

▼ Basic Python

▼ 1. Split this string

```
s = "Hi there Sam!"  
  
s.split()  
  
['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  
  
op="The diameter of () is () kilometers"  
print(op.format(planet, diameter))  
  
The diameter of () is () kilometers
```

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

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

▼ Numpy

```
import numpy as np
```

▼ 4.1 Create an array of 10 zeros?

4.2 Create an array of 10 fives?

```
np.zeros((10,10))
```

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

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

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

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

```
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

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

```
[[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])
```

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

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

▼ Pandas

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

```
import pandas as pd
```

```
ip=[['apple',20],['cherry',15],['mango', 10]]  
result=pd.DataFrame(ip,columns= ['fruits name', 'no. of fruits'])  
print (result)
```

| | fruits name | no. of fruits |
|---|-------------|---------------|
| 0 | apple | 20 |
| 1 | cherry | 15 |
| 2 | mango | 10 |

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

```
d=pd.date_range(start='01-01-2023', end='02-10-2023')  
res=pd.Series(d)  
print(res)
```

| | |
|----|------------|
| 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]]
```

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

```
res=pd.DataFrame(lists)
print(res)
```

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

