

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

Answer

```
In [18]: import pandas as pd

try :
    data = [4,8,12,16,23,42]
    ser = pd.Series(data)
    print(ser)
except Exception as e :
    print(e)

type(ser)
```

```
0    4
1    8
2   12
3   16
4   23
5   42
dtype: int64
```

```
Out[18]: pandas.core.series.Series
```

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

Answer

```
In [19]: N = ["Hare Krishna", 2,3,4,5,0.3,"@", [23],6,7]

ser1 = pd.Series(N)
print(ser1)
```

```

0    Hare Krishna
1              2
2              3
3              4
4              5
5              0.3
6              @
7             [23]
8              6
9              7
dtype: object

```

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

Name : Alice Bob Claire

Age : 25 30 27

Gender : Female Male Female

```

In [21]: data1 = {"Name":["Alice", "Bob", "Claire"],
                  "Age":[25,30,27],
                  "Gender":["Female", "Male", "Female"]}

df = pd.DataFrame(data1)
df

```

```

Out[21]:
   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

In pandas, dataframe can have more than one columns while on the other side series can have only 1 column. It means that dataframe can store more complex data than "Series".

Another difference is that dataframe can have different data types for each and every column while on the other side series can store only one type of data for the whole array. This means that Dataframe can store mixed type of data while series can only one type of data at a time.

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

1.Head() :

This function gives us the top 5 rows as the output. If we have a very big size data but we only have to focus only on the top 5 or 10 rows, then in this situation Head() function in pandas helps us a lot.

2.Tail() :

This function in pandas gives us the last 5 rows of our dataset as the output.

3.describe() :

If we want to have a rough idea or analysis of our dataset then in that situation, we use describe function in pandas. If we have a data set of mixed data types then in that situation it works only on those columns with "int" or "float" data types.

4.columns():

In any situation, if we have a very big size data with a lot of rows and columns and we want to check the column names, then the columns() function in pandas helps us a lot.

5.Nunique() :

The Nunique() Function in pandas shows us the total number of unique values a column has.

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

Answer

All the data structures in pandas are mutable in nature but except series all others are size mutable.

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

Answer

```
In [31]: data2 = {"Student_Name" :["Harshit", "Madhav", "Gopal", "Nimai", "Nitai"],  
                "CS_Marks" :[23,34,45,56,67]}
```

```
df3 = pd.DataFrame(data2)  
df3
```

We all know that a series is List type data array, Here we can see how i used 2 d

```
Out[31]:
```

	Student_Name	CS_Marks
0	Harshit	23
1	Madhav	34
2	Gopal	45
3	Nimai	56
4	Nitai	67