ASSIGNMENT-01

BASIC PYTHON

Assignment Date	12 September 2022
Student Name	SAHANA J M
Student Roll Number	113219071033
Maximum Marks	2 Marks

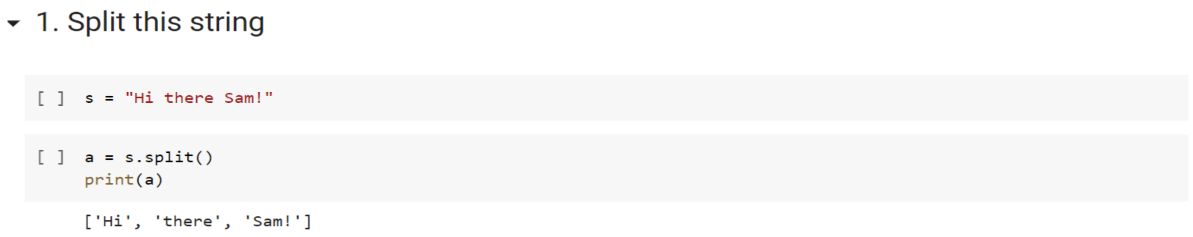
QUESTION-01:

1. Split this string

s = "Hi there Sam!"

SOLUTION:

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



1. Split this string

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

```
[ ]  a = s.split()
     print(a)

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

QUESTION-02:

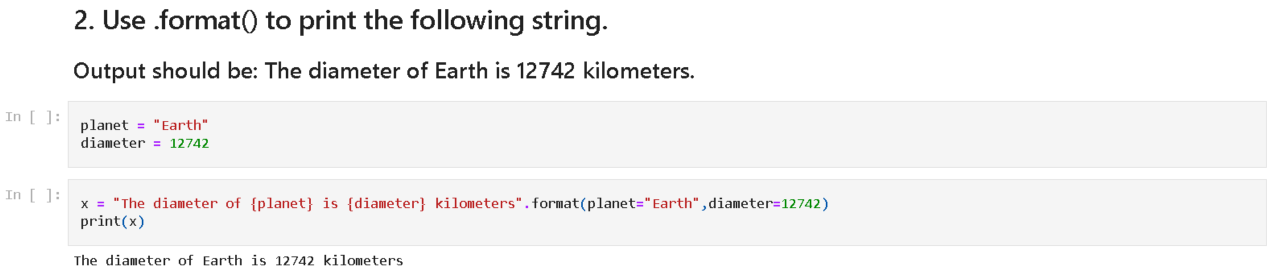
2. Use .format() to print the following string.

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

planet = "Earth"
diameter = 12742

SOLUTION:

The diameter of Earth is 12742 kilometers



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

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

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

```
In [ ]:  x = "The diameter of {planet} is {diameter} kilometers".format(planet="Earth",diameter=12742)
         print(x)

         The diameter of Earth is 12742 kilometers
```

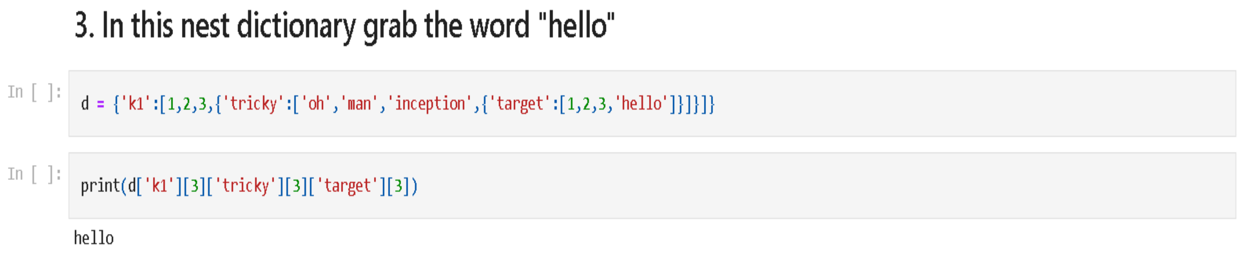
QUESTION-03:

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

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

SOLUTION:

hello



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

hello
```

QUESTION-04:

Numpy

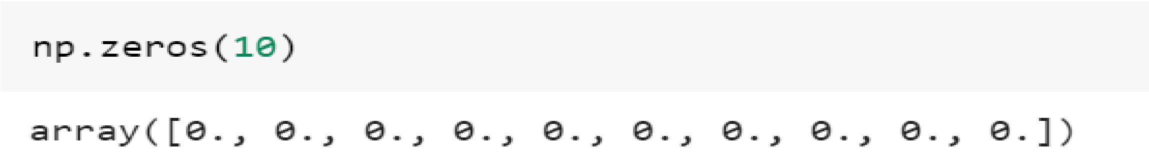
// Importing Numpy

import numpy **as** np

4.1 Create an array of 10 zeros?

SOLUTION:

array([0., 0., 0., 0., 0., 0., 0., 0., 0., 0.])

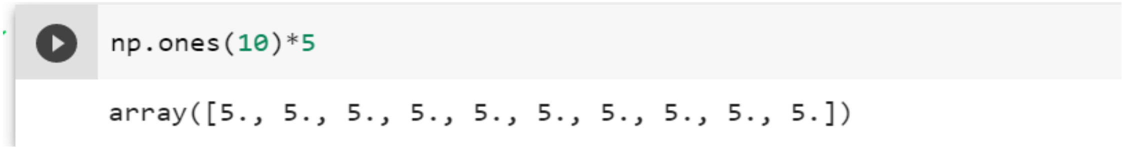


```
np.zeros(10)

array([0., 0., 0., 0., 0., 0., 0., 0., 0., 0.])
```

4.2 Create an array of 10 fives?

SOLUTION:

array([5., 5., 5., 5., 5., 5., 5., 5., 5., 5.])



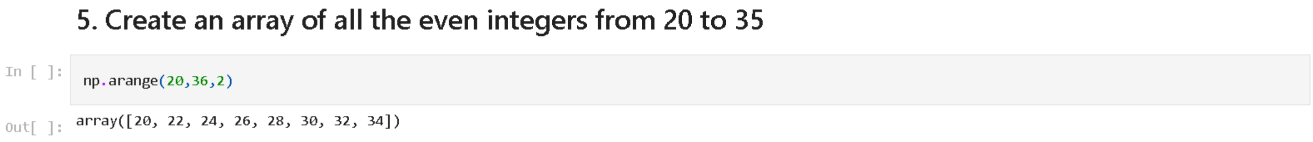
```
np.ones(10)*5

array([5., 5., 5., 5., 5., 5., 5., 5., 5., 5.])
```

QUESTION-05:

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

SOLUTION:

array([20, 22, 24, 26, 28, 30, 32, 34])



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

```
np.arange(20,36,2)
```

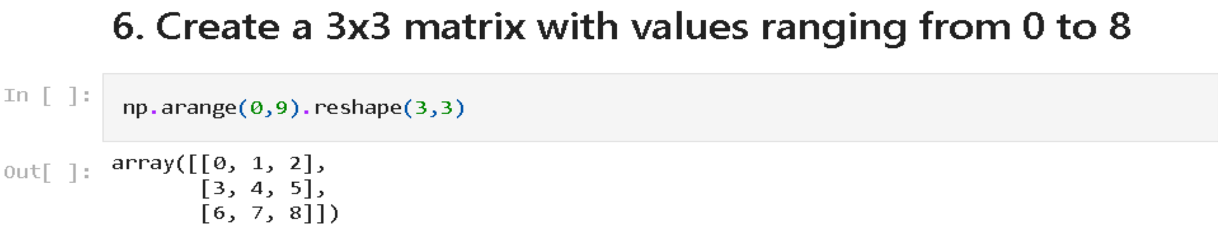
```
array([20, 22, 24, 26, 28, 30, 32, 34])
```

QUESTION-06:

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

SOLUTION:

array([[0, 1, 2],
       [3, 4, 5],
       [6, 7, 8]])



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

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

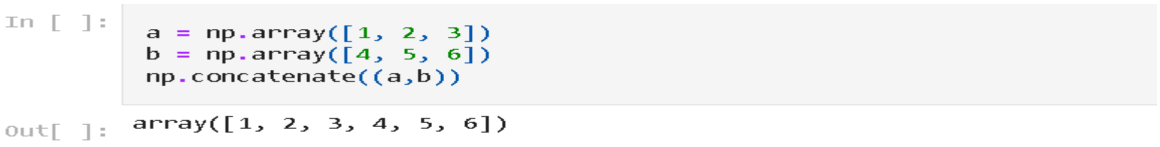
```
array([[0, 1, 2],
       [3, 4, 5],
       [6, 7, 8]])
```

