1 Python Pandas-I

Fastrack Revision

- **Introduction to Python Libraries:** Python libraries contains a collection of built-in modules that allow us to perform many actions without writing detailed programs for it.
- NumPy, Pandas and Matplotlib are three well-established Python libraries for scientific and analytical use. These libraries allow us to manipulate, transform and visualise data easily and efficiently.
- **NumPy:** NumPy stands for 'Numerical Python', is a library package that can be used for numerical data analysis and scientific computing.
- **Pandas:** Pandas stands for 'panel data'. It is a high-level data manipulation tool used for analysing data.
- **Matplotlib:** The Matplotlib library in Python is used for plotting graphs and visualisation. Using Matplotlib, with just a few lines of code we can generate publication quality plots, histograms, bar charts, scatter plots, etc.
- **Difference between Pandas and NumPy:** Following are some of the difference between Pandas and NumPy:
  - A NumPy array requires homogeneous data, while a Pandas DataFrame can have different data types (float, int, string, etc).
  - Pandas DataFrames (with column names) make it very easy to keep track of data.
  - Pandas is used when data is in tabular format, whereas NumPy is used for numeric array based data manipulation.
- **Installing Pandas:** To install Pandas from command line, we need to type in:

```
pip install pandas
```

- **Importing Pandas:** In order to work with Pandas in Python, we need to import Pandas library in Python environment. We can do this either on the shell prompt or in our script file (.py) by writing:

```
import pandas as pd
```

- **Pandas Data Structure:** A data structure is a particular way of storing and organising data in a computer to suit a specific purpose so that it can be accessed and worked with in appropriate ways.
  Two commonly used data structures in Pandas are:
  - **Series:** It is one-dimensional data structure of Python Pandas.
  - **DataFrame:** It is two-dimensional data structure of Python Pandas.
- **Series Data Structure:** A series is a one-dimensional array containing a sequence of values of any data type (int, float, list, string, etc.) which by default have numeric data labels starting from zero. The data label associated with a particular value is called its index.

Example:

Index	Value
0	Rohit
1	Ankit
2	Deepak
3	Ayush

Index	Value
Jan	31
Feb	28
Mar	31
Apr	30

- **Creation of Series:** A series can be created in many ways using Pandas library's series ( ). Make sure that we have imported Pandas and NumPy modules with import statements.
  - **Create Empty Series Object by using just Series ( ) with no Parameter:** To create an empty objects *i.e.,* having no values, we can just use the series ( ) as:
    <Series Object> = Panda. Series ( )
  - **Creating Non-empty Series Object:** To create non-empty series, the we need to specify arguments for data and indexes as per the following syntax:
    <Series Object> = pd. Series (data, index = Idx)
    where Idx is a valid NumPy datatype and data is the data part of the series object, it can be one of the following:
  - **Creation of Series from Scalar Value:** A series can be created using scalar values as:

```
>>>import pandas as pd
>>>series 1 = pd.Series ([10,20,30])
>>>print (series 1)
```

Output:

```
Index    Data values
0        10
1        20
2        30
dtypo : int 64
```

  - **Creation of Series from NumPy Arrays:** We can create a series from one-dimensional NumPy array as:

```
>>> import numpy as np
>>> import pandas as pd
>>> array 1 = np.array ([11, 22, 33, 44])
>>> series 1= pd.Series (array1)
>>> print (series 1)
```

Output:

```
Index    Data values
0        11
1        22
2        33
3        44
dtypo : int 32
```

➤ **Creation of Series from Dictionary:** We can create a series by specifying indexes and values through a dictionary as:

```
>>> dict 1 = {'Uttar Pradesh': 'Lucknow', 'Rajasthan': 'Jaipur'}
>>> print (dict 1)
{'Uttar Pradesh': 'Lucknow', 'Rajasthan': 'Jaipur'}
>>> Series 1 = pd.Series (dict 1)
>>> print (series 1)
```

Output:

Uttar Pradesh	Lucknow
Rajasthan	Jaipur
dtype : object	

▶ **Accessing Elements of a Series:** There are two common ways for accessing the elements of a series: Indexing and Slicing.

➤ **Indexing:** Indexing in series is similar to that for NumPy arrays and is used to access elements in a series. Indexes are of two types: positional index and labelled index. Positional index takes an integer value that corresponds to its position in the series starting from 0, whereas labelled index takes any user-defined label as index.

Example:
```
>>> seriesNum = pd.Series([11,22,33])
>>> seriesNum[1]
22
```

Here, the value 30 is displayed for the positional index 2.

➤ **Slicing:** This is similar to slicing used with NumPy arrays. We can define which part of the series is to be sliced by specifying the start and end parameters [start : end] with the series name. When we use positional indices for slicing, the value at the endindex position is excluded, *i.e.*, only (end - start) number of data values of the series are extracted.

Example:
```
>>> seriesCapState = pd.Series
(['Dispur', 'Patna','Panaji'], index=
['Assam', 'Bihar', 'Goa'])
>>> seriesCapState[1:2]
Bihar Patna
dtype: object
```

Here, only data values at indices 1 is displayed *i.e.* excludes the value at index position 2.

▶ **Attributes of Series:** We can access certain properties called attributes of a series by using that property with the series name.

Attribute Name	Purpose
name	assigns a name to the series
index.name	assigns a name to the index of the series
values	prints a list of the values in the series
size	prints the number of values in the series object
empty	prints True if the series is empty and False otherwise

▶ **Methods of Series:**

Method	Explanation
Head(n)	Returns the first n members of the series. If the value for n is not passed, then by default n takes 5 and the first five members are displayed.
Count()	Returns the number of non-NaN values in the series.
Tail(n)	Returns the last n members of the series. If the value for n is not passed, then by default n takes 5 and the last five members are displayed.

▶ **Mathematical Operations on Series:**

➤ **Addition:** We can use the '+' Operator or add() method of series to perform addition between two series objects.

➤ **Subtraction:** We can use the '–' Operator or sub() method of series to perform subtraction between two series objects.

➤ **Division:** We can use the '/' Operator or div() method of series to perform division between two series objects.

➤ **Multiplication:** We can use the '×' Operator or mul() method of series to perform multiplication between two series objects.

➤ **Exponential Power:** We can use the '**' Operator or pow() method of series to put each element of passed series as exponential power of caller series and return the results.

▶ **DataFrame Data Structure:** A DataFrame is a two-dimensional labelled data structure like a table of MySQL. It contains rows and columns and therefore has both a row and column index. The row index is known as **index** and the column index is called the **column-name**.

▶ **Creation of DataFrame:** There are a number of ways to create a DataFrame. Some of them are listed in this section.

➤ **Creation of an empty DataFrame:** An empty DataFrame can be created as follows:

```
>>> import pandas as pd
>>> dFrameEmt = pd.DataFrame()
>>> dFrameEmt
```

Output:

Empty DataFrame
Columns: [ ]
Index: [ ]

➤ **Creation of DataFrame from NumPy ndarrays:** Consider the following three NumPy ndarrays. Let us create a simple DataFrame without any column labels, using a single ndarray:

```
>>> import numpy as np
>>> array1 = np.array([11,22,33])
>>> array2 = np.array([110,210,310])
>>> array3 = np.array([-100,-200,-300,
-400])
>>> dFrame4 = pd.DataFrame(array1)
>>> dFrame4
    0
```

Output:

```
0  11
1  22
2  33
```

- **Creation of DataFrame from List of Dictionaries:** We can create DataFrame from a list of Dictionaries as:

```
>>> listDict = [{'a':11, 'b':22},
{'a':5, 'b':10, 'c':20}]
>>> dFrameListDict = pd.DataFrame
(listDict)
>>> dFrameListDict
```

Output:

	a	b	c
0	11	22	NaN
1	5	10	20.0

- **Creation of DataFrame from Dictionary of Lists:** DataFrames can also be created from a dictionary of lists.

```
>>> dictForest = {'State': ['Kanpur',
'Delhi', 'Udaipur'], 'GArea': [96838,
7583, 44552],
'VDF' : [3197, 4.42, 2563]}
>>> dFrameForest= pd.DataFrame
(dict Forest)
>>> dFrameForest
```

Output:

	State	GArea	VDF
0	Kanpur	96838	3197.00
1	Delhi	7583	4.42
2	Udaipur	44552	2563.00

- **Creation of DataFrame from Dictionary of Series:** A dictionary of series can also be used to create a DataFrame as:

```
>>> ResultSheet= 'Rohit':
pd.Series([80, 92, 87],
index=['English','Science','Maths']),
'Ayush': pd.Series([72, 81, 94],
index=['English','Science','Maths']),
'Priya': pd.Series([84, 86, 78],
index=['English','Science','Maths']),
>>> ResultDF=pd.DataFrame(ResultSheet)
>>> ResultDF
```

Output:

	Rohit	Ayush	Priya
English	80	72	84
Science	92	81	86
Maths	87	94	78

- **Operations on Rows and Columns in DataFrames:**
  - **Adding a New Column to a DataFrame:** We can easily add a new column to a DataFrame.
  - **Adding a New Row to a DataFrame:** We can add a new row to a DataFrame using the DataFrame.loc[ ] method.
  - **Deleting Rows or Columns from a DataFrame:** We can use the DataFrame.drop() method to delete rows and columns from a DataFrame.
  - **Renaming Row Labels of a DataFrame:** We can change the labels of rows and columns in a DataFrame using the DataFrame.rename() method.
  - **Renaming Column Labels of a DataFrame:** To alter the column names of ResultDF, we can again use the rename() method.
- **Accessing DataFrames Element through Indexing:** Data elements in a DataFrame can be accessed using indexing. There are two ways of indexing DataFrames: Label Based Indexing and Boolean Indexing.
  - **Label Based Indexing:** There are several methods in Pandas to implement label based indexing. DataFrame.loc[ ] is an important method that is used for label based indexing with DataFrames.
  - **Boolean Indexing:** In boolean indexing, we can select the subsets of data based on the actual values in the DataFrame rather than their row/column labels. Thus, we can use conditions on column names to filter data values.
- **Accessing DataFrames Element through Slicing:** We can use slicing to select a subset of rows and/or columns from a DataFrame. To retrieve a set of rows, slicing can be used with row labels.
- **Attributes of DataFrames:**

Attribute Name	Purpose
DataFrame.index	to display row labels
DataFrame.columns	to display column labels
DataFrame.dtypes	to display data type of each column in the DataFrame
DataFrame.values	to display a NumPy ndarray having all the values in the DataFrame, without the axes labels
DataFrame.shape	to display a tuple representing the dimensionality of the DataFrame
DataFrame.size	to display a tuple representing the dimensionality of the DataFrame
DataFrame.T	to transpose the DataFrame, means, row indices and column labels of the DataFrame replace each other's position
DataFrame.head(n)	to display the first n rows in the DataFrame
DataFrame.tail(n)	to display the last n rows in the DataFrame

Practice Exercise

Multiple Choice Questions

Q 1. To create an empty series object, you can use:
a. pd.Series(empty) b. pd.Series(np.NaN)
c. pd.Series( ) d. All of these

Q 2. To specify datatype int16 for a series object, you can write:
a. pd.Series(data = array, dtype = int16)
b. pd.Series(data = array, dtype = numpy.int16)
c. pd.Series(data = array, dtype = pandas.int16)
d. All of the above

Q 3. To get the number of dimensions of a series object, ... attribute is displayed.
a. Index b. size c. itemsize d. ndim

Q 4. To get the size of the datatype of the items in series object, you can display ..... attribute.
a. index b. size c. itemsize d. ndim

Q 5. To get the number of elements in a series object, ... attribute may be used.
a. Index b. size c. itemsize d. ndim

Q 6. To get the number of bytes of the series data, ... attribute is displayed.
a. hasnans b. nbytes
c. ndim d. dtype

Q 7. To check if the series object contains NaN values, ... attribute is displayed.
a. hasnans b. nbytes
c. ndim d. dtype

Q 8. To display third element of a series object S, you will write .....
a. S[:3] b. S[2] c. S[3] d. S[:2]

Q 9. To display first three elements of a series object S, you may write .....
a. S[:3] b. S[3]
c. S [3rd] d. All of these

Q 10. To display last five rows of a series object S, you may write .....
a. head( ) b. tail(5)
c. tail( ) d. Either b. or c.

