

**SUMMER BREAK HOLIDAY HOMEWORK 2019-20**

**CLASS-XII SUBJECT-INFORMATICS PRACTICES**

**1. What is pandas series?**

**Ans.** Series is a one-dimensional labeled array capable of holding data of any type (integer, string, float, python objects, etc.).

**2. In pandas S is a series with following result:**

**S = pd . Series ( [ 5 , 10 , 15 , 20 , 25 ] )**

**The Series object is automatically indexed as 0, 1, 2, 3, 4. Write a statement to assign the series as a, b, c, d, e explicitly.**

**Ans.** S . index = [ 'a' , 'b' , 'c' , 'd' , 'e' ]

**3. In pandas S is a series with following result:**

**S = pd . Series ( [ 5 , 10 , 15 , 20 , 25 ] )**

**Find the output of the following:**

**(i) S [ [ 1 , 2 ] ]**

**(ii) S [ 1 : 3 ]**

**4. What is DataFrame?**

**Ans.** DataFrame is a two-dimensional array with heterogeneous data, usually represented in the tabular format. The data is represented in rows and columns. Each column represents an attribute and each row represents a person.

**5. A dictionary s\_marks contains the following data:**

**s\_marks = { 'name' : [ 'Rashmi' , 'Harsh' , 'Ganesh' , 'Priya' , 'Vivek' ] , 'Grade' : [ 'A1' , 'A2' , 'B1' , 'A1' , 'B2' ] }**

**Write a statement to create DataFrame called df. Assume that pandas has been imported as pd.**

**Ans.** df = pd.DataFrame(s\_marks)

**6. What is the purpose axis option in pandas concat() function?**

**Ans.** By default axis = 0 thus the new DataFrame will be added row-wise. If a column is not present then

in one of the DataFrames it creates NaNs. To join column wise we set axis = 1

**7. A dictionary Grade contains the following data:**

**Grade = { 'Name' : [ 'Rashmi' , 'Harsh' , 'Ganesh' , 'Priya' , 'Vivek' , 'Anita' , 'Karthik' ] , 'Grade' : [ 'A1' , 'A2' , 'B1' , 'A1' , 'B2' , 'A2' , 'A1' ] }**

**Write statements for the following:**

**(a) Create a DataFrame called Gr.**

**(b) Find the output of Gr.iloc[0:5] and Gr[0:5]**

**(c) Add a column called Percentage with following data:**

**[92, 89, None, 95, 68, None, 93]**

**(d) Rearrange the columns as Name, Percentage and Grade.**

**(e) Add following 3 rows with following data: (Note: None is a null value).**

**Ishan 86 B1**

**Amrita 97 A1**

**None None None**

**(f) Drop the column (i.e., Grade) by name.**

**(g) Delete the 3rd and 5th rows rows.**

**(h) What does the following will do?**

**(i) Gr.drop(0,axis = 0)**

**(ii) Gr.drop(0,axis = "index")**

**(iii) Gr.drop([0,1,2,3],axis = 0)**

**Ans. (a) Gr = pd.DataFrame(Grade)**

(b) Output for both the commands are same:

Grade Name

0 A1 Rashmi

1 A2 Harsh

2 B1 Ganesh

3 A1 Priya

4 B2 Vivek

(c) `Gr["Percentage"] = [92, 89, None, 95, 68, None, 93]`

(d) `Gr = Gr[['Name', 'Percentage', 'Grade']]`

(e) `TGr = pd.DataFrame({'Name': ['Ishan', 'Amrita', None],`

`'Percentage': [86, 97, None],`

`'Grade': ['B1', 'A1', None]},`

`columns=['Name', 'Percentage', 'Grade'])`

`Gr = Gr.append(TGr, ignore_index=True)`

(f) `Gr.drop('Grade',axis=1)`

(g) `Gr.drop([2, 4])`

(h) (i) First row will be dropped.

(ii) First row will be dropped.