QUESTION-07:

7. Concatenate a and b

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

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



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

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

QUESTION-08:

Pandas

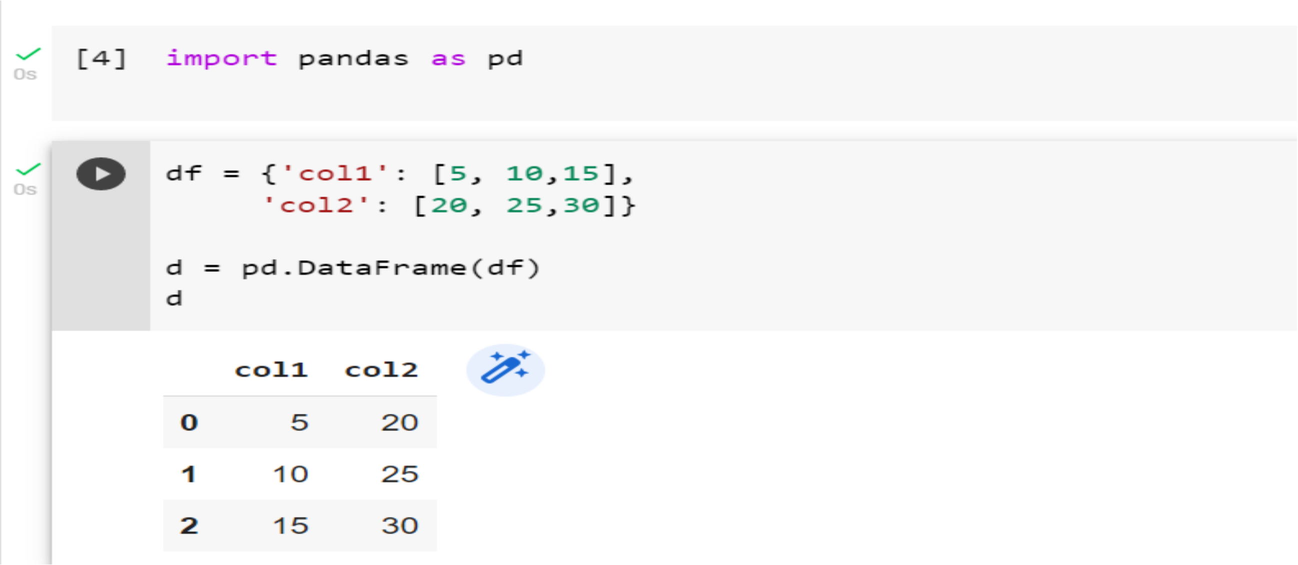
// importing Pandas library

import pandas **as** pd

8. Create a dataframe with 3 rows and 2 columns

SOLUTION:

```
col1  col2
0    1    4
1    2    5
2    3    6
```



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

```
df = {'col1': [5, 10,15],
      'col2': [20, 25,30]}

d = pd.DataFrame(df)
d
```

```
   col1  col2
0     5    20
1    10    25
2    15    30
```

QUESTION-09:

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

SOLUTION:

```
2023-01-01 00:00:00
2023-01-02 00:00:00
2023-01-03 00:00:00
2023-01-04 00:00:00
2023-01-05 00:00:00
2023-01-06 00:00:00
2023-01-07 00:00:00
2023-01-08 00:00:00
2023-01-09 00:00:00
```