Q 11. Missing data in Pandas object is represented through:
a. null b. none
c. missing d. NaN

Q 12. Given a Pandas series called Sequences, the command which will display the first 4 rows is .....
a. print(Sequences.head(4))
b. print (Sequences. Head(4))
c. print(Sequences.heads(4))
d. print(Sequences.Heads(4))

Q 13. If a DataFrame is created using a 2D dictionary, then the indexes/row labels are formed from .....
a. dictionary's values b. inner dictionary's keys
c. outer dictionary's keys d. None of these

Q 14. If a DataFrame is created using a 2D dictionary, then the column labels are formed from .....
a. dictionary's values b. inner dictionary's keys
c. outer dictionary's keys d. None of these

Q 15. The axis 0 identifies a DataFrame's .....
a. rows b. columns
c. values d. datatype

Q 16. The axis 1 identifies a DataFrame's .....
a. rows b. columns
c. values d. datatype

Q 17. To get the number of elements in a DataFrame, ... attribute may be used.
a. size b. shape
c. values d. ndim

Q 18. To get NumPy representation of a DataFrame, ... attribute may be used.
a. size b. shape
c. values d. ndim

Q 19. To get a number representing number of axes in a DataFrame, ... attribute may be used.
a. size b. shape
c. values d. ndim

Q 20. The name "Pandas" is derived from the term: *[CBSE SQP 2021 Term-1]*
a. Panel Data b. Panel Series
c. Python Document d. Panel DataFrame

Q 21. The command to install the Pandas is: *[CBSE SQP 2021 Term-1]*
a. install pip Pandas b. install Pandas
c. pip Pandas d. pip install Pandas

Q 22. Python Pandas was developed by: *[CBSE SQP 2021 Term-1]*
a. Guido van Rossum b. Travis Oliphant
c. Wes McKinney d. Brendan Eich

Q 23. Pandas Series is: *[CBSE SQP 2021 Term-1]*
a. 2 Dimensional b. 3 Dimensional
c. 1 Dimensional d. Multidimensional

Q 24. Pandas is a: *[CBSE SQP 2021 Term-1]*
a. Package b. Language
c. Library d. Software

Q 25. We can analyse the data in Pandas with *[CBSE SQP 2021 Term-1]*
a. Series b. Data Frame
c. Both a. and b. d. None of these

Q 26. Method or function to add a new row in a DataFrame is: *[CBSE SQP 2021 Term-1]*

a. .loc() b. .iloc() c. join d. add()

Q 27. Which of the following import statement is not correct? *[CBSE SQP 2021 Term-1]*

a. import Pandas as class12
b. import Pandas as 1pd
c. import Pandas as pd1
d. import Pandas as pd

Q 28. While accessing the column from the dataframe, we can specify the column name. In case column does not exist, which type of error it will raise: *[CBSE SQP 2021 Term-1]*

a. Key Error b. Syntax Error
c. Name Error d. Runtime Error

Q 29. Function to display the first n rows in the DataFrame: *[CBSE SQP 2021 Term-1]*

a. tail (n) b. head (n)
c. top (n) d. first (n)

Q 30. Pandas DataFrame cannot be created using: *[CBSE SQP 2021 Term-1]*

a. Dictionary of tuples b. Series
c. Dictionary of List d. List of Dictionaries

Q 31. Which function will be used to read data from a CSV file into Pandas DataFrame? *[CBSE SQP 2021 Term-1]*

a. readcsv() b. to_csv()
c. read_csv() d. csv_read()

Q 32. Which of the following is not an attribute of Pandas DataFrame? *[CBSE SQP 2021 Term-1]*

a. length b. T
c. Size d. Shape

Q 33. What will be the output of the given code?

```
import Pandas as pd
s = pd.Series([1,2,3,4,5],
index=['akram', 'brijesh', 'charu', 'deepika', 'era'])
print(s['charu'])
```

[CBSE SQP 2021 Term-1]

a. 1 b. 2 c. 3 d. 4

Q 34. Assuming the given series, named stud, which command will be used to print 5 as output?

Amit	90
Ramesh	100
Mahesh	50
john	67
Abdul	89

Name: Student, dtype: int64 *[CBSE SQP 2021 Term-1]*

a. stud.index b. stud.length
c. stud.values d. stud.size

Q 35. A Social Science teacher wants to use a Pandas series to teach about Indian historical monuments and its states. The series should have the monument names as values and state names as indexes which are stored in the given lists, as shown in the code. Choose the statement which will create the series:

```
import pandas as pd
Monument=['Qutub Minar', 'Gateway of India', 'Red Fort', 'Taj Mahal']
State=['Delhi', 'Maharashtra', 'Delhi', 'Uttar Pradesh']
```

[CBSE SQP 2021 Term-1]

a. S=df.Series(Monument,index=State)
b. S=pd.Series(State,Monument)
c. S=pd.Series(Monument,index=State)
d. S=pd.series(Monument,index=State)

Q 36. Consider the following series named animal:

L	Lion
B	Bear
E	Elephant
T	Tiger
W	Wolf
dtype:	Object

Write the output of the command:

print(animal[::-3]) *[CBSE SQP 2021 Term-1]*

a. L Lion
T Tiger
dtype: object

b. W Wolf
B Bear
dtype: object

c. B Bear
E Elephant
dtype: Object

d. W Wolf
T Tiger
dtype: object

Q 37. What is a correct syntax to return the values of first row of a Pandas DataFrame? Assuming the name of the DataFrame is dfRent. *[CBSE SQP 2021 Term-1]*

a. dfRent[0] b. dfRent.loc[1]
c. dfRent.loc[0] d. dfRent.iloc[1]

Q 38. Difference between loc() and iloc().: *[CBSE SQP 2021, Term-1]*

a. Both are Label Indexed based functions.
b. Both are Integer position-based functions.
c. loc() is label based function and iloc() integer position based function.
d. loc() is integer position based function and iloc() index position based function.

Q 39. Write the output of the given program:

```
import Pandas as pd
S1=pd.Series([5,6,7,8,10],index=['v', 'w', 'x', 'y', 'z'])
1=[2,6,1,4,6]
S2=pd.Series(1,index=['z', 'y', 'a', 'w', 'v'])
print(S1-S2)
```

[CBSE SQP 2021 Term-1]

a.
a	0
v	–1.0
w	2.0
x	NaN
y	2.0
z	8.0

dtype: float64

b.
a	NaN
v	–1.0
w	2.0
x	NaN
y	2.0
z	8.0

dtype: float64

c.
v	–1.0
w	2.0
y	2.0
z	8.0

dtype: float64

d.
a	NaN
v	–1.0
w	2.0
x	3.0
y	2.0
z	8.0

dtype: float64

Q 40. Which command will be used to delete 3 and 5 rows of the DataFrame. Assuming the DataFrame name as DF. *[CBSE SQP 2021 Term-1]*

a. DF.drop([2,4],axis=0) b. DF.drop([2,4],axis=1)
c. DF.drop([3,5],axis=1) d. DF.drop([3,5])

Q 41. Write the output of the given command:

```
import Pandas as pd
s=pd.Series([1,2,3,4,5,6],index=['A','B','C','D','E','F'])
print(s[s%2==0])
```

[CBSE SQP 2021 Term-1]

a.
```
B  0
D  0
F  0
dtype: int64
```

b.
```
B  2
D  4
F  6
dtype: int64
```

c.
```
A  1
B  2
C  5
dtype: int64
```

d.
```
B  1
D  2
F  3
dtype: int64
```

Q 42. Ritika is a new learner for the Python Pandas and she is aware of some concepts of Python. She has created some lists, but is unable to create the DataFrame from the same. Help her by identifying the statement which will create the DataFrame.

```
import pandas as pd
Name=['Manpreet','Kavil','Manu','Ria']
Phy=[70,60,76,89]
Chem=[30,70,50,65]
```

a. df=pd.DataFrame({'Name':Name,'Phy':Phy,'Chem':Chem})
b. d={'Name':Name,'Phy':Phy,'Chem':Chem}
df=pd.DataFrame(d)
c. df=pd.DataFrame([Name,Phy,Chem],columns=['Name','Phy','Chem','Total'])
d. df=pd.DataFrame({Name:'Name', Phy :'Phy',Chem:'Chem'})

Q 43. Assuming the given structure, which command will give us the given output:

Flight No.		Airline	Passenger
0	1	Indigo	230000
1	2	SpiceJet	12000
2	3	Indian Airlines	240000
3	4	Lufthansa	245000
4	5	Air Asia	210000

Output Required: (3,5) *[CBSE SQP 2021 Term-1]*

a. print(df.shape()) b. print(df.shape)
c. print(df.size) d. print(df.size())

Q 44. Write the output of the given command:

```
df1.loc[:0,'Sal']
```

Consider the given DataFrame.

	EName	Airline	Passenger
0	Kavita	50000	3000
1	Sudha	60000	4000
2	Garima	55000	5000

[CBSE SQP 2021 Term-1]

a. 0 Kavita 50000 3000 b. 50000
c. 3000 d. 60000

Q 45. Consider the following DataFrame name df

	Name	Age	Marks
0	Amit	15	90.0
1	Bhavdeep	16	NaN
2	Reema	17	87.0

Write the output of the given command:

```
print(df.marks/2)
```

[CBSE SQP 2021 Term-1]

a.
```
0  45.0
1   NaN
2  43.5
Name: Marks, dtype: float64
```

b.
```
0  45.0
1   NaN
2   43
Name: Marks, dtype: float64
```

c.
```
0   45
1  NaN
2  43.5
Name: Marks, dtype: float64
```

d.
```
0  45.0
1    0
2  43.5
Name: Marks, dtype: float64
```

Q 46. Read the statements given below. Identify the right option from the following for Attribute and method/function.

Statement 1: Attribute always ends without parenthesis.

Statement 2: Function/Method cannot work without arguments. *[CBSE SQP 2021 Term-1]*

a. Both statements are correct.
b. Both statements are incorrect.
c. Statement 1 is correct, but Statement 2 is incorrect.
d. Statement 1 is incorrect, but Statement 2 is correct.

