Worksheet-1&2 (2024-2025) (ANSWERS)

Name of the Student : **Sub : Informatics Practices (065)**

Class: XII **Date of Issue : 30/03/2024**

Topic: Unit 1 – Pandas (Series & DataFrame) **Date of Submission:**

I. Multiple Choice Questions

1. Which of the following statement will import pandas library?

(a) Import pandas as pd (b) import Pandas as py

(c) import pandas as pd (d) import panda as pd

2. 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.

3. 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.

4. To get the number of dimensions of a Series object, _____ attribute is displayed.

(a) index (b) size (c) itemsize (d) ndim

5. 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

6. To get the number of elements in a Series object, _____ attribute may be used.

(a) index (b) size (c) itemsize (d) ndim

7. To get the number of bytes of the Series data, _____ attribute is displayed.

(a) hasnans (b) nbytes (c) ndim (d) dbyte

8. To check if the Series object contains NaN values, _____ attribute is displayed.

(a) hasnans (b) nbytes (c) ndim (d) dbyte

9. To display third element of a Series object S, you will write _____

(a) S[:3] (b) S[2] (c) S[3] (d) S[:2]

10. 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

11. To display last five rows of a Series object S, you may write _____

(a) head( ) (b) head(5) (c) tail( ) (d) Tail(5)

12. Which of the following statement is wrong?

(a) We 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.

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

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

(a) Value Error (b) Syntax Error (c) Name Error (d) Logical Error

14. To display last five rows of a series object 'S', you may write:

(a) S.Head( ) (b) S.Tail(5) (c) S.Head(5) (d) S.tail( )

15. Missing data in Pandas object is represented through:

(a) Null (b) None (c) Missing (d) NaN

16. 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))

Answers

1.c **2.**c **3.**b **4.** d **5.**c **6.**b **7.**b **8.**a **9.**b **10.**a **11.**c,d **12.**a **13.**a **14.**d **15.**d **16.**a

13. ValueError: Length of values (4) does not match length of index (3)

II. (a) Create the following Series Object "SCapital".

```
Hyd          Telangana
Chennai      Tamilnadu
Mumbai     Maharastra
dtype: object
```

Ans:

SCapital=pd.Series(['Telangana','Tamilnadu','Maharastra'],index=['Hyd','Chennai','Mumbai'])

(b) Create a Series "MyWord" with the individual characters of the word "Welcome"

```
0    W
1    e
2    l
3    c
4    o
5    m
6    e
dtype: object
```

Ans: MyWord=pd.Series(["W","e","l","c","o","m","e"])

(c) Consider the following Series "Market". Write the outputs:

```
45     Carrot
20     Potato
25      Onion
60     Chilly
dtype: object
```

(i) Market[1:4]

```
20     Potato
25      Onion
60     Chilly
dtype: object
```

(ii) Market[0:4:2]

```
45     Carrot
25      Onion
dtype: object
```

(iii) Market[3:1:-1]

```
60     Chilly
25      Onion
dtype: object
```

(d) Create a Series Object "FullMarks" with the following specifications.
Index should be multiples of 3 between 10 to 20, use range function.
Give all the values as 500.

Ans:

FullMarks=pd.Series(500, index=range(12,20,3))

```
12    500
15    500
18    500
dtype: int64
```

(e)Create a series by using following ndarrays.

(i) Use arange( ) function to get values from 10 to 20, with each difference 2.5

(ii) Use linspace( ) function to get values from 10 to 20. Total values should be 5

(iii)Use tile( ) function to repeat [2,3,4] for 2 times.

i)

```
import pandas as pd
import numpy as np
One=np.arange(10,20,2.5)
First=pd.Series(One)
```

```
0    10.0
1    12.5
2    15.0
3    17.5
dtype: float64
```

(ii)

```
import pandas as pd
import numpy as np
Two=np.linspace(10,20,5)
Second=pd.Series(Two)
```

```
0    10.0
1    12.5
2    15.0
3    17.5
4    20.0
dtype: float64
```

