

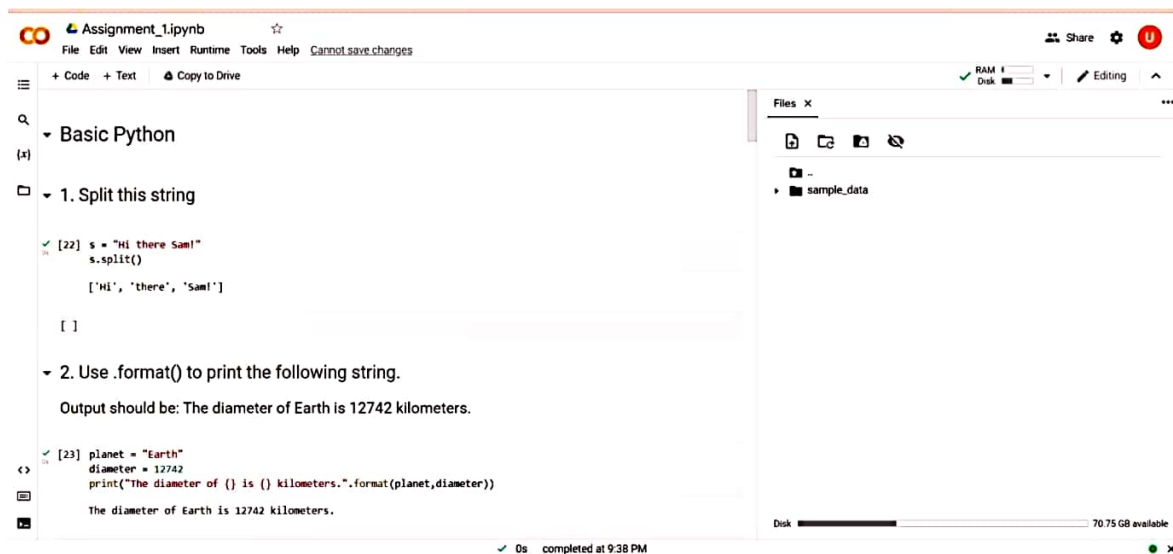
# ASSIGNMENT 1

Date : 11 October 2022  
Team ID : A.Naga Prathibha

Project Name : A NOVEL METHOD FOR HANDWRITTEN DIGIT RECOGNITION SYSTEM

Maximum Marks : 2 Marks

## Basic Python Program:



The screenshot shows a Jupyter Notebook titled "Assignment\_1.ipynb". The interface includes a menu bar (File, Edit, View, Insert, Runtime, Tools, Help) and a toolbar with options like "+ Code", "+ Text", and "Copy to Drive". The notebook content is organized into sections: "Basic Python" and "1. Split this string".

**Section 1: Split this string**

```
[22] s = "Hi there Sam!"  
s.split()  
  
['Hi', 'there', 'Sam!']  
  
[ ]
```

**Section 2: Use .format() to print the following string.**

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

```
[23] planet = "Earth"  
diameter = 12742  
print("The diameter of {} is {} kilometers.".format(planet,diameter))  
  
The diameter of Earth is 12742 kilometers.
```

The bottom status bar indicates "0s completed at 9:38 PM". On the right, a "Files" panel shows a directory structure with "sample\_data".

Assignment\_1.ipynb

File Edit View Insert Runtime Tools Help Cannot save changes

+ Code + Text Copy to Drive

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

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

      'hello'

      [ ]

      [ ]
```

Numpy

4.1 Create an array of 10 zeros?

4.2 Create an array of 10 fives?

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

Files

sample\_data

RAM 70.75 GB available

0s completed at 9:38 PM

Assignment\_1.ipynb

File Edit View Insert Runtime Tools Help Cannot save changes

+ Code + Text Copy to Drive

RAM 100% Disk 100% Editing

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

[26] 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.]

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

[10] 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]

Files

sample\_data

Assignment\_1.ipynb

File Edit View Insert Runtime Tools Help Cannot save changes

+ Code + Text Copy to Drive

RAM 100% Disk 100% Editing

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

import numpy as np  
x = np.arange(2, 11).reshape(3,3)  
print(x)

[[ 2 3 4]  
 [ 5 6 7]  
 [ 8 9 10]]

7. Concatenate a and b

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

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

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

Pandas

Disk 70.75 GB available

0s completed at 9:38 PM

Assignment\_1.ipynb

File Edit View Insert Runtime Tools Help Cannot save changes

+ Code + Text Copy to Drive

RAM 100% Disk 100% Editing

8. Create a dataframe with 3 rows and 2 columns

[15] import pandas as pd

# initialize data of lists.  
data = {'Name': ['Tom', 'Jack', 'nick', 'juli'],  
 'marks': [99, 98, 95, 90]}

df

	Name	Age
0	tom	10
1	nick	15
2	juli	14

[ ]

Files

sample\_data

Assignment\_1.ipynb

File Edit View Insert Runtime Tools Help Cannot save changes

+ Code + Text Copy to Drive

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

```
[1]: # import datetime module
import datetime

# consider the start date as 2021-february 1 st
start_date = datetime.date(2023, 1, 1)

# consider the end date as 2021-march 1 st
end_date = datetime.date(2023, 2, 10)

# delta time
delta = datetime.timedelta(days=1)

# iterate over range of dates
while (start_date <= end_date):
    print(start_date, end="\n")
    start_date += delta
```

Files X

sample\_data

Assignment\_1.ipynb

File Edit View Insert Runtime Tools Help Cannot save changes

+ Code + Text Copy to Drive

```
[21]: 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
```

Files X

sample\_data

Disk 70.75 GB available

Assignment\_1.ipynb

File Edit View Insert Runtime Tools Help Cannot save changes

+ Code + Text Copy to Drive

```
[21]: 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
```

10. Create 2D list to DataFrame

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

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

[18]: # importing pandas as pd
import pandas as pd

# dictionary of lists
lists = {'s.no': ["1", "2", "3"],
        'Name': ["aaa", "bbb", "ccc"],
        'age': [22, 25, 24]}

df = pd.DataFrame(lists)

df
```

Files X

sample\_data

Disk 70.75 GB available

Assignment\_1.ipynb

File Edit View Insert Runtime Tools Help Cannot save changes

Share Settings User

+ Code + Text Copy to Drive

RAM 1 Disk

Editing

10. Create 2D list to DataFrame

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

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

```
[18] # importing pandas as pd
import pandas as pd

# dictionary of lists
lists = {'S.No': [1, 2, 3],
        'Name': ['aaa', 'bbb', 'ccc'],
        'age': [22, 25, 24]}

df = pd.DataFrame(lists)

df
```

	S.No	Name	age
0	1	aaa	22
1	2	bbb	25
2	3	ccc	24

Files

..

sample\_data

Disk 70.75 GB available