

Q1. Create a Pandas Series that contains the following data: 4, 8, 15, 16, 23, and 42. Then, print the series.

Answer.

```
import pandas as pd
data = [4,8,15,16,23,42]
ser = pd.Series(data)
print(ser)
```

OUTPUT:

```
0      4
1      8
2     15
3     16
4     23
5     42
dtype: int64
```

Q2. Create a variable of list type containing 10 elements in it, and apply pandas.Series function on the variable print it.

Answer:

```
data = ["Ashish", 'Singh', 8.9, True, False, 4, 8, 15, 16, 23, [1, 2, 3, 4, 5]]
ser = pd.Series(data)
print(ser)
```

```
import pandas as pd
data = ["Ashish", 'Singh', 8.9, True, False, 4, 8, 15, 16, 23, [1, 2, 3, 4, 5]]
ser = pd.Series(data)
print(ser)
```

Output:

```
0      Ashish
1      Singh
2        8.9
3       True
4      False
5         4
6         8
7        15
8        16
9        23
10    [1, 2, 3, 4, 5]
dtype: object
```

Q3. Create a Pandas DataFrame that contains the following data:

Name	Age	Gender
Alice	25	Female
Bob	30	Male
Claire	27	Female

Answer:

```
import pandas as pd
```

```
Name = ["Alice","Bob","Claire"]
Age = [25,30,27]
Gender= ["Female","Male","Female"]
```

```
NameSer = pd.Series(Name)
AgeSer = pd.Series(Age)
GenderSer=pd.Series(Gender)
```

```
frame = {'Name': NameSer,
         'Age': AgeSer,
         'Gender':GenderSer}
```

```
result = pd.DataFrame(frame)
```

```
print(result)
```

Output:

	Name	Age	Gender
0	Alice	25	Female
1	Bob	30	Male
2	Claire	27	Female

Q4. What is 'DataFrame' in pandas and how is it different from pandas.series? Explain with an example.

Answer: Series is a type of list in Pandas that can take integer values, string values, double values, and more. But in Pandas Series we return an object in the form of a list, having an index starting from 0 to n, Where n is the length of values in the series.

Pandas DataFrame is two-dimensional size-mutable, potentially heterogeneous tabular data structure with labeled axes (rows and columns). A Data frame is a two-dimensional data structure, i.e., data is aligned in a tabular fashion in rows and columns.

Q5. What are some common functions you can use to manipulate data in a Pandas DataFrame? Can you give an example of when you might use one of these functions?

Answer: *We can read the dataframe by using head() function also which is having an argument (n) i.e. number of rows to be displayed.

* Counting the rows and columns in DataFrame using shape(). It returns the no. of rows and columns enclosed in a tuple.

* Summary of Statistics of DataFrame using describe() method.

* Dropping the missing values in DataFrame, it can be done using the dropna() method, it removes all the NaN values in the dataframe.

* Merging DataFrames using merge(), arguments passed are the dataframes to be merged along with the column name.

* Renaming the columns of dataframe using rename(), arguments passed are the columns to be renamed & inplace.

* Sorting the DataFrame using sort_values() method.

Q6. Which of the following is mutable in nature Series, DataFrame, Panel?

Answer: All Pandas data structures are value mutable (can be changed) and except Series all are size mutable.

Q7. Create a DataFrame using multiple Series. Explain with an example.

Answer:

```
import pandas as pd
```

```
Subject = ["Data Science","C++","Java","Python","Neet","JEE"]  
Duration = [8,7,9,6,12,12]
```

```
SubSer = pd.Series(Subject)  
DurSer = pd.Series(Duration)
```

```
frame = {'Subject': SubSer,  
        'Duration(In Months)': DurSer}
```

```
result = pd.DataFrame(frame)
```

```
print(result)
```

OutPut:

	Subject	Duration(In Months)
0	Data Science	8
1	C++	7
2	Java	9
3	Python	6
4	Neet	12
5	JEE	12