(iii)

```
import pandas as pd
import numpy as np
Three=np.tile([2,3,4],2)
Third=pd.Series(Three)
```

```
0    2
1    3
2    4
3    2
4    3
5    4
dtype: int32
```

III. Mention attribute names and its values for the following Series "S"

```
45    Carrot
20    Potato
25     Onion
60    Chilly
dtype: object
```

(i) S.index

Int64Index([45, 20, 25, 60], dtype='int64')

(ii) S.values

array(['Carrot', 'Potato', 'Onion', 'Chilly'], dtype=object)

(iii) S.dtype

dtype('O')

(iv) S.shape

(4,)

(v) S.nbytes

32

(vi) S.ndim

1

(vii) S.size

4

(viii) S.hasnans

False

(ix) S.empty

False

(x) S.name="Vegetables"

S.name

'Vegetables'

S

45 Carrot
20 Potato
25 Onion
60 Chilly
Name: Vegetables, dtype: object

(xi) S.index.name="Vegnames"

S.index.name

'Vegnames'

S

Vegnames

45 Carrot
20 Potato
25 Onion
60 Chilly
Name: Vegetables, dtype: object

Worksheet 2 – DataFrames Concept

I. Multiple Choice Questions

(1) Which of the following statement is wrong in context of DataFrame?
(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.

(2) When we create a DataFrame from a list of Dictionaries the columns labels are formed by the
(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.

(3) 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.

(4) 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.

(5) Which of the following can be used to specify the data while creating a DataFrame?
(a) Series (b) List of Dictionaries (c) Structured ndarray **(d)** All of these.

(6) The axis 0 identifies a dataframe's _____
(a) rows (b) columns (c) values (d) datatypes

(7) The axis 1 identifies a dataframe's ____
(a) rows **(b)** columns (c) values (d) datatypes

(8) To get the number of elements in a dataframe, _____ attribute may be used.
(a) size (b) shape (c) values (d) ndim

(9) To get NumPy representation of a dataframe, _____ attribute may be used.
(a) size (b) shape **(c)** values (d) ndim

(10) To get a number representing number of axes in a dataframe, _____ attribute may be used.
(a) size (b) shape (c) values **(d)** ndim

(11) Which attribute is not used with DataFrame?
(a) Size **(b)** Type (c) Empty (d) Columns

(12) To get the transpose of a dataframe D1, you can write _____
(a) D1.T (b) D1.Transpose (c) D1.Swap (d) All of these

(13) To extract row/column from a database, _____ function may be used.
(a) row( ) (b) column( ) **(c)** loc( ) (d) All of these

(14) 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]

(15) To change the 5th column's value at 3rd row as 35 in dataframe DF, you can wirte _____
(a) DF[4,6]=35 (b) DF[3,5]=35 (c) DF.iat[4,6]=35 **(d)** DF.iat[3,5]=35

(16) Which among the following options can be used to create a DataFrame in Pandas?
(a) A scalar value (b) An ndarray (c) A python dict **(d)** All of these

(17) Identify the correct statement:
(a) The standard marker for missing data in Pandas is NaN.

(b) Series act in a way similar to that of an array.
(c) Both (a) and (b)
(d) None of the above.

(18) Identify the correct option to select first four rows and second to fourth columns from a DataFrame : "Data" :

(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])

(19) To delete a column from a DataFrame, you may use _____ statement.

(a) remove **(b)** del (c) drop (d) cancel

(20) To delete a row from a DataFrame, you may use _____statement.