2023-01-10 00:00:00

2023-01-11 00:00:00

2023-01-12 00:00:00

2023-01-13 00:00:00

2023-01-14 00:00:00

2023-01-15 00:00:00

2023-01-16 00:00:00

2023-01-17 00:00:00

2023-01-18 00:00:00

2023-01-19 00:00:00

2023-01-20 00:00:00

2023-01-21 00:00:00

2023-01-22 00:00:00

2023-01-23 00:00:00

2023-01-24 00:00:00

2023-01-25 00:00:00

2023-01-26 00:00:00

2023-01-27 00:00:00

2023-01-28 00:00:00

2023-01-29 00:00:00

2023-01-30 00:00:00

2023-01-31 00:00:00

2023-02-01 00:00:00

2023-02-02 00:00:00

2023-02-03 00:00:00

2023-02-04 00:00:00

2023-02-05 00:00:00

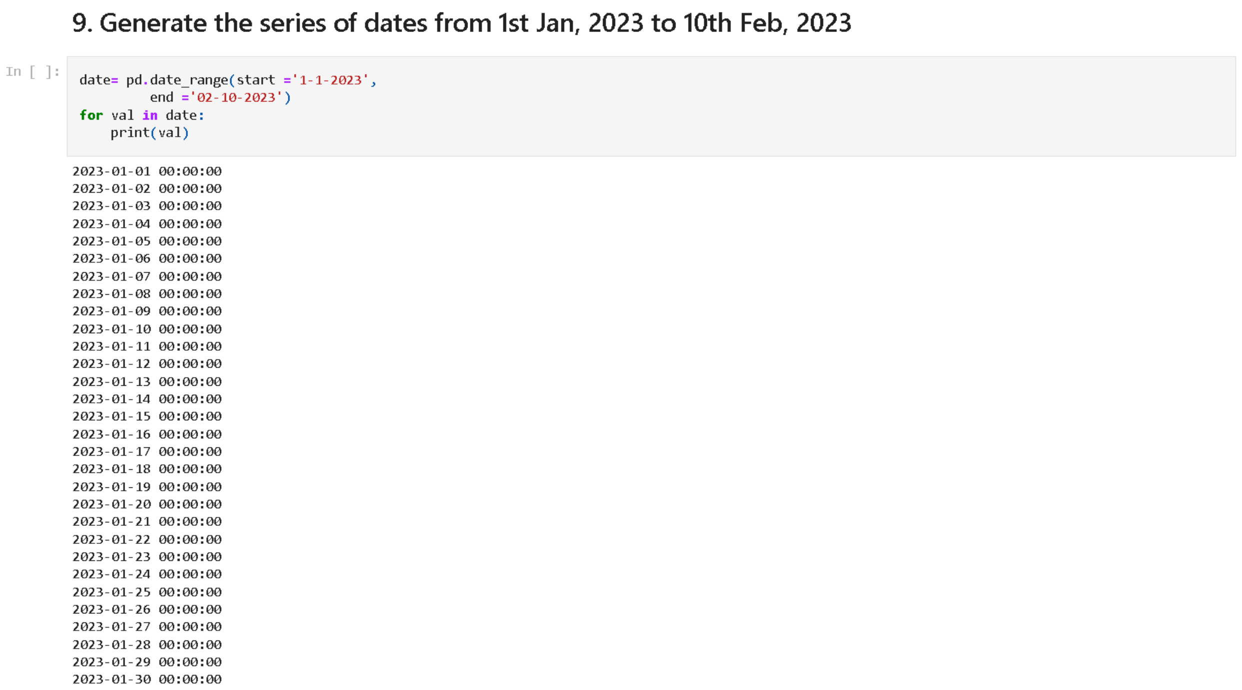
2023-02-06 00:00:00

2023-02-07 00:00:00

2023-02-08 00:00:00

2023-02-09 00:00:00

2023-02-10 00:00:00

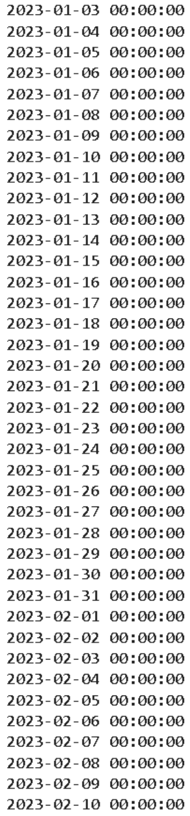


9. Generate the series of dates from 1st Jan, 2023 to 10th Feb, 2023

In [ ]:

```
date= pd.date_range(start ='1-1-2023',
         end ='02-10-2023')
for val in date:
    print(val)
```

```
2023-01-01 00:00:00
2023-01-02 00:00:00
2023-01-03 00:00:00
2023-01-04 00:00:00
2023-01-05 00:00:00
2023-01-06 00:00:00
2023-01-07 00:00:00
2023-01-08 00:00:00
2023-01-09 00:00:00
2023-01-10 00:00:00
2023-01-11 00:00:00
2023-01-12 00:00:00
2023-01-13 00:00:00
2023-01-14 00:00:00
2023-01-15 00:00:00
2023-01-16 00:00:00
2023-01-17 00:00:00
2023-01-18 00:00:00
2023-01-19 00:00:00
2023-01-20 00:00:00
2023-01-21 00:00:00
2023-01-22 00:00:00
2023-01-23 00:00:00
