Assignment -1

Assignment Date	17 September 2022
Team ID	PNT2022TMID38845
Project Name	AI Based Discourse for Banking Industry
Student Name	Charulatha S
Student Roll Number	421219104003
Maximum Marks	2 Marks

Question-1.

Split this string

s = "Hi there Sam!"

Solution:

s.split(' ')

```
[2]  s = "Hi there Sam!"

[3]  s.split(' ')

     ['Hi', 'there', 'Sam!']
```

Question-2.

Use .format() to print the following string.

Output should be: The diameter of Earth is 12742 kilometers.

Solution:

```
planet = "Earth"
diameter = 12742
print( 'The diameter of {} is {} kilometers.' .format(planet,diameter) ) ;
```

```
[5]  planet = "Earth"
     diameter = 12742
```

```
print( 'The diameter of {} is {} kilometers.' .format(planet,diameter));
```

```
The diameter of Earth is 12742 kilometers.
```

Question-3.

In this nest dictionary grab the word "hello"

d = {'k1':[1,2,3,{'tricky':['oh','man','inception',{'target':[1,2,3,'hello']}]}]}

Solution:

d['k1'][3]['tricky'][3]['target'][3]

```
[7]  d = {'k1':[1,2,3,{'tricky':['oh','man','inception',{'target':[1,2,3,'hello']}]}]}
```

```
d['k1'][3]['tricky'][3]['target'][3]
```

```
'hello'
```

Question-4.

4.1 Create an array of 10 zeros?

Solution:

```
import numpy as np
array=np.zeros(10)
print("An array of 10 zeros:")
print(array)
```

```
[11]  import numpy as np
```

```
array=np.zeros(10)
print("An array of 10 zeros:")
print(array)
```

```
An array of 10 zeros:
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
```

4.2 Create an array of 10 fives?

Solution:

```
import numpy as np
array=np.ones(10)*5
print("An array of 10 fives:")
print(array)
```

```
[11] import numpy as np
```

```
array=np.ones(10)*5
print("An array of 10 fives:")
print(array)
```

```
An array of 10 fives:
[5. 5. 5. 5. 5. 5. 5. 5. 5. 5.]
```

Question-5.

Create an array of all the even integers from 20 to 35

Solution:

```
import numpy as np
array=np.arange(20,35,2)
print("Array of all the even integers from 20 to 35")
print(array)
```

```
import numpy as np
array=np.arange(20,35,2)
print("Array of all the even integers from 20 to 35")
print(array)
```

```
Array of all the even integers from 20 to 35
[20 22 24 26 28 30 32 34]
```

Question-6.

Create a 3x3 matrix with values ranging from 0 to 8

Solution:

```
import numpy as np
matrix = np.arange(0, 9).reshape(3,3)
matrix
```

```
import numpy as np
matrix =  np.arange(0, 9).reshape(3,3)
matrix
```

```
array([[0, 1, 2],
       [3, 4, 5],
       [6, 7, 8]])
```

Question-7.

Concatenate a and b

a = np.array([1, 2, 3]), b = np.array([4, 5, 6])

Solution:

```
import numpy as np
a = np.array([1, 2, 3])
b= np.array([4, 5, 6])
array = np.concatenate((a, b))
array
```

```
import numpy as np

a = np.array([1, 2, 3])

b= np.array([4, 5, 6])

array = np.concatenate((a, b))
array
```

```
array([1, 2, 3, 4, 5, 6])
```

Question-8.

Create a dataframe with 3 rows and 2 columns

Solution:

```
import pandas as pd
d = {'a': [1, 'A'],'b': [2, 'B'],'c': [3, 'C']}
f = pd.DataFrame(d)
f
```

```
import pandas as pd
```

```
d = {'a': [1, 'A'],'b': [2, 'B'],'c': [3, 'C']}
f = pd.DataFrame(d)
f
```

	a	b	c
0	1	2	3
1	A	B	C

Question-9.

Generate the series of dates from 1st Jan, 2023 to 10th Feb, 2023

Solution:

```
dates = pd.date_range("1/1/2023", "10/02/2023")
dates
```

```
dates = pd.date_range("1/1/2023", "10/02/2023")
dates
```

```
DatetimeIndex(['2023-01-01', '2023-01-02', '2023-01-03', '2023-01-04',
               '2023-01-05', '2023-01-06', '2023-01-07', '2023-01-08',
               '2023-01-09', '2023-01-10',
               ...
               '2023-09-23', '2023-09-24', '2023-09-25', '2023-09-26',
               '2023-09-27', '2023-09-28', '2023-09-29', '2023-09-30',
               '2023-10-01', '2023-10-02'],
              dtype='datetime64[ns]', length=275, freq='D')
```

Question-10.

Create 2D list to DataFrame

lists = [[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]]

Solution:

```
lists = [[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]]
df = pd.DataFrame(lists)
df
```

```
lists = [[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]]
```

```
[22] df = pd.DataFrame(lists)
     df
```

	0	1	2
0	1	aaa	22
1	2	bbb	25
2	3	ccc	24