

PYTHON PANDA ASSIGNMENT

With Solutions

1. Write a Pandas program to create and display a one-dimensional array-like object containing an array of data.

Sample Output:

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

2. Write a Pandas program to get the Array values element-wise.

Sample data: {'X':[78,85,96,80,86], 'Y':[84,94,89,83,86], 'Z':[86,97,96,72,83]}

3. Write a Pandas program to create and display a DataFrame from a specified dictionary data which has the index labels.

Sample DataFrame:

```
exam_data = {'name': ['Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'],
'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],
'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],
'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']}

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

4.

Write a Pandas program to compare the elements of the two Pandas Series.

Sample Series: [2, 4, 6, 8, 10], [1, 3, 5, 7, 9]

(>,<,<=)

5.

Write a Pandas program to count the number of rows and columns of a DataFrame.

Sample Python dictionary data and list labels:

```
exam_data = {'name': ['Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'],
'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],
'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],
'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']}
labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']
```

Expected Output:

Number of Rows: 10

Number of Columns: 4

6. Write a Pandas program to calculate the mean score for each different student in data frame.

USING ABOVE DATA

7.

Write a Pandas program to rename columns of a given DataFrame

Sample data:

Original DataFrame

```
col1 col2 col3
```

```
0 1 4 7
```

```
1 2 5 8
```

```
2 3 6 9
```

New DataFrame after renaming columns:

```
Column1 Column2 Column3
```

```
0 1 4 7
```

```
1 2 5 8
```

```
2 3 6 9
```

8.

Write a Pandas program to select rows from a given DataFrame based on values in some columns.

Sample data:

Original DataFrame

```
col1 col2 col3
```

```
0 1 4 7
```

1 4 5 8

2 3 6 9

3 4 7 0

4 5 8 1

Rows for column1 value == 4

col1 col2 col3

1 4 5 8

3 4 7 0

9.

Write a Pandas program to convert a dictionary to a Pandas series.

Sample dictionary: d1 = {'a': 100, 'b': 200, 'c':300, 'd':400, 'e':800}

10.

Write a Pandas program to add one row in an existing DataFrame.

Sample data:

Original DataFrame

col1 col2 col3

0 1 4 7

1 4 5 8

2 3 6 9

3 4 7 0

4 5 8 1

After add one row:

col1 col2 col3

0 1 4 7

1 4 5 8

2 3 6 9

3 4 7 0

4 5 8 1

5 10 11 12

11.

Write a Pandas program to convert a given Series to an array.

Sample output:

```
Original Data Series:
0      100
1      200
2    python
3    300.12
4      400
dtype: object
Series to an array
['100' '200' 'python' '300.12' '400']
```

12.

Write a Pandas program to create the mean and standard deviation of the data of a given Series.

Sample Output:

Original Data Series:

```
0 1
1 2
2 3
3 4
4 5
5 6
6 7
7 8
8 9
9 5
10 3
```

dtype: int64

Mean of the said Data Series:

4.81818181818

Standard deviation of the said Data Series:

2.52262489555

13.

Write a Pandas program to find the positions of numbers that are multiples of 5 of a given series.

Sample Output:

Original Series:

```

0 1
1 9
2 8
3 6
4 9
5 7
6 1
7 1
8 1
dtype: int64
Positions of numbers that are multiples of 5:
[]
14.

```

Write a Pandas program to calculate the number of characters in each word in a given series.

Sample Output:

Original Series:

```

0 Php
1 Python
2 Java
3 C#
dtype: object

```

Number of characters in each word in the said series:

```

0 3
1 6
2 4
3 2
dtype: int64

```

SOLUTIONS

1. `import pandas as pd`

```

ds = pd.Series([2, 4, 6, 8, 10])
print(ds)

```

2.

```

import pandas as pd
df = pd.DataFrame({'X':[78,85,96,80,86],
'Y':[84,94,89,83,86], 'Z':[86,97,96,72,83]});
print(df)

```

3.