Q 47. To get the transpose of a DataFrame D1, you can write .....

a. D1.T b. D1. Transpose
c. D1.Swap d. All of these

Q 48. Which of the following is a two-dimensional labelled data structure of Python? *[CBSE 2023]*

a. Relation b. DataFrame
c. Series d. Square

Q 49. To display the 3rd, 4th and 5th columns from the 6th to 9th rows of a DataFrame DF, you can write .....

a. DF.loc[6:9, 3:5]
b. DF.loc[6:10, 3:6]
c. DF.iloc[6:10, 3:6]
d. DF.iloc[6:9, 3:5]

Q 50. To change the 5th column's value at 3rd row as 35 in DataFrame DF, you can write .....

a. DF[4, 6] = 35 b. DF[3, 5] = 35
c. DF.at[4, 6] = 35 d. DF.at[3, 5] = 35

Q 51. What will be the output of the following code?

```
Import pandas as pd
myser = pd. Series ([0, 0, 0])
print (myser)
```

(CBSE 2023)

a.
```
0  0
0  0
0  0
```
b.
```
0  1
0  1
0  2
```
c.
```
0  0
1  0
2  0
```
d.
```
0  0
1  1
2  2
```

Q 52. Which of the following command will show the last 3 rows from a Pandas Series named NP?

[CBSE SQP 2023-24]

a. NP.Tail( )
b. NP.tail(3)
c. NP.TAIL(3)
d. All of these

Q 53. Which of the following statement is wrong?

[CBSE 2021 Term-1]

a. Can't change the Index of the Series.
b. We can easily convert the list, tuple and dictionary into a Series.
c. A Series represents a single column in memory
d. We can create empty Series.

Q 54. What type of error is returned by the following statement?

```
import pandas as pa
pa.Series ([1,2,3,4], index = ['a', 'b', 'c'])
```

[CBSE 2021 Term-1]

a. Value error
b. Syntax error
c. Name error
d. Logical error

Q 55. Which is incorrect statement for the python package Numpy?

[CBSE SQP 2021 Term-1]

a. It is a general-purpose array-processing package
b. Numpy arrays are faster and more compact
c. It is multi-dimensional arrays
d. It is proprietary software

Q 56. The data of any CSV file can be shown in which of the following software?

[CBSE 2021 Term-1]

a. MS Word
b. Notepad
c. Spreadsheet
d. All of these

Q 57. Which Python library is not used for data science?

[CBSE 2021 Term-1]

a. Panda
b. Numpy
c. Matplotlib
d. Tkinter

Q 58. Which method is used to Delete row(s) from DataFrame?

[CBSE 2021 Term-1]

a. .drop() method
b. .del() method
c. .remove() method
d. .delete() method

Q 59. Consider the following code:

```
Import numpy as np
import pandas as pd
L = np.array ([10,201])
xpd.Series (.....)
print(x)
```

Output of the above code is:

```
0  1000
1  8000
dtype: int64
```

What is the correct statement for the above output in the following statement 1?

[CBSE 2021 Term-1]

a. d=L**3
b. data=L**3
c. L**3
d. [10,20]**3

Q 60. Which of the following would give the same output as DF/DF1 where DF and DF1 are DataFrames?

a. DF.div(DF1)
b. DF1.div(DF)
c. Divide(DF,DF1)
d. Div(DF,DF1)

Q 61. Which of the following statement is wrong in context of DataFrame?

[CBSE 2021 Term-1]

a. Two dimensional size is Mutable
b. Can perform Arithmetic operations on rows and columns
c. Homogeneous tabular data structure
d. Create DataFrame from numpy ndarray

Q 62. Which attribute is not used with DataFrame?

[CBSE 2021 Term-1]

a. size
b. type
c. empty
d. columns

Q 63. When we create a DataFrame from a list of Dictionaries the columns labels are formed by the

[CBSE 2021 Term-1]

a. Union of the keys of the dictionaries
b. Intersection of the keys of the dictionaries
c. Union of the values of the dictionaries
d. Intersection of the values of the dictionaries

Q 64. Identify the correct option to select first four rows and second to fourth columns from a DataFrame 'Data'

[CBSE 2021 Term-1]

a. display(Data.iloc[1 : 4, 2 : 4])
b. display(Data.iloc[1 : 5, 2 : 5])
c. print(Data.iloc[0 : 4, 1 : 4])
d. print(Data.iloc[1 : 4, 2 : 4])

Q 65. Which attribute is used with Series to count the total number of NaN values.

[CBSE 2021 Term-1]

a. size
b. len
c. count
d. count total

Q 66. Consider the following Series in Python:

```
data = pd.Series([5,2,3,7], index=['a', 'b', 'c', 'd'])
```

Which statement will display all odd values?

[CBSE 2021 Term-1]

a. print(data%2==0)
b. print(data(data%2!=0))
c. print(data mod 2!=0)
d. print(data(data%2!=0))

Q 67. What will be the output of the following code?

```
import pandas as pd
import numpy
s=pd.Series(data=[31,54,34,89,12,23],
dtype=numpy.int) print (>50)
```

[CBSE 2021 Term-1]

a.	b.	c.	d.
0 False	1 54	0 31	1 True
1 True	3 89	1 54	3 True
2 False	dtype: int64	2 34	dtype: bool
3 True		3 89	
4 False		4 12	
5 False		5 23	
dtype:bool		dtype:int64	

Q 68. Consider a following DataFrame:

```
import pandas as pd
s=pd.Series (data=[31, 54, 34, 89, 12, 23])
df=pd.DataFrame(s)
```

Which statement will be used to get the output as 2?

[CBSE 2021 Term-1]

a. print(df.index) b. print(df.shape())
c. print(df.ndim) d. print(df.values)

Q 69. Sandhya wants to display the last four rows of the dataframe df and she has written the following command:

```
df.tail()
```

But the first 5 rows are being displayed. To rectify this problem, which of the following statements should be written? *[CBSE 2021, Term-1]*

a. df.head() b. df.last(4)
c. df.tail(4) d. df.rows(4)

Q 70. Consider the following series:

```
ser=pd.Series([`C', `O', 'M', 'F', 'O', 'R', 'T', 'A', 'B', 'L', 'E',])
index=([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11])
print(ser[4:])
```

[CBSE 2021 Term-1]

a.	b.	c.	d.
4 F	4 F	4 F	5 O
5 O	5 O	5 O	6 R
6 R	6 R	6 R	7 T
7 T	7 T	7 T	8 A
8 A	8 A	8 A	9 B
9 B	dtype: object	9 B	10 L
10 L		dtype: object	11 E
11 E			
dtype: object			dtype: object

Q 71. Now-a-days for developing Machine learning projects programmers rely on CSV files rather than databased. Why? *[CBSE 2021 Term-1]*

a. CSV can be used with proprietary softwares only.
b. CSV files can be downloaded from open source websites free of cost.
c. CSV files need not be imported while creating the projects.
d. CSV is a simple and well formatted mode for data storage.

Q 72. DataFrames can be created from: *[CBSE 2021 Term-1]*

a. lists b. dictionaries
c. series d. All of these

Q 73. Consider the following statements:
Statement 1: .loc() is a label based data selecting method to select a specific row(s) or column(s) which we want to select.
Statement 2: .iloc() can not be used with default indices if customised indices are provided. *[CBSE 2021, Term-1]*

a. Statement 1 is True, but Statement 2 is False.
b. Statement 1 is False, but Statement 2 is True.
c. Statement 1 and Statement 2 both are False.
d. Statement 1 and Statement 2 both are True.

Q 74. Abhay is a student of class 'XII' and he is aware of some concepts of python. He has created the DataFrame, but he is getting errors after executing the code. Help him by identifying the correct statement that will create the DataFrame.

Code:

```
import pandas as pd
stuname= [ 'Muskan', 'Radhika', 'Gopar', 'Pihu' ]
term1 = [70, 63, 74, 90]
term2 = [67, 70, 86, 95]
```

[CBSE 2021 Term-1]

a. df=pd.DataFrame('Name':stuname,"marks1":term1, "marks2": term2))
b. df=pd.DataFrame([stuname, term1, term2], columns=['stuname', "marks1", "marks2"])
c. df=pd.DataFrame([stuname,term1, term2])
d. df=PD.dataframe ([stuname, term1, term2])

Q 75. Mr. Raman created a DataFrame from a Numpy array

```
arr = np.array([2, 4, 8], [3, 9, 271, [4, 16, 64]])
df=pd.DataFrame (arr, index=['one', 'two', 'three'],
.....)
print (df)
```

Help him to add a customised column labels to the above DataFrame. *[CBSE 2021 Term-1]*

a. columns='no', 'sq', 'cube')
b. column=['no', 'sq', 'cube']
c. columns=['no', 'sq', 'cube']
d. columns=[['no', 'sq', 'cube']]

Q 76. What will be the output of the following program

```
Import pandas as pd?
dic={'Name' : ['Sapna', 'Anmol', 'Rishul', 'Sameep'],
'Aggregate' : [56, 67, 75, 76], 'Age' : [16, 18, 16, 19]}
df=pd.DataFrame (dic, columns=['Name', 'Age'])
print (df)
```

[CBSE SQP 2021 Term-1]

a.

	Name	Agg	Age
101	Sapna	56	16
102	Anmol	67	18
103	Rishul	75	16
104	Sameep	76	19

b.

	Name	Agg	Age
0	Sapna	56	16
1	Anmol	67	18
2	Rishul	75	16
3	Sameep	76	19

c.

	Name
0	Sapna
1	Anmol
2	Rishul
3	Sameep

d.

	Name	Age
0	Sapna	16
1	Anmol	18
2	Rishul	16
3	Sameep	19

Q 77. Consider the following code:

```
import pandas as pd
S1=pd . Series ([23, 24, 35, 56], index=['a', 'b', 'c', 'd'])
S2=pd.Series ([27, 12, 14, 15],
index=['b', 'y', 'c', 'ab'])
df=pd.DataFrame (S1+S2)
print (df)
```

Output for the above code will be: *[CBSE 2021, Term-1]*

a.
```
    0
a   NaN
ab  NaN
b   51.0
c   49.0
d   NaN
y   NaN
```

b.
```
    0
a   50
b   36
c   49
d   71
```

c.
```
    0
b   50
y   36
c   49
ab  71
```

d.
```
    0
a   NaN
ab  NaN
b   NaN
c   NaN
d   NaN
y   NaN
```

Q 78. Sudhanshu has written the following code to create a DataFrame with boolean index:

```
import numpy as np
import pandas as pd
df=pd.DataFrame (data=[5, 6, 7], index=[true, false, true])
print (df)
```

While executing the code, she is getting an error, help her to rectify the code: *[CBSE 2021 Term-1]*

a. df=pd.DataFrame([True, False, True], data=[5, 6, 7])
b. df=pd.DataFrame(data=[5, 6, 7],index=[True, False, True])
c. df=pd.DataFrame([true,false,true],data=[[5,6,7]])
d. df=pd.DataFrame(index=[true,false,true],data=[[5,6,7]])

Q 79. Sushila has created a DataFrame with the help of the following code:

```
import pandas
EMP='EMPID' : ['E01', 'E02', 'E03', 'E04', 'E05'],
'EMPNAME' : ['KISHORI', 'PRIYA', 'DAMODAR', 'REEMA', 'MANOJ'],
'EMP_SALARY' : [67000, 34000, 68000, 90000, 43000]
df=pandas.DataFrame(EMP,index=['001', '002', '003', '004', '005'])
print(df.loc[0:3,:])
```

and she wants to get the following output

```
EMPID EMPNAME EMP_SALARY
001 E01 KISHORI      67000
002 E02 PRIYA        34000
003 E03 DAMODAR      68000
```

Help her to correct the code *[CBSE 2021 Term-1]*

a. print(df.iloc['001':'003',:])
b. print(df.loc['001':'003',:])
c. print(EMP[loc[[0:3,:]])
d. print(df.loc['001':'004',:])

Fill in the Blanks Type Questions

Q 80. ..... is a popular data-science library of Python.