(iii) First four rows will be dropped.

## REVIEW QUESTIONS

1. If a list of array contains the following elements:

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

Write the statements to create a series from using NumPy array.

2. What will be the output of the following:

`D = {'a': 0., 'b': 1., 'c': 2.}`

`S = pd.Series(D,index=['b','c','d','a'])`

`print(S)`

3. A dictionary contains first 10 states with their Per Capita Income as follows:

`d = {'Goa': 224138, 'Delhi': 212219, 'Sikkim': 176491, 'Chandigarh': None, 'Puducherry': 143677,`

`'Haryana': 133427, 'Maharashtra': None, 'Tamil Nadu': 112664, 'A. & N. Islands': 107418, 'Gujarat': 106831}`

Answer the following:

(a) Create a series called Income.

(b) List the states with the income below 130000.

(c) What will be the output of the commands?

`Less_Than = income < 130000`

`print(Less_Than)`

4. Suppose we make a DataFrame as

`df = pd.DataFrame({'Book_ID': ['C0001', 'F0001', 'T0001', 'T0002', 'F0002'],`

`'Book_name': ['Fast Cook', 'The Tears', 'My First C++', 'C++ Brainworks', 'Thunderbolts'],`

`'Author_Name': ['Lata Kapoor', 'William Hopkins', 'Brain & Brooke', 'A.W. Rossaine', 'Anna Roberts'],`

`'Price': [540, 450, 670, 548, 750]},`

`columns = ['Book_ID', 'Book_name', 'Author_Name', 'Price'])`

Note that pandas has been imported as pd.

What are the purposes of the following commands?

(i) `print(df.loc[:, 'Book_Name'])`

(ii) `print(df['Book_Name'])`

(iii) `print(df[['Book_Name']])`

(iv) `print(df[['Book_Name', 'Price']])`

(v) `print(df[0:3])`

(vi) print(df[2:4])

(vii) print(df.iloc[2])

## **CHAPTER-2**

1. Why pivot() method is more restrictive than pivot\_table method in pandas?

2. State one difference between stacking and unstacking.

3. A dataframe contains following data:

**Name Qualification Experience**

0 Ms. Mittal Masters 8

1 Minu Arora Graduate 11

2 Sharmila Kaur Post Graduate 7

3 Sangeeta Vats Masters 9

4 Ramesh Kumar Graduate 6

5 Jatin Ghosh Post Graduate 8

6 Yash Sharma Masters 10

Write the command for the following (assume that the dataframe name is dfT):

(a) Find the average experience for each qualification.

(b) Find the total experience for each qualification.

(c) Find the average experience for each qualification and name.

(d) What will be the following will display:

```
pva = pd.pivot_table(dfT, index = 'Qualification', columns = 'Qualification',  
values = 'Experience', aggfunc = np.sum)
```

```
pva.stack()
```

4. A sample dataset is given with different columns as given below:

**Item\_ID ItemName Manufacturer Price CustomerName City**

PC01 Personal Computer HCL India 42000 N Roy Delhi

LC05 Laptop HP USA 55000 H Singh Mumbai

PC03 Personal Computer Dell USA 32000 R Pandey Delhi

PC06 Personal Computer Zenith USA 37000 C Sharma Chennai

LC03 Laptop Dell USA 57000 K Agarwal Bengalur

AL03 Monitor HP USA 9800 S C Gandhi Delhi

CS03 Hard Disk Dell USA 5400 B S Arora Mumbai

PR06 Motherboard Intel USA 17500 A K Rawat Delhi

BA03 UPS Microtek India 4300 C K Naidu Chennai

MC01 Monitor HCL India 6800 P N Ghosh Bengalur

Write the command for the following (assume that the dataframe name is dfA):

(a) Create a city wise customer table.

(b) Create a pivot table for manufacturer wise item names and their price.

(c) Arrange the DataFrame in ascending order of CustomerName.

(d) Arrange the DataFrame in ascending order of City and Price.