

**Assignment -1**  
**Python Programming**

Assignment Date	12 november 2022
Student Name	Shalini A
Student Roll Number	211419106249
Maximum Marks	2 Marks

**Question-1:**

**Split this string**

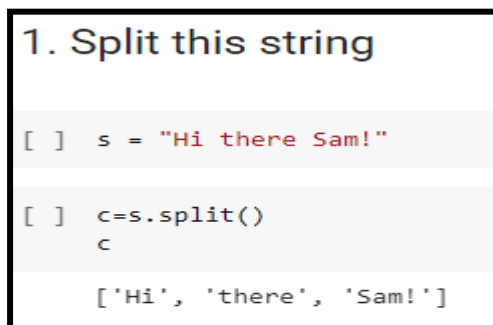
**Solution:**

```
s = "Hi there Sam!"  
c=s.split()  
c
```

**Output:**

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

**Screenshot:**



**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))
```

**Output:**

The diameter of Earth is 12742 kilometers.

**Screenshot:**

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

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

```
[ ] 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"

Solution:

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

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

Output:

'hello'

Screenshot:

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

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

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

```
'hello'
```

Question-4:

Numpy

4.1:Create an array of 10 zeros

Solution:

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

Output:

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

4.2:Create an array of 10 fives

Solution:

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

Output:

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

Screenshot:

## Numpy

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

4.1 Create an array of 10 zeros?

4.2 Create an array of 10 fives?

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

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

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

[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(20,36,2)
```

```
print(array)
```

**Output:**

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

**Screenshot:**

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

```
[ ] array=np.arange(20,36,2)
    print(array)

[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)
```

**Output:**

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

**Screenshot:**

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

**Output:**

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

**Screenshot:**

7. Concatenate a and b

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)))

[1 2 3 4 5 6]
```

## PANDAS

**Question-8:**

**Create a dataframe with 3 rows and 2 columns**

**Solution:**

```
import pandas as pd
data = [['raj', 10], ['suresh', 15], ['juli', 14]]
d= pd.DataFrame(data, columns=['Name', 'Age'])
print(d)
```

**Output:**

	Name	Age
0	raj	10
1	suresh	15
2	juli	14

## Screenshot:

```
8. Create a dataframe with 3 rows and 2 columns
```

```
[ ] import pandas as pd
```

```
[ ] data = [['raj', 10], ['suresh', 15], ['juli', 14]]
d = pd.DataFrame(data, columns=['Name', 'Age'])
print(d)
```

	Name	Age
0	raj	10
1	suresh	15
2	juli	14

## Question-9:

Generate the series of datas from jan,2023 to feb,2023

## Solution:

```
pd.date_range(start='1/1/2023', periods=41)
```

## Output:

```
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', '2023-01-11', '2023-01-12',
               '2023-01-13', '2023-01-14', '2023-01-15', '2023-01-16',
               '2023-01-17', '2023-01-18', '2023-01-19', '2023-01-20',
               '2023-01-21', '2023-01-22', '2023-01-23', '2023-01-24',
               '2023-01-25', '2023-01-26', '2023-01-27', '2023-01-28',
               '2023-01-29', '2023-01-30', '2023-01-31', '2023-02-01',
               '2023-02-02', '2023-02-03', '2023-02-04', '2023-02-05',
               '2023-02-06', '2023-02-07', '2023-02-08', '2023-02-09',
               '2023-02-10'],
              dtype='datetime64[ns]', freq='D')
```

Code

Text

## Screenshot:

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

```
[ ] pd.date_range(start='1/1/2023', periods=41)
```

```
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', '2023-01-11', '2023-01-12',
               '2023-01-13', '2023-01-14', '2023-01-15', '2023-01-16',
               '2023-01-17', '2023-01-18', '2023-01-19', '2023-01-20',
               '2023-01-21', '2023-01-22', '2023-01-23', '2023-01-24',
               '2023-01-25', '2023-01-26', '2023-01-27', '2023-01-28',
               '2023-01-29', '2023-01-30', '2023-01-31', '2023-02-01',
               '2023-02-02', '2023-02-03', '2023-02-04', '2023-02-05',
               '2023-02-06', '2023-02-07', '2023-02-08', '2023-02-09',
               '2023-02-10'],
              dtype='datetime64[ns]', 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]]
```

```
s = pd.DataFrame(lists)
```

s

Output:

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

Screenshot:

#### 10. Create 2D list to DataFrame

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

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

```
[ ] s = pd.DataFrame(lists)
s
```

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

