Assignment -1

Basic Python

Assignment Date	12 September 2022
Student Name	Mr. Gunasekaran A
Student Roll Number	2127190801023
Maximum Marks	2 Marks

1. Split this string

s = "Hi there Sam!"

Solution:

```
s.split(" ")
```

```
In [5]: s.split(" ")

Out[5]: ['Hi', 'there', 'Sam!']
```

2. Use .format() to print the following string.

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

planet = "Earth"diameter = 12742

Solution:

```
txt = "The diameter of {planet} is {diameter} kilometers.".format(planet = planet, diameter = diameter)
txt
```

```
In [9]: txt = "The diameter of {planet} is {diameter} kilometers.".format(planet = planet, diameter = diameter)
        txt

Out[9]: 'The diameter of Earth is 12742 kilometers.'
```

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])
```

```
In [2]: print(d["k1"][3]["tricky"][3]["target"][3])
hello
```

Import Numpy

Solution:

```
import numpy as np
```

4.1 Create an array of 10 zeros?

Solution:

```
array=np.zeros(10)array
```

```
In [15]: array=np.zeros(10)
         array
Out[15]: array([0., 0., 0., 0., 0., 0., 0., 0., 0., 0.])
```

4.2 Create an array of 10 fives?

Solution:

```
array2 = np.ones(10)*5array2
```

```
In [16]: array2 = np.ones(10)*5
         array2

Out[16]: array([5., 5., 5., 5., 5., 5., 5., 5., 5., 5.])
```

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

Solution:

```
even_nos = []for i in range(20,36):
    if(i % 2 == 0):
        even_nos.append(i)

array3 = np.array(even_nos)array3
```

```
In [19]: even_nos = []
         for i in range(20,36):
             if(i % 2 == 0):
                 even_nos.append(i)


         array3 = np.array(even_nos)
         array3

Out[19]: array([20, 22, 24, 26, 28, 30, 32, 34])
```

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

Solution:

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

```
In [20]: x = np.arange(0,9).reshape(3,3)
         x

Out[20]: array([[0, 1, 2],
                [3, 4, 5],
                [6, 7, 8]])
```

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

Solution:

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

```
In [20]: x = np.arange(0,9).reshape(3,3)
         x

Out[20]: array([[0, 1, 2],
                [3, 4, 5],
                [6, 7, 8]])
```

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])
c = np.concatenate((a, b))c
```

In [27]:

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

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

Out[27]:

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

Pandas

8. Create a dataframe with 3 rows and 2 columns

Solution:

```
import pandas as pd
data = [['Cr7', 7], ['Lm10', 10], ['Ney', 11]]
df = pd.DataFrame(data, columns=['footie', 'no'])
df
```

In [26]:

```
data = [['Cr7', 7], ['Lm10', 10], ['Ney', 11]]

# Create the pandas DataFrame
df = pd.DataFrame(data, columns=['footie', 'no'])

# print dataframe.
df
```

Out[26]:

	footie	no
0	Cr7	7
1	Lm10	10
2	Ney	11

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

Solution:

```
base = datetime.datetime(2023, 1, 1)
date_list = [base + datetime.timedelta(days=x) for x in range(41)]
series = pd.Series(date_list)
Series
```

In [35]:

```
base = datetime.datetime(2023, 1, 1)
date_list = [base + datetime.timedelta(days=x) for x in range(41)]

series = pd.Series(date_list)
series
```

Out[35]:

```
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]
```

10. Create 2D list to DataFrame

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

Solution:

```
df1 = pd.DataFrame(lists, columns = ["Col1", "Col2", "Col3"])
df1
```

In [29]:

```
df1 = pd.DataFrame(lists, columns = ["Col1", "Col2", "Col3"])
df1
```

Out[29]:

	Col1	Col2	Col3
0	1	aaa	22
1	2	bbb	25
2	3	ccc	24