

```
import pandas as pd
import numpy as np
df=pd.DataFrame(np.random.randn(5,3),index=['a','c','e','f','h'],columns=['one','two','three'])
df=df.reindex(['a','b','c','d','e','f','g','h'])
print(df['one'].isnull())
```

```
a    False
b     True
c    False
d     True
e    False
f    False
g     True
h    False
Name: one, dtype: bool
```

```
df=pd.DataFrame(np.random.randn(5,3),index=['a','c','e','f','h'],columns=['one','two','three'])
print(df)
df=df.reindex(['a','b','c','d','e','f','g','h'])
print(df)
```

```
➡
```

	one	two	three
a	1.865662	-0.469540	-0.092190
c	0.583589	0.452245	-0.821016
e	0.491013	0.664013	-1.306760
f	0.549234	-0.245976	-0.491084
h	0.593099	-0.345669	1.014453

	one	two	three
a	1.865662	-0.469540	-0.092190
b	NaN	NaN	NaN
c	0.583589	0.452245	-0.821016
d	NaN	NaN	NaN
e	0.491013	0.664013	-1.306760
f	0.549234	-0.245976	-0.491084
g	NaN	NaN	NaN
h	0.593099	-0.345669	1.014453

```
df=pd.DataFrame(np.random.randn(5,3),index=['a','c','e','f','h'],columns=['one','two','three'])
df=df.reindex(['a','b','c','d','e','f','g','h'])
print(df)
print("Nan replaced with '0: ")
print(df.fillna(0))
```

	one	two	three
a	-1.211633	1.402231	1.199110
b	NaN	NaN	NaN
c	0.882627	0.769118	-0.527793
d	NaN	NaN	NaN
e	-0.927164	0.001684	1.651462
f	1.191803	-0.793633	1.135883
g	NaN	NaN	NaN
h	-0.439390	1.254829	0.031908

Nan replaced with '0:

	one	two	three
a	-1.211633	1.402231	1.199110
b	0.000000	0.000000	0.000000
c	0.882627	0.769118	-0.527793
d	0.000000	0.000000	0.000000
e	-0.927164	0.001684	1.651462
f	1.191803	-0.793633	1.135883
g	0.000000	0.000000	0.000000
h	-0.439390	1.254829	0.031908

```
df=pd.DataFrame(np.random.randn(5,3),index=['a','c','e','f','h'],columns=['one','two','three'])
df=df.reindex(['a','b','c','d','e','f','g','h'])
print(df)
print('-----')
print(df.fillna(method='pad'))
```

	one	two	three
a	1.218140	1.448274	-1.147499
b	NaN	NaN	NaN
c	1.454349	-0.341040	-0.629953
d	NaN	NaN	NaN
e	-1.306844	0.902404	-1.750844
f	1.079966	1.178667	2.576581
g	NaN	NaN	NaN
h	-1.036838	0.190790	1.069643

	one	two	three
a	1.218140	1.448274	-1.147499
b	1.218140	1.448274	-1.147499
c	1.454349	-0.341040	-0.629953
d	1.454349	-0.341040	-0.629953

```
e -1.306844 0.902404 -1.750844
f 1.079966 1.178667 2.576581
g 1.079966 1.178667 2.576581
h -1.036838 0.190790 1.069643
```

```
df=pd.DataFrame(np.random.randn(5,3),index=['a','c','e','f','h'],columns=['one','two','three'])
df=df.reindex(['a','b','c','d','e','f','g','h'])
print(df.fillna(method='bfill'))
```

```
      one      two      three
a -0.367954  1.700383  1.663467
b -0.807544 -0.738556  1.158562
c -0.807544 -0.738556  1.158562
d -0.617482 -0.010213  0.367905
e -0.617482 -0.010213  0.367905
f -0.353869 -0.417613 -0.133084
g -0.544576 -1.586791  1.051521
h -0.544576 -1.586791  1.051521
```

```
df=pd.DataFrame(np.random.randn(5,3),index=['a','c','e','f','h'],columns=['one','two','three'])
df=df.reindex(['a','b','c','d','e','f','g','h'])
print(df)
print('-----')
print(df.dropna())
```

```
      one      two      three
a  0.079064  0.671803 -0.543138
b      NaN      NaN      NaN
c  0.151800  0.791420  0.639957
d      NaN      NaN      NaN
e -0.725952  1.401040 -0.312698
f -1.050159  0.862232 -0.739136
g      NaN      NaN      NaN
h -0.218216 -0.665878  0.909793
-----
      one      two      three
a  0.079064  0.671803 -0.543138
c  0.151800  0.791420  0.639957
e -0.725952  1.401040 -0.312698
f -1.050159  0.862232 -0.739136
h -0.218216 -0.665878  0.909793
```

```
df=pd.read_csv("/content/2,1 dataset titanic.csv")
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   PassengerId  891 non-null    int64
1   Survived     891 non-null    int64
2   Pclass       891 non-null    int64
3   Name         891 non-null    object
4   Sex          891 non-null    object
5   Age         714 non-null    float64
6   SibSp        891 non-null    int64
7   Parch        891 non-null    int64
8   Ticket       891 non-null    object
9   Fare         891 non-null    float64
10  Cabin        204 non-null    object
11  Embarked     889 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

