pandas.DataFrame.drop

`DataFrame.`**`drop`**(*labels=None*, *axis=0*, *index=None*, *columns=None*, *level=None*, *inplace=False*, *errors='raise'*) [source]

Drop specified labels from rows or columns.

Remove rows or columns by specifying label names and corresponding axis, or by specifying directly index or column names. When using a multi-index, labels on different levels can be removed by specifying the level.

Parameters:

labels : *single label or list-like*
Index or column labels to drop.

axis : *{0 or 'index', 1 or 'columns'}, default 0*
Whether to drop labels from the index (0 or 'index') or columns (1 or 'columns').

index, columns : *single label or list-like*
Alternative to specifying axis (`labels, axis=1` is equivalent to `columns=labels`).
New in version 0.21.0.

level : *int or level name, optional*
For MultiIndex, level from which the labels will be removed.

inplace : *bool, default False*
If True, do operation inplace and return None.

errors : *{'ignore', 'raise'}, default 'raise'*
If 'ignore', suppress error and only existing labels are dropped.

Returns:

dropped : *pandas.DataFrame*

Raises:

KeyError
If none of the labels are found in the selected axis

See also:

`DataFrame.loc`
Label-location based indexer for selection by label.

`DataFrame.dropna`
Return DataFrame with labels on given axis omitted where (all or any) data are missing.

`DataFrame.drop_duplicates`
Return DataFrame with duplicate rows removed, optionally only considering certain columns.

`Series.drop`
Return Series with specified index labels removed.

Examples

```
>>> df = pd.DataFrame(np.arange(12).reshape(3,4),
...                   columns=['A', 'B', 'C', 'D'])
>>> df
   A  B   C   D
0  0  1   2   3
1  4  5   6   7
2  8  9  10  11
```

Drop columns

```
>>> df.drop(['B', 'C'], axis=1)
   A   D
0  0   3
1  4   7
2  8  11
```

```
>>> df.drop(columns=['B', 'C'])
   A   D
0  0   3
1  4   7
2  8  11
```

Drop a row by index

```
>>> df.drop([0, 1])
   A  B   C   D
2  8  9  10  11
```

Drop columns and/or rows of MultiIndex DataFrame

```
>>> midx = pd.MultiIndex(levels=[['lama', 'cow', 'falcon'],
...                              ['speed', 'weight', 'length']],
...                      codes=[[0, 0, 0, 1, 1, 1, 2, 2, 2],
...                             [0, 1, 2, 0, 1, 2, 0, 1, 2]])
>>> df = pd.DataFrame(index=midx, columns=['big', 'small'],
...                   data=[[45, 30], [200, 100], [1.5, 1], [30, 20],
...                         [250, 150], [1.5, 0.8], [320, 250],
...                         [1, 0.8], [0.3,0.2]])
>>> df
                big     small
lama    speed   45.0    30.0
        weight  200.0   100.0
        length  1.5     1.0
cow     speed   30.0    20.0
        weight  250.0   150.0
        length  1.5     0.8
falcon  speed   320.0   250.0
        weight  1.0     0.8
        length  0.3     0.2
```

```
>>> df.drop(index='cow', columns='small')
                big
lama    speed   45.0
        weight  200.0
        length  1.5
falcon  speed   320.0
        weight  1.0
        length  0.3
```

```
>>> df.drop(index='length', level=1)
                big     small
lama    speed   45.0    30.0
        weight  200.0   100.0
cow     speed   30.0    20.0
        weight  250.0   150.0
falcon  speed   320.0   250.0
        weight  1.0     0.8
```