

### Pandas Data Structure

**Series** A one-dimensional labeled array capable of holding any data type

```
>>> s = pd.Series([3, -5, 7, 4], index=['a', 'b', 'c', 'd'])
```

**DataFrame** A two-dimensional labeled data structure with columns of potentially different types

```
>>> data = {'Country': ['Belgium', 'India', 'Brazil'], 'Capital': ['Brussels', 'New Delhi', 'Brasília'], 'Population': [11190846, 1303171035, 207847528]}
```

### I/O

**Read and Write to CSV**

```
>>> pd.read_csv(, header=None, nrows=5)

>>> df.to_csv('myDataFrame.csv')
```

**Read and Write to Excel**

```
>>> pd.read_excel( )

>>> df.to_excel('dir/myDataFrame.xlsx', sheet_name='Sheet1')
```

### Applying Functions

**Apply function element-wise**

```
>>> f = lambda x: x*2

>>> df.apply(f)

>>> df.applymap(f)
```

### Basic Information

```
(rows,columns) >>> df.shape
Describe index >>> df.index
Describe DataFrame columns >>> df.columns
Info on DataFrame >>> df.info()
```

### Data Alignment

NA values are introduced in the indices that don't overlap:

```
>>> s3 = pd.Series([7, -2, 3], index=['a', 'c', 'd'])

>>> s + s3
a 10.0
b NaN
c 5.0
d 7.0
```

### Data Alignment

NA values are introduced in the indices that don't overlap:

```
>>> s3 = pd.Series([7, -2, 3], index=['a', 'c', 'd'])

>>> s + s3
a 10.0
b NaN
c 5.0
d 7.0
```

### Data Alignment

NA values are introduced in the indices that don't overlap:

```
>>> s3 = pd.Series([7, -2, 3], index=['a', 'c', 'd'])

>>> s + s3
a 10.0
b NaN
c 5.0
d 7.0
```

### Data Alignment

NA values are introduced in the indices that don't overlap:

```
>>> s3 = pd.Series([7, -2, 3], index=['a', 'c', 'd'])

>>> s + s3
a 10.0
b NaN
c 5.0
d 7.0
```

C

By **suchismita**  
[cheatography.com/suchismita/](http://cheatography.com/suchismita/)

Not published yet.  
Last updated 14th July, 2020.  
Page 1 of 1.

Sponsored by **CrosswordCheats.com**  
Learn to solve cryptic crosswords!  
<http://crosswordcheats.com>