Q 81. A ..... is a Pandas data structure that represents a 1D array like object.

Q 82. A ..... is a Pandas data structure that represents a 2D array like object.

Q 83. You can use numpy..... for missing data.

Q 84. To specify datatype for a series object,..... argument is used.

Q 85. The ..... function on series object returns total elements in it including NaNs.

Assertion & Reason Type Questions

Directions (Q. Nos. 86-94): *In the questions given below, there are two statements marked as Assertion (A) and Reason (R). Read the statements and choose the correct option.*

a. Both Assertion (A) and Reason (R) are true and Reason (R) is the correct explanation of Assertion (A).
b. Both Assertion (A) and Reason (R) are true, but Reason (R) is not the correct explanation of Assertion (A).
c. Assertion (A) is true, but Reason (R) is false.
d. Assertion (A) is false, but Reason (R) is true.

Q 86. Assertion (A): To use the Pandas library in a Python program, one must import it.
Reason (R): The only alias name that can be used with the Pandas library is pd. *(CBSE SQP 2023-24)*

Q 87. Assertion (A): A Data structure is a Pandas data structure that represents a 2D array like object.
Reason (R): You can use numpy NaN for missing data.

Q 88. Assertion (A): To specify datatype for a series object, dtype argument is used.
Reason (R): The len( ) function on series object returns total elements in it including NaNs.

Q 89. Assertion (A): The count( ) function on series object returns only the count of non-NaN values in it.
Reason (R): Series is value mutable.

Q 90. Assertion (A): Series is not size mutable.
Reason (R): DataFrame is size mutable as well as value mutable.

Q 91. Assertion (A): In a DataFrame, Axis = 1 represents the column elements.
Reason (R): To access values using row labels you can use DF.open.

Q 92. Assertion (A): To access individual value, you can use DF on using row/column index labels.
Reason (R): To access individual value, you can use DF. lat using row/column integer position.

Q 93. Assertion (A): The rename( ) function requires inplace argument to make changes in the original DataFrame.
Reason (R): To access values using row labels you can use DF. loc.

Q 94. Assertion (A): The output of addition of two series will be NaN, if one of the elements or both the elements have no value(s).
Reason (R): While performing mathematical operations on a series by default all missing values are filled in with 0. *[CBSE 2023]*

Answers

1. (c)	2. (b)	3. (d)	4. (c)	5. (b)
6. (b)	7. (a)	8. (b)	9. (a)	10. (d)
11. (d)	12. (a)	13. (b)	14. (c)	15. (a)
16. (b)	17. (a)	18. (c)	19. (d)	20. (a)
21. (d)	22. (c)	23. (c)	24. (c)	25. (c)
26. (a)	27. (b)	28. (a)	29. (b)	30. (a)
31. (c)	32. (a)	33. (c)	34. (d)	35. (c)
36. (b)	37. (c)	38. (c)	39. (b)	40. (a)
41. (b)	42. (a)	43. (b)	44. (b)	45. (a)
46. (c)	47. (a)	48. (b)	49. (c)	50. (d)
51. (c)	52. (c)	53. (a)	54. (a)	55. (d)
56. (d)	57. (d)	58. (a)	59. (b)	60. (a)
61. (c)	62. (b)	63. (a)	64. (c)	65. (c)
66. (d)	67. (a)	68. (c)	69. (c)	70. (d)
71. (d)	72. (d)	73. (a)	74. (a)	75. (c)
76. (d)	77. (a)	78. (b)	79. (b)	

80. Pandas 81. series
82. DataFrame 83. NaN
84. dtype 85. len( )

86. (b)	87. (d)	88. (b)	89. (b)	90. (b)
91. (c)	92. (d)	93. (b)	94. (b)	

Case Study Based Questions

Case Study 1

Pandas Data Structure: A data structure is a particular way of storing and organising data in a computer to suit a specific purpose so that it can be accessed and worked with in appropriate ways. Pandas offer many data structures to handle a variety of data. At very basic level, Pandas data structure can be thought of somewhat as enhanced version of NumPy structured arrays in which the rows and columns can be identified and accessed with labels rather than simple integer indices.

Q 1. Out of many data structures of Pandas, two basic data structures ..... and ..... are universally popular for their dependability.
a. series, DataFrame
b. matplotlib, NumPy
c. 1D array and 2D array
d. None of the above

Q 2. DataFrame is a ..... data structure of Python Pandas.
a. two-Dimensional b. one-Dimensional
c. Both a. and b. d. None of these

Q 3. A series is a Pandas data structure that represents a one-dimensional array like object containing an array of data and an associated array of data labels, called its .....
a. row b. column
c. label d. index

Q 4. When numeric indexes can depict positions of data in the objects (for example, when indexes go as 0, 1, 2, ...), they are called .....
a. labelled Indexes b. positional Indexes
c. Both a. and b. d. None of these

Q 5. A series type objects can be created in many ways using Pandas library's .....
a. series() b. seriesObject()
c. series.Object() d. None of these

Answers

1. (a) 2. (a) 3. (d) 4. (a) 5. (a)

Case Study 2

Consider the following DataFrame df and answer any four questions from (a) – (e):

Roll No.	Name	UT1	UT2	UT3	UT4
1.	Prerna Singh	24	24	20	22
2.	Manish Arora	18	17	19	22
3.	Tanish Goel	20	22	18	24
4.	Falguni Jain	22	20	24	20
5.	Kanika Bhatnagar	15	20	18	22
6.	Ramandeep Kaur	20	15	22	24

Q 1. Write down the command that will give the following output:

```
rollno      6
name        Tanish Goel
UT1         24
UT2         24
UT3         24
UT4         24
dtype: object
```

a. print(df.max) b. print(df.max())
c. print(df.max(axis=1)) d. print(df.max. axis=1)

Q 2. The teacher needs to know the marks scored by the student with roll number 4. Help her identify the correct set of statement(s) from the given options:

a. df1 = df[df['rollno'] == 4]
print(df1)
b. df1 = df[rollno == 4]
print(df1)
c. df1 = df[df.rollno=4]
print(df1)
d. df1 = df[df.rollno == 4]
print(df1)

Q 3. Which of the following statement(s) will give the exact number of values in each column of the DataFrame?

I. print(df.count())
II. print(df.count(0))
III. print(df.count)
IV. print(df.count(axis='index'))

Choose the correct option:

a. I and II b. Only II
c. I, II and III d. I, II and IV

Q 4. Which of the following command will display the column labels of the DataFrame?

a. print(df.columns( )) b. print(df.column( ))
c. print(df.column) d. print(df.columns)

Q 5. Mrs. Sharma, the class teacher wants to add a new column, the scores of Grade with the values, 'A', B', 'A', 'A', 'B', 'A', to the DataFrame. Help her choose the command to do so: *(CBSE SQP 2020-21)*

a. df.column= ['A', 'B', 'A', 'A', 'B', 'A']
b. df['Grade'] =['A', 'B', 'A', 'A', 'B', 'A']
c. df.loc['Grade']= ['A', B', 'A', 'A', 'B', 'A']
d. Both b. and c. are correct

Answers

1. (b) 2. (a) 3. (a) 4. (d) 5. (b)

Case Study 3

Mr. Ankit is working in an organisation as data analyst. He uses Python Pandas and Matplotlib for the same. He got a dataset of the passengers for the year 2010 to 2012 for January, March and December. His manager wants certain information from him, but he is facing some problems. Help him by answering few questions given below:

	Year	Month	Passengers
0	2010	Jan	25
1	2010	Mar	50
2	2012	Jan	35
3	2010	Dec	55
4	2012	Dec	65

Code to create the above DataFrame:

```
import pandas as ..... #Statement 1
data = {"Year": [2010,2010,2012,2010,2012],
 "Month":
["Jan", "Mar", "Jan", "Dec", "Dec"],
 "Passengers":[25,50,35,55,65]}
df=pd. ..... (data) #Statement 2
print(df)
```

Q 1. Choose the right code from the following for statement 1.

a. pd b. df c. data d. p

Q 2. Choose the right code from the following for the statement 2.

a. DataArray b. DataFrame
c. Series d. Dictionary

Q 3. Choose the correct statement/method for the required output: (5,3)

a. df.index b. df.shape()
c. df.shape d. df.size

Q 4. He wants to print the details of "January" month along with the number of passengers. Identify the correct statement:

	Month	Passengers
0	Jan	25
2	Jan	35

a. df.loc[['Month','Passengers']][df['Month']=='Jan']
b. df[['Month','Passengers']][df['Month']=='Jan']]
c. df.iloc[['Month','Passengers']][df['Month']=='Jan']
d. df(['Month','Passengers'][df['Month']=='Jan'])

Q 5. Mr. Ankit wants to change the index of the DataFrame and the output for the same is given below. Identify the correct statement to change the index.

	Year	Month	Passengers
Air India	2010	Jan	25
Indigo	2010	Mar	50
SpiceJet	2012	Jan	35
Jet	2010	Dec	55
Emirates	2012	Dec	65

a. df.index[]=["Air India","Indigo","SpiceJet","Jet", "Emirates"]
b. df.index{"Air India","Indigo","SpiceJet","Jet", "Emirates"]
c. df.index=["Air India","Indigo","SpiceJet","Jet", "Emirates"]
d. df.index()=["Air India","Indigo","SpiceJet","Jet", "Emirates"]

Answers

1. (a) 2. (b) 3. (c) 4. (b) 5. (c)

Case Study 4

Answer the following based on the series given below.

```
import pandas as pd
list1=[1,2,3,4,5,6,7,8]
list2=['swimming','tt','skating','kho kho','bb','chess','football',"cricket"]
school=pd.Series(list1,index=list2)
school.name=("little")
print (school^2) #statement 1
print (school.tail(3)) #statement 2
print (school["tt"]) #statement 3
print (school[2:4]) #statement 4
```

Q 1. Choose the correct name of the series object given above.

a. list1 b. list2
c. school d. little

Q 2. Choose the correct output for the statement.
print (school.tail(3)) # statement 2

a. swimming 1
tt 2
skating 3

b. chess 6
football 7
cricket 8

c. 4

d. kho kho 4
bb 5
chess 6
football 7
cricket 8

Q 3. Choose the correct output for the statement.
print (school["tt"]). # statement 3

a. 2
b. 3
c. tt 2
d. true

Q 4. Identify the correct output for:
print (school[2:4]) # statement 4.

a. skating 3
kho kho 4

b. tt 2
skating 3
kho kho 4

c. skating 3
kho kho 4
bb 5

d. skating 3
kho kho 4
bb 5
chess 6
football 7
cricket 8

Q 5. The correct output of the statement:
print (school*2) # statement 1 will be:

a. swimming 3
tt 4
skating 5
kho kho 6
bb 7
chess 8
football 9
cricket 10

b. swimming False
tt False
skating True
kho kho True
bb True
chess True
football True
cricket True

c. swimming 2
tt 4
skating 6
kho kho 8
bb 10
chess 12
football 14
cricket 16

d. swimming 1
tt 4
skating 9
kho kho 16
bb 25
chess 36
football 49
cricket 64

Answers

1. (d) **2.** (b) **3.** (a) **4.** (a) **5.** (c)

