0.848077 | 0.605965 | 2 |
| **C** | -2.018168 | 0.740122 | 0.528813 | -0.589001 | 3 |
| **D** | 0.188695 | -0.758872 | -0.933237 | 0.955057 | 4 |
| **E** | 0.190794 | 1.978757 | 2.605967 | 0.683509 | 5 |

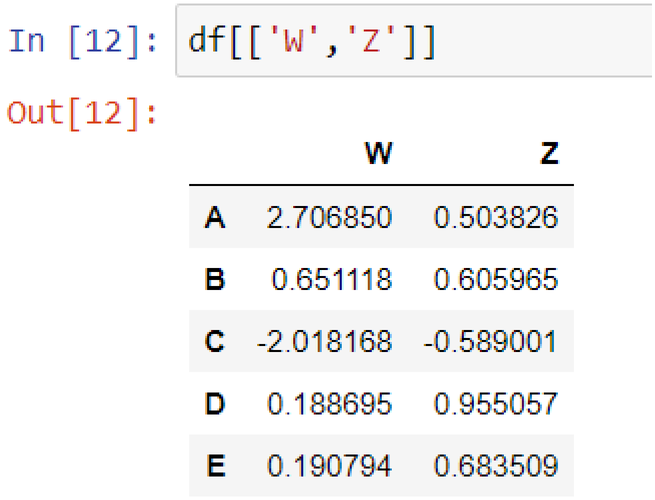
```
In [18]: df.min
```

```
Out[18]: <bound method DataFrame.min of            W         X         Y         Z  min
A  2.706850  0.628133  0.907969  0.503826    1
B  0.651118 -0.319318 -0.848077  0.605965    2
C -2.018168  0.740122  0.528813 -0.589001    3
D  0.188695 -0.758872 -0.933237  0.955057    4
E  0.190794  1.978757  2.605967  0.683509    5>
```

# Selection and Indexing



```
In [19]: df['min']

Out[19]: A     1
         B     2
         C     3
         D     4
         E     5
         Name: min, dtype: int64
```

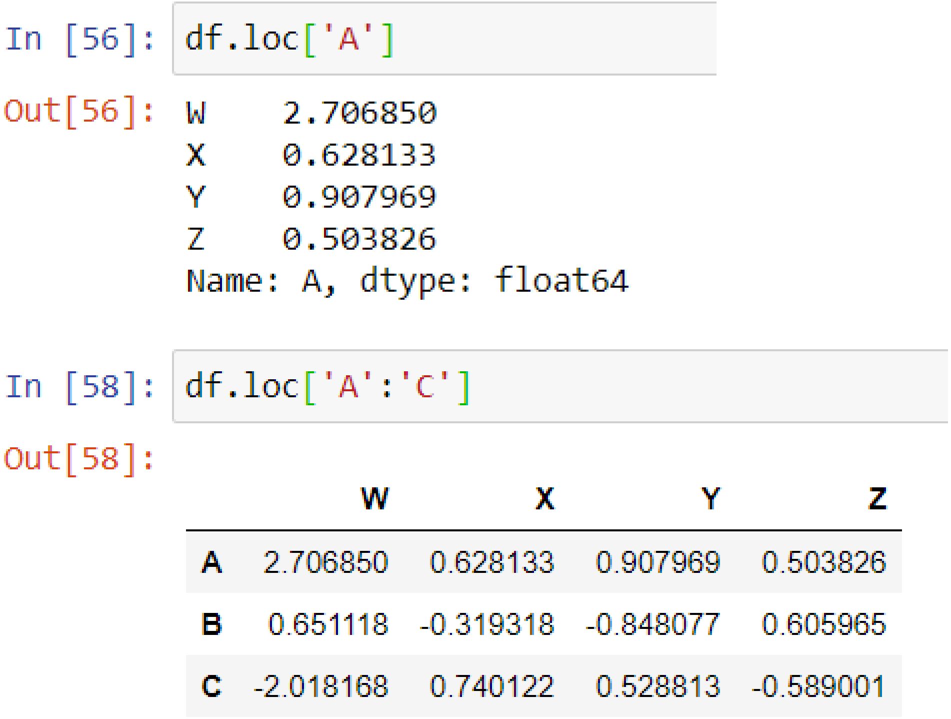


```
In [12]: df[['W','Z']]

Out[12]:
```

