Assignment -1
Basic Python

Assignment Date	08 November 2022
Student Name	Gomathi V
Student Roll Number	620619106008
Maximum Marks	2 Marks

1) SPLIT THE STRING:

s = "Hi there Sam!"

s.split(" ")

OUTPUT:

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

CODE:

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

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

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

```
In [3]: planet = "Earth"
        diameter = 12742
```

```
In [4]: print("The diameter of Earth is {} kilometers.".format(diameter))
```

The diameter of Earth is 12742 kilometers.

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

CODE:

d =

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

result=d['k1'][3]['tricky'][3]['target'][3] print(result)

OUTPUT:

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

```
In [5]: d = {'k1':[1,2,3,{'tricky':['oh','man','inception',{'target':[1,2,3,'hello']}]}]}
```

```
In [6]: result=d['k1'][3]['tricky'][3]['target'][3]
        print(result)
```

hello

4) Create an array of 10 zeros and Create an array of 10 fives

CODE:
import numpy as np
array=np.zeros(10)
print(array)
arr1=np.ones(10)*5
print(arr1) OUTPUT:

Numpy

```
In [2]: import numpy as np
```

4.1 Create an array of 10 zeros?

4.2 Create an array of 10 fives?

```
In [3]: array=np.zeros(10)
        print(array)

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

In [4]: arr1=np.ones(10)*5
        print(arr1)

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

5) Create an array of 10 zeros and Create an array of 10 fives?

CODE:
array=np.zeros(10)
print(array)
arr1=np.ones(10)*5
print(arr1)

OUTPUT:

Numpy

```
In [2]: import numpy as np
```

4.1 Create an array of 10 zeros?

4.2 Create an array of 10 fives?

```
In [3]: array=np.zeros(10)
        print(array)

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

```
In [4]: arr1=np.ones(10)*5
        print(arr1)

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

6) Create an array of all the even integers from 20 to 35 and Create a 3x3 matrix with values ranging from 0 to 8

CODE:

arr2=np.arange(20,35,2)

print(arr2)

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

print(x)

OUTPUT:

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

```
In [5]: arr2=np.arange(20,35,2)
        print(arr2)
```

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

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

```
In [6]: x=np.arange(0,9).reshape(3,3)
        print(x)
```

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

7) Create a dataframe with 3 rows and 2 columns CODE:

import pandas as pd import numpy as

np

array=np.random.randint(10,size=(3,

2)) array=([9,2,2],

   [3,5,1],

    [4,9,8],

   [3,4,5])

df=pd.DataFrame(array)

df

OUTPUT:

Pandas

8. Create a dataframe with 3 rows and 2 columns

```python
In [8]: import pandas as pd
        import numpy as np
```

```python
In [ ]: array=np.random.randint(10,size=(3,2))
        array=([9,2,2],
              [3,5,1],
               [4,9,8],
               [3,4,5])
        df=pd.DataFrame(array)
        df
```

8) Generate the series of dates from 1st Jan, 2023 to 10th Feb, 2023 CODE:

import pandas as pd

dRan3 = pd.date_range(start ='2023-01-01', end ='2023-02-10').to_pydatetime().tolist()

print(dRan3)

OUTPUT:

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

```python
In [2]: import pandas as pd
        dRan3 = pd.date_range(start ='2023-01-01', end ='2023-02-10').to_pydatetime().tolist()
        print(dRan3)

[datetime.datetime(2023, 1, 1, 0, 0), datetime.datetime(2023, 1, 2, 0, 0), datetime.datetime(2023, 1, 3, 0, 0), datetime.dateti
me(2023, 1, 4, 0, 0), datetime.datetime(2023, 1, 5, 0, 0), datetime.datetime(2023, 1, 6, 0, 0), datetime.datetime(2023, 1, 7,
0, 0), datetime.datetime(2023, 1, 8, 0, 0), datetime.datetime(2023, 1, 9, 0, 0), datetime.datetime(2023, 1, 10, 0, 0), datetim
e.datetime(2023, 1, 11, 0, 0), datetime.datetime(2023, 1, 12, 0, 0), datetime.datetime(2023, 1, 13, 0, 0), datetime.datetime(20
23, 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.datetime(2023, 1, 20, 0, 0), datetime.
datetime(2023, 1, 21, 0, 0), datetime.datetime(2023, 1, 22, 0, 0), datetime.datetime(2023, 1, 23, 0, 0), datetime.datetime(202
3, 1, 24, 0, 0), datetime.datetime(2023, 1, 25, 0, 0), datetime.datetime(2023, 1, 26, 0, 0), datetime.datetime(2023, 1, 27, 0,
0), datetime.datetime(2023, 1, 28, 0, 0), datetime.datetime(2023, 1, 29, 0, 0), datetime.datetime(2023, 1, 30, 0, 0), datetime.
datetime(2023, 1, 31, 0, 0), datetime.datetime(2023, 2, 1, 0, 0), datetime.datetime(2023, 2, 2, 0, 0), datetime.datetime(2023,
2, 3, 0, 0), datetime.datetime(2023, 2, 4, 0, 0), datetime.datetime(2023, 2, 5, 0, 0), datetime.datetime(2023, 2, 6, 0, 0), dat
etime.datetime(2023, 2, 7, 0, 0), datetime.datetime(2023, 2, 8, 0, 0), datetime.datetime(2023, 2, 9, 0, 0), datetime.datetime(2
023, 2, 10, 0, 0)]
```

9) Create 2D list to DataFrame

```
lists = [[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]] df =
    pd.DataFrame(lists, columns =['ID', 'number','no'])
    print(df)
```

OUTPUT:

10. Create 2D list to DataFrame

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

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

```
4]: df = pd.DataFrame(lists, columns =['ID', 'number','no'])
    print(df)
```

```
   ID number  no
0   1    aaa  22
1   2    bbb  25
2   3    ccc  24
```