Case Study 5

Mr. Sharma is working with an IT company and he has provided some data. On which he wants to do some operations, but he is facing some problem, help him:

Code:

```
import pandas as pd
ResultSheet={
'Naveen': pd.Series ([90, 91, 97],
index=['Maths', 'Science', 'Hindi']),
'Rehana' : pd.Series ([92, 81, 96],
index=['Maths', 'Science', 'Hindi']),
'John' : pd.Series ([89, 91, 88],
index=['Maths', 'Science', 'Hindi']),
'Roja' : pd.Series ([81, 71, 67]),
index = ['Maths', 'Science', 'Hindi']),
'Mannat' : pd.Series ([94, 95, 99],
index=['Maths', 'Science', 'Hindi'])}
DF=pd.DataFrame(ResultSheet)
print(DF)
```

Output of the above code:

Sub	Naveen	Rehana	John	Roja	Mannat
Maths	90	92	89	81	94
Science	91	81	91	71	95
Hindi	97	96	88	67	99

Q 1. He wants to add a new column with name of student 'Prem' in above dataframe, choose the right command to do so:

a. DF['Prem']=(89,78,76)
b. df['Prem']=(89,78,76)
c. DF['Prem']=(89,78,76,67)
d. DF['Name']=(89,78,76)

Q 2. He wants to set all the values to zero in dataframe, choose the right command to do so:

a. DF=0
b. DF[ ]=0
c. DF[:]=0
d. DF[:]= =0

Q 3. He wants to delete the row of science marks:

a. DF.drop('Science', axis=1)
b. DF.drop('Science', axis=0)
c. DF.drop('Science', axis=-1)
d. DF.drop('Science', axis= =0)

Q 4. The following code is to create another dataframe, which he wants to add to the existing DataFrame. Choose the right command to do so:

```
Sheet1={
'Aaradhya': pd.Series([90,91,97]
        index=['Maths','Science','Hindi'])}
S1=pd.DataFrame(Sheet1)
```

a. DF.append(S1,axis=0)
b. DF.append(S1)
c. DF.Insert(S1)
d. DF.Join(S1)

Q 5. What will be the output of the given command?

DF.index=['A','B','C']

a.

		Naveen	Rehana	John	Roja	Mannat
A	Maths	90	92	89	81	94
B	Science	91	81	91	71	95
C	Hindi	97	96	88	67	99

b.

	Naveen	Rehana	John	Roja	Mannat
A	90	92	89	81	94
B	91	81	91	71	95
C	97	96	88	67	99

c.

	A	B	C	Roja	Mannat
Maths	90	92	89	81	94
Science	91	81	91	71	95
Hindi	97	96	88	67	99

d. Error. Index already exists and cannot be overwritten.

Q 6. What will be the output of the given command?

	Naveen	Rehana	John	Roja	Mannat
Maths	90	92	89	81	94
Science	91	81	91	71	95
Hindi	97	96	88	67	99

print(DF.size)

a. 15 b. 18 c. 21 d. 23

Answers

1. (a) 2. (c) 3. (b) 4. (b) 5. (b)
6. (a)

Case Study 6

Ms. Ramdeep Kaur maintains the records of all students of her class. She wants to perform some operation on the data:

Code:

```
import pandas as pd
t={'Rollno':[101,102,103,104,105,106,107],
'Name' : [Shubrato' 'Krishna', 'Pranshu', 'Gurpreet',
'Arpit', 'Sanidhya', 'Aurobindo'],
'Age': [15,14,14,15,16,15,16],
'Marks' : [77.9, 70.4, 60.9, 80.3, 86.5, 67.7, 85.0]
'Grade': ['11B','11A','11B','11C','11E','11A','11C']}
df=pd.DataFrame(t,index=[10,20,30,40,50,60,70])
print(df)
```

Output of the above code:

	RollNo.	Name	Age	Marks	Grade
10	101	Shubrato	15	77.9	11B
20	102	Krishna	14	70.4	11A
30	103	Pranshu	14	60.9	11B
40	104	Gurpreet	15	80.3	11C
50	105	Arpit	16	86.5	11E
60	106	Sanidhya	15	67.7	11A
70	107	Aurobindo	16	85.0	11C

Q 1. Select the correct statement for the below output:

```
Name        Krishna
Age         14
Marks       70.4
Grade       11A
Name : 20,  dtype: object
```

a. print(df.iloc[2]) b. print(df.loc[2])
c. print(df.iloc[20]) d. print(df.loc[20])

Q 2. The teacher wants to know the marks secured by the second last student only. Which statement would help her to get the correct answer?

a. print(df.loc[60:70,'Marks'])
b. print(df.loc[60:60,'Marks'])
c. print(df.iloc[-2:-2],['Marks'])
d. print(df[-2:-2] ['Marks'])

Q 3. Which of the following statement(s) will add a new column 'fee' at second position with values [3200, 3400,4500,3100,3200,4000,3700] in DataFrame df?

a. df.insert(loc=2,column='fee',value= [3200,3400, 4500,3100,3200,4000,3700])
b. df.add(2,column='fee', [3200,3400,4500,3100, 3200,4000,3700])
c. df.append(loc=2,'fee', [3200,3400,4500,3100, 3200,4000,3700])
d. df.insert(loc=2,'fee', [3200,3400,4500,3100,32 00,4000,3700])

Q 4. Which of the following commands is used to delete the column 'Grade' in the DataFrame df?

a. df.drop('Grade',axis=1,inplace=True)
b. df.drop('Grade',axis=0,inplace=True)
c. df.drop('Grade',axis=1,inplace=True)
d. df.delete('Grade',axis=1,inplace=True)

Q 5. Which of the following commands would rename the column 'Marks' to 'Halfyearly' in the DataFrame df?

a. df.rename(('Marks', 'Halfyearly'),inplace=True)
b. df.rename(('Marks', 'Halfyearly'),inplace=True)
c. df.rename(columns={'Marks':'Halfyearly'}, inplace=True)
d. df.rename({'Marks' : 'Halfyearly'},inplace=True)

Q 6. Which of the following commands will display the Names and Marks of all students getting more than 80 marks?

a. print(df.loc['Marks'>80,['Name', 'Marks']])
b. print(df.loc[df['Marks']<80,'Name', 'Marks'])
c. print(df.loc[df['Marks']<80,['Name', 'Marks']])
d. print(df.loc[df['Marks']>80,['Name', 'Marks']])

Answers

1. (d) 2. (b) 3. (d) 4. (a) 5. (c)
6. (a)

Case Study 7

Sanyukta is the event incharge in a school. One of her students gave her a suggestion to use Python Pandas and Matplotlib for analysing and visualising the data, respectively. She has created a DataFrame "SportsDay" to keep track of the number of First, Second and Third prizes won by different houses in various events.

	House	First	Second	Third
0	Chenab	5	7	6
1	Ganges	10	5	4
2	Jamuna	8	13	15
3	Jhelum	12	9	12
4	Ravi	5	11	10
5	Satluj	10	5	3

Write Python commands to do the following:

Q 1. Display the house names where the number of Second Prizes are in the range of 12 to 20.

a. df['Name'][(df['Second']>=12) & (df['Second']<=20)]
b. df[Name][(df['Second']>=12) & (df['Second']<=20)]
c. df['Name'][(df['Second']>=12) & (df['Second']<=20)]
d. df[(df['Second']>=12) & (df['Second']<=20)]

Q 2. Display all the records in the reverse order.

a. print(df[::1]) b. print(df.iloc[::-1])
c. print(df[-1:]+df[:-1]) d. print(df.reverse())

Q 3. Display the bottom 3 records.

a. df.last(3) b. df.bottom(3)
c. df.next(3) d. df.tail(3)

Q 4. Choose the correct output for the given statements.

```
x=df.columns[:1]
print(x)
```

a. 0 b. Name
c. First d. Error

Q 5. Which command will give the output 24?

a. print(df.size) b. print(df.shape)
c. print(df.index) d. print(df.axes)

Answers

1. (c) 2. (b) 3. (d) 4. (b) 5. (a)

Case Study 8

Pandas is an open source library that is made mainly for working with relational or labeled data both easily and intuitively. It provides various data structures and operations for manipulating numerical data and time series. This library is built on top of the NumPy library. Pandas is fast and it has high performance and productivity for users. Pandas were initially developed by Wes McKinney in 2008 while he was working at AQR Capital Management. He convinced the AQR to allow him to open source the Pandas. Another AQR employee, Chang She, joined as the second major contributor to the library in 2012. Over time many versions of pandas have been released. The latest version of the pandas is 1.4.0

Q 1. What do you mean by Pandas in Python?
Q 2. Name three data structures available in Pandas.
Q 3. Write command to install Pandas in Python.
Q 4. What do you mean by Series in Python?
Q 5. Define data structure in Python.

Answers

1. PANDAS (PANEL DATA) is a high-level data manipulation tool used for analysing data. Pandas library has a very rich set of functions.
2. Three data structures available in Pandas are: Series, DataFrame and Panel
3. pip install Pandas
4. A Series is a one-dimensional array containing a sequence of values of any data type (int, float, list, string, etc.) which by default have numeric data labels (called index) starting from zero.
5. A data structure is a collection of data values and operations that can be applied to that data.

Case Study 9

Pandas is a fast, powerful, flexible and easy to use data analysis library built on top of NumPy and provides features not available in it. Pandas stands for panel data, a reference to the tabular format. It adopts significant parts of NumPy's idiomatic style of array-based computing. While Pandas adopts many coding idioms from NumPy, the biggest difference is that Pandas is designed for working with tabular, heterogeneous data.
Numpy, by contrast, is best suited for working with homogeneous numerical array data. The key to learning Pandas is to understand its data structures. A data structure is a collection of data values and defines the relationship between the data and the operations that can be performed on the data. There are three main data structures in Pandas: Series–1D, DataFrame –2D and Panel – 3D.

Q 1. Name a method which is used to create Series in Python.
Q 2. Write the code in Python to create an empty Series.
Q 3. Name any two attributes of Series in Python.
Q 4. Which property of series return all the index value?
Q 5. Write a program in Python to create a series of first five even numbers.

Answers

1. Series( )
2. import pandas as pd
   S1 = pd.Series( )
   print(S1)
   Output: Series([ ], dtype: float64)

3. Two attributes of Series in Python are: index and values.
4. Index property return all the index value.
5. Import pandas as pd
S1 = pd.Series([2, 4, 6, 8, 10])
print(S1)

Output:

```
0    2
1    4
2    6
3    8
4    10
dtype:int64
```

Very Short Answer Type Questions

Q 1. What is the significance of Pandas library?

Ans. The significance of Pandas library are as follows:
(i) It can read or write in many different data formats (integer, float, double, etc.).
(ii) It can calculate in all the possible ways data is organised *i.e.* across rows and down columns.

Q 2. Name some common data structures of Python's Pandas library.

Ans. Some common data structures of Python's Pandas library are DataFrame, Series and Panel.

Q 3. How is a series object different from and similar to ndarrays?

Ans. Series is 1D structure. It is a homogeneous data structure. Mutable type but size is immutable while ndarray is an N Dimensional array.

Q 4. Write commands to print following details of a series object seal:
(i) If the series is empty. (ii) Indexes of the series.