|   | W | Z |
|---|---|---|
| **A** | 2.706850 | 0.503826 |
| **B** | 0.651118 | 0.605965 |
| **C** | -2.018168 | -0.589001 |
| **D** | 0.188695 | 0.955057 |
| **E** | 0.190794 | 0.683509 |

# Selection and Indexing

## loc



```
In [56]: df.loc['A']

Out[56]: W    2.706850
         X    0.628133
         Y    0.907969
         Z    0.503826
         Name: A, dtype: float64
```

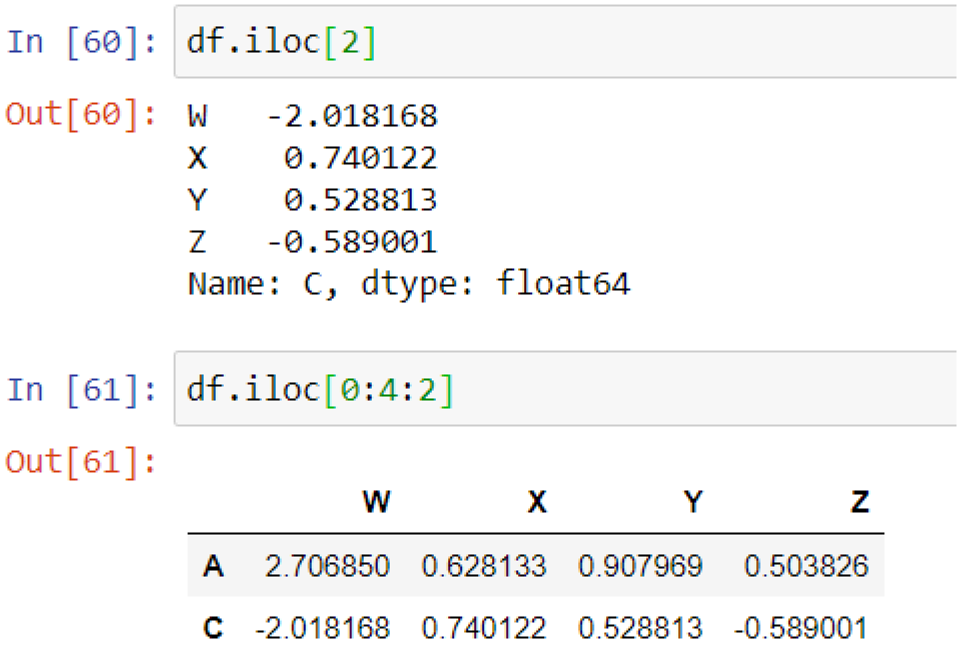
```
In [58]: df.loc['A':'C']

Out[58]:
```

|   | W | X | Y | Z |
|---|---|---|---|---|
| **A** | 2.706850 | 0.628133 | 0.907969 | 0.503826 |
| **B** | 0.651118 | -0.319318 | -0.848077 | 0.605965 |
| **C** | -2.018168 | 0.740122 | 0.528813 | -0.589001 |

Purwadhika
Startup and Coding School

# Selection and Indexing

## iloc



```
In [60]: df.iloc[2]
```

```
Out[60]: W   -2.018168
         X    0.740122
         Y    0.528813
         Z   -0.589001
         Name: C, dtype: float64
```