```
import pandas as pd
import numpy as np

exam_data = {'name': ['Anastasia', 'Dima', 'Katherine', 'James',
                      'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'],
             'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],
             'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],
             'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes',
                        'no', 'no', 'yes']}
labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']

df = pd.DataFrame(exam_data , index=labels)
print(df)
```

4.

```
import pandas as pd
ds1 = pd.Series([2, 4, 6, 8, 10])
ds2 = pd.Series([1, 3, 5, 7, 10])
print("Series1:")
print(ds1)
print("Series2:")
print(ds2)
print("Compare the elements of the said Series:")
print("Equals:")
print(ds1 == ds2)
print("Greater than:")
print(ds1 > ds2)
print("Less than:")
print(ds1 < ds2)
```

5.

```
import pandas as pd
import numpy as np
exam_data = {'name': ['Anastasia', 'Dima', 'Katherine', 'James',
                      'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'],
             'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],
             'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],
             'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes',
                        'no', 'no', 'yes']}
labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']
df = pd.DataFrame(exam_data , index=labels)

total_rows=len(df.axes[0])
```

```
total_cols=len(df.axes[1])
print("Number of Rows: "+str(total_rows))
print("Number of Columns: "+str(total_cols))
```

```
6. df = pd.DataFrame(exam_data , index=labels)

print("\nMean score for each different student in data frame:")

print(df['score'].mean())
```

7. `import pandas as pd`

```
d = {'col1': [1, 2, 3], 'col2': [4, 5, 6], 'col3': [7, 8, 9]}
df = pd.DataFrame(data=d)
print("Original DataFrame")
print(df)
df.columns = ['Column1', 'Column2', 'Column3']
df = df.rename(columns={'col1': 'Column1', 'col2': 'Column2', 'col3':
'Column3'})
print("New DataFrame after renaming columns:")
print(df)
```

8.

```
import pandas as pd
import numpy as np
d = {'col1': [1, 4, 3, 4, 5], 'col2': [4, 5, 6, 7, 8], 'col3': [7, 8,
9, 0, 1]}
df = pd.DataFrame(data=d)
print("Original DataFrame")
print(df)
print('Rows for column1 value == 4')
print(df.loc[df['col1'] == 4])
```

9.

```
import pandas as pd
d1 = {'a': 100, 'b': 200, 'c':300, 'd':400, 'e':800}
print("Original dictionary:")
print(d1)
new_series = pd.Series(d1)
print("Converted series:")
print(new_series)
```

10.

```
import pandas as pd
import numpy as np
d = {'col1': [1, 4, 3, 4, 5], 'col2': [4, 5, 6, 7, 8], 'col3': [7, 8, 9, 0, 1]}
df = pd.DataFrame(data=d)
print("Original DataFrame")
print(df)
print('After add one row:')
df2 = {'col1': 10, 'col2': 11, 'col3': 12}
df = df.append(df2, ignore_index=True)
print(df)
```

11.

```
import pandas as pd
import numpy as np
s1 = pd.Series(['100', '200', 'python', '300.12', '400'])
print("Original Data Series:")
print(s1)
print("Series to an array")
a = np.array(s1.values.tolist())
print(a)
```

12.

```
import pandas as pd
s = pd.Series(data = [1,2,3,4,5,6,7,8,9,5,3])
print("Original Data Series:")
print(s)
print("Mean of the said Data Series:")
print(s.mean())
print("Standard deviation of the said Data Series:")
print(s.std())
```

13.

```
import pandas as pd
import numpy as np
num_series = pd.Series(np.random.randint(1, 10, 9))
print("Original Series:")
print(num_series)
result = np.argwhere(num_series % 5==0)
print("Positions of numbers that are multiples of 5:")
print(result)
```


14.

```
import pandas as pd
series1 = pd.Series(['Php', 'Python', 'Java', 'C#'])
print("Original Series:")
print(series1)
result = series1.map(lambda x: len(x))
print("\nNumber of characters in each word in the said series:")
print(result)
```

X-----X-----X