Ans. (i) seaLempty( ) (ii) seaLindex( )

Q 5. Why does following code cause error?
s1 = pd. Series (range(1, 15, 3), index = list('abcd'))

Ans. range (1, 15, 3) output contains 5 elements and Index has only 4 elements.

COMMON ERROR

Students are unable to find error and the concerned reason for it i.e., they ignore the number of elements in range and index.

Q 6. Why does following code cause error?
s1 = pd.Series(range(1, 15, 3), index = list('ababa'))
print(s1['ab'])

Ans. Because there are no rows by the names 'ab' in the series.

Q 7. Write code statements to list the following, from a DataFrame namely sales.
(I) List only columns 'Item' and 'Revenue'.
(ii) List the value of cell in 5th row, 'Item' column.

Ans. (i) sales['Item']
sales['Revenue']
(II) sales['Item'][4]

Q 8. Hitesh wants to display the last four rows of the DataFrame df and has written the following code:
df.tail()
But last 5 rows are being displayed. Identify the error and rewrite the correct code so that last 4 rows get displayed. *[CBSE SQP 2019-20]*

Ans. df.tail(4)

Q 9. How would you add a new column namely 'val' to a DataFrame df that has 10 rows in it and has columns as 'Item', 'Qty', 'Price'? You can choose to put any values of your choice.

Ans. df["val"] = [0,10,20,30,40,50,60,70,80,90]

Q 10. Write code statements for a DataFrame df for the following:
(i) Delete an existing column from it.
(ii) Delete rows from 3 to 6 from it.

Ans. (I) del df['existing column name(")
(ii) df.drop([2,3,4,5])

Q 11. Write code statements for a DataFrame df for the following:
(i) Check if the DataFrame has any missing values.
(II) Fill all missing values with 999 in it.

Ans. (I) df.isnull() (ii) df.fillna(999)

Q 12. Write statement(s) to delete a row from a DataFrame.

Ans. df.drop([row Index/rowLabel])

Q 13. Write statement(s) to delete a column from a DataFrame.

Ans. del df ['Column Name']

Q 14. Write statement(s) to change the value at 5th row, 6th column in a DataFrame df.

Ans. df ['6th column name"] [5th row index/row label] = New value

Q 15. Write statement(s) to change the values to 750 at 4th row to 9th row, 7th column in a DataFrame df.

Ans. df ['7th column name "] [4:7] = 750

Q 16. What is the difference between 'iloc' and 'loc' with respect to a DataFrame?

Ans. The difference between 'iloc' and 'loc' with respect to a DataFrame is as follows:
(i) loc gets rows (or columns) with particular labels from the index.
(ii) iloc gets rows (or columns) at particular positions in the index (so it only takes integers).

Short Answer Type-I Questions

Q 1. What do you understand by axes in a NumPy array? Define axes for a 2d ndarray.

Ans. In a NumPy array (also called ndarray), dimensions are called axes. The number of axes is called rank. A two-dimensional ndarray has two corresponding axes:
(i) The first axis running vertically downwards across rows (axis 0).
(ii) The second axis running horizontally across column (axis 1).

Q 2. How are DataFrames related to Series? What do you understand by the size of a Series and a DataFrame?

Ans. DataFrame and Series both are data structures from the Pandas library. Series is a one-dimensional structure whereas DataFrame is a two-dimensional structure.

Knowledge BOOSTER

Size attribute gives the number of elements present in Series or DataFrames.

Q 3. Create the following series and do the specified operations:
EngAlph, having 26 elements with the alphabets as values and default index values.

Ans.
```
import pandas as pd
EngAlph=pd.Series(['a','b','c','d','e','f','g','h','i','j','k','l','m',
'n','o','p','q','r','s','t','u','v','w','x','y','z'])
print(EngAlph)
```

Q 4. Consider a given Series, M1:

Index	Marks
Term 1	45
Term 2	65
Term 3	24
Term 4	89

Write a program in Python Pandas to create the series. *[CBSE SQP 2020-21]*

Ans.
```
import pandas as pd
m1=pd.Series ([45,65,24,89], index = ['term1', 'term2',
'term3', 'term4'])
```

Q 5. Create the following series and do the specified operations:
MonthDays, from a numpy array having the number of days in the 12 months of a year. The labels should be the month numbers from 1 to 12.

Ans.
```
import pandas as pd
import numpy as np
MonthDays=np.
array([31,28,31,30,31,30,31,31,30,31,30,31])
month=pd.Series(MonthDays,index=np.arange(1,13))
print(month)
```

Q 6. Write a Python program to create a series object, country using a list that stores the capital of each country. *[CBSE 2023]*
Note: Assume four countries to be used as index of the series object are India, UK, Denmark and Thailand having their capitals as New Delhi, London, Copenhagen, and Bangkok respectively.

Ans.
```
import pandas as pd
a = ['New Delhi', 'London', 'Copenhagen', 'Bangkok']
country = pd.series (a, index = ['India', 'UK', 'Denmark',
          'Thailand']
print (country)
```

Q 7. Write a program to create a series object using a dictionary that stores the number of students in each section of class 12 in your school.

Ans.
```
import pandas as pd
Stu = {'A' : 40 , 'B': 38, 'C': 46, 'D':43}
s8 = pd.Series(stu)
print(s8)
```

Output:
```
A    40
B    38
C    46
D    43
dtype: int64
```

Q 8. Given the following Series objects:

	S1		S2
0	3	0	12
1	5	2	10
2	6	3	15
4	10	4	20
5	12	6	27

(i) What will be the result of S1 + S2?
(ii) What will be the result of S1 – S2?
[CBSE SQP 2020-21]

Ans. (i)
```
0    15.0
1    NaN
2    16.0
3    25.0
4    32.0
6    NaN
```
(ii)
```
0    -9.0
1    NaN
2    -4.0
3    -5.0
4    -8.0
6    NaN
```

Q 9. Predict the output of the given Python code:

```
import pandas as pd
list 1 = (-10, -20, -30]
ser = pd.Series (list 1*2)
print (ser)
```
[CBSE SQP 2023-24]

Ans. The output of the given code is as shown:
```
0    -10
1    -20
2    -30
3    -10
4    -20
5    -30
```

Q 10. What will be the output of the following code:

```
import pandas as pd
s1=pd.Series (data=2* (3, 10))
print (s1)
```
[CBSE 2023]

Ans. Output of the given code is:
```
0    3
1    10
2    3
3    10
dtype : int 64
```

Q 11. Consider following Series object namely S:

```
0    0.430271
1    0.617328
2   –0.265421
3   –0.836113
dtype: float64
```

What will be returned by following statements?

(i) S * 100 **(ii) S > 0**

What will be the values of Series objects S1 and S2 created above?

Ans. (i) S*100

```
0    43.0271
1    61.7328
2   -26.5421
3   -83.6113
dtype: float64
```

(ii) S>0

```
0     True
1     True
2    False
3    False
dtype: bool
```

Q 12. What will be the output produced by following code, considering the Series object S given above?

(i) print(S[1:1]) **(ii) print(S[0:1])**

(iii) print (S[0:2]) **(iv) S[0:2] = 12**
print(S)

Ans. (i) Series([], dtype: int64)

(ii)
```
pencil    20
dtype:    int64
```

(iii)
```
pencil      20
notebook0   33
dtype:      int64
```

(iv)
```
pencil      12
notebook0   12
scale       52
eraser      10
dtype:      int64
```

Q 13. Complete the given Python code to get the required output as: Rajasthan import ..... as pd

di = { 'Corbett' : 'Uttarakhand', 'Sariska' : 'Rajasthan', 'Kanha' : 'Madhya Pradesh', 'Gir' : 'Gujarat'}

NP = ... Series (.....)

print (NP [.....]) *[CBSE SQP 2023-24]*

Ans.
```
import pandas as pd
di = {'Corbett': 'Uttarakhand', 'Sariska':'Rajasthan',
'Kanha': 'Madhya Pradesh', 'Gir':'Gujarat'}
NP = pd.Series (di)
print(NP['Sariska'])
```

Q 14. Carefully observe the following code:

```
import pandas as pd
dic= { 'paid' : [101, 102, 103, 104, 105],
'pname' : ('Shyam','Roushan','Archit','Medha','Lalit'],
'sports' : ['Cricket', 'Tennis', 'Football', 'Cricket',
'Cricket']
'points' : [45000, 20000, 15000, 53000, 60000] }
player=pd.DataFrame (dic)
print (player)
```

Write Python statements for the following:

(i) In the dataframe player created above, set the row labels as 'Player1', 'Player2', 'Player3', 'Player4', Player5'.

(ii) Rename the column 'points' to 'netpoint' in the DataFrame player'. *[CBSE 2023]*

Ans. (i) Player = pd.DataFrame (dic, index = ['Player1', 'Player2', 'Player3', 'Player4', 'Player5'])

(ii) Player = pd.rename ({'points' : 'netpoint'}, axis = 'columns')

Q 15. The python code written below has syntactical errors. Rewrite the correct code and underline the corrections made.

```
Import pandas as pd
df = {"Technology" : ["Programming", "Robotics",
"3D Printing"], "Time (in months)" : [4, 4, 3] }
df = Pd.dataframe (df)
Pint (df)
```

[CBSE SQP 2023-24]

Ans. The correct code is as follows:

```
import pandas as pd
df = {'Technology' : ['Programming', 'Robotics', '3D
Printing'], 'Time (in months)' : [4, 4, 3] }
df = pd.DataFrame (df)
print (df)
```

Q 16. If Ser is a Series type object having 30 values, then how are statements (i), (ii) and (iii), (iv) similar and different?

(i) print(Ser.head( ))

(ii) print(Ser.head(8))

(iii) print(Ser.tail())

(iv) print(Ser.tail(11))

Ans. (i) Prints all series

(ii) Prints first 8 elements of series

(iii) Print all series - Ser.tail() equal to Ser.head()

(iv) Prints last 11 elements of series

Q 17. Write a program to create a Series object from an ndarray that stores characters from 'a' to 'g'.

Ans.
```
import pandas as pd
import numpy as np
arr = np.array(list('abcdef'))
print(arr)
s1 = pd.Series(arr)
```

Q 18. How does DataFrame objects specify indexes to its data rows?

Ans. If the indexes are not specified with the DataFrame(), then these are generated from the input data in an intuitive fashion. For example, from the keys of dictionary (in the case of column labels) or by using np.range(n) (in the case of row labels), where n corresponds to the number of rows.

Short Answer Type-II Questions

Q 1. What is a Series and how is it different from a 1-D array, a list and a dictionary?

Ans. **Series:** A Series is a one-dimensional array having a sequence of values of any data type (int, float, list, string, etc.). By default, series have numeric data labels starting from zero.

Series vs 1-D array: Series can have default as well as predefined index labels whereas a numpy 1-d array has only default indexes. Series can contain values of any datatype whereas arrays can contain elements of the same datatype.

Series vs List: Series can have default as well as predefined index labels whereas a list has only default indexes.

Series vs Dictionary: Series elements can be accessed using default indexes as well as its row labels whereas dictionary elements cannot be accessed using default indexes. They have to be accessed using the predefined keys.

Q 2. Distinguish between Series and DataFrame.

Ans.

