Python Assignment – Pandas

1. Define and create a Dataframe using Pandas.

2. How can we calculate the standard deviation from the series.

3. Create a Dataframe using List given below:

   lis=[‘Python’,’Numpy’,’Pandas’]

4. Create a Series from dictionary given below:

   info = {'x' : 0., 'y' : 1., 'z' : 2.}

5. How will you create an empty Dataframe in Pandas?

6. i) Add a column to that pandas Dataframe that you created in Q1
   ii) Change index of that Dataframe.

7. Get the following from this dataframe:

   ```
   import random
   A = [ random.randint(0,100) for i in range(10) ]
   B = [ random.randint(0,100) for i in range(10) ]
   df = pd.DataFrame({ 'field_A': A, 'field_B': B })
   ```

   i) mean
   ii) median
   iii) mode
   iv) 75th percentile
   v) 90th percentile
   vi) 99th percentile

8. How do you count how many unique rows a DataFrame has (i.e. ignore all rows that are duplicates)?

9. How to keep only top 2 most frequent values as it is and replace everything else as ‘Other’? Take the following as input.

```
np.random.RandomState(100)
ser = pd.Series(np.random.randint(1, 5, [12]))
```

10. A DataFrame has a column of groups 'grps' and and column of integer values 'vals':

```
df = pd.DataFrame({'grps':
list('aaabbcaabcccbbc'),
'vals': [12, 345, 3, 1, 45, 14 , 4, 52, 54, 23, 235,
21, 57, 3, 87]})
```

For each *group*, find the sum of the three greatest values. You should end up with the answer as follows:

```
grps
a   409
b   156
c   345
```

*Create a dataframe for the following to answer the next few questions

```
        data = {'animal': ['cat', 'cat', 'snake', 'dog', 'dog', 'cat',
    'snake', 'cat', 'dog', 'dog'],
        'age': [2.5, 3, 0.5, np.nan, 5, 2, 4.5, np.nan, 7, 3],
        'visits': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],
        'priority': ['yes', 'yes', 'no', 'yes', 'no', 'no', 'no', 'yes', 'no',
    'no']}

        labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']
```

11. Display a summary of the basic information about this DataFrame and its data (*hint: there is a single method that can be called on the DataFrame*).

12. Select the rows where the age is between 2 and 4 (inclusive).

13. Select the data in rows [3, 4, 8] *and* in columns ['animal', 'age'].

14. Select the rows where the age is missing, i.e. it is NaN.

15. Select the rows where the animal is a cat *and* the age is less than 3.

16. Change the age in row 'f' to 1.5.

17. In the 'animal' column, change the 'snake' entries to 'python'.

18. Count the number of each type of animal in df.

19. The 'priority' column contains the values 'yes' and 'no'. Replace this column with a column of boolean values: 'yes' should be True and 'no' should be False.

20. For each animal type and each number of visits, find the mean age. In other words, each row is an animal, each column is a number of visits and the values are the mean ages (*hint: use a pivot table*).

Following is the link for MCQ questions:
https://forms.gle/571iAjCoCKgwcTDR6