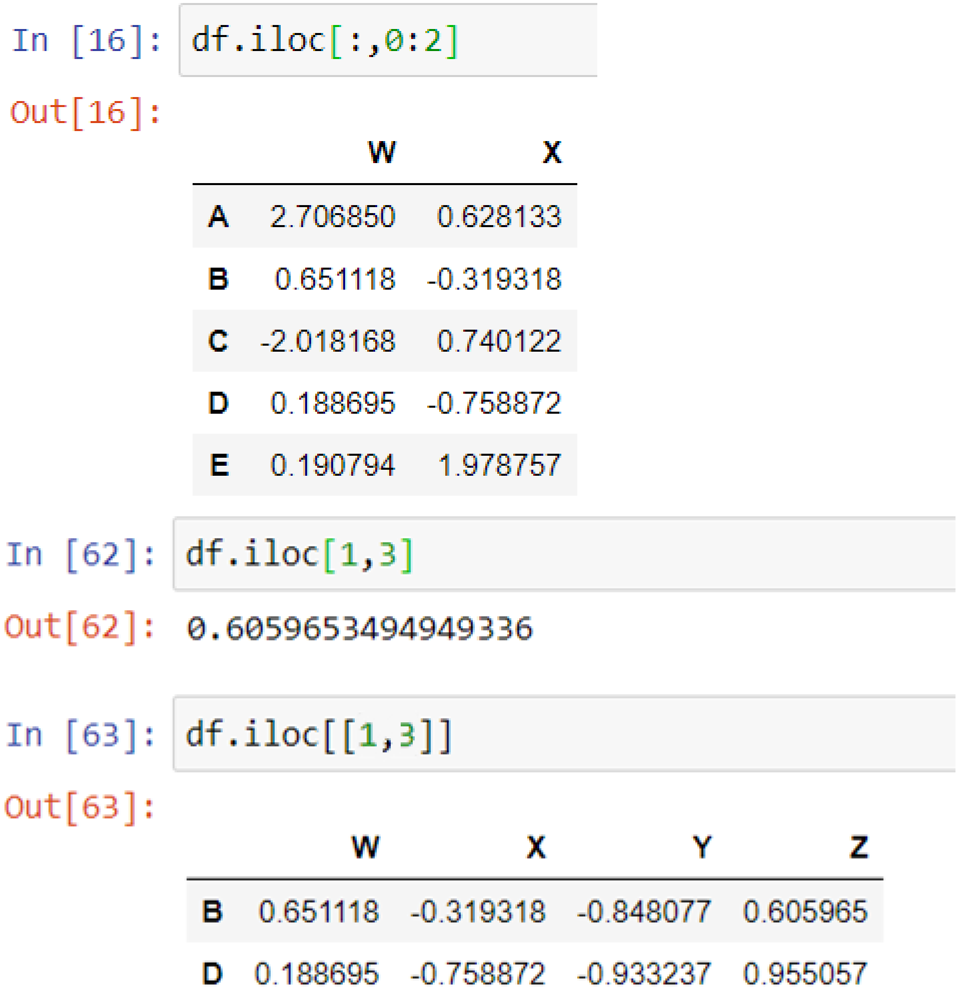
```
In [61]: df.iloc[0:4:2]
```

Out[61]:

|   | W | X | Y | Z |
|---|---|---|---|---|
| **A** | 2.706850 | 0.628133 | 0.907969 | 0.503826 |
| **C** | -2.018168 | 0.740122 | 0.528813 | -0.589001 |

# Selection and Indexing

## iloc



```
In [16]: df.iloc[:,0:2]

Out[16]:
          W          X
A   2.706850   0.628133
B   0.651118  -0.319318
C  -2.018168   0.740122
D   0.188695  -0.758872
E   0.190794   1.978757
```

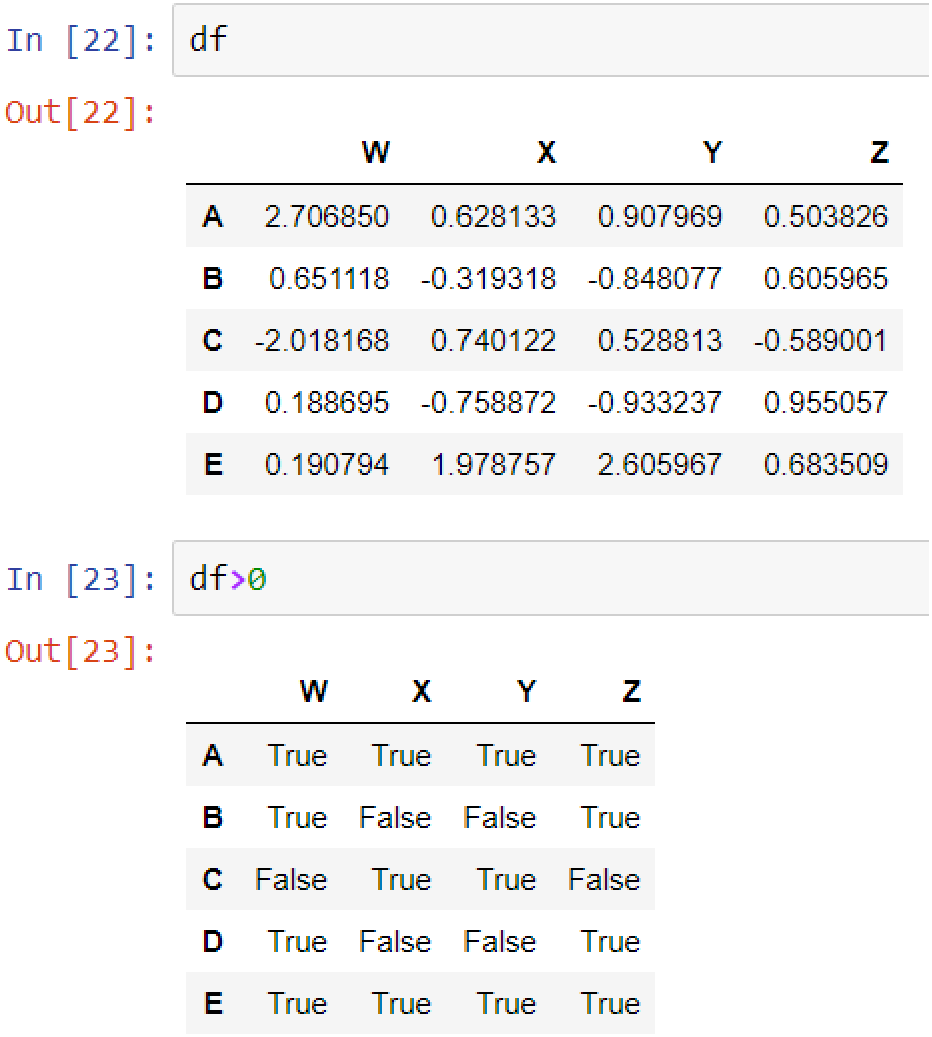
```
In [62]: df.iloc[1,3]

Out[62]: 0.6059653494949336
```

```
In [63]: df.iloc[[1,3]]

Out[63]:
          W          X          Y         Z
B  0.651118  -0.319318  -0.848077  0.605965
D  0.188695  -0.758872  -0.933237  0.955057
```

Purwadhika
Startup and Coding School

# Conditional Selection

# Conditional Selection



In [22]:

```
df
```

Out[22]:

|   | W | X | Y | Z |
|---|---|---|---|---|
| **A** | 2.706850 | 0.628133 | 0.907969 | 0.503826 |
| **B** | 0.651118 | -0.319318 | -0.848077 | 0.605965 |
| **C** | -2.018168 | 0.740122 | 0.528813 | -0.589001 |
| **D** | 0.188695 | -0.758872 | -0.933237 | 0.955057 |
| **E** | 0.190794 | 1.978757 | 2.605967 | 0.683509 |

In [23]:

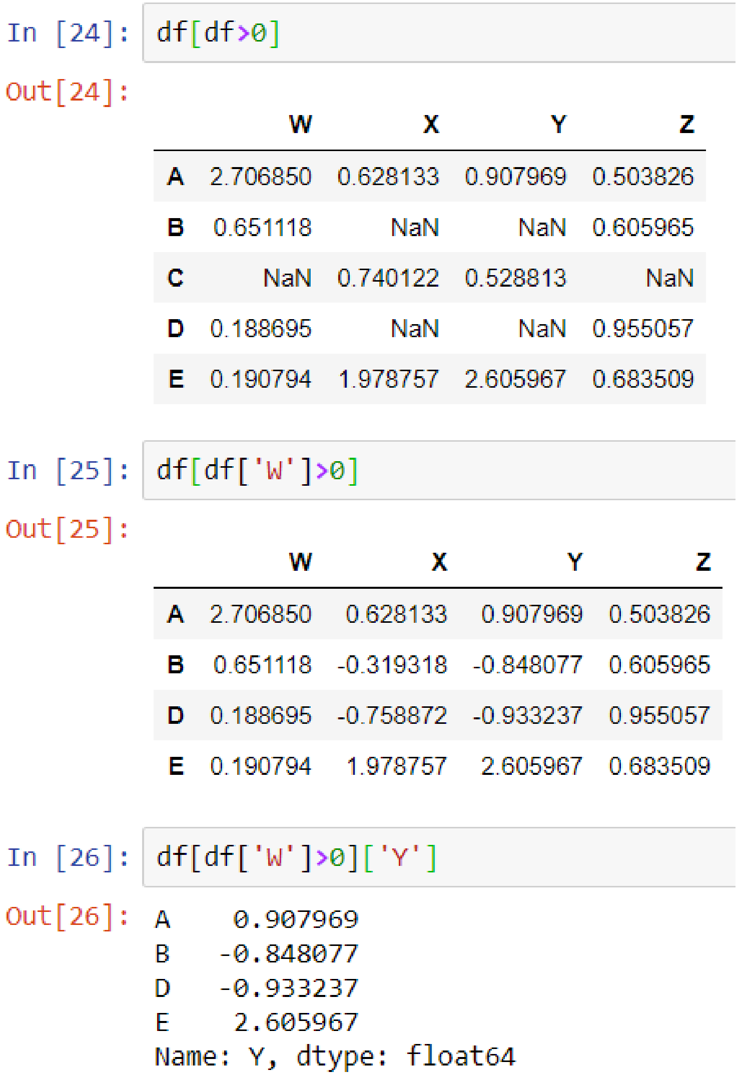
```
df>0
```

Out[23]:

|   | W | X | Y | Z |
|---|---|---|---|---|
| **A** | True | True | True | True |
| **B** | True | False | False | True |
| **C** | False | True | True | False |
| **D** | True | False | False | True |
| **E** | True | True | True | True |

Purwadhika
Startup and Coding School

# Conditional Selection



```
In [24]: df[df>0]
Out[24]:
```

| | W | X | Y | Z |
|---|---|---|---|---|
| **A** | 2.706850 | 0.628133 | 0.907969 | 0.503826 |
| **B** | 0.651118 | NaN | NaN | 0.605965 |
| **C** | NaN | 0.740122 | 0.528813 | NaN |
| **D** | 0.188695 | NaN | NaN | 0.955057 |
| **E** | 0.190794 | 1.978757 | 2.605967 | 0.683509 |