Property	Series	DataFrame
Dimensions	One-dimensional	Two-dimensional
Type of data	**Homogeneous,** i.e., all the elements must be of same data type in a Series object.	**Heterogeneous,** i.e., a DataFrame object can have elements of different data types.
Mutability	• **Value Mutable,** i.e., their elements' value can change. • **Size-immutable** i.e., size of a Series object, once created, cannot change. If you want to add/drop an element, internally a new Series object will be created.	• **Value Mutable,** i.e., their elements' value can change. • **Size-mutable** i.e., size of a DataFrame object, once created, can change in place. This is, you can add/drop elements in an existing DataFrame object.

Q 3. Consider two objects x and y. x is a list whereas y is a series. Both have values 20, 40, 90, 110.
What will be the output of the following two statements considering that the above objects have been created already?
(i) print (x*2) (ii) print(y*2)
Justify your answer. *[CBSE SQP 2020-21]*

Ans. (i) This will give the output as:
[20,40,90,110,20,40,90,110]
(ii) This will give the output as:

```
0    40
1    80
2   180
3   220
```

Justification: In the first statement, x represents a list so when a list is multiplied by a number, it is replicated that many number of times.
The second y represents a series. When a series is multiplied by a value, then each element of the series is multiplied by that number.

Q 4. Create a DataFrame in Python from the given list:
[['Divya','HR',95000],['Mamta','Marketing',97000], ['Payal','IT',980000],['Deepak','Sales',79000]]
Also give appropriate column headings as shown below: *[CBSE SQP 2023-24]*

	Name	Department	Salary
0	Divya	HR	95000
1	Mamta	Marketing	97000
2	Payal	IT	980000
3	Deepak	Sales	79000

Ans. The DataFrame in Python is as under:

```
Import pandas as pd
df = [ 'Divya', 'HR', 95000], ['Mamta', 'Marketing', 97000], ['Payal', 'IT', 980000], ['Deepak', 'Sales', 79000] ]
df=pd. DataFrame (df, columns=['Name', 'Department', 'Salary'])
print (df)
```

Q 5. Write code to print all the information about a Series object.

Ans.

```
Import pandas as pd
s = pd.Series([40,35,25,20,22,26,28,29,23,28])
print("shape of Series :", s.shape)
print("dtype of Series :", s.dtype)
print("Index of Series :", s.index)
print("nbytes of Series :", s.nbytes)
print("Ndimension of Series :", s.ndim)
print("size of Series :", s.size)
print("NaN values of Series :", s.hasnans)
```

TiP
Be careful towards the syntax in all the statements while writing the code.

Q 6. Consider the following DataFrame and answer the questions given below:

```
import pandas as pd
df = pd.DataFrame({"Quarter1":[2000, 4000, 5000, 4400, 10000],
 "Quarter2":[5800, 2500, 5400, 3000, 2900],
 "Quarter3":[20000, 16000, 7000, 3600, 8200],
 "Quarter4":[1400, 3700, 1700, 2000, 6000]})
```

(i) Write the code to find mean value from above DataFrame df over the index and column axis.

(ii) Use sum() function to find the sum of all the values over the index axis.

(iii) Find the median of the DataFrame df.

[CBSE SQP 2019-20]

Ans.

```
(i) print(df.mean(axis = 1))
    print(df.mean(axis = 0))
(ii) print(df.sum(axis = 1))
(iii) print(df.median())
```

Q 7. Kavyanjali, a chemical analyst, needs to arrange data of few elements in the form of two series containing symbols and their atomic numbers respectively. Thereafter, the data of these two series has to be arranged and displayed in the form of DataFrame as shown below:

	Symbol	Atomic Number
Hydrogen	H	1
Helium	He	2
Lithium	Li	3
Beryllium	Be	4

Help her in writing suitable Python code to complete the task. *[CBSE 2023]*

Ans.

```
import pandas as pd
elements = {'Symbol' : ['H', 'He', 'Li', 'Be'],
            'Atomic Number' : [1, 2, 3, 4]}
df = pd . DataFrame (elements, index = ['Hydrogen', 'Helium', 'Lithium', 'Beryllium'])
print (df)
```

Q 8. Consider the given DataFrame 'health':

	Disease name	Agent
0	Common cold	Virus
1	Chickenpox	Virus
2	Cholera	Bacteria
3	Tuberculosis	Bacteria

Write suitable Python statements for the following:

(i) Remove the row containing details of disease named Tuberculosis.

(ii) Add a new disease named 'Malaria' caused by 'Protozoa'.

(iii) Display the last 2 rows. *[CBSE 2023]*

Ans.

```
(i) health = health . drop (3, axis=0)
(ii) health.loc [4] = ['Malaria', 'Protozoa']
(iii) health.tail (2)
```

Q 9. What will be the output produced by following code?

```
Stationery = ['pencils', 'notebooks', 'scales', 'erasers']
S = pd. Series ([20, 33, 52, 10], index = Stationery)
S2 = pd. Series ([17, 13, 31, 32], index = Stationery)
print (S + S2)
S = S + S2
print (S + S2)
```

Ans.

```
pencils      37
notebooks    46
scales       83
erasers      42
dtype: int   64
pencils      54
notebooks    59
scales       114
erasers      74
dtype: int64
```

Q 10. Consider the given DataFrame 'Genre':

	Type	Code
0	Fiction	F
1	Non Fiction	NF
2	Drama	D
3	Poetry	P

Write suitable Python statements for the following:

(i) Add a column called Num_Copies with the following data: [300, 290, 450, 760].

(ii) Add a new genre of type 'Folk Tale' having code as "FT" and 600 number of copies.

(iii) Rename the column 'Code' to 'Book_Code'.

[CBSE SQP 2023-24]

Ans.

```
(i) Genre ["Num_Copies"] = [300. 290. 450. 760]
(ii) Genre.loc [4] = ["Folk Tale", "FT", 600)
(iii) Genre=Genre.rename ( {"Code" : "Book_Code" }. axis-1)
                    Or
Genre=Genre.rename ({'Code':'Book_Code'}, axis="columns')
```

Long Answer Type Questions

Q 1. Write a program that stores the sales of 5 fast moving items of a store for each month in 12 Series objects, *i.e.*, S1 Series object stores sales of these 5 items in 1st month, S2 stores sales of these 5 items in 2nd month and so on.

The program should display the summary sales report like this:

Total Yearly Sales, item-Series (should display stem of items' sales over the months)
Maximum sales of item made: <name of item that was sold the maximum in whole year>
Maximum sales for individual items
Maximum sales of item 1 made: <month in which that item sold the maximum>
Maximum sales of item 2 made: <month in which that item sold the maximum>

Maximum sales of item 3 made: <month in which that item sold the maximum>
Maximum sales of item 4 made: <month in which that item sold the maximum>
Maximum sales of item 5 made: <month in which that item sold the maximum>

Ans.
```
import pandas as pd
Indx=["Item1", 'Item2', "Item3", "Item4', "Item5']
s1 = pd.Series( [ 160, 75, 89, 75, 85 ] , index=indx )
s2 = pd.Series( [ 86, 89, 70, 85, 90 ] , index=Indx )
s3 = pd.Series( [ 85, 75, 60, 75, 72 ] , index=indx )
s4 = pd.Series( [ 372, 92, 85, 107, 85 ] , index=indx )
s5 = pd.Series( [ 60, 75, 90, 75, 77 ] , index=Indx )
s6 = pd.Series( [ 60, 85, 45, 60, 85 ], index=Indx )
s7 = pd.Series( [ 286, 75, 66, 75, 86 ], index=indx )
s8 = pd.Series( [ 60, 72, 200, 70, 75 ], index=indx )
s9 = pd.Series( [ 86, 75, 60, 85, 70 ], Index=Indx )
s10 = pd.Series( [ 60, 89, 90, 75, 85 ] , index=indx )
s11 = pd.Series( [ 70, 75, 78, 86, 55 ] , index=indx )
s12 = pd.Series( [ 86, 85, 85, 75, 53 ] , index=Indx )
yearlySal = s1 + s2 + s3 + s4 + s5 + s6 + s7 + s8 + s9
+ s10 + s11 + s12
print( 'Yearly Sales Item-' ,yearlySal )
print('Maximum sales of item made' , end='> ' )
itemlist = list ( yearlySal )
print( indx [ itemlist.index ( max ( itemlist ) ) ] )
print()
List = [ s1 , s2 , s3 , s4 , s5 , s6 , s7 , s8 , s9 , s10 , s11 ,
s12 ] #list of all series
lst=[]
print( "Maximum sales for individual items" )
print()
dic={ 1:s1 , 2:s2 , 3:s3 , 4:s4 , 5:s5 , 6:s6 , 7:s7 , 8:s8 ,
9:s9 , 10:s10 , 11:s11 , 12:s12 }
max = 0
for j in range( 5 ) :
 print ( f'Maximum sales of {Indx[i]} made: ' , end=' ' )
 max = 0
 for j in dic:
 if max < list ( dic [ j ] ) [ i ] :
    max = list( dic[ j ])[ i ]
    s = j
 print(s)
```

Q 2. Answer the following questions:

(i) Consider the below given two code fragments. Will they produce the same output? Why/Why not?

(a)
```
fst = [9, 10, 11]
ob1 = pd.Series(data = fst * 2)
print(ob1)
```

(b)
```
fst = pd.Series (data = [9, 10, 11])
ob2 = pd.Series(data = fst * 2)
print(ob2)
```

(ii) Given are two objects, a list object namely 1st1 and a Series object namely ser1, both are having similar values i.e., 2, 4, 6, 8. Find out the output produced by following statements:

(a) print(1st1 * 2)
(b) print(ser1 * 2)

(iii) Can you specify the reason behind the output produced by the code statements of previous questions?

Ans. (i) No, both codes will produce different results.

This is because, in the part (a) Series object ob1's data contains **a Python list (fst)*2** which will repeat the values of the list two times and the data will contain : **9, 10, 11, 9, 10, 11**.

In part (b), Series object ob2's data contains a **Series object (fst) *2**, which will perform the vectorised operation on the values of **fst** and then make it as data of ob2, *i.e.*, **18, 20, 22.**

(ii) The output produced by given statement (a) will be:

[2, 4, 6, 8, 2, 4, 6, 8]

The output produced by given statement (b) will be:

0	4
1	8
2	12
3	16

where 0, 1, 2, 3 are the indexes.

(iii) The reason behind the output produced by statement (a) is that:

When some scalar value is multiplied with a list object, then the list is replicated that many times, thus the result is having the list elements repeated twice.

The reason behind the output produced by statement (b) is that:

When some scalar value is multiplied with a Series object, then the given operation gets performed on every single element of the Series because for a Series, the operations are carried in a vectorised way, *i.e.*, applied on each element one by one.

Q 3. Answer the following questions:

(i) Write a program to create data series and then changes the indexes of the Series object in any random order.

(ii) Write a program to sort the values of a Series object s1 in ascending order of its values and stores it into Series object s2.

(iii) Write a program to create a Series object with 6 random integers and having indexes as : ['p', 'q', 'r', 'n', 't', 'v']

Ans. (i)
```
import pandas as pd
import numpy as np
s1 = pd.Series (data = [100, 200, 300, 400, 500],
index = ['I', 'J', 'K', 'L', 'M'])
print("Original Data Series:")
print(s)
s1 = s1. reindex(Index = ['K', 'I', 'M', 'L', 'J' ])
print ("Data Series after changing the order of
Index:")
print(s)
```

(ii) Import pandas as pd
: # Series object s1 created or loaded
s2 = s1.sort_values ( )
print("Series object s1:")
print(s1)
print("Series object s2:")
print(s2)

(iii) import pandas as pd
import numpy as np
s = pd.Series (np.random.randint(6, size = 6),
index = ['p', 'q', 'r', 'n', 't', 'v'])
print(s)

Q 4. Consider the following DataFrame 'mdf'.