(a) remove (b) del **(c)** drop (d) cancel

(21) 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:

(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])

Answers

1. c 2.a 3.b 4.c 5.d 6.a 7.b 8.a 9.c 10.d 11.b 12.a
13.c 14.c 15.d 16.d 17.c 18.c 19.b 20.c 21.b

II. Create the following DataFrames:

	Teacher	Subject	Exp
One	Akash	Maths	3
Two	Suresh	English	15
Three	Mohan	IP	5

	Private	Aided	Govt	ZP
AP	100	75	125	89
TN	98	92	130	92
TS	110	85	110	91

	RNo	SName	Marks
First	51	Lahari	55
Second	52	Chanakya	62
Third	53	Harish	52
Fourth	54	Neha	75

(i) DFTeacher (ii) DFSchool (iii) DF Student

(i Ans)

```
import pandas as pd
D={"Teacher":["Akash","Suresh","Mohan"],"Subject":["Maths","English","IP"],"Exp":[3,15,5]}
DFTeacher=pd.DataFrame(D,index=["One","Two","Three"])
DFTeacher
```

III. 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:

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

Answer:

```
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)
```

IV. Mention attribute names and its values for the following DataFrame "DF"

	P1	P2
Mon	IP	Maths
Tue	Che	Phy
Wed	IP	Phy

(i) DF.index
Index(['Mon', 'Tue', 'Wed'], dtype='object')
(ii) DF.columns
Index(['P1', 'P2'], dtype='object')
(iii) DF.axes
[Index(['Mon', 'Tue', 'Wed'], dtype='object'), Index(['P1', 'P2'], dtype='object')]
(iv) DF.dtypes
P1 object
P2 object
dtype: object
(v) DF.values #numpy representation
array([['IP', 'Maths'],
['Che', 'Phy'],
['IP', 'Phy']], dtype=object)
(vi) DF.shape
(3, 2)
(vii) DF.size
6
(viii) DF.empty
False
(ix) DF.ndim
2
(x) DF.T

	Mon	Tue	Wed
P1	IP	Che	IP
P2	Maths	Phy	Phy

(A) DF.count()
P1 3
P2 3
dtype: int64
(B) DF.shape[0] #No. of rows
3
(c) DF.shape[1] # No.of columns
2

V. Question on DataFrame Operations.

	P1	P2	P3
Mon	IP	Maths	Eng
Tue	Che	Phy	Acc
Wed	IP	Phy	Maths

(1) Create the above DataFrame "TT"

```
import pandas as pd
D={"P1":["IP","Che","IP"],"P2":["Maths","Phy","Phy"],"P3":["Eng","Acc","Maths"]}
TT=pd.DataFrame(D,index=["Mon","Tue","Wed"])
print(TT)
```

(2) Display only the details of First 2 Rows.

	P1	P2	P3
Mon	IP	Maths	Eng
Tue	Che	Phy	Acc

```
TT.loc["Mon":"Tue"]
TT.head(2)
TT.iloc[0:2]
```

(3) Display the details of rows Wed and Mon (in the order Wed, Mon)

	P1	P2	P3
Wed	IP	Phy	Maths
Mon	IP	Maths	Eng

```
TT.loc[['Wed','Mon']]
TT.iloc[[2,0]]
```

(4) Display the details of Columns “P1” and “P3”

	P1	P3
Mon	IP	Eng
Tue	Che	Acc
Wed	IP	Maths

```
TT.loc[:,["P1","P3"]]
TT.iloc[:,[0,2]]
```

(5) Display the following output.

	P3	P1	P2
Tue	Acc	Che	Phy
Mon	Eng	IP	Maths

```
TT.loc[[“Tue”,”Mon”],[“P3”,”P1”,”P2”]]
TT.iloc[[1,0]]
```

(6) Display the following Output.

	P3	P2
Mon	Eng	Maths
Tue	Acc	Phy
Wed	Maths	Phy

```
TT.loc[:,["P3","P2"]]
TT.loc["Mon":"Wed",["P3","P2"]]
TT.iloc[:,[2,1]]
TT.iloc[0:3,[2,1]]
```

(7) Change Tuesday’s 3rd Period to “Eng”

	P1	P2	P3
Mon	IP	Maths	Eng
Tue	Che	Phy	Eng
Wed	IP	Phy	Maths

```
TT.at["Tue","P3"]="Eng"
TT.iat[1,2]="Eng"
```

(8) Add a new column "P4" with values Acc,BS,Maths

	P1	P2	P3	P4
Mon	IP	Maths	Eng	Acc
Tue	Che	Phy	Eng	BS
Wed	IP	Phy	Maths	Maths

TT["P4"]="Acc","BS","Maths"

(9) Add a row with index Thu, values Bio,Che,Eng,Acc

	P1	P2	P3	P4
Mon	IP	Maths	Eng	Acc
Tue	Che	Phy	Eng	BS
Wed	IP	Phy	Maths	Maths
Thu	Bio	Che	Eng	Acc

TT.loc["Thu"]="Bio","Che","Eng","Acc"

(10) Rename Mon to Monday

	P1	P2	P3	P4
Monday	IP	Maths	Eng	Acc
Tue	Che	Phy	Eng	BS
Wed	IP	Phy	Maths	Maths
Thu	Bio	Che	Eng	Acc

TT.rename(index={"Mon":"Monday"},inplace=True)

(11) Rename P2 to Period2

	P1	Period2	P3	P4
Monday	IP	Maths	Eng	Acc
Tue	Che	Phy	Eng	BS
Wed	IP	Phy	Maths	Maths
Thu	Bio	Che	Eng	Acc

TT.rename(columns={"P2":"Period2"},inplace=True)

(12) Delete the Column P3
TT.drop("P3",axis=1,inplace=True)

	P1	Period2	P4
Monday	IP	Maths	Acc
Tue	Che	Phy	BS
Wed	IP	Phy	Maths
Thu	Bio	Che	Acc

(13) Delete Row with Index "Tue"

	P1	Period2	P4
Monday	IP	Maths	Acc
Wed	IP	Phy	Maths
Thu	Bio	Che	Acc

TT=TT.drop("Tue")
TT.drop("Tue",inplace=True)
TT.drop("Tue",axis=0,inplace=True)

(14) Modify the Entire P4 values to “Acc”

	P1	Period2	P4
Monday	IP	Maths	Acc
Wed	IP	Phy	Acc
Thu	Bio	Che	Acc

TT.loc[:,"P4"]="Games"
TT.iloc[:,1:3]=”Games”

(15) Delete complete DataFrame
del TT

Differences between Series and List, Series and DataFrame, Series and NDArrays.

Series	Lists
It is essentially a 1D data structure	It can be 1D and even multi-dimensional with nested lists in it.
It can have numeric indexes as well as labels	It can take numeric indexes only
It supports explicit indexing, i.e., we can programmatically choose, provide and change indexes in terms of numbers or labels	It does not support explicit indexing; only supports implicit indexing whereby the indexes are implicitly given 0 onwards in forward indexing and -1 onwards in backward indexing.
Indexes can be duplicate	Indexes cannot be duplicate
Homogeneous elements. Series objects store elements of same data type (values may be different but their datatype is the same for each element)	Heterogeneous elements: Lists can store elements of different data types.

ndarrays	Series Objects
We can perform vectorised operations only if the shapes of two ndarrays match, otherwise it returns an error (ValueError)	In case of vectorised operations, the data of two Series objects is aligned as per matching indexes and operation is performed on them and for non-matching indexes, NaN is returned.
The indexes are always numeric starting from 0 onwards	Series objects can have any type of indexes, including numbers (not necessarily starting from 0), letters, labels, strings, etc.

Property	Series	DataFrame
Dimensions	**1 Dimensional**	**2-Dimensional**
Type of Data	**Homegeneous**, i.e., all the elements must be of same 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	**Value mutable**, i.e., their elements value can change
	Size-immutable, i.e., size of a Series object, once created, cannot change. If we want to add/drop an element, internally a new Series object will be created	**Size-mutable**,i.e., size of a Dataframe object, once created, can change in place. That is, you can add/drop elements in an existing dataframe object.

Pandas Series	NumPy Arrays
In series we can define our own labeled index to access elements of an array. These can be numbers or letters.	NumPy arrays are accessed by their integer position using numbers only.
The elements can be indexed in descending order also.	The indexing starts with zero for the first element and the index is fixed.
If two series are not aligned, NaN or missing values are generated.	There is no concept of NaN values and if there are no matching values in arrays, alignment fails.
Series require more memory.	NumPy occupies lesser memory.