

# Assignment -1

## Python Programming

Assignment Date	10 September 2022
Student Name	R. Vignesh Kumar
Student Roll Number	912419104038
Maximum Marks	2 Marks

### Basic Python

#### Question-1:

1. Split this string

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

#### Solution:

```
# Split a string into a list using split() where each word is a list item
x = s.split()
print(x)
```

▼ Basic Python

▼ 1. Split this string

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

```
[ ] # Split a string into a list using split() where each word is a list item
    x = s.split()
    print(x)
```

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

#### Question-2:

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

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

```
planet = "Earth"
diameter = 12742
```

### Solution:

#The format() method formats the specified value(s) and inserts them inside the string's placeholder.

```
string = "The diameter of {} is {} kilometers.".format(planet, diameter)
print(string)
```

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

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

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

```
[ ] #The format() method formats the specified value(s) and insert them inside the string's placeholder.
    string = "The diameter of {} is {} kilometers.".format(planet, diameter)
    print(string)
```

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

### Question-3:

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

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

### Solution:

```
# grap the word "hello" in nested dictionary
x = d['k1'][3]['tricky'][3]['target'][3]
print(x)
```

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

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

```
[ ] # grap the word "hello" in nested distionary
    x = d['k1'][3]['tricky'][3]['target'][3]
    print(x)
```

```
hello
```

# Numpy

## Question-4:

4.1 Create an array of 10 zeros?

### Solution:

```
import numpy as np

array=np.zeros(10)
print("An array of 10 zeros")
print(array)
```

4.2 Create an array of 10 fives?

### Solution:

```
array=np.ones(10)*5
print("An array of 10 fives")
print(array)
```

## Question-5:

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

### Solution:

```
array=np.arange(20,35,2)
print("An array of all the even integers from 20 to 35")
print(array)
```

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

```
[ ] array=np.arange(20,35,2)
print("An array of all the even integers from 20 to 35")
print(array)
```

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

### Question-6:

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

#### Solution:

```
matrix = np.arange(0, 9).reshape(3,3)
print("A 3X3 matrix with values ranging from 0 to 8\n")
print(matrix)
```

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

```
[ ] matrix = np.arange(0, 9).reshape(3,3)
print("A 3X3 matrix with values ranging from 0 to 8\n")
print(matrix)
```

A 3X3 matrix with values ranging from 0 to 8

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

### Question-7:

7. Concatenate a and b

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

#### Solution:

```
a = np.array([1, 2, 3])
b = np.array([4, 5, 6])
cab=np.concatenate((a,b), axis=None)
print("Concatenation of a and b")
print(cab)
```

▼ 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])
cab=np.concatenate((a,b), axis=None)
print("Concatenation of a and b")
print(cab)
```

☞ Concatenation of a and b  
[1 2 3 4 5 6]

# Pandas

## Question-8:

8. Create a dataframe with 3 rows and 2 columns

### Solution:

```
import pandas as pd
```

```
data = {'student_name': ['muthamizhan', 'karthik', 'Ravi'], 'cgpa_marks': [9.4, 8.9, 8.7]}
```

```
df = pd.DataFrame(data)
df=df.to_string(index=False)
print("A dataframe with 3 rows and 2 columns\n")
print (df)
```

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

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

```
data = {'student_name': ['muthamizhan', 'karthik', 'Ravi'],
        'cgpa_marks': [9.4, 8.9, 8.7]}
```

```
df = pd.DataFrame(data)
df=df.to_string(index=False)
```

```
print("A dataframe with 3 rows and 2 columns\n")
print (df)
```

📄 A dataframe with 3 rows and 2 columns

student_name	cgpa_marks
muthamizhan	9.4
karthik	8.9
Ravi	8.7

## Question-9:

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

### Solution:

```
date=pd.date_range(start='1/1/2023', end='2/10/2023').date #getting dates from  
range 1st Jan, 2023 to 10th Feb, 2023
```

```
date = pd.to_datetime(pd.Series(date))  
date = date.dt.strftime('%d-%m-%Y')  
print("The series of dates from 1st Jan, 2023 to 10th Feb, 2023 \n\t(01-01-2023 to  
10-02-2023)\n")
```

```
for s_date in date:  
    print(s_date)
```

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

```
▶ date=pd.date_range(start='1/1/2023', end='2/10/2023').date #getting dates from range 1st Jan, 2023 to 10th Feb, 2023  
date = pd.to_datetime(pd.Series(date))  
date = date.dt.strftime('%d-%m-%Y')  
print("The series of dates from 1st Jan, 2023 to 10th Feb, 2023 \n\t(01-01-2023 to 10-02-2023)\n")  
for s_date in date:  
    print(s_date)
```

📄 The series of dates from 1st Jan, 2023 to 10th Feb, 2023  
(01-01-2023 to 10-02-2023)

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

## Question-10:

10. Create 2D list to DataFrame

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

### Solution:

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

df = pd.DataFrame(lists, columns=['no', 'name', 'd_no'])
df=df.to_string(index=False)
print("Given 2D list")
print(lists)
print("\n2D list to dataframe")
print(df)
```

#### ▼ 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]]
```

```
[ ] df = pd.DataFrame(lists, columns=['no', 'name', 'd_no'])
df=df.to_string(index=False)
print("Given 2D list")
print(lists)
print("\n2D list to dataframe")
print(df)
```

```
Given 2D list
[[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]]
```

```
2D list to dataframe
no name d_no
1 aaa 22
2 bbb 25
3 ccc 24
```