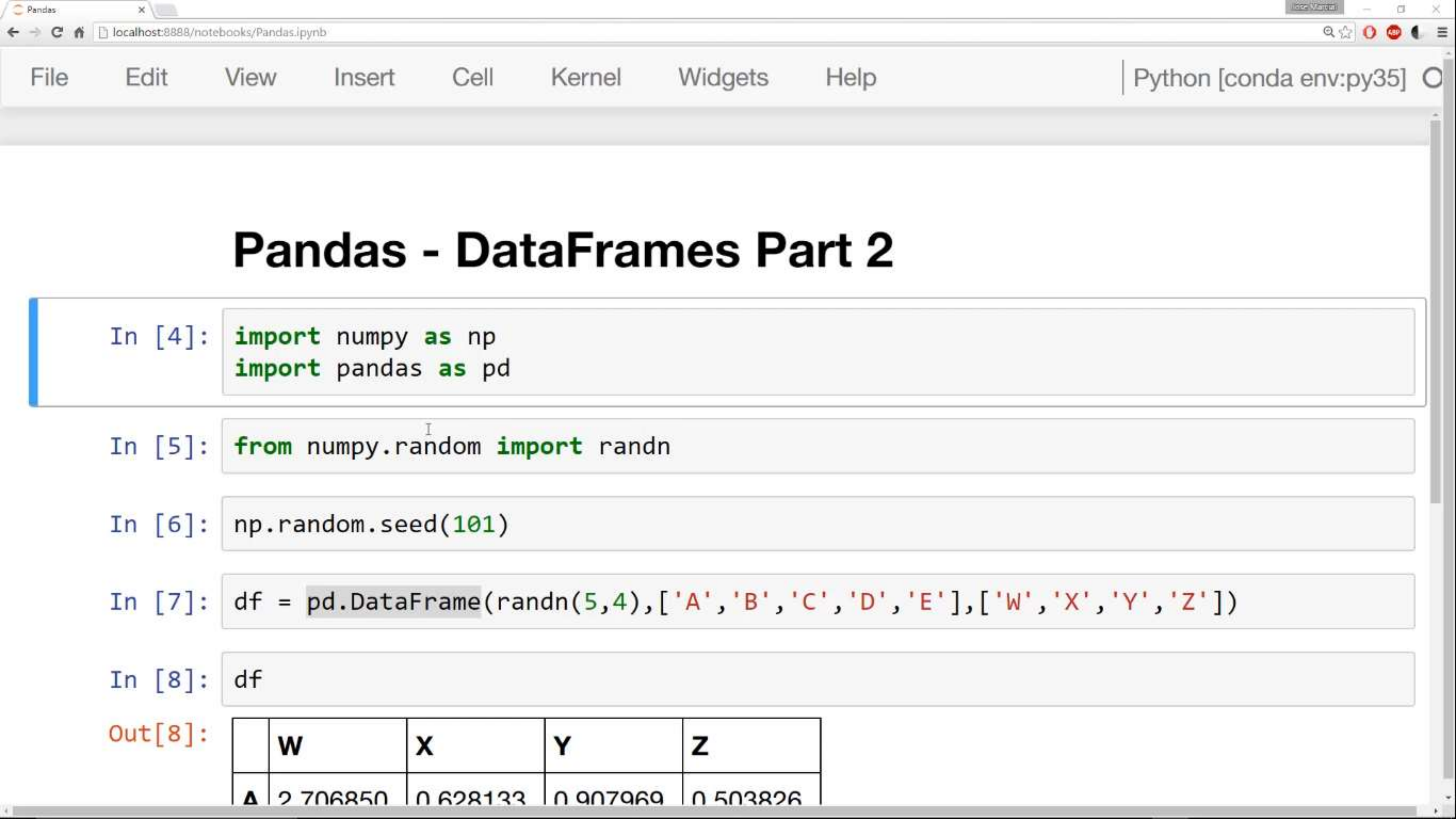
Python for Data Science

DataFrames - Part 2

PIERIAN DATA
www.pieriandata.com



Pandas - DataFrames Part 2

In [4]:
```
import numpy as np
import pandas as pd
```

In [5]:
```
from numpy.random import randn
```

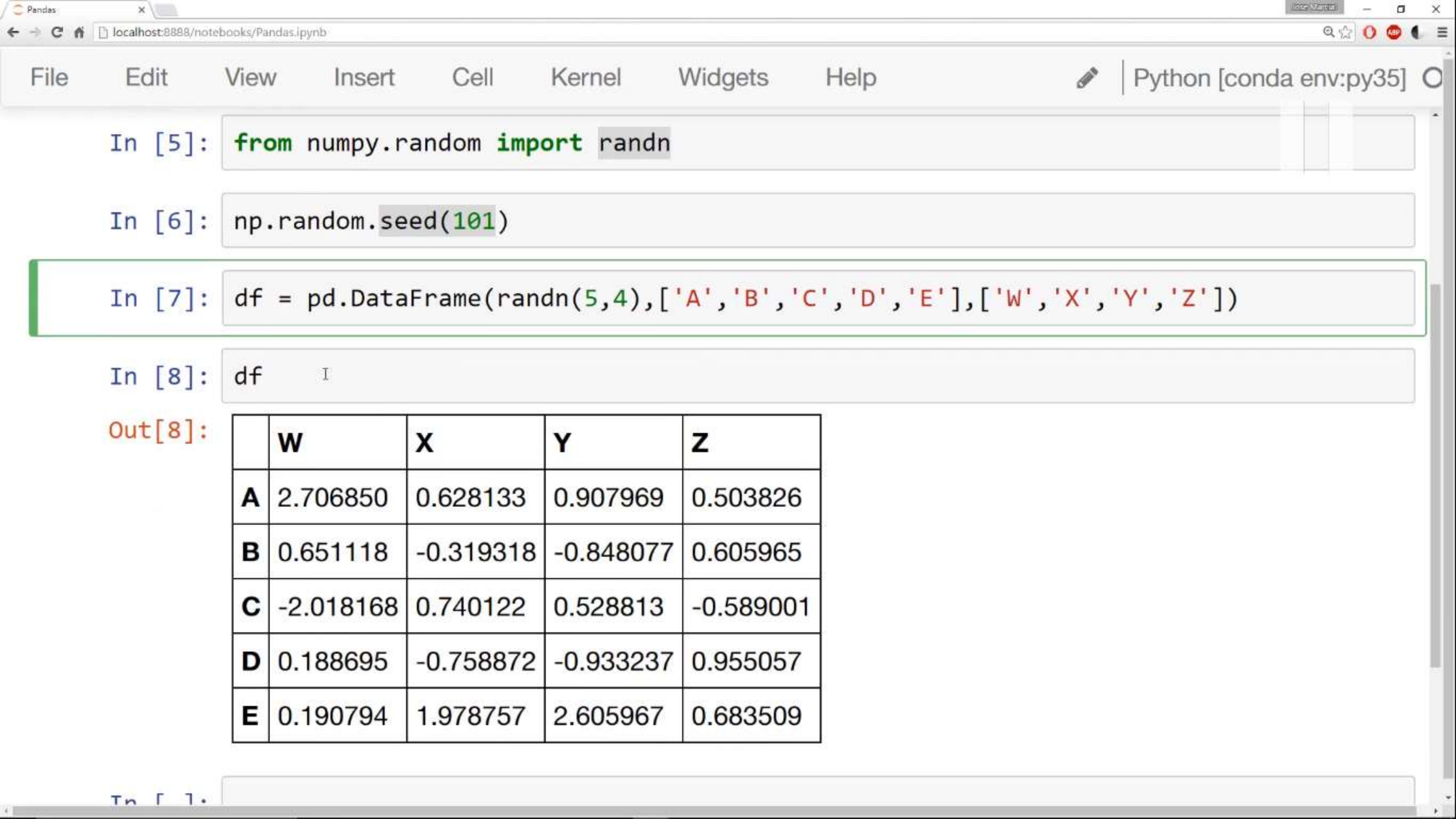
In [6]:
```
np.random.seed(101)
```

In [7]:
```
df = pd.DataFrame(randn(5,4),['A','B','C','D','E'],['W','X','Y','Z'])
```

In [8]:
```
df
```

Out[8]:

	W	X	Y	Z
A	2.706850	0.628133	0.907969	0.503826



In [5]:
```
from numpy.random import randn
```

In [6]:
```
np.random.seed(101)
```

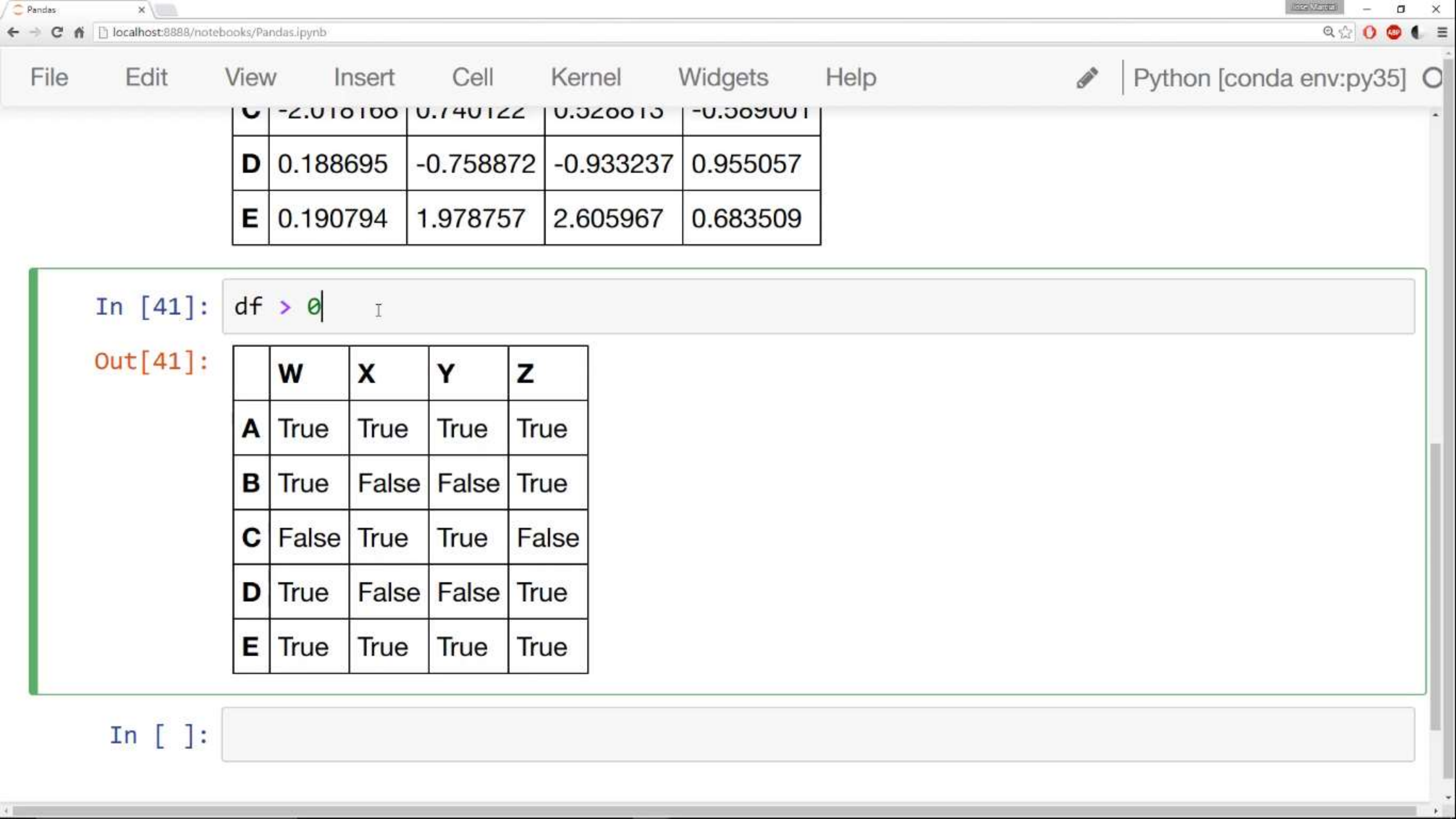
In [7]:
```
df = pd.DataFrame(randn(5,4),['A','B','C','D','E'],['W','X','Y','Z'])
```

In [8]:
```
df
```

Out[8]:

	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 [ ]:



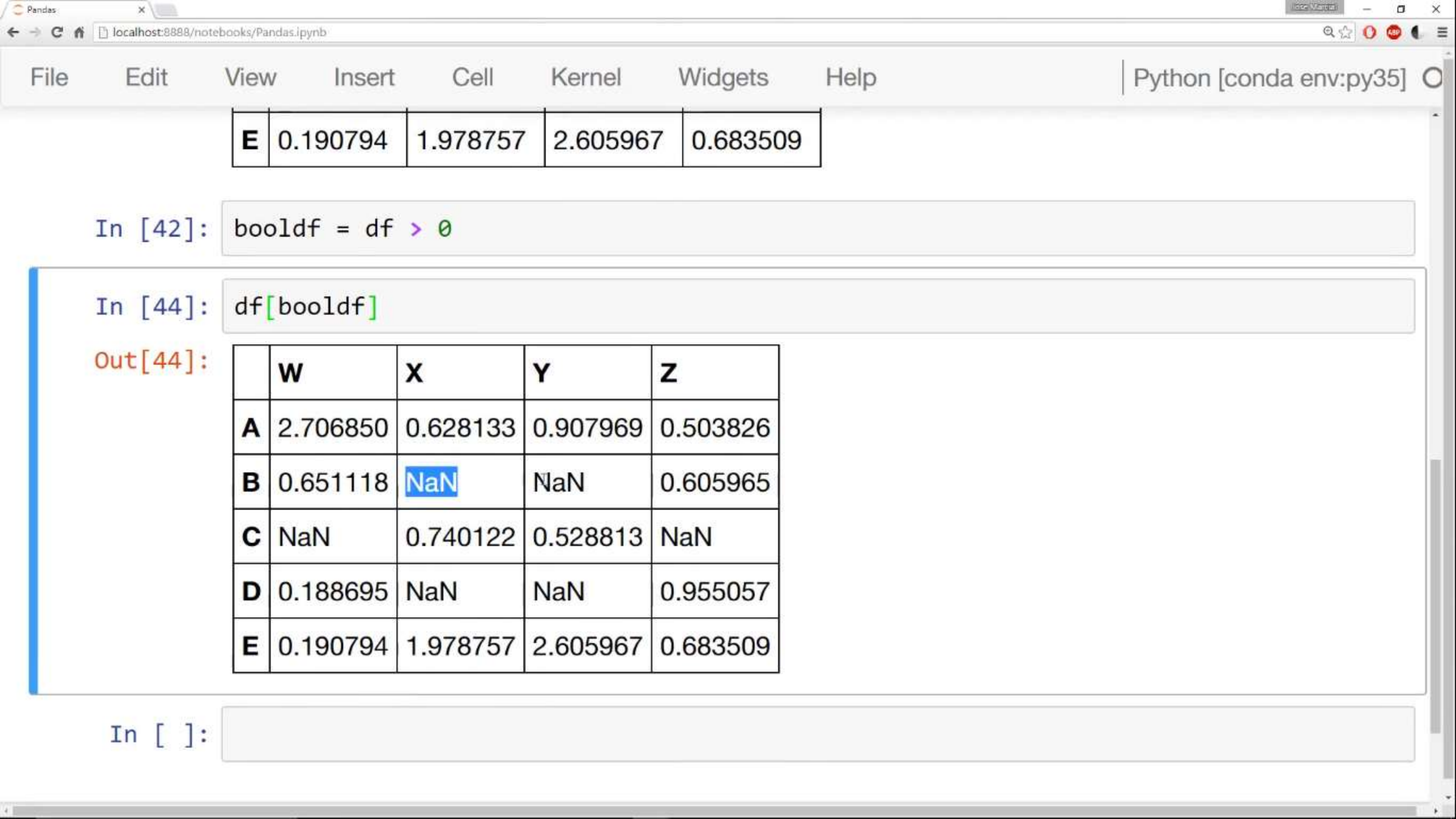
C	-2.018168	0.740122	0.528813	-0.589001
D	0.188695	-0.758872	-0.933237	0.955057
E	0.190794	1.978757	2.605967	0.683509

In [41]:
```
df > 0
```

Out[41]:

	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

In [ ]:



E	0.190794	1.978757	2.605967	0.683509

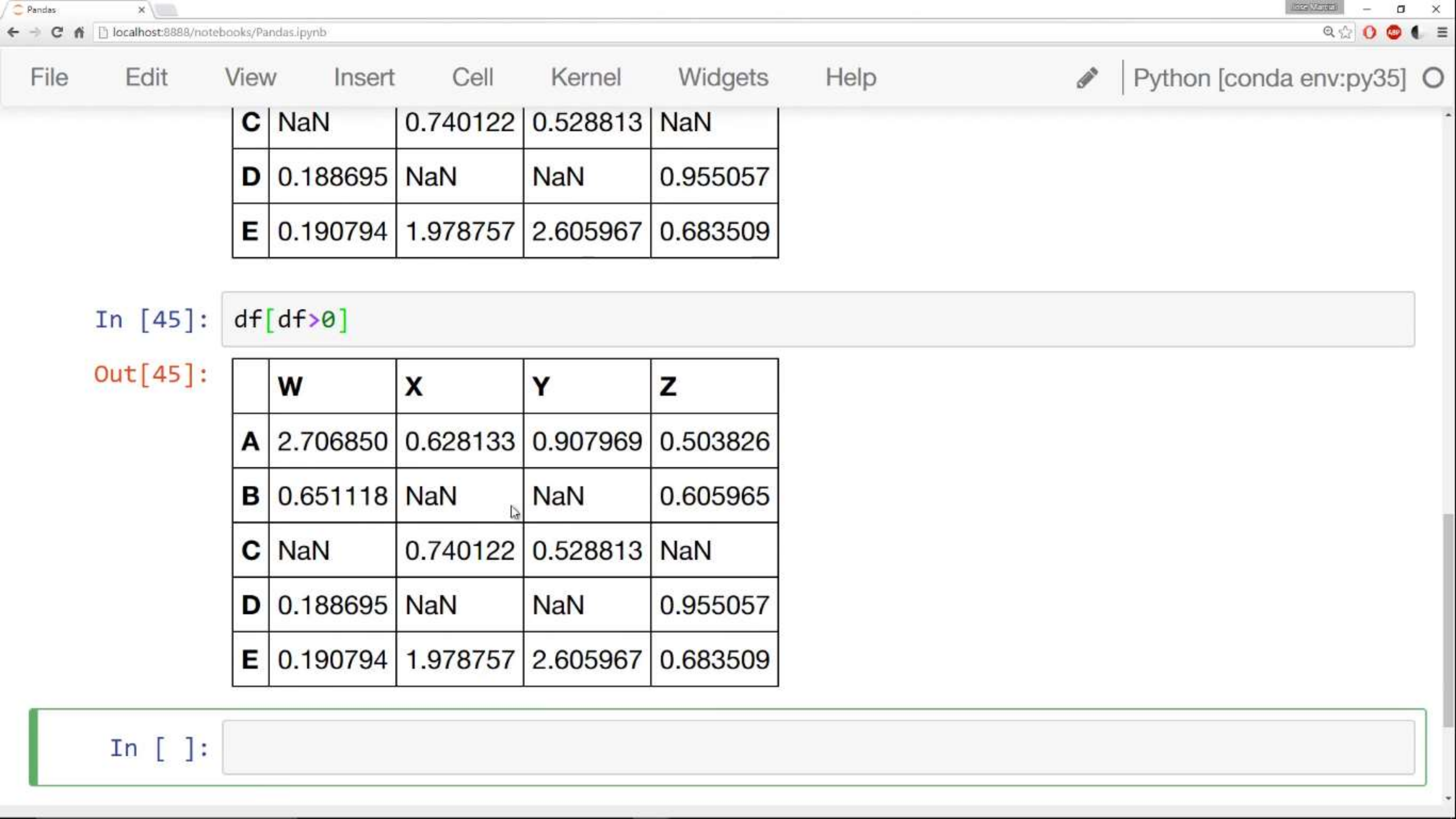
In [42]:
```
booldf = df > 0
```

In [44]:
```
df[booldf]
```

Out[44]:

	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 [ ]:



C	NaN	0.740122	0.528813	NaN
D	0.188695	NaN	NaN	0.955057
E	0.190794	1.978757	2.605967	0.683509

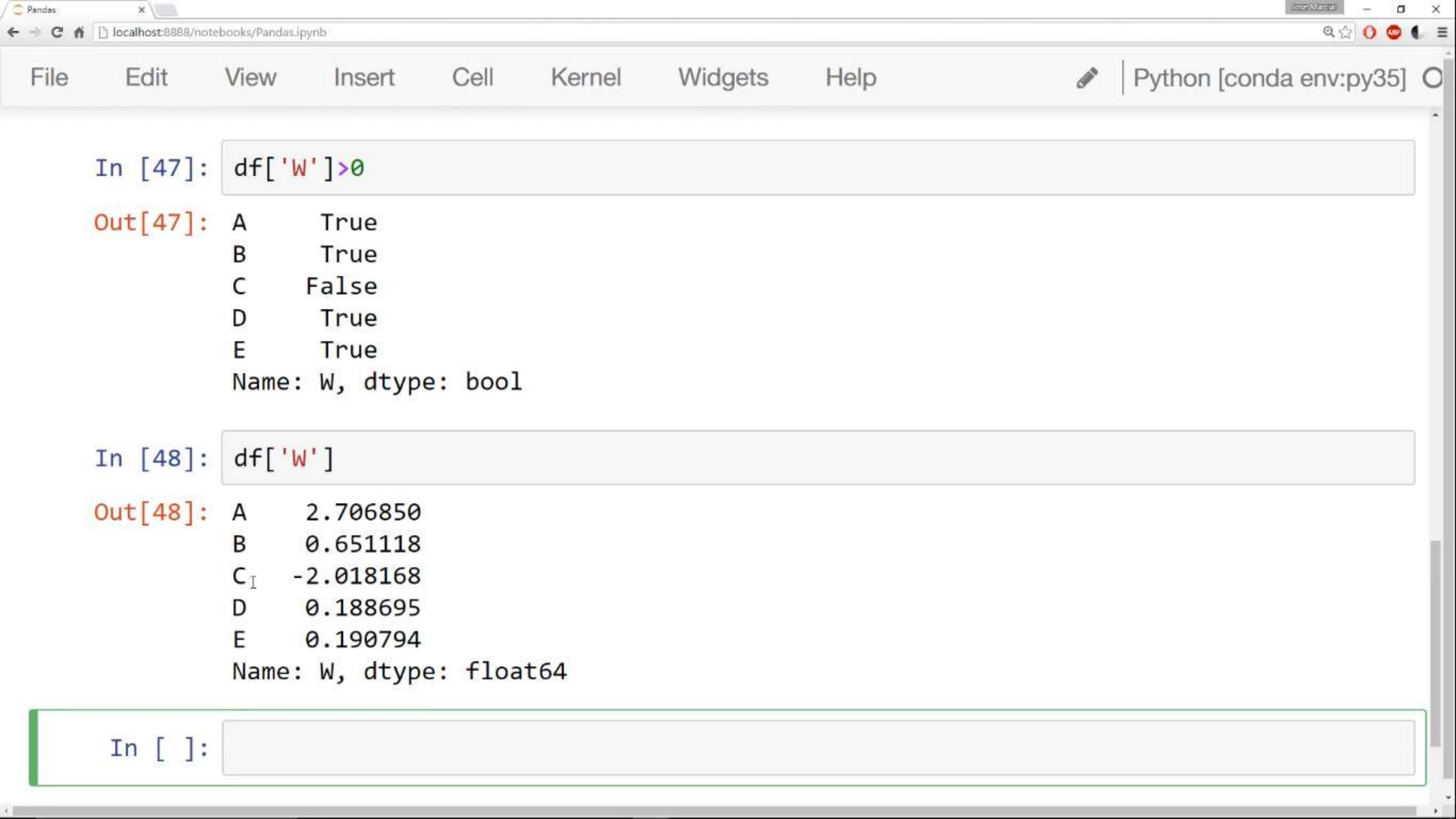
In [45]:

```
df[df>0]
```

Out[45]:

	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 [ ]:



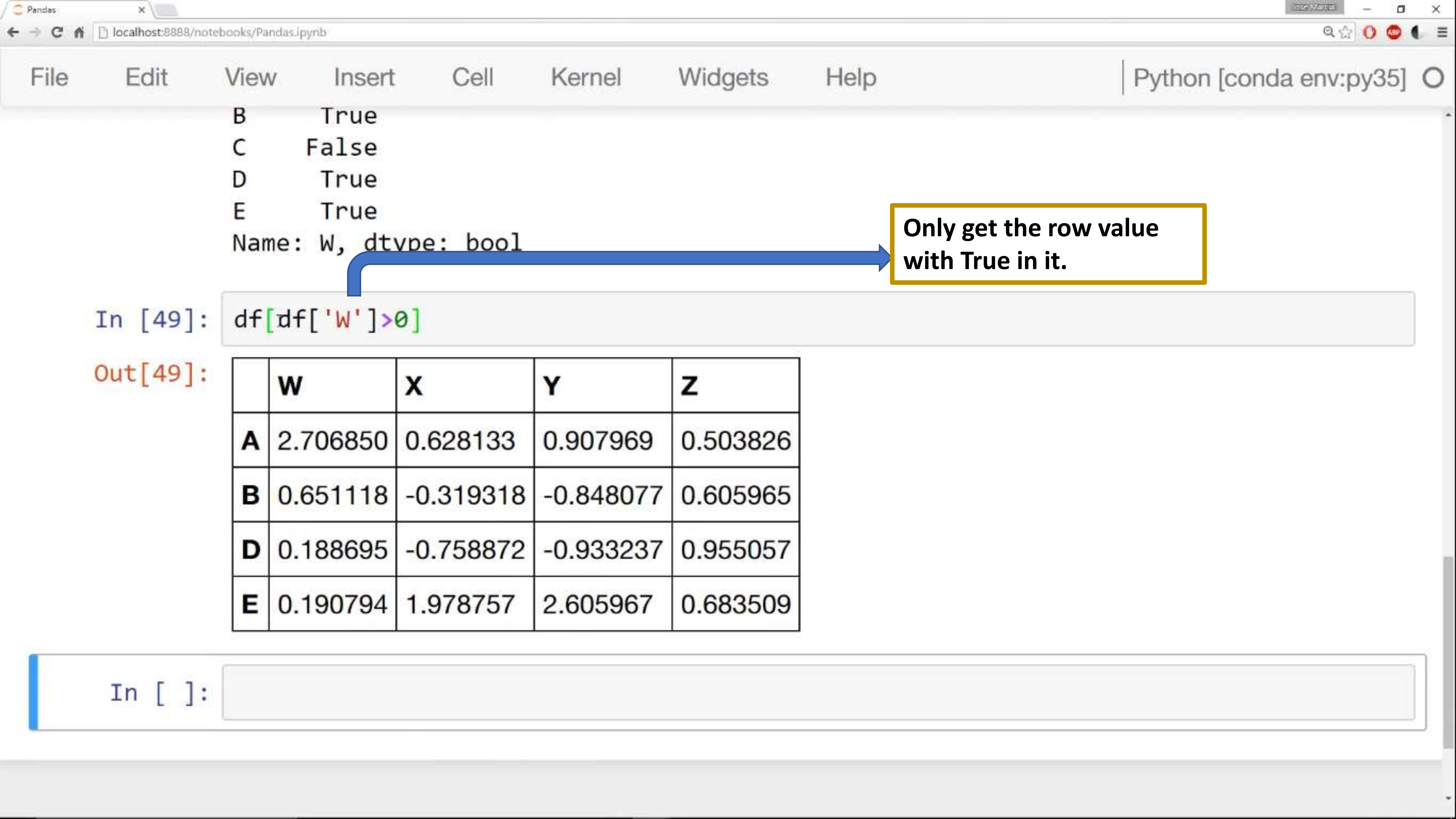
```
In [47]: df['W']>0

Out[47]: A     True
         B     True
         C    False
         D     True
         E     True
         Name: W, dtype: bool

In [48]: df['W']

Out[48]: A    2.706850
         B    0.651118
         C   -2.018168
         D    0.188695
         E    0.190794
         Name: W, dtype: float64

In [ ]:
```



```
B     True
C    False
D     True
E     True
Name: W, dtype: bool
```

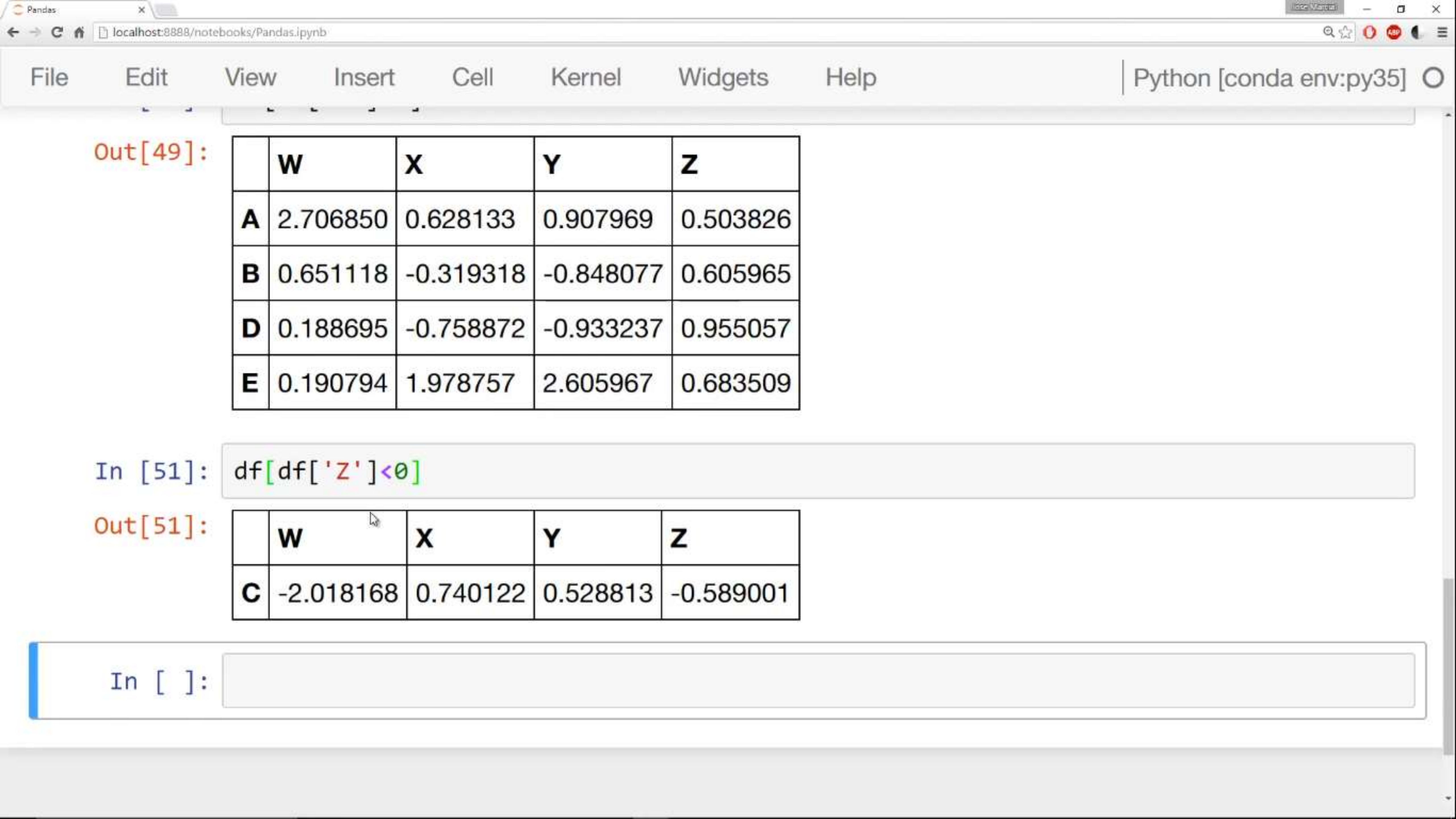
Only get the row value with True in it.

```
In [49]: df[df['W']>0]
```

Out[49]:

	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 [ ]:
```



Out[49]:

	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

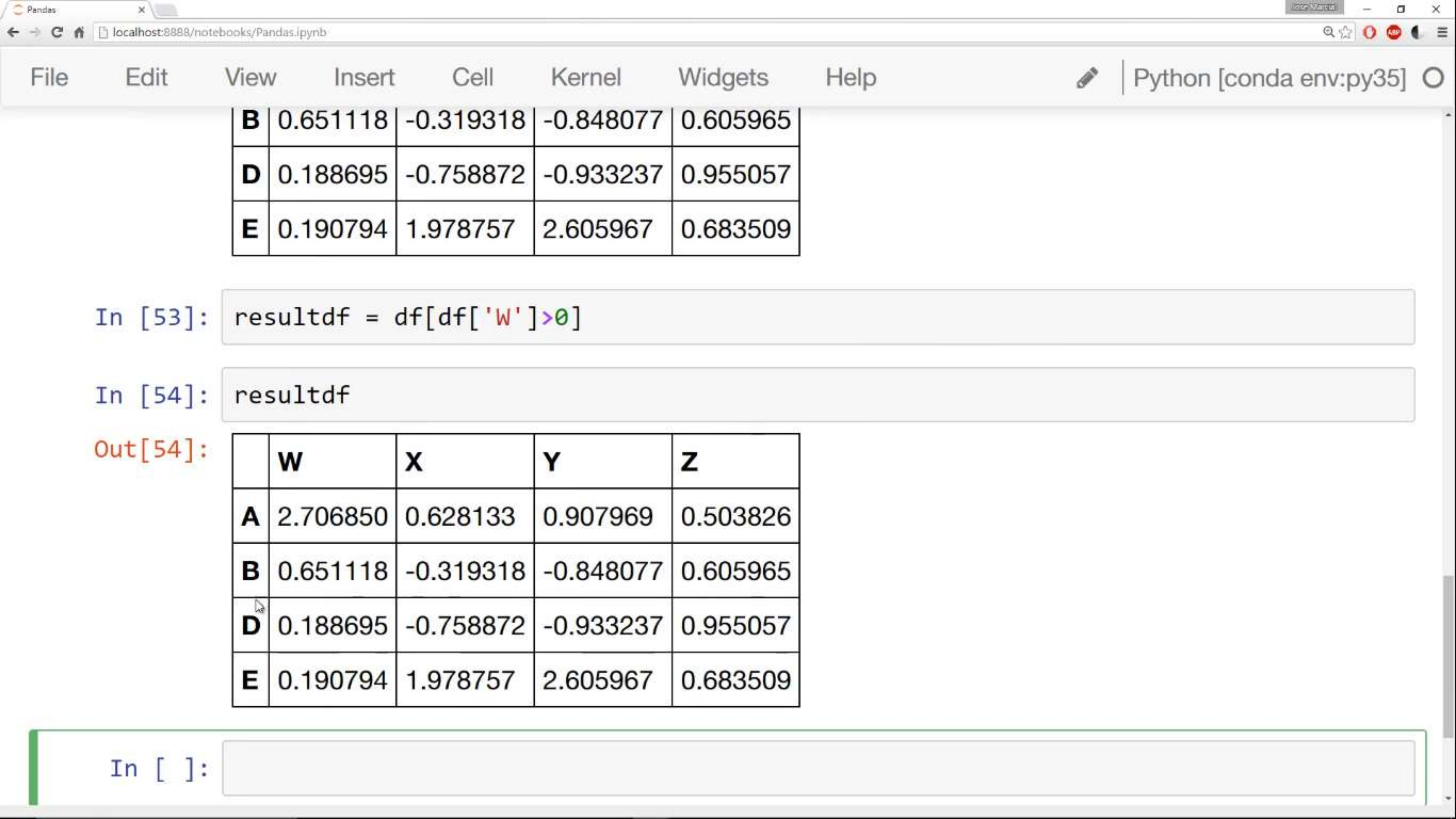
In [51]:

```
df[df['Z']<0]
```

Out[51]:

	W	X	Y	Z
C	-2.018168	0.740122	0.528813	-0.589001

In [ ]:



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

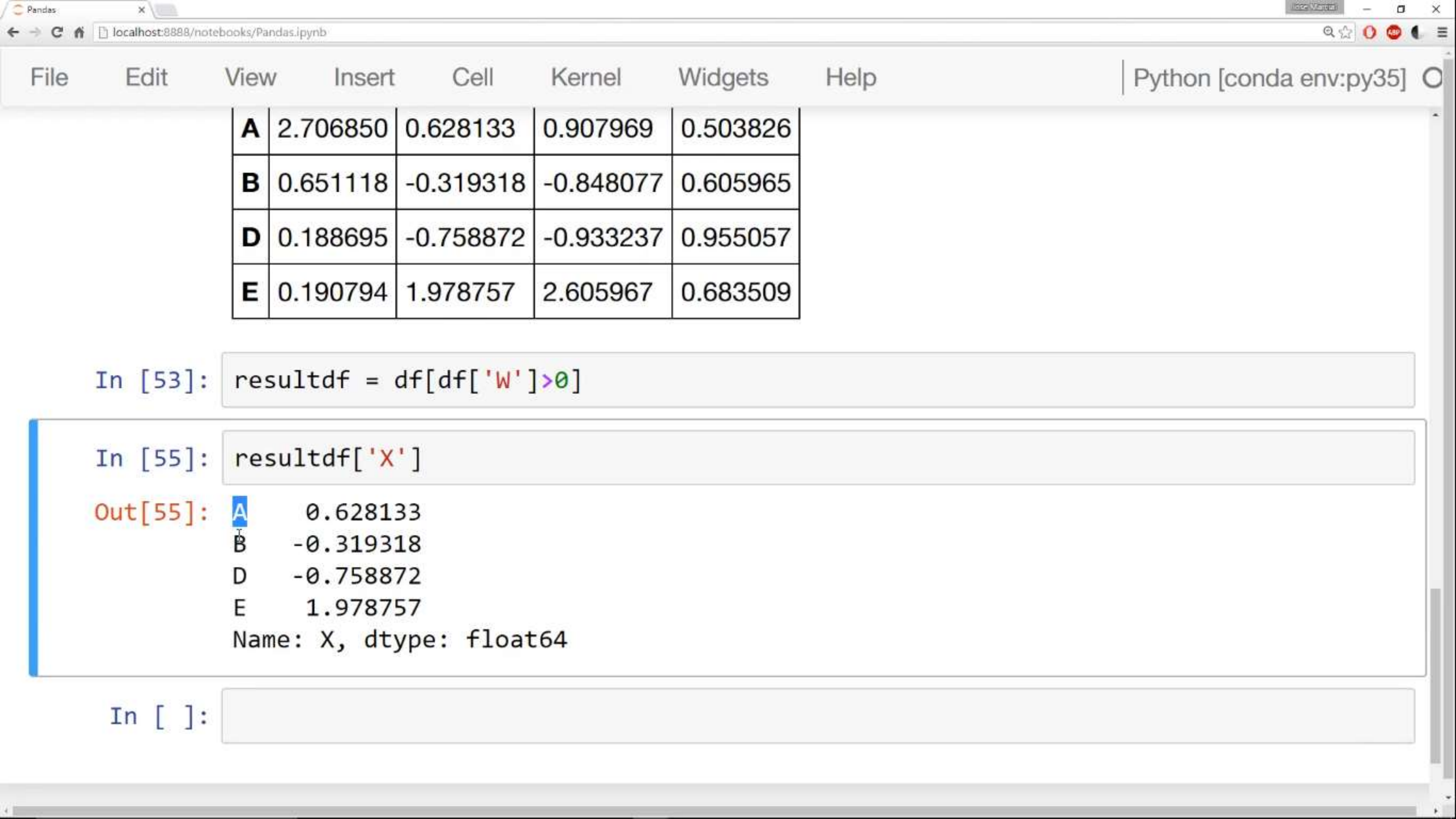
```
In [53]: resultdf = df[df['W']>0]
```

```
In [54]: resultdf
```

Out[54]:

	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 [ ]:
```



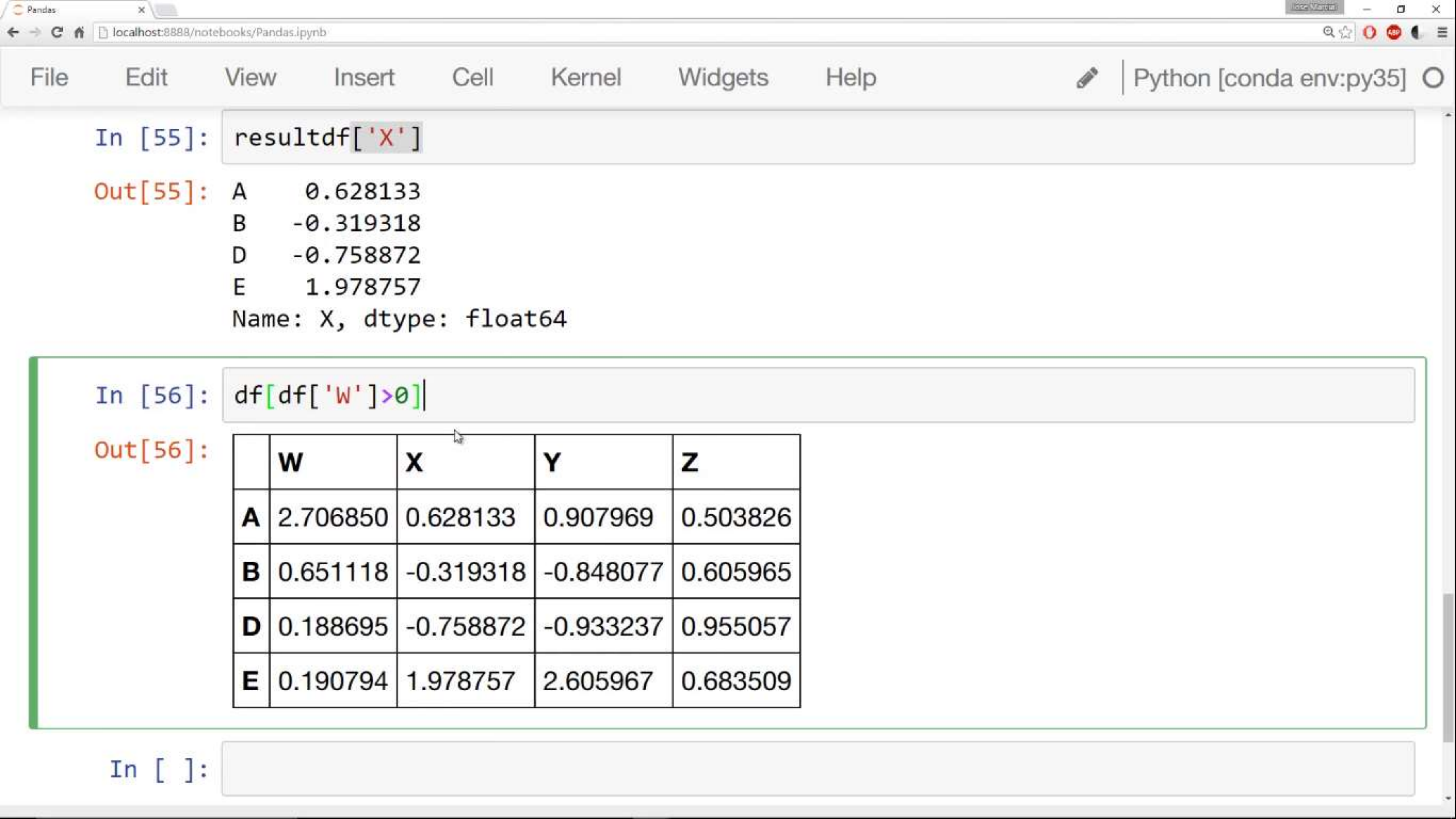
A	2.706850	0.628133	0.907969	0.503826
B	0.651118	-0.319318	-0.848077	0.605965
D	0.188695	-0.758872	-0.933237	0.955057
E	0.190794	1.978757	2.605967	0.683509

```
In [53]: resultdf = df[df['W']>0]
```

```
In [55]: resultdf['X']
```

```
Out[55]: A    0.628133
         B   -0.319318
         D   -0.758872
         E    1.978757
         Name: X, dtype: float64
```

```
In [ ]:
```



```
In [55]: resultdf['X']
```

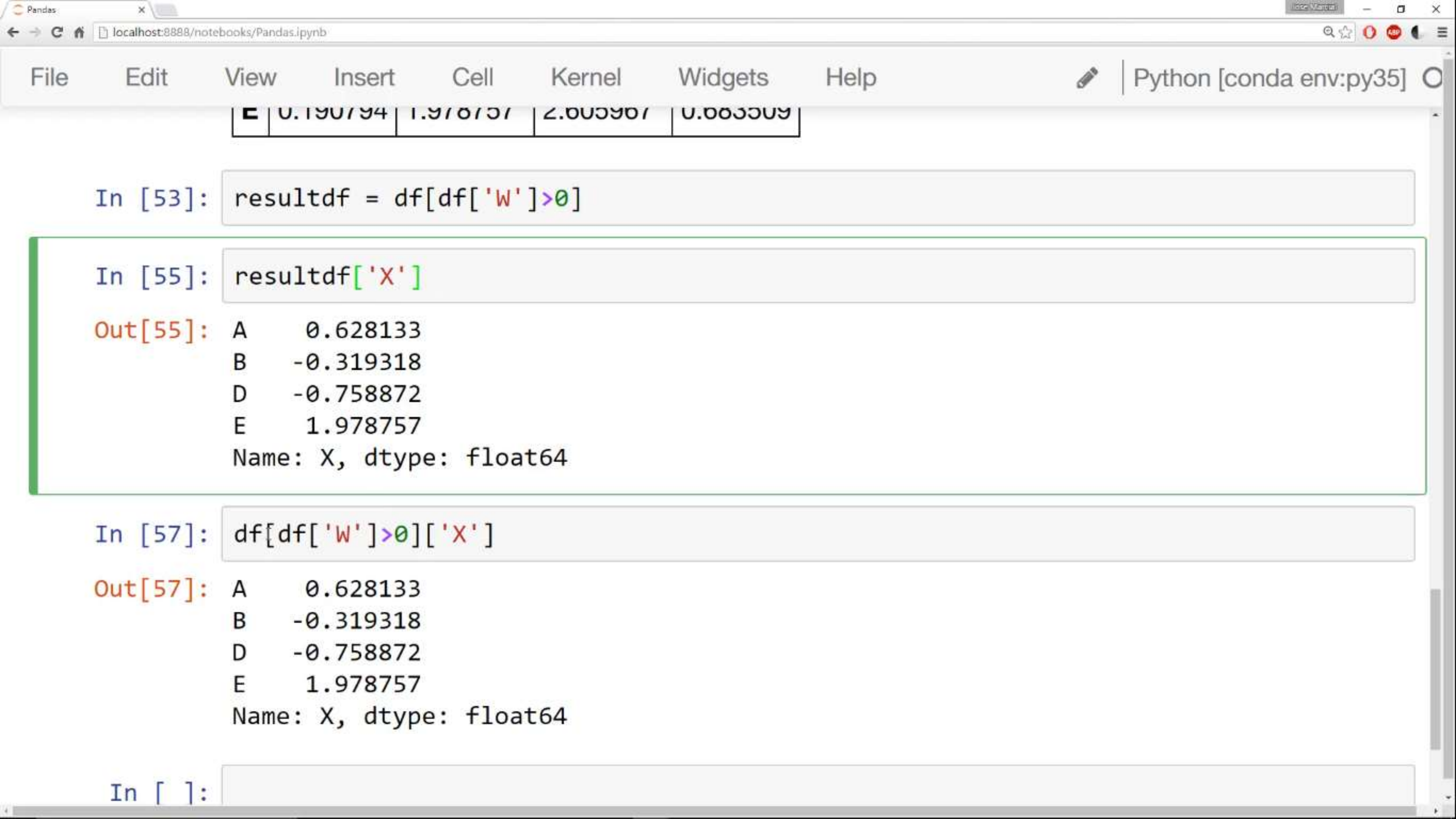
```
Out[55]: A    0.628133
         B   -0.319318
         D   -0.758872
         E    1.978757
         Name: X, dtype: float64
```

```
In [56]: df[df['W']>0]
```

Out[56]:

	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 [ ]:
```



E	0.190794	1.978757	2.605967	0.683509

In [53]:
```
resultdf = df[df['W']>0]
```

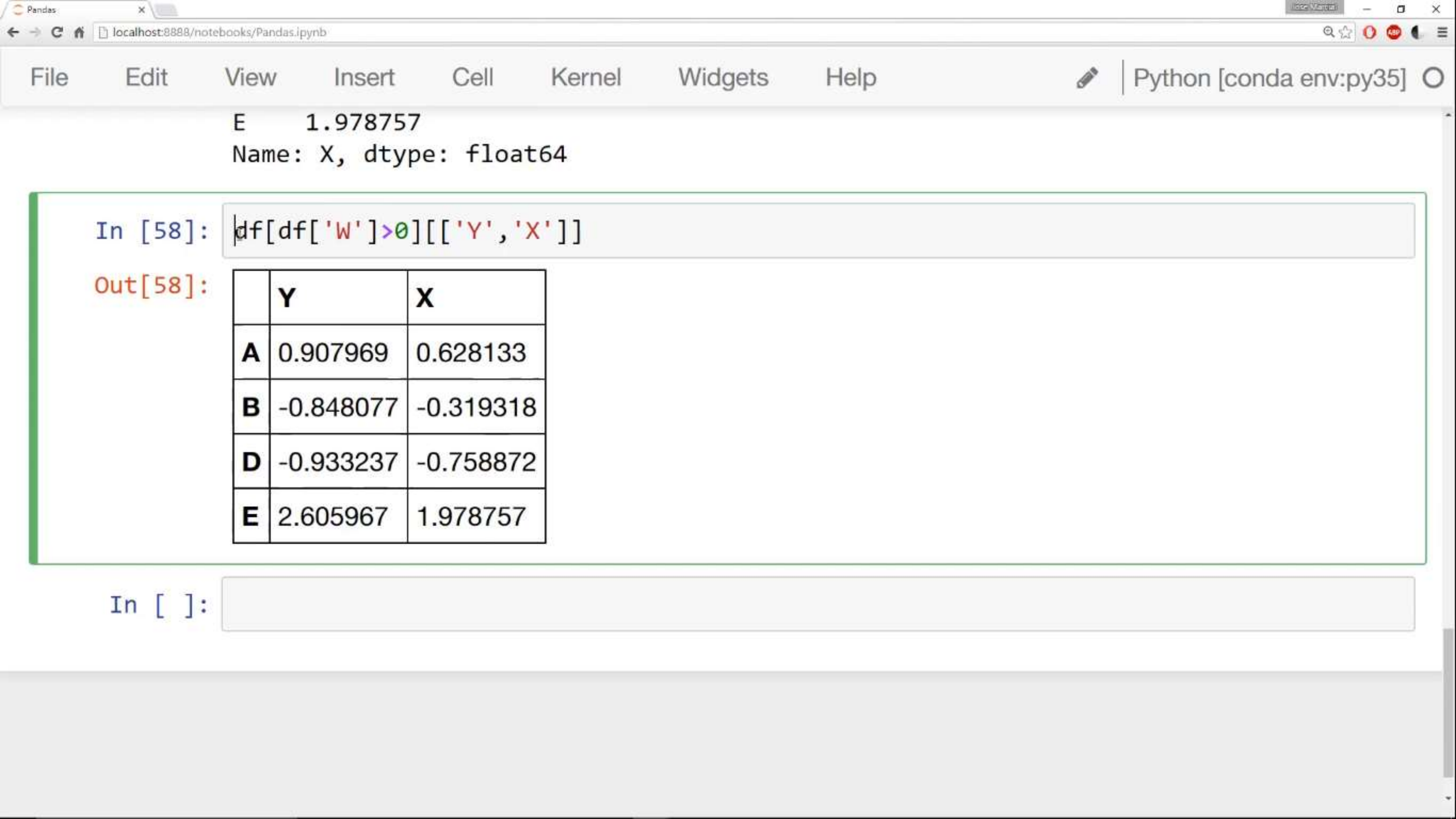
In [55]:
```
resultdf['X']
```

Out[55]:
```
A    0.628133
B   -0.319318
D   -0.758872
E    1.978757
Name: X, dtype: float64
```

In [57]:
```
df[df['W']>0]['X']
```

Out[57]:
```
A    0.628133
B   -0.319318
D   -0.758872
E    1.978757
Name: X, dtype: float64
```

In [ ]:



```
E    1.978757
Name: X, dtype: float64
```

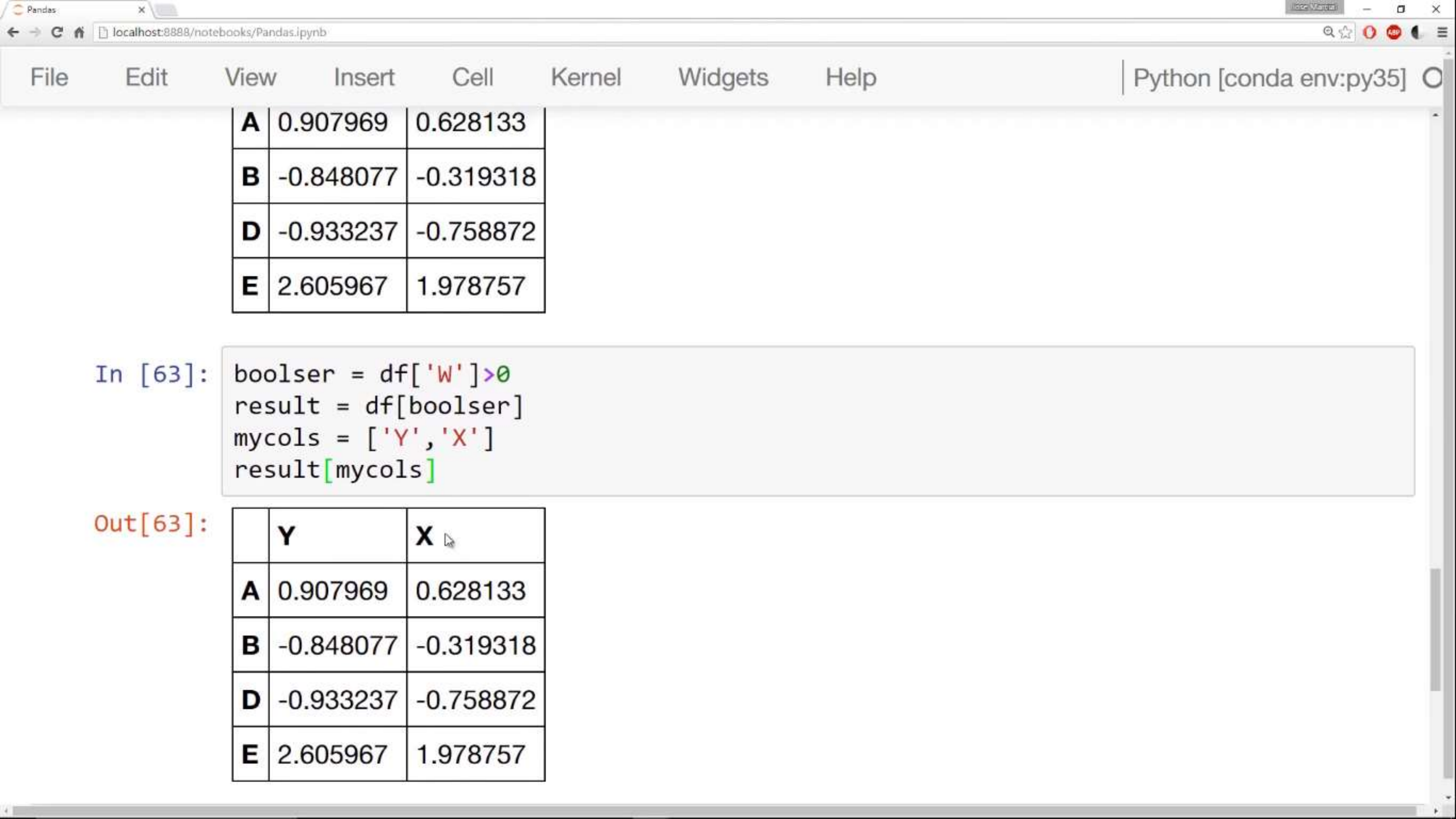
In [58]:

```
df[df['W']>0][['Y','X']]
```

Out[58]:

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

In [ ]:



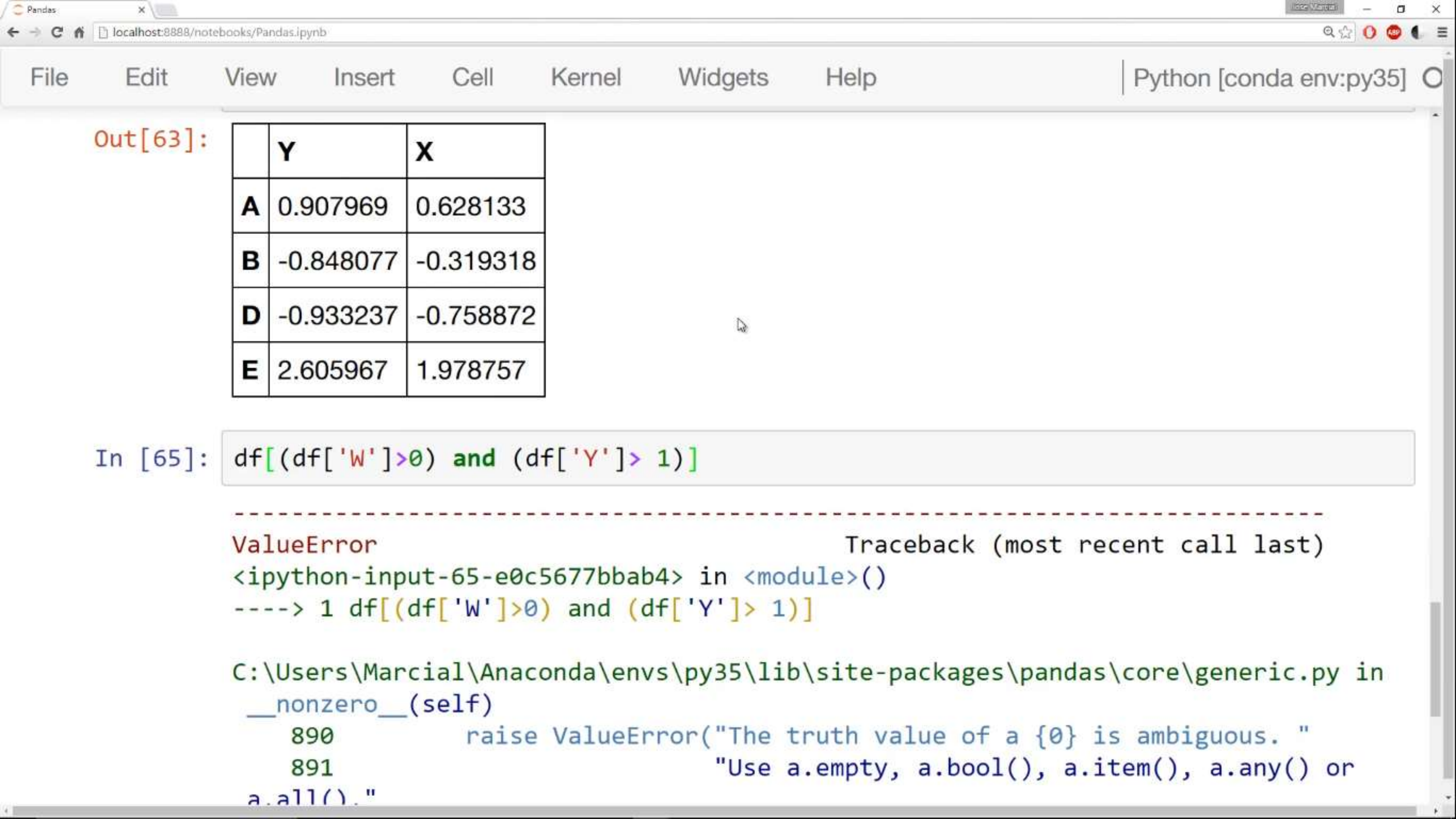
A	0.907969	0.628133
B	-0.848077	-0.319318
D	-0.933237	-0.758872
E	2.605967	1.978757

In [63]:

```
boolser = df['W']>0
result = df[boolser]
mycols = ['Y','X']
result[mycols]
```

Out[63]:

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



Out[63]:

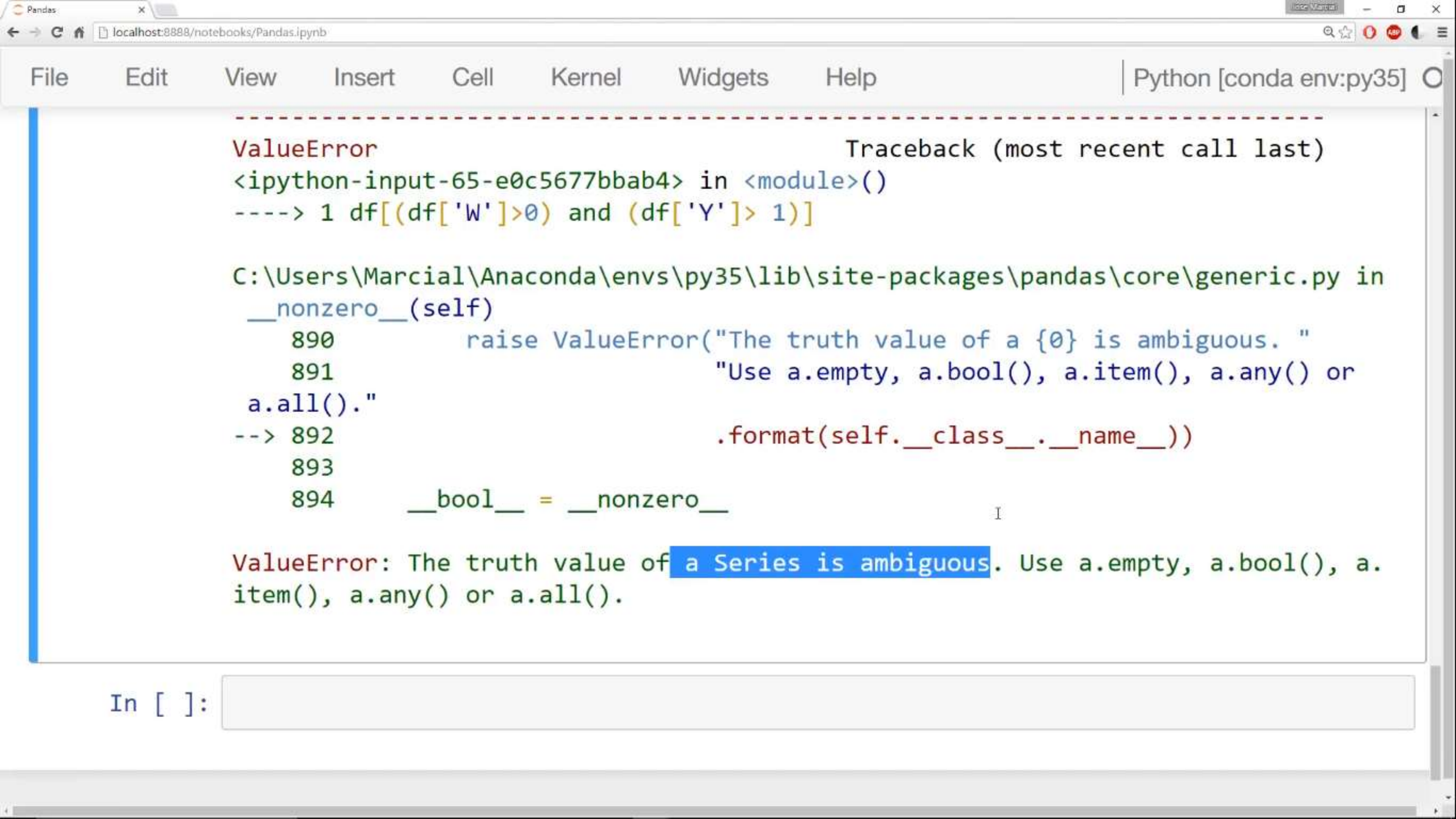
	Y	X
A	0.907969	0.628133
B	-0.848077	-0.319318
D	-0.933237	-0.758872
E	2.605967	1.978757

In [65]:

```
df[(df['W']>0) and (df['Y']> 1)]
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-65-e0c5677bbab4> in <module>()
----> 1 df[(df['W']>0) and (df['Y']> 1)]

C:\Users\Marcial\Anaconda\envs\py35\lib\site-packages\pandas\core\generic.py in
 __nonzero__(self)
    890         raise ValueError("The truth value of a {0} is ambiguous. "
    891                          "Use a.empty, a.bool(), a.item(), a.any() or
 a.all()."
```

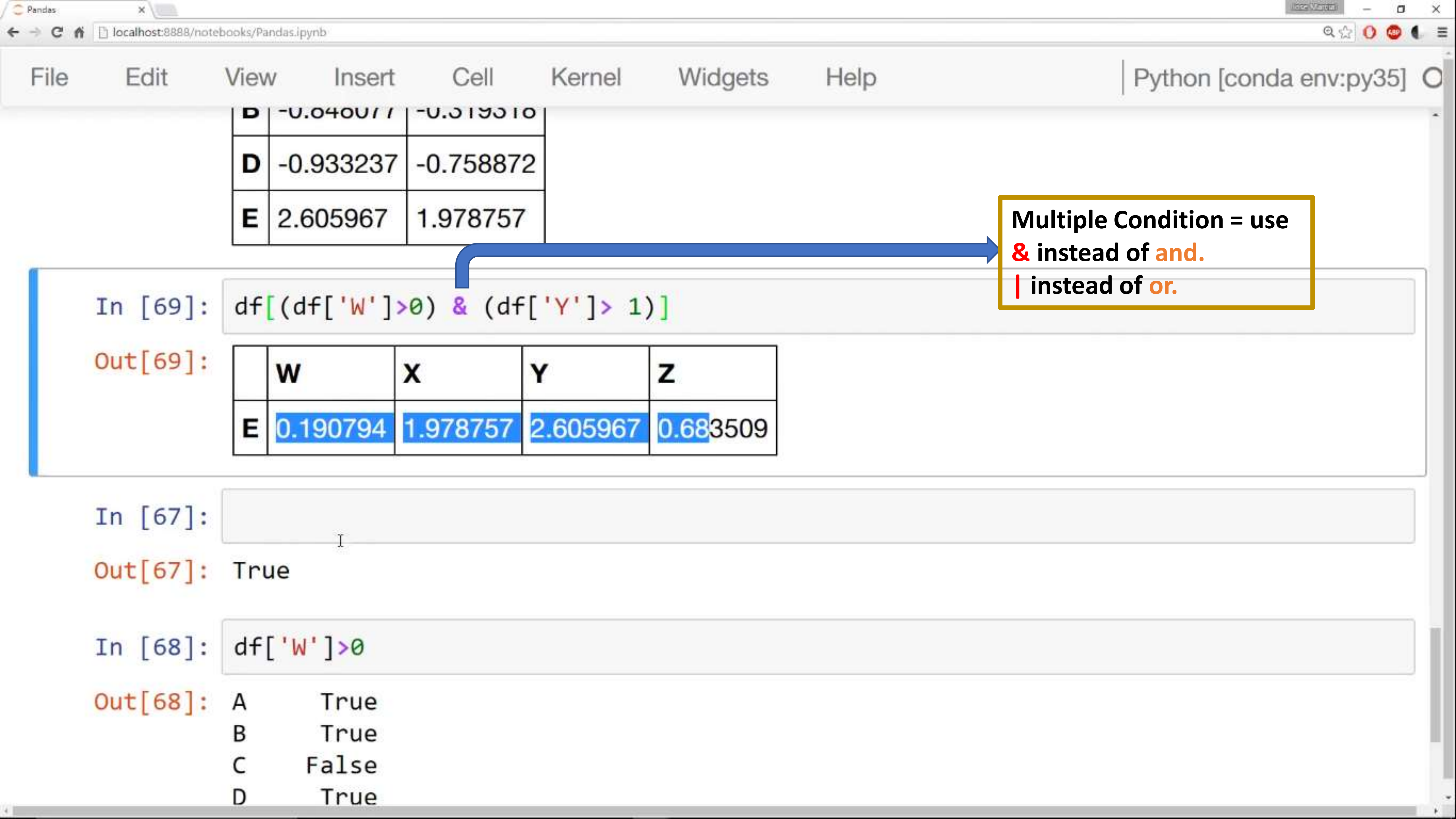


```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-65-e0c5677bbab4> in <module>()
----> 1 df[(df['W']>0) and (df['Y']> 1)]

C:\Users\Marcial\Anaconda\envs\py35\lib\site-packages\pandas\core\generic.py in
 __nonzero__(self)
    890         raise ValueError("The truth value of a {0} is ambiguous. "
    891                          "Use a.empty, a.bool(), a.item(), a.any() or
 a.all()."
--> 892                          .format(self.__class__.__name__))
    893
    894     __bool__ = __nonzero__

ValueError: The truth value of a Series is ambiguous. Use a.empty, a.bool(), a.
item(), a.any() or a.all().
```

```
In [ ]:
```



B	-0.848077	-0.319318
D	-0.933237	-0.758872
E	2.605967	1.978757

Multiple Condition = use
& instead of and.
| instead of or.

```
In [69]: df[(df['W']>0) & (df['Y']> 1)]
```

Out[69]:

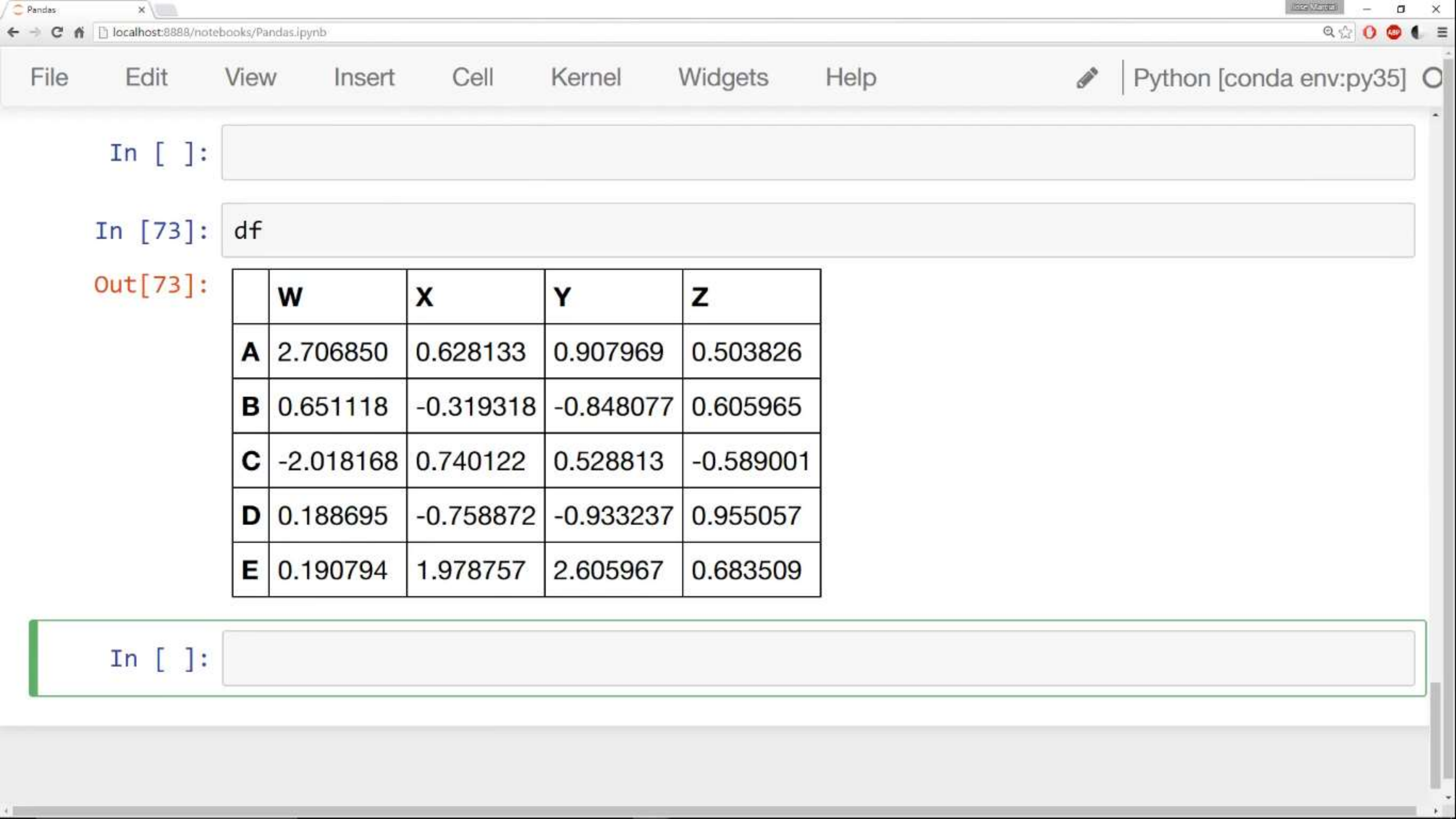
	W	X	Y	Z
E	0.190794	1.978757	2.605967	0.683509

```
In [67]:

Out[67]: True
```

```
In [68]: df['W']>0

Out[68]: A     True
         B     True
         C    False
         D     True
```



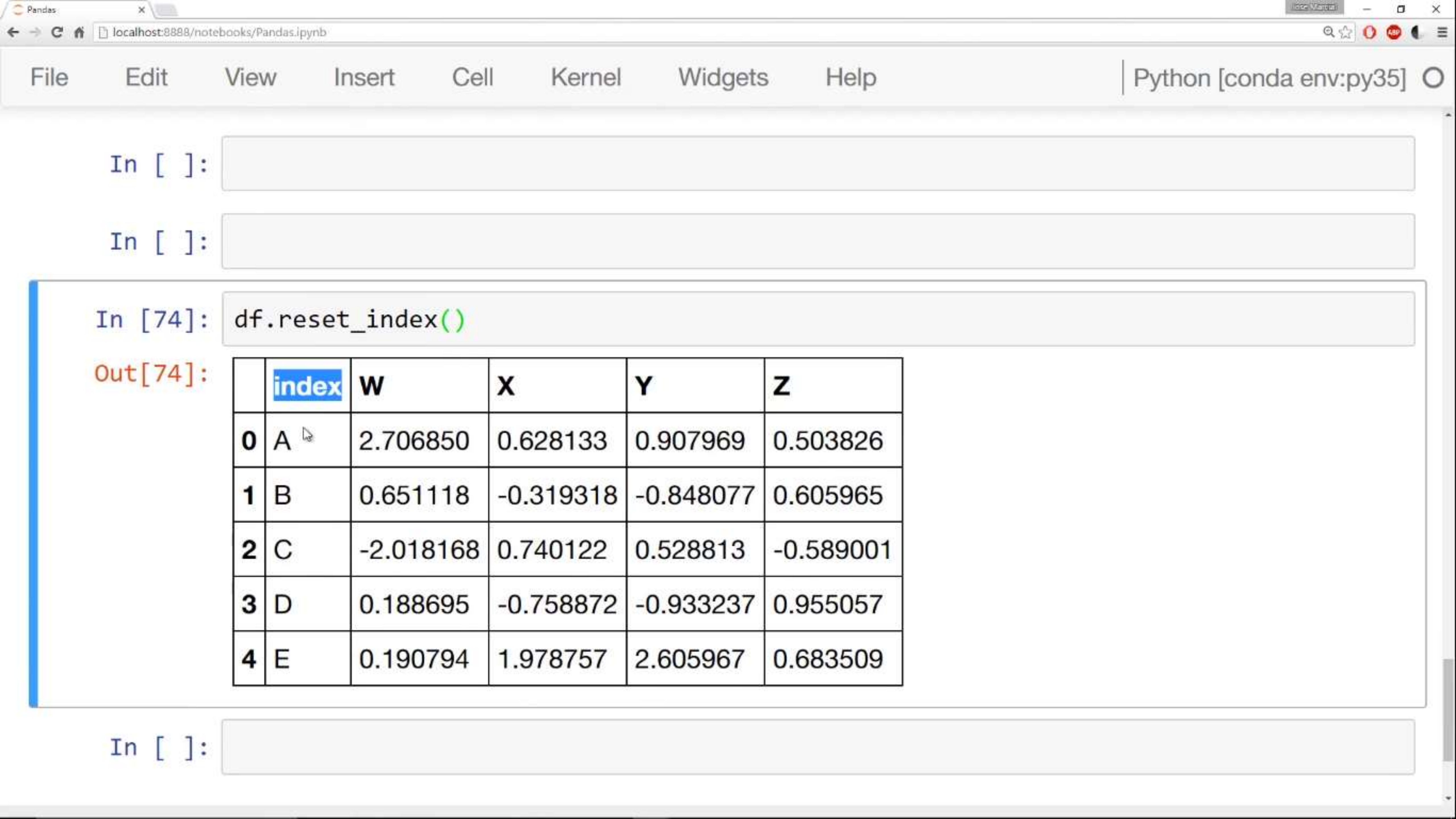
In [ ]:

In [73]:
```
df
```

Out[73]:

	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 [ ]:



```
In [ ]:
```

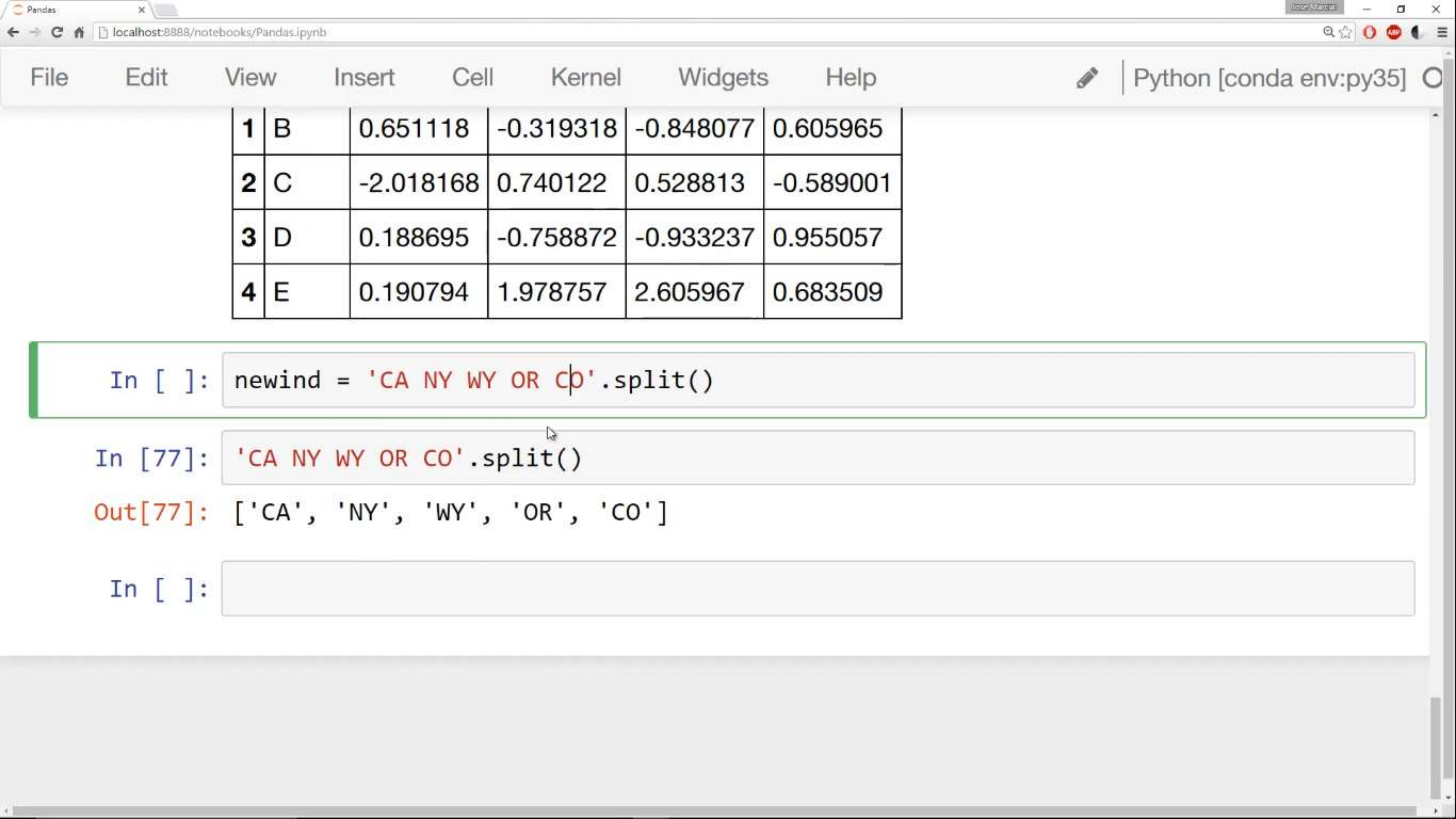
```
In [ ]:
```

```
In [74]: df.reset_index()
```

Out[74]:

	index	W	X	Y	Z
0	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

```
In [ ]:
```



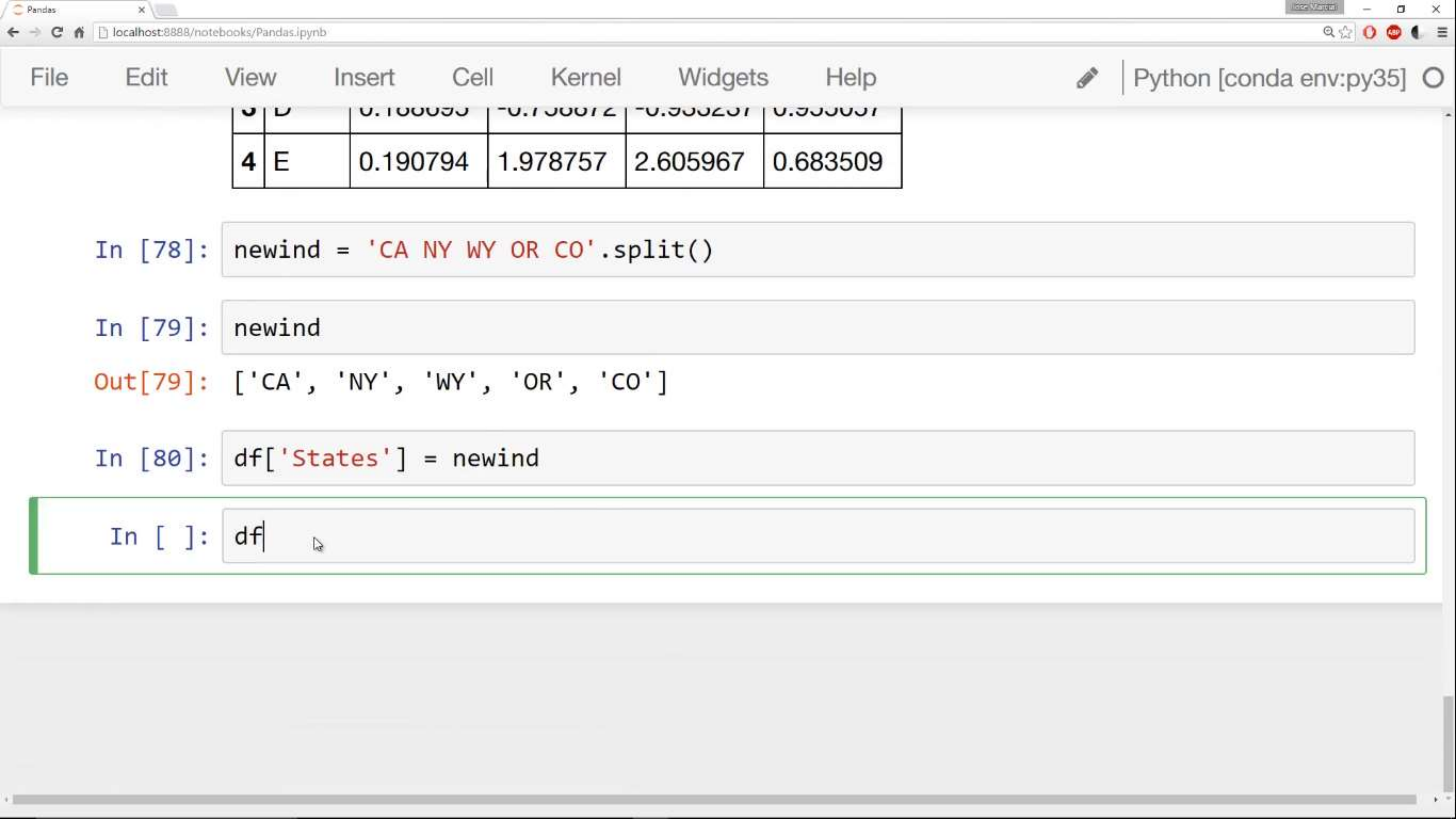
1	B	0.651118	-0.319318	-0.848077	0.605965
2	C	-2.018168	0.740122	0.528813	-0.589001
3	D	0.188695	-0.758872	-0.933237	0.955057
4	E	0.190794	1.978757	2.605967	0.683509

```
In [ ]: newind = 'CA NY WY OR CO'.split()
```

```
In [77]: 'CA NY WY OR CO'.split()