```
cols=['Name','Ticket','Cabin']
df=df.drop(cols,axis=1)
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 9 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   PassengerId  891 non-null    int64
1   Survived     891 non-null    int64
2   Pclass       891 non-null    int64
3   Sex          891 non-null    object
4   Age         714 non-null    float64
5   SibSp        891 non-null    int64
6   Parch        891 non-null    int64
7   Fare         891 non-null    float64
8   Embarked     889 non-null    object
```

df.dropna()

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	1	0	3	male	22.0	1	0	7.2500	S
1	2	1	1	female	38.0	1	0	71.2833	C
2	3	1	3	female	26.0	0	0	7.9250	S
3	4	1	1	female	35.0	1	0	53.1000	S
4	5	0	3	male	35.0	0	0	8.0500	S
...
885	886	0	3	female	39.0	0	5	29.1250	Q
886	887	0	2	male	27.0	0	0	13.0000	S
887	888	1	1	female	19.0	0	0	30.0000	S
889	890	1	1	male	26.0	0	0	30.0000	C
890	891	0	3	male	32.0	0	0	7.7500	Q

712 rows × 9 columns

```
dummies=[]
cols=['Pclass','Sex','Embarked']
for col in cols:
    dummies.append(pd.get_dummies(df[col]))
print(df)
```

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	\
0	1	0	3	male	22.0	1	0	7.2500	
1	2	1	1	female	38.0	1	0	71.2833	
2	3	1	3	female	26.0	0	0	7.9250	
3	4	1	1	female	35.0	1	0	53.1000	
4	5	0	3	male	35.0	0	0	8.0500	
..	
886	887	0	2	male	27.0	0	0	13.0000	
887	888	1	1	female	19.0	0	0	30.0000	
888	889	0	3	female	NaN	1	2	23.4500	
889	890	1	1	male	26.0	0	0	30.0000	
890	891	0	3	male	32.0	0	0	7.7500	

Embarked	
0	S
1	C
2	S
3	S
4	S
..	...
886	S
887	S
888	S
889	C
890	Q

[891 rows x 9 columns]

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	\
0	1	0	3	male	22.0	1	0	7.2500	
1	2	1	1	female	38.0	1	0	71.2833	
2	3	1	3	female	26.0	0	0	7.9250	
3	4	1	1	female	35.0	1	0	53.1000	
4	5	0	3	male	35.0	0	0	8.0500	
..	
886	887	0	2	male	27.0	0	0	13.0000	
887	888	1	1	female	19.0	0	0	30.0000	
888	889	0	3	female	NaN	1	2	23.4500	
889	890	1	1	male	26.0	0	0	30.0000	
890	891	0	3	male	32.0	0	0	7.7500	

Embarked	
0	S
1	C
2	S
3	S
4	S
..	...
886	S
887	S
888	S
889	C
890	Q

[891 rows x 9 columns]

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	\
0	1	0	3	male	22.0	1	0	7.2500	
1	2	1	1	female	38.0	1	0	71.2833	
2	3	1	3	female	26.0	0	0	7.9250	