	Roll.No	Name	English	Hindi	Maths
0	1	Aditya	23	20	28
1	2	Balwant	18	1	25
2	3	Chirag	27	23	30
3	4	Deepak	11	3	7
4	5	Eva	17	21	24

(i) Write Python statements for the DataFrame 'mdf':

(a) To display the records of the students having roll numbers 2 and 3.

(b) To increase the marks of subject Math by 4, for all students.

(ii) Write Python statement to display the Rollno and Name of all students who secured less than 10 marks in Maths.

Or

(Option for Part B only)

Write Python statement to display the total marks *i.e.*, sum of marks secured in English, Hindi and Maths for all students. *[CBSE 2023]*

Ans. (I) (a) mdf . loc [1 : 2]

(b) mdf . loc [:, 'Maths'] = mdf . loc [:, 'Maths'] + 4

(II) mdf . loc ['Marks'<10, ['Rollno', 'Name']]

Or

mdf ['Total marks'] = sum(mdf ['English'], mdf['Hindi'], mdf['Maths'])
print (mdf)

Q 5. Write a program in Python Pandas to create the following DataFrame batsman from a Dictionary:

B_NO	Name	Score1	Score2
1.	Sunil Pillai	90	80
2.	Gaurav Sharma	65	45
3.	Piyush Goel	70	90
4.	Kartik Thakur	80	76

Perform the following operations on the DataFrame:

(i) Add both the scores of a batsman and assign to column "Total".

(ii) Display the highest score in both Score1 and Score2 of the DataFrame.

(iii) Display the DataFrame. *[CBSE SQP 2020-21]*

Ans. import pandas as pd
d1={'B_NO':[1,2,3,4],
'Name':["Sunil Pillai","Gaurav Sharma",
"Piyush Goel","Kartik Thakur"],
'Score1':[90,65,70,80],
'Score2':[80,45,95,76] }
df=pd.DataFrame(d1)
print(df)
df['Total'] = df['Score1']+ df['Score2']

Alternative Answer:

Scheme
df['Total'] = sum(df['Score1'], df['Score2'])
print(df)
print("Maximum scores are :" ,
max(df['Score1']), max(df['Score2']))

Q 6. Ekam, a Data Analyst with a multinational brand has designed the DataFrame df that contains the four quarter's sales data of different stores as shown below:

	Store	Qtr1	Qtr2	Qtr3	Qtr4
0	Store1	300	240	450	230
1	Store2	350	340	403	210
2	Store3	250	180	145	160

Answer the following questions:

(i) Predict the output of the following python statement:

(a) print(df.size)

(b) print(df[1:3])

(ii) Delete the last row from the DataFrame.

(iii) Write Python statement to add a new column Total_Sales which is the addition of all the 4 quarter sales. *[CBSE SQP 2023-24]*

Ans. (i) The output of the python statements are:

(a) 15

(b)

	Store	Qtr1	Qtr2	Qtr3	Qtr4
1	Store2	350	340	403	210
2	Store3	250	180	145	160

(II) df = df . drop (2)

Or

df . drop (2, axis=0)

(III) df ['total'] = df ['Qtr1'] + df ['Qtr2'] + df ['Qtr3'] + df ['Qtr4']

Chapter Test

Multiple Choice Questions

Q 1. To delete a column from a DataFrame, you may use ..... statement.
a. remove b. del c. drop d. cancel

Q 2. To delete a row from a DataFrame, you may use ..... statement.
a. remove b. del c. drop d. cancel

Q 3. ..... is a popular data-science library of Python.
a. numpy b. pandas
c. Both a. and b. d. None of these

Q 4. A ..... is a Pandas data structure that represents a 1D array like object.
a. DataFrame b. vector
c. series d. All of these

Q 5. A ..... is a Pandas data structure that represents a 2D array like object.
a. DataFrame b. vector
c. series d. All of these

Fill in the Blanks

Q 6. The ..... function on series object returns only the count of non-Nan value in it.

Q 7. DataFrame is ..... mutable as well as ..... mutable.

Q 8. To access value using row labels you can use DF .....

Assertion & Reason Type Questions

Directions (Q. Nos. 9-11): *In the questions given below, there are two statements marked as Assertion (A) and Reason (R). Read the statements and choose the correct option.*

a. Both Assertion (A) and Reason (R) are true and Reason (R) is the correct explanation of Assertion (A).
b. Both Assertion (A) and Reason (R) are true, but Reason (R) is not the correct explanation of Assertion (A).
c. Assertion (A) is true, but Reason (R) is false.
d. Assertion (A) is false, but Reason (R) is true.

Q 9. Assertion (A): A Pandas series object can be thought of as a column or a row, essentially.
Reason (R): Both series and DataFrame are one-dimensional data structure objects.

Q 10. Assertion (A): While series is a one-dimensional data structure object, DataFrame is a multi-dimensional data structure object.
Reason (R): A series object is value mutable.

Q 11. Assertion (A): A series object is size mutable.
Reason (R): A DataFrame value mutable.

Case Study Based Questions

Q 12. Zeenat has created the following DataFrame, DataFrame1 to keep track of data Rollno, Name, Marks1 and Marks2 for various students of her class where row indexes are taken as the default values:

RollNo.	Name	Marks1	Marks2
1	Swapnil Sharma	30	50
2	Raj Batra	75	45
3	Bhooml Singh	85	95
4	Jay Gupta	90	95

(i) Which among the following option will give 90, 95 as output?
a. print(max(DataFrame1['Marks1','Marks2']))
b. print((DataFrame1.Marks1.max(),(DataFrame1.Marks2.max())))
c. print(max(DataFrame1['Marks1']))
d. print(max(DataFrame1['Marks2']))

(ii) She needs to know the marks scored by Rollno 2. Help her to identify the correct set of statement/s from the given options.
a. print(DataFrame1[DataFrame1['Rollno']= =2])
b. print(DataFrame1['Rollno']= =2)
c. print(DataFrame1[DataFrame1. Rollno = =2])
d. print(DataFrame1[DataFrame1['Rollno']])

(iii) Which of the following statement/s will delete the 3rd column?
A. del DataFrame1['Marks1']
B. DataFrame1.pop('Marks1')
C. drop DataFrame1['Marks1']
D. pop DataFrame1['Marks1']
Choose the correct option.
a. Both A. and B. b. Only B.
c. A. B. and C. d. A, B. and D.

(iv) Which of the following command will display the total number of elements in the DataFrame?
a. print(DataFrame1.shape)
b. print(DataFrame1.num)
c. print(DataFrame1.size)
d. print(DataFrame1.elements)

(v) Now she wants to add a new column Marks3 with relevant data. Help her choose the command to perform this task.
a. DataFrame1.column=[45,52,90,95]
b. DataFrame1 ['Marks3']= [45,52,90,95]
c. DataFrame1.loc['Marks3']= [45,52,90,95]
d. Both b. and c. are correct

Q 13. Pandas DataFrame is a two-dimensional size-mutable, potentially heterogeneous tabular data structure with labeled axes (rows and columns). A DataFrame is a two-dimensional data structure, i.e., data is aligned in a tabular fashion in rows and columns like a spreadsheet or SQL table or a dict of Series objects. Pandas DataFrame consists of three principle components, the data, rows and columns. In the real world, a Pandas DataFrame will be created by loading the datasets from existing storage, storage can be SQL Database, CSV file and Excel file. Pandas DataFrame can be created from the lists, dictionary and from a list of dictionary, etc.

(i) Write a program in Python to create series of vowels.

(ii) Write a program in Python to create series of given tuple: A = (11, 22, 33, 44, 55)

(iii) Write the output of the following:

```
import numpy as num
import pandas as pd
arr=num.array([1,7,21])
S1 = pd.Series(arr)
print(S1)
```

(iv) Write the output of the following:

```
import pandas as pd
S1 = pd.Series(range(100, 150, 10), index=[x for x in 'My name is Amit Gandhi". split ()])
print(S1)
```

(v) Write the output of the following:

```
import pandas as pd
S1 = pd.Series(12, index = [4, 6, 8])
print(S1)
```

Very Short Answer Type Questions

Q 14. What is the difference between 'lat' and 'at' with respect to a DataFrame?

Q 15. How would you delete columns from a DataFrame?

Q 16. How would you delete rows from a DataFrame?

Q 17. Which function would you use to rename the index/ column names in a DataFrame?

Short Answer Type-I Questions

Q 18. What is a DataFrame and how is it different from a 2-D array?

Q 19. Write a program to create a DataFrame from a 2-D list. Specify own index labels.

Short Answer Type-II Questions

Q 20. Given a DataFrame mdf as shown below:

	A	B	C
0	1	2	3
1	4	5	6

Find out the errors in following statements.

(i) mdf.drop(["Total", "Order"], axis = 1)

(ii) mdf.drop (["A", "D"])

(iii) mdf.drop(["A", "D"], axis = 1) *[CBSE SQP 2019-20]*

Q 21. Why is the following code not renaming the index and columns even when code is executing without any error, considering that the saleDf DataFrame is as shown on the right?

>>> saleDF

	Target	Sales
zoneA	56000	58000
zoneB	70000	68000
zoneC	75000	78000
zoneD	60000	61000

The code:

```
saleDf.rename(index = {'zoneC' : 'Central', 'zoneD' : Dakshin'},\columns = {'Target' : 'Targeted', 'Sales' : 'Achieved'} )
print(saleDf)
```

What output would be produced by the above code and what is the problem with the code?

Long Answer Type Questions

Q 22. Answer the following questions:

(i) Which property of series returns the number of elements in the series?

(ii) Give an example of creating series from NumPy array.

(iii) Which property of Series help of check whether a Series is empty or not? Explain with example.

(iv) Fill in the blanks in the given code:

```
import pandas as pd
..... = ..... Series ([1, 2, 3, 4, 5])
print(S1)
```

(v) Fill in the blank of given code, if the output is 71.

```
import pandas as pd
S1 = pd.Series([10, 20, 30, 40, 71, 50])
print(S1[.....])
```

Q 23. Answer the following questions:

(i) Write the output of the following:

```
import pandas as pd
S1 = pd.Series(range(1, 20, 3))
print(S1)
```

(ii) Write the output of the following:

```
import pandas as pd
D = { }
for i in range(25, 95, 10):
D[i]=i+10
S1 = pd.Series(D)
print(S1)
```

(iii) Complete the code to get the required output:

```
import ..... as pd
..... = pd.Series([31, 28, 31], index = ["Jan", "Feb", "Mar"] )
print(S1[".....])
```

Output:

28

(iv) Write the output of the following code:

```
import pandas as pd
S1 = pd.Series([31, 28, 31, 30, 31], index = ["Jan", "Feb", "Mar", "Apr", "May"] )
print(S1["Jun"])
```

Or

Write the output of the following code:

```
import pandas as pd
S1 = pd.Series([31, 28, 31, 30, 31], index = ["Jan", "Feb", "Mar", "Apr", "May"] )
print(S1["May"])
```