Basic Python

1. Split this string

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

```
['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
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"

```
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 num
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 num
array=np.ones(10)
```

```
print("an array of 10 fives:")
print(array)
```

```
an array of 10 fives:
[1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]
```

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

```
import numpy as np
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

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

```
[[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])

```
import numpy as np
a=np.array([1,2,3])
b=np.array([4,5,6])
np.concatenate((a,b),axis=0)
```

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

Pandas

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
```

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

```
dates=pd.date_range('2023-01-01',periods=41,freq='D')
dates
```

```
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')
```

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
lists = [[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]]
df=pd.DataFrame(lists,columns =['serial number','Name','Age'],
                                            dtype= float)
print(df)
```

```
   serial number 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)
```