ASSIGNMENT 1

Assignment Date	**03/09/2022**
Student Name	**S.Saiusha**
Student Roll No	**960519104070**
Maximum Marks	**2 Marks**

Basic Python

1. Split this string

In [ ]:

```
s = "Hi there Sam!"
```

In [24]:

```
s.split()
```

Out[24]:

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

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

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

In [ ]:

```
planet = "Earth" diameter = 12742
```

In [ ]:

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

3. In this nest dictionary grab the word "hello"

In [ ]:

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

In [25]:

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

Numpy

In [ ]:

```
import numpy as np
```

4.1 Create an array of 10 zeros?

4.2 Create an array of 10 fives?

In [26]:

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

```
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
```

In [27]:

```
array=np.ones(10)*5 print(array)
```

```
[5. 5. 5. 5. 5. 5. 5. 5. 5. 5.]
```

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

In [28]:

```
array=np.arange(20,35,2)
 print(array)
```

```
[20 22 24 26 28 30 32 34]
```

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

In [29]:

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

```
[[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])

In [30]:

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

Out[30]:

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

Pandas

8. Create a dataframe with 3 rows and 2 columns

In [32]:

```
import pandas as pd
```

In [31]:

```
data = [['tom', 10], ['nick', 15], ['juli', 14]] df = pd.DataFrame(data,
columns=['Name', 'Age']) df
```

Out[31]:

	Name	Age
0	tom	10
1	nick	15
2	juli	14

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

In [34]:

```
import datetime test_date = datetime.datetime(2023, 1, 1) print("The original
date is : " + str(test_date)) K = 40 res = [test_date +
datetime.timedelta(days=idx) for idx in range(K)] print("Next K dates list :
" + str(res))
```

```
The original date is : 2023-01-01 00:00:00
Next K dates list : [datetime.datetime(2023, 1, 1, 0, 0), datetime.datetime
(2023, 1, 2, 0, 0), datetime.datetime(2023, 1, 3, 0, 0), datetime.datetime(
2023, 1, 4, 0, 0), datetime.datetime(2023, 1, 5, 0, 0), datetime.datetime(2
023, 1, 6, 0, 0), datetime.datetime(2023, 1, 7, 0, 0), datetime.datetime(20
23, 1, 8, 0, 0), datetime.datetime(2023, 1, 9, 0, 0), datetime.datetime(202
3, 1, 10, 0, 0), datetime.datetime(2023, 1, 11, 0, 0), datetime.datetime(20
23, 1, 12, 0, 0), datetime.datetime(2023, 1, 13, 0, 0), datetime.datetime(2
023, 1, 14, 0, 0), datetime.datetime(2023, 1, 15, 0, 0), datetime.datetime(
2023, 1, 16, 0, 0), datetime.datetime(2023, 1, 17, 0, 0), datetime.datetime
(2023, 1, 18, 0, 0), datetime.datetime(2023, 1, 19, 0, 0), datetime.datetim
e(2023, 1, 20, 0, 0), datetime.datetime(2023, 1, 21, 0, 0), datetime.dateti
me(2023, 1, 22, 0, 0), datetime.datetime(2023, 1, 23, 0, 0), datetime.datet
ime(2023, 1, 24, 0, 0), datetime.datetime(2023, 1, 25, 0, 0), datetime.date
time(2023, 1, 26, 0, 0), datetime.datetime(2023, 1, 27, 0, 0), datetime.dat
etime(2023, 1, 28, 0, 0), datetime.datetime(2023, 1, 29, 0, 0), datetime.da
tetime(2023, 1, 30, 0, 0), datetime.datetime(2023, 1, 31, 0, 0), datetime.d
atetime(2023, 2, 1, 0, 0), datetime.datetime(2023, 2, 2, 0, 0), datetime.da
tetime(2023, 2, 3, 0, 0), datetime.datetime(2023, 2, 4, 0, 0), datetime.dat
etime(2023, 2, 5, 0, 0), datetime.datetime(2023, 2, 6, 0, 0), datetime.date
time(2023, 2, 7, 0, 0), datetime.datetime(2023, 2, 8, 0, 0), datetime.datet
ime(2023, 2, 9, 0, 0)]
```

10. Create 2D list to DataFrame

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

In [35]:

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

In [37]:

```
df = pd.DataFrame(lists, columns =['s.no','name','rollno'])
print(df )
```

```
   s.no name  rollno 0
1  aaa      22
1     2  bbb      25
2     3  ccc      24
```