

Assignment -1

Python Programming

Assignment Date	16 September 2022
Student Name	P.Keerthi
Student Roll Number	713119104005
Maximum Marks	2 Marks

BASIC PYTHON

1.Split this string

```
#Split this string s
= "Hi there Sam!"
s.split()
```

OUTPUT



The screenshot shows a Jupyter Notebook interface with a code cell containing the following Python code:

```
[1] import pandas as pd
import numpy as np

#Split this string
s = "Hi there Sam!"
s.split()
```