Out[77]: ['CA', 'NY', 'WY', 'OR', 'CO']
```

```
In [ ]:
```



3	D	0.188695	-0.758872	-0.933237	0.955057
4	E	0.190794	1.978757	2.605967	0.683509

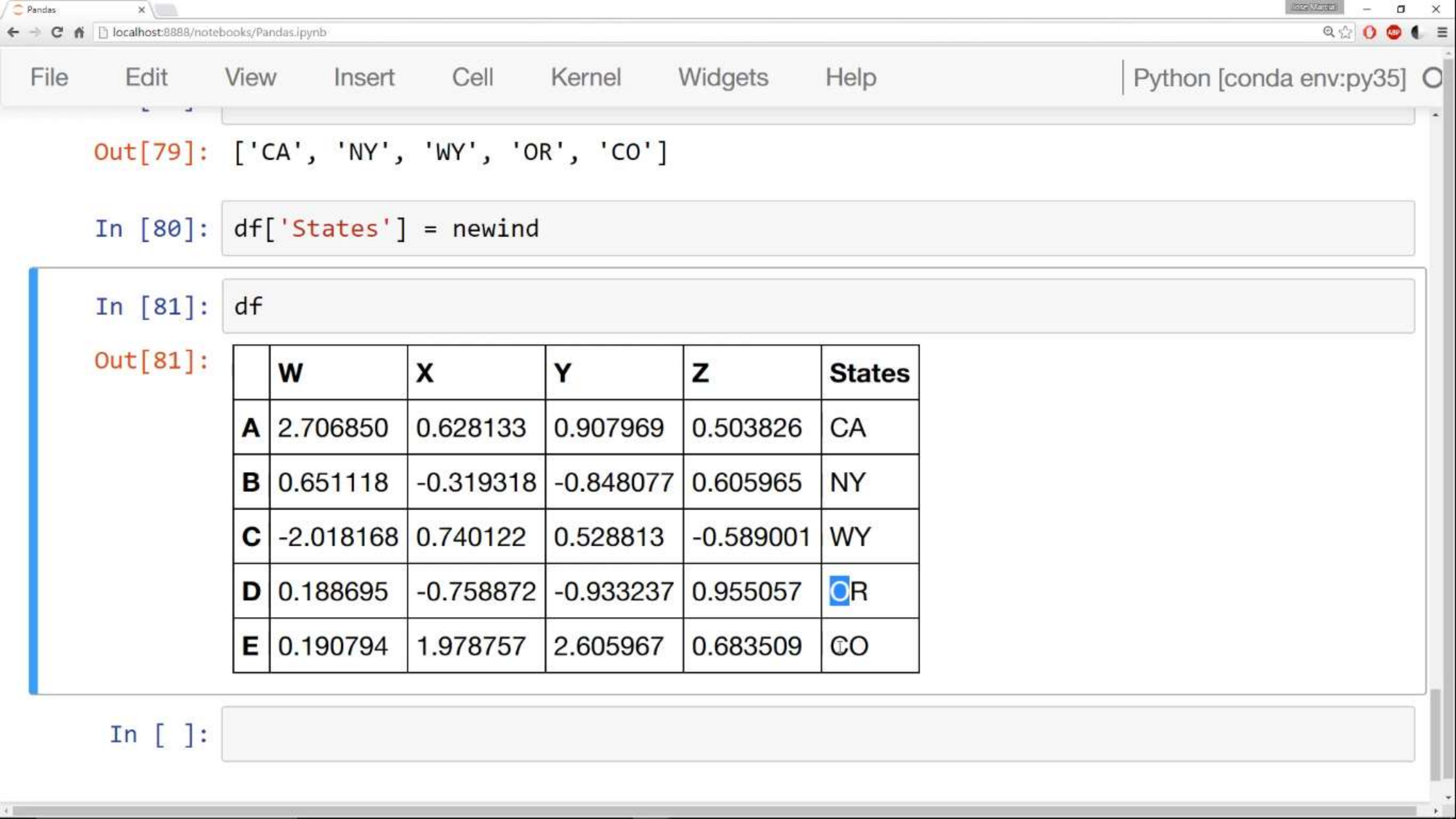
```
In [78]: newind = 'CA NY WY OR CO'.split()
```

```
In [79]: newind
```

```
Out[79]: ['CA', 'NY', 'WY', 'OR', 'CO']
```

```
In [80]: df['States'] = newind
```

```
In [ ]: df
```



```
Out[79]: ['CA', 'NY', 'WY', 'OR', 'CO']
```

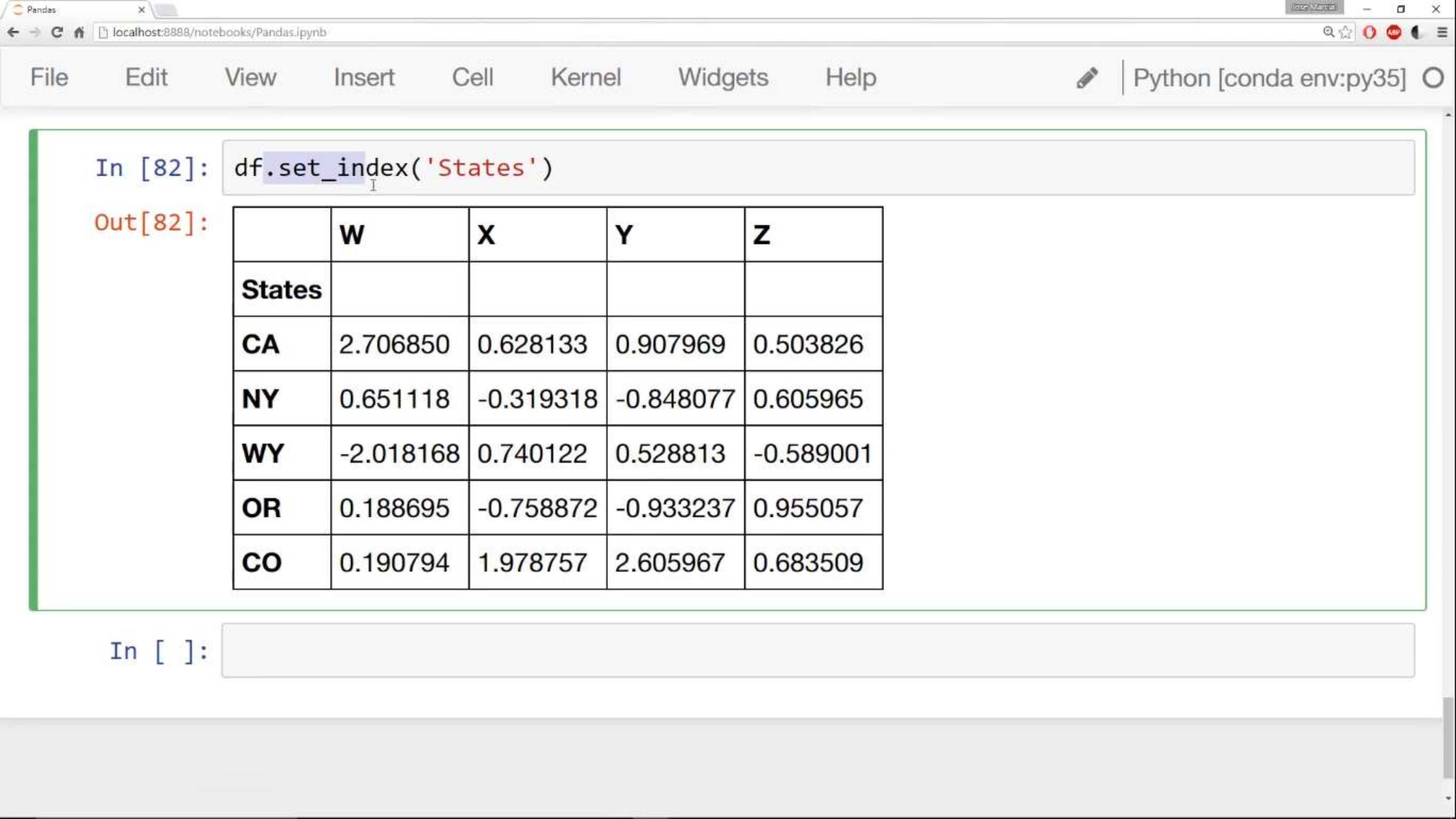
```
In [80]: df['States'] = newind
```

```
In [81]: df
```

Out[81]:

	W	X	Y	Z	States
A	2.706850	0.628133	0.907969	0.503826	CA
B	0.651118	-0.319318	-0.848077	0.605965	NY
C	-2.018168	0.740122	0.528813	-0.589001	WY
D	0.188695	-0.758872	-0.933237	0.955057	OR
E	0.190794	1.978757	2.605967	0.683509	CO

```
In [ ]:
```



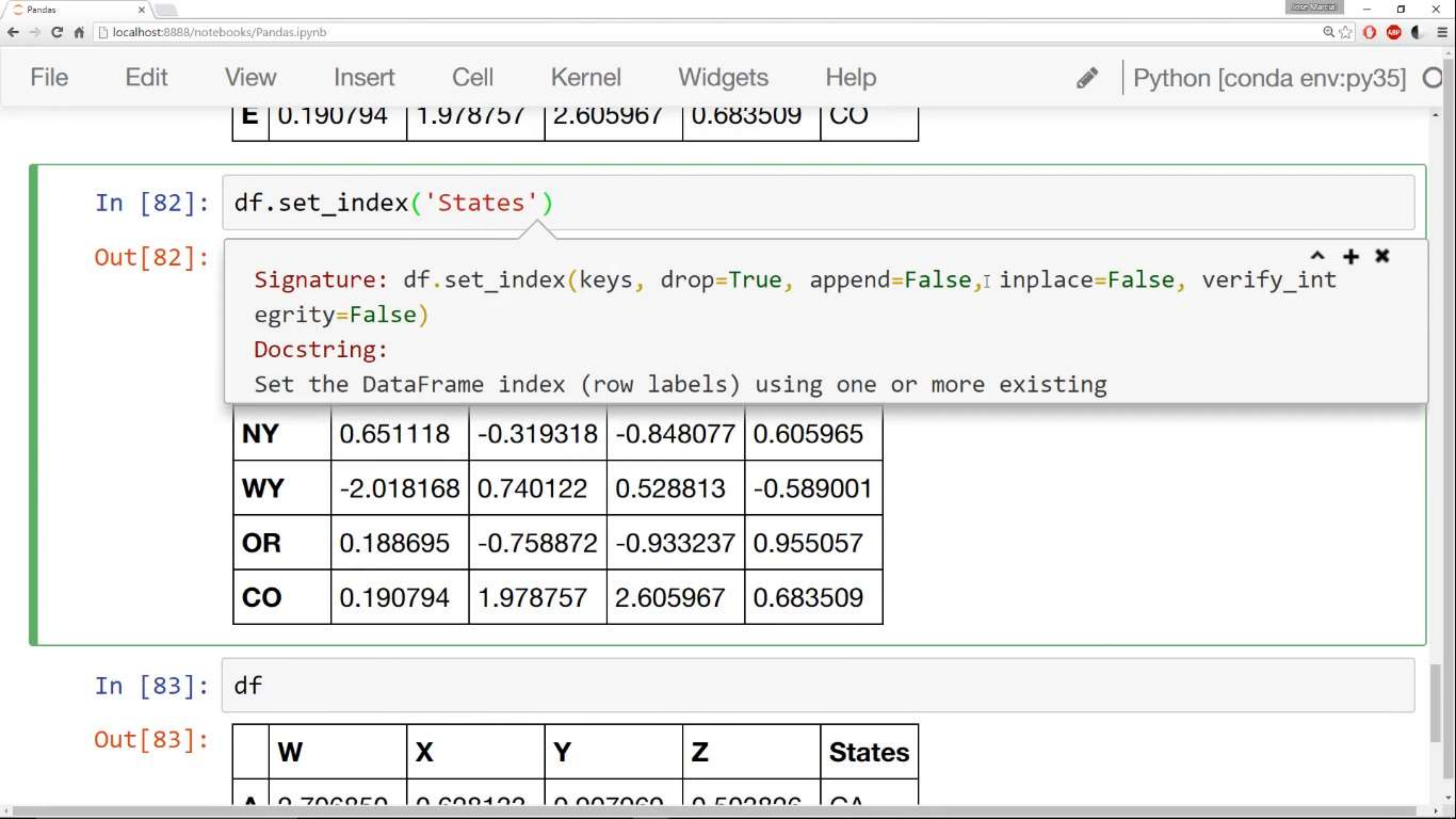
In [82]:

```
df.set_index('States')
```

Out[82]:

	W	X	Y	Z
States				
CA	2.706850	0.628133	0.907969	0.503826
NY	0.651118	-0.319318	-0.848077	0.605965
WY	-2.018168	0.740122	0.528813	-0.589001
OR	0.188695	-0.758872	-0.933237	0.955057
CO	0.190794	1.978757	2.605967	0.683509

In [ ]:



E	0.190794	1.978757	2.605967	0.683509	CO

In [82]:
```
df.set_index('States')
```

Out[82]:

```
Signature: df.set_index(keys, drop=True, append=False, inplace=False, verify_integrity=False)
Docstring:
Set the DataFrame index (row labels) using one or more existing
```

NY	0.651118	-0.319318	-0.848077	0.605965
WY	-2.018168	0.740122	0.528813	-0.589001
OR	0.188695	-0.758872	-0.933237	0.955057
CO	0.190794	1.978757	2.605967	0.683509

In [83]:
```
df
```

Out[83]:

	W	X	Y	Z	States
A	2.706850	0.628133	0.907969	0.503826	CA