Basic Python

1. Split this string

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

```
['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
```

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

```
The diameter of Earth is 12742 kilometers.
```

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

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

```
hello
```

Numpy

```
import numpy as np
```

4.1 Create an array of 10 zeros?

4.2 Create an array of 10 fives?

```
import numpy as np
array=np.zeros(10)
print("An array of 10 zeros:")
print(array)
```

```
An array of 10 zeros:
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
```

```
import numpy as np
array=np.ones(10)*5
print("An array of 10 five:")
print(array)
```

```
An array of 10 five:
[5. 5. 5. 5. 5. 5. 5. 5. 5. 5.]
```

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

```
import numpy as np
array=np.arange(20,36,2)
print("Array of all the even integers from 20 to 35")
print(array)
```

```
Array of all the even integers from 20 to 35
[20 22 24 26 28 30 32 34]
```

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

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

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

7. Concatinate 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])
c = np.concatenate((a,b))
print(c)
```

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

Pandas

8. Create a dataframe with 3 rows and 2 columns

```
import pandas as pd

# Import pandas library
import pandas as pd

#initialize list of lists
data = [[0,'aaa',10], [1,'bbb',15]]

#Create the pandas DataFrame
df = pd.DataFrame(data, columns=['Sno','Name','Age'])
```

```
#print dataframe.
df
```

```
   Sno Name  Age
0    0  aaa   10
1    1  bbb   15
```

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

```
import pandas as pd
from datetime import datetime
pd.date_range(start="2023-01-01",end="2023-02-
10").to_pydatetime().tolist()
```

```
[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(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),
 datetime.datetime(2023, 1, 11, 0, 0),
 datetime.datetime(2023, 1, 12, 0, 0),
 datetime.datetime(2023, 1, 13, 0, 0),
 datetime.datetime(2023, 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(2023, 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),
 datetime.datetime(2023, 2, 7, 0, 0),
 datetime.datetime(2023, 2, 8, 0, 0),
 datetime.datetime(2023, 2, 9, 0, 0),
 datetime.datetime(2023, 2, 10, 0, 0)]
```

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]]

import pandas as pd

#List1
lst = [[1,'aaa',22], [2,'bbb',25], ['3', 'ccc', 24]]

df = pd.DataFrame(lst,columns=['Sno', 'Name', 'Age'], dtype = float)
print(df)
```

```
   Sno Name   Age
0  1.0  aaa  22.0
1  2.0  bbb  25.0
2  3.0  ccc  24.0

/usr/local/lib/python3.7/dist-packages/IPython/core/
interactiveshell.py:3326: FutureWarning: Could not cast to float64,
falling back to object. This behavior is deprecated. In a future
version, when a dtype is passed to 'DataFrame', either all columns
will be cast to that dtype, or a TypeError will be raised
  exec(code_obj, self.user_global_ns, self.user_ns)
```