Assignment -1

Python Programming

Assignment Date	09 November 2022
Student Name	Mr.S.Dhilip Kumar
Student Roll Number	E1194015
Maximum Marks	2 Marks

Question-1:

Split this string

s = "Hi there Sam!"

Solution:

```
print(s.split())
```

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

Question-2:

Use .format() to print the following string.

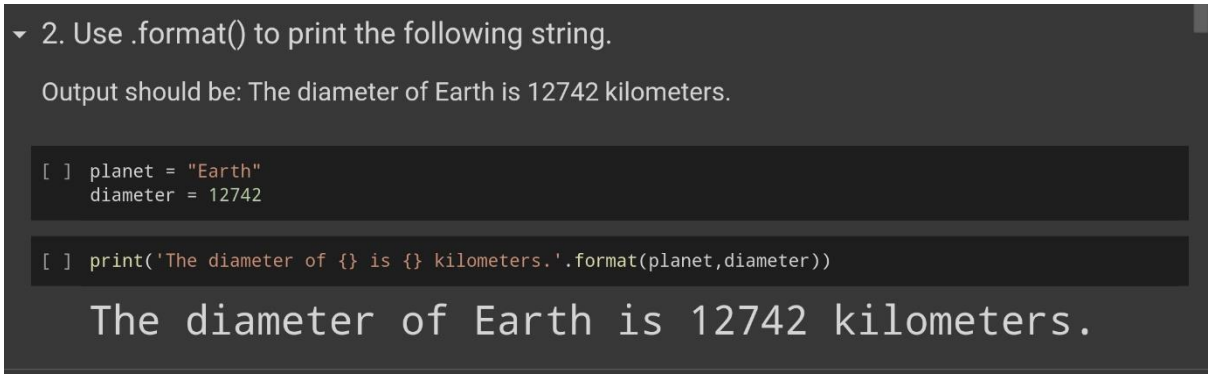
Output should be: The diameter of Earth is 12742 kilometers.

planet = "Earth" diameter = 12742

Solution:

```
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:

```
print(d['k1'][3]['tricky'][3]['target'][3])
```

hello

Question-4.1:

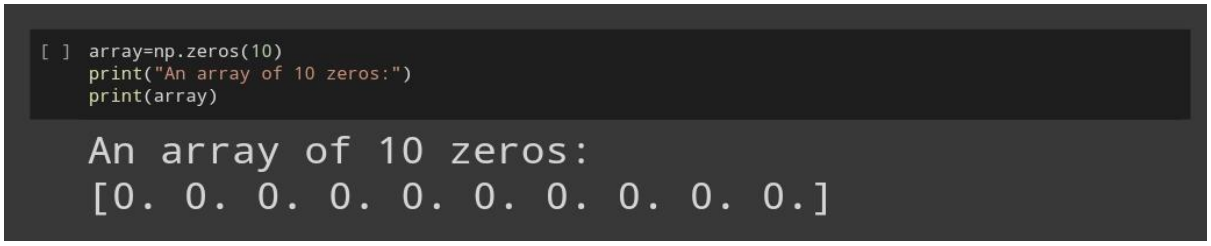
import numpy as np

Create an array of 10 zeros?

Solution:

```
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.]



Question-4.2:

import numpy as np

Create an array of 10 fives?

Solution:

```
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.]

```
[ ] 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:

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]

```
[ ] 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:

x=np.arange(0,9).reshape((3,3))

print(x)

[[0 1 2] [3 4 5] [6 7 8]]

```
[ ] x=np.arange(0,9).reshape((3,3))
    print(x)

[[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:

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

print(np.concatenate((a, b), axis=0))

[1 2 3 4 5 6]

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

```
[ ]  a = np.array([1,2,3])
     b = np.array([4,5,6])
     print(np.concatenate((a, b), axis=0))
```

```
[1 2 3 4 5 6]
```

Question-8:

Create a dataframe with 3 rows and 2 columns

Solution:

import pandas as pd

import matplotlib.pyplot as plt

import numpy as np

data=[["Joshua","Leader"],["Senthil","Member"],["Kirthika","Mentor"]]
print(pd.DataFrame(data,columns=["Name","Role"]))

Name Role

0 Joshua Leader

1 Senthil Member

2 Kirthika Mentor

```
[ ]  import pandas as pd
     import matplotlib.pyplot as plt
     import numpy as np
```

```
[ ]  data=[["Joshua","Leader"],["Senthil","Member"],["Kirthika","Mentor"]]
     print(pd.DataFrame(data,columns=["Name","Role"]))
```

```
       Name    Role
0    Joshua  Leader
1   Senthil  Member
2  Kirthika  Mentor
```

Question-9:

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

Solution:

```
from datetime import datetime
date=pd.date_range(start="2023-01-01",end="2023-02-10")
print(pd.Series(date))
```

0 2023-01-01

1 2023-01-02

2 2023-01-03

3 2023-01-04

4 2023-01-05

5 2023-01-06

6 2023-01-07

7 2023-01-08

8 2023-01-09

9 2023-01-10

10 2023-01-11

11 2023-01-12

12 2023-01-13

13 2023-01-14

14 2023-01-15

15 2023-01-16

16 2023-01-17

17 2023-01-18

18 2023-01-19

19 2023-01-20

20 2023-01-21

21 2023-01-22

22 2023-01-23

23 2023-01-24

24 2023-01-25

25 2023-01-26

26 2023-01-27

27 2023-01-28

28 2023-01-29

29 2023-01-30

30 2023-01-31

31 2023-02-01

32 2023-02-02

33 2023-02-03

34 2023-02-04

35 2023-02-05

36 2023-02-06

37 2023-02-07

38 2023-02-08

39 2023-02-09

40 2023-02-10

dtype: datetime64[ns]

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

```
[ ] from datetime import datetime

    date=pd.date_range(start="2023-01-01",end="2023-02-10")
    print(pd.Series(date))
```

```
0    2023-01-01
1    2023-01-02
2    2023-01-03
3    2023-01-04
4    2023-01-05
5    2023-01-06
6    2023-01-07
7    2023-01-08
8    2023-01-09
9    2023-01-10
10   2023-01-11
11   2023-01-12
12   2023-01-13
13   2023-01-14
14   2023-01-15
15   2023-01-16
16   2023-01-17
17   2023-01-18
18   2023-01-19
19   2023-01-20
20   2023-01-21
21   2023-01-22
22   2023-01-23
23   2023-01-24
24   2023-01-25
25   2023-01-26
26   2023-01-27
27   2023-01-28
28   2023-01-29
29   2023-01-30
30   2023-01-31
31   2023-02-01
32   2023-02-02
33   2023-02-03
34   2023-02-04
35   2023-02-05
36   2023-02-06
37   2023-02-07
38   2023-02-08
39   2023-02-09
40   2023-02-10
dtype: datetime64[ns]
```

Question-10:

Create 2D list to DataFrame

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

Solution: