**Assignment -1**
Basic Python Programming in ipynb

| Student Name | SUBASHINI R |
|---|---|
| Team ID | PNT2022TM1D43508 |

# Basic Python

## 1. Split this string

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

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

```
Hi there Sam!
['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"

```
d =
{'k1':[1,2,3,{'tricky':['oh','man','inception',{'target':[1,2,3,'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 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)
array=np.ones(10)*5
print("An array of 10 fives:")
print(array)
```

```
An array of 10 fives:
[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(20,35,2)
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. 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])
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 as pd
data=[['ammu',40],['ravi',53],['sankar',70]]
df=pd.DataFrame(data,columns=['Name','Age'])
df
```

```
     Name  Age
0    ammu   40
1    ravi   53
2  sankar   70
```

## 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="2023-02-01").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)]
```

## 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=['s.no', 'alphabet', 'number'])
print(df)
```

```
   s.no alphabet  number
0     1      aaa      22
1     2      bbb      25
2     3      ccc      24
```