```
titanic_dummies=pd.concat(dummies,axis=1)
print(df)
```

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	\
0	1	0	3	male	22.0	1	0	7.2500	
1	2	1	1	female	38.0	1	0	71.2833	
2	3	1	3	female	26.0	0	0	7.9250	
3	4	1	1	female	35.0	1	0	53.1000	
4	5	0	3	male	35.0	0	0	8.0500	
..	
886	887	0	2	male	27.0	0	0	13.0000	
887	888	1	1	female	19.0	0	0	30.0000	
888	889	0	3	female	NaN	1	2	23.4500	
889	890	1	1	male	26.0	0	0	30.0000	
890	891	0	3	male	32.0	0	0	7.7500	

Embarked

0	S
1	C
2	S
3	S
4	S
..	...
886	S
887	S
888	S
889	C
890	Q

[891 rows x 9 columns]

```
df=pd.concat((df,titanic_dummies),axis=1)
print(df)
```

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	\
0	1	0	3	male	22.0	1	0	7.2500	
1	2	1	1	female	38.0	1	0	71.2833	
2	3	1	3	female	26.0	0	0	7.9250	
3	4	1	1	female	35.0	1	0	53.1000	
4	5	0	3	male	35.0	0	0	8.0500	
..	
886	887	0	2	male	27.0	0	0	13.0000	
887	888	1	1	female	19.0	0	0	30.0000	
888	889	0	3	female	NaN	1	2	23.4500	
889	890	1	1	male	26.0	0	0	30.0000	
890	891	0	3	male	32.0	0	0	7.7500	

Embarked

	Embarked	1	2	3	female	male	C	Q	S
0	S	0	0	1	0	1	0	0	1
1	C	1	0	0	1	0	1	0	0
2	S	0	0	1	1	0	0	0	1
3	S	1	0	0	1	0	0	0	1
4	S	0	0	1	0	1	0	0	1
..
886	S	0	1	0	0	1	0	0	1
887	S	1	0	0	1	0	0	0	1
888	S	0	0	1	1	0	0	0	1
889	C	1	0	0	0	1	1	0	0
890	Q	0	0	1	0	1	0	1	0

[891 rows x 17 columns]

```
df=df.drop(['Pclass','Sex','Embarked'],axis=1)
print(df)
```

	PassengerId	Survived	Age	SibSp	Parch	Fare	1	2	3	female	\
0	1	0	22.0	1	0	7.2500	0	0	1	0	
1	2	1	38.0	1	0	71.2833	1	0	0	1	
2	3	1	26.0	0	0	7.9250	0	0	1	1	
3	4	1	35.0	1	0	53.1000	1	0	0	1	
4	5	0	35.0	0	0	8.0500	0	0	1	0	
..	
886	887	0	27.0	0	0	13.0000	0	1	0	0	
887	888	1	19.0	0	0	30.0000	1	0	0	1	
888	889	0	NaN	1	2	23.4500	0	0	1	1	
889	890	1	26.0	0	0	30.0000	1	0	0	0	
890	891	0	32.0	0	0	7.7500	0	0	1	0	

	male	C	Q	S
0	1	0	0	1
1	0	1	0	0

```

2      0  0  0  1
3      0  0  0  1
4      1  0  0  1
..     ... .. ..
886    1  0  0  1
887    0  0  0  1
888    0  0  0  1
889    1  1  0  0
890    1  0  1  0

```

[891 rows x 14 columns]

```
df['Age']=df['Age']
```

```

from sklearn.preprocessing import MinMaxScaler
from numpy import asarray
data=[[-1,2],[-0.5,6],[0,10],[1,18]]
scaler=MinMaxScaler()
print(scaler.fit(data))
print('-----')
MinMaxScaler()
print(scaler.data_max_)
print('-----')
print(scaler.transform(data))

```

```

MinMaxScaler()
-----
[ 1. 18.]
-----
[[0.  0. ]
 [0.25 0.25]
 [0.5  0.5 ]
 [1.   1.  ]]

```

```

from sklearn.preprocessing import StandardScaler
data=asarray([[100,0.001],[8,0.05],[50,0.005],[88,0.07],[4,0.1]])
print(data)
scaler=StandardScaler()
scaled=scaler.fit_transform(data)
print(scaled)

```

```

[[1.0e+02 1.0e-03]
 [8.0e+00 5.0e-02]
 [5.0e+01 5.0e-03]
 [8.8e+01 7.0e-02]
 [4.0e+00 1.0e-01]]
[[ 1.26398112 -1.16389967]
 [-1.06174414  0.12639634]
 [ 0.         -1.05856939]
 [ 0.96062565  0.65304778]
 [-1.16286263  1.44302493]]

```