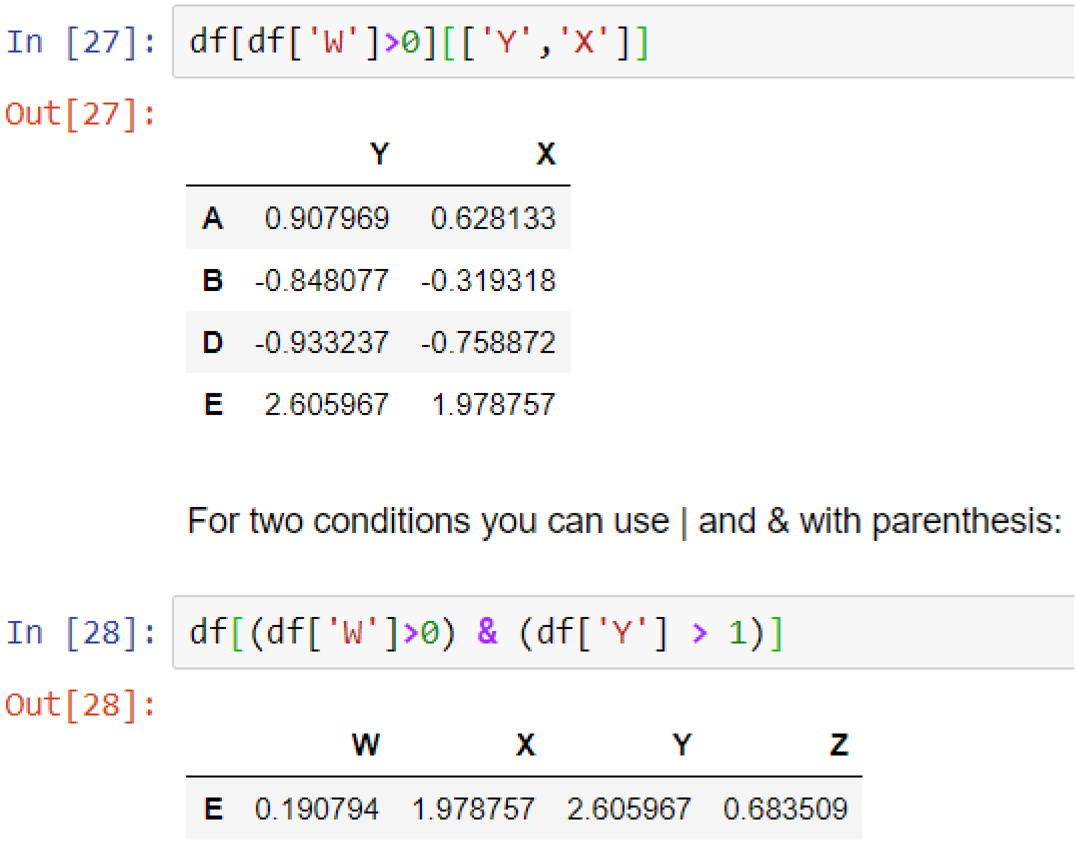
```
In [25]: df[df['W']>0]
Out[25]:
```

| | W | X | Y | Z |
|---|---|---|---|---|
| **A** | 2.706850 | 0.628133 | 0.907969 | 0.503826 |
| **B** | 0.651118 | -0.319318 | -0.848077 | 0.605965 |
| **D** | 0.188695 | -0.758872 | -0.933237 | 0.955057 |
| **E** | 0.190794 | 1.978757 | 2.605967 | 0.683509 |

```
In [26]: df[df['W']>0]['Y']
Out[26]: A    0.907969
         B   -0.848077
         D   -0.933237
         E    2.605967
         Name: Y, dtype: float64
```

Purwadhika
Startup and Coding School

# Conditional Selection



```
In [27]: df[df['W']>0][['Y','X']]
```

Out[27]:

|   | Y | X |
|---|---|---|
| **A** | 0.907969 | 0.628133 |
| **B** | -0.848077 | -0.319318 |
| **D** | -0.933237 | -0.758872 |
| **E** | 2.605967 | 1.978757 |

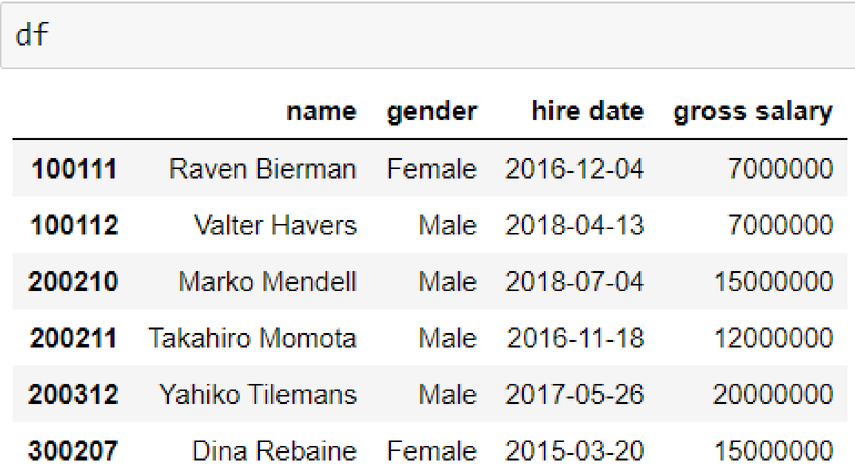
For two conditions you can use | and & with parenthesis:

```
In [28]: df[(df['W']>0) & (df['Y'] > 1)]
```

Out[28]:

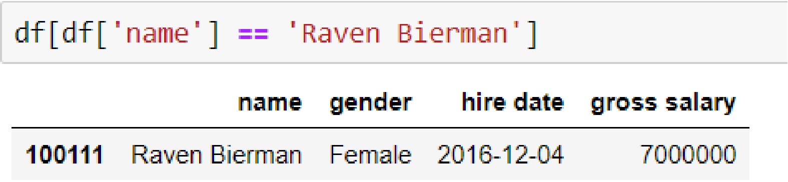
|   | W | X | Y | Z |
|---|---|---|---|---|
| **E** | 0.190794 | 1.978757 | 2.605967 | 0.683509 |

# Conditional Selection



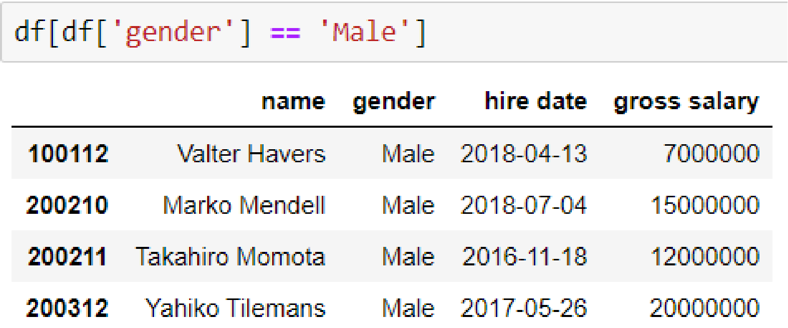
```
df
```

|  | name | gender | hire date | gross salary |
|---|---|---|---|---|
| **100111** | Raven Bierman | Female | 2016-12-04 | 7000000 |
| **100112** | Valter Havers | Male | 2018-04-13 | 7000000 |
| **200210** | Marko Mendell | Male | 2018-07-04 | 15000000 |
| **200211** | Takahiro Momota | Male | 2016-11-18 | 12000000 |
| **200312** | Yahiko Tilemans | Male | 2017-05-26 | 20000000 |
| **300207** | Dina Rebaine | Female | 2015-03-20 | 15000000 |



```
df[df['name'] == 'Raven Bierman']
```

|  | name | gender | hire date | gross salary |
|---|---|---|---|---|
| **100111** | Raven Bierman | Female | 2016-12-04 | 7000000 |



```
df[df['gender'] == 'Male']
```

|  | name | gender | hire date | gross salary |
|---|---|---|---|---|
| **100112** | Valter Havers | Male | 2018-04-13 | 7000000 |
| **200210** | Marko Mendell | Male | 2018-07-04 | 15000000 |
| **200211** | Takahiro Momota | Male | 2016-11-18 | 12000000 |
| **200312** | Yahiko Tilemans | Male | 2017-05-26 | 20000000 |

# References

- Pandas User Guide: Indexing and selecting data https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html