

Python Pandas

Data analysis using Pandas

Pandas is the most popular python library that is used for data analysis. It provides highly optimized performance with back-end source code is purely written in **C** or **Python**

We can analyze data in pandas with:

1. **Series**
2. **DataFrames**

Series:

Series is one dimensional(1-D) array defined in pandas that can be used to store any data type.

Code #1: Creating Series

Program to create series

```
import pandas as pd # Import Panda Library
```

```
# Create series with Data, and Index
```

```
a = pd.Series(Data, index = Index)
```

Here, **Data** can be:

1. A **Scalar value** which can be integerValue, string
2. A **Python Dictionary** which can be Key, Value pair
3. A **Ndarray**

Note: Index by default is from 0, 1, 2, ...(n-1) where n is length of data.

Code #2: When Data contains scalar values

Program to Create series with scalar values

```
Data =[1, 3, 4, 5, 6, 2, 9] # Numeric data
```

```
# Creating series with default index values
```

```
s = pd.Series(Data)
```

```
# predefined index values
```

```
Index =['a', 'b', 'c', 'd', 'e', 'f', 'g']
```

```
# Creating series with predefined index values
```

```
si = pd.Series(Data, Index)
```

```
In [4]: s
Out[4]:
0    1
1    3
2    4
3    5
4    6
5    2
6    9
dtype: int64
```

Scalar Data with default Index

```
In [6]: si
Out[6]:
a    1
b    3
c    4
d    5
e    6
f    2
g    9
dtype: int64
```

Code #3: When Data contains Dictionary

```
# Program to Create Dictionary series
```

```
dictionary ={'a':1, 'b':2, 'c':3, 'd':4, 'e':5}
```

```
# Creating series of Dictionary type
```

```
sd = pd.Series(dictionary)
```

```
In [8]: sd
Out[8]:
a    1
b    2
c    3
d    4
e    5
dtype: int64
```

Code #4: When Data contains Ndarray

```
In [13]: snd
Out[13]:
0      [2, 3, 4]
1      [5, 6, 7]
dtype: object
```

DataFrames:

DataFrames is two-dimensional(2-D) data structure defined in pandas which consists of rows and columns.

Code #1: Creation of DataFrame

Program to Create DataFrame

```
import pandas as pd # Import Library
```

```
a = pd.DataFrame(Data) # Create DataFrame with Data
```

Here, Data can be:

1. One or more **dictionaries**
2. One or more **Series**
3. **2D-numpy Ndarray**

Code #2: When Data is Dictionaries

Program to Create Data Frame with two dictionaries

```
dict1 = {'a':1, 'b':2, 'c':3, 'd':4} # Define Dictionary 1
```

```
dict2 = {'a':5, 'b':6, 'c':7, 'd':8, 'e':9} # Define Dictionary 2
```

```
Data = {'first':dict1, 'second':dict2} # Define Data with dict1 and dict2
```

```
df = pd.DataFrame(Data) # Create DataFrame
```

```
In [15]: df
Out[15]:
   first  second
a    1.0      5
b    2.0      6
c    3.0      7
d    4.0      8
e    NaN      9
```

Code #3: When Data is Series

Program to create Dataframe of three series

```
import pandas as pd
```

```
s1 = pd.Series([1, 3, 4, 5, 6, 2, 9]) # Define series 1
```

```
s2 = pd.Series([1.1, 3.5, 4.7, 5.8, 2.9, 9.3]) # Define series 2
```

```
s3 = pd.Series(['a', 'b', 'c', 'd', 'e']) # Define series 3
```

```
Data = {'first':s1, 'second':s2, 'third':s3} # Define Data
```

```
dfseries = pd.DataFrame(Data) # Create DataFrame
```

```
In [5]: dfseries
```

```
Out[5]:
```

| | first | second | third |
|---|-------|--------|-------|
| 0 | 1 | 1.1 | a |
| 1 | 3 | 3.5 | b |
| 2 | 4 | 4.7 | c |
| 3 | 5 | 5.8 | d |
| 4 | 6 | 2.9 | e |
| 5 | 2 | 9.3 | NaN |
| 6 | 9 | NaN | NaN |

Code #4: When Data is 2D-numpy ndarray

Note: One constraint has to be maintained while creating DataFrame of 2D arrays – Dimensions of 2D array must be same.

Program to create DataFrame from 2D array

```
import pandas as pd # Import Library
```

```
d1 = [[2, 3, 4], [5, 6, 7]] # Define 2d array 1
```

```
d2 = [[2, 4, 8], [1, 3, 9]] # Define 2d array 2
```

```
Data = {'first': d1, 'second': d2} # Define Data
```

```
df2d = pd.DataFrame(Data) # Create DataFrame
```

```
In [5]: df2d
```

```
Out[5]:
```

| | first | second |
|---|-----------|-----------|
| 0 | [2, 3, 4] | [2, 4, 8] |
| 1 | [5, 6, 7] | [1, 3, 9] |