

Assignment -1
Python Programming

Assignment Date	19 September 2022
Student Name	NIVEDHA K
Student Roll Number	820419104047
Maximum Marks	2 Marks

Basic Python

Question-1:

1. Split this string

s = "Hi there Sam!"

Solution:

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

Output:

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

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

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

```
['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:

```
planet = "Earth"
diameter = 12742
print ('The diameter of {planet} is {measure}
kilometers'.format(planet="Earth", measure=12742.34))
```

Output:

The diameter of Earth is 12742.34 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.
```

Question-3:

In this nest dictionary grab the word "hello"

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

Solution:

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

Output:

hello

```
✓ [3] d = {'k1': [1,2,3,{'tricky': ['oh', 'man', 'inception', {'target': [1,2,3, 'hello']}]}]}
0s

✓ [1] d = {'k1': [1,2,3,{'tricky': ['oh', 'man', 'inception', {'target': [1,2,3, 'hello']}]}]}
0s
    print(d['k1'][3]['tricky'][3]['target'][3])

hello
```

Numpy

Question-4:

```
import numpy as np
```

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)
```

Output:

An array of 10 zeros:

[0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]

```
[ ] np.zeros(10)

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

4.2 Create an array of 10 fives?

Solution:

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

Output:

An array of 10 fives:

[5. 5. 5. 5. 5. 5. 5. 5. 5. 5.]

```
[5] import numpy as np

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.]
```

Question-5:

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

Solution:

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

Output:

Array of all the even integers from 20 to 35

[20 22 24 26 28 30 32 34]

```
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]

Question-6:

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

Solution:

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

```
print(arr)
```

Output:

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

Solution:

In []:

```
import numpy as np
a = np.array([1, 2, 3])
b = np.array([4, 5, 6])
c = np.concatenate((a, b),axis = None)
print (c )
```

Output:

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

▼ 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])
    c = np.concatenate((a, b),axis = None)
    print (c )
```

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

Pandas

8. Create a dataframe with 3 rows and 2 columns

Solution:

```
import pandas as pd
data = [[10,20],[30,40],[50,60]]
# Create the pandas DataFrame with column name is provided explicitly
df = pd.DataFrame(data, columns=['Numbers','Numbers2'])
# print dataframe.
Print(df)
```

Output:



```
import pandas as pd
data = [[10,20],[30,40],[50,60]]
# Create the pandas DataFrame with column name is provided explicitly
df = pd.DataFrame(data, columns=['Numbers','Numbers2'])
# print dataframe.
print(df)
```

	Numbers	Numbers2
0	10	20
1	30	40
2	50	60

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

In [2]:

```
import datetime
import pandas as pd
start = datetime.datetime.strptime("01-01-2023", "%d-%m-%Y")
date_generated = pd.date_range(start, periods=41)
print(date_generated.strftime("%d-%m-%Y"))
```

Output:

```
Index(['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'],
      dtype='object')
```

```

import datetime

import pandas as pd

start = datetime.datetime.strptime("01-01-2023", "%d-%m-%Y")

date_generated = pd.date_range(start, periods=41)

print(date_generated.strftime("%d-%m-%Y"))

Index(['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'],
      dtype='object')

```

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]]
import pandas as pd
df = pd.DataFrame(lists, columns=['key', 'name', 'age'])

print(df)

```

Output

	key	name	age
0	1	aaa	22
1	2	bbb	25
2	3	ccc	24