2023-01-24 00:00:00
2023-01-25 00:00:00
2023-01-26 00:00:00
2023-01-27 00:00:00
2023-01-28 00:00:00
2023-01-29 00:00:00
2023-01-30 00:00:00
```



```
2023-01-03 00:00:00
2023-01-04 00:00:00
2023-01-05 00:00:00
2023-01-06 00:00:00
2023-01-07 00:00:00
2023-01-08 00:00:00
2023-01-09 00:00:00
2023-01-10 00:00:00
2023-01-11 00:00:00
2023-01-12 00:00:00
2023-01-13 00:00:00
2023-01-14 00:00:00
2023-01-15 00:00:00
2023-01-16 00:00:00
2023-01-17 00:00:00
2023-01-18 00:00:00
2023-01-19 00:00:00
2023-01-20 00:00:00
2023-01-21 00:00:00
2023-01-22 00:00:00
2023-01-23 00:00:00
2023-01-24 00:00:00
2023-01-25 00:00:00
2023-01-26 00:00:00
2023-01-27 00:00:00
2023-01-28 00:00:00
2023-01-29 00:00:00
2023-01-30 00:00:00
2023-01-31 00:00:00
2023-02-01 00:00:00
2023-02-02 00:00:00
2023-02-03 00:00:00
2023-02-04 00:00:00
2023-02-05 00:00:00
2023-02-06 00:00:00
2023-02-07 00:00:00
2023-02-08 00:00:00
2023-02-09 00:00:00
2023-02-10 00:00:00
```

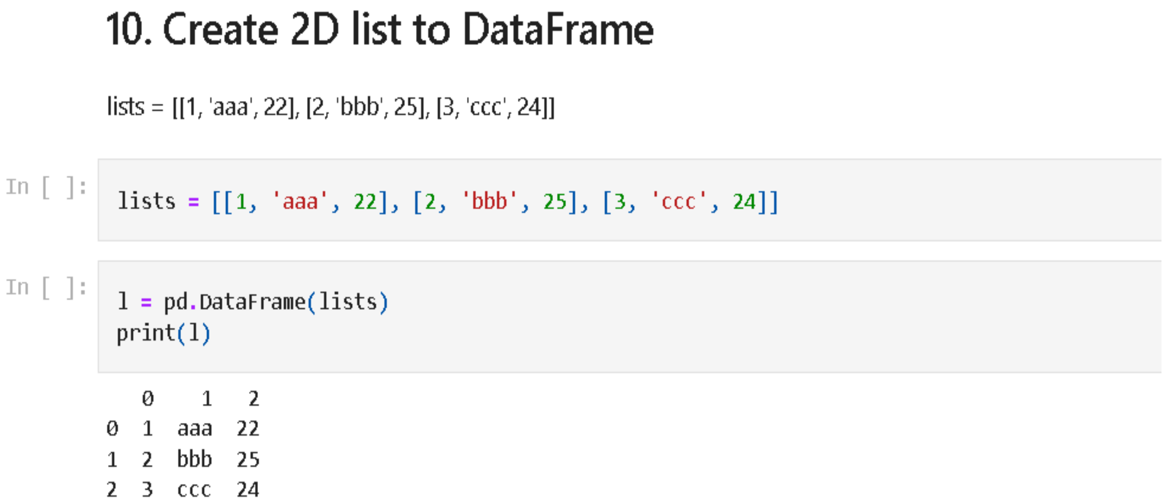
QUESTION-10:

10. Create 2D list to DataFrame

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

SOLUTION:

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



10. Create 2D list to DataFrame

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

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

```
In [ ]: l = pd.DataFrame(lists)
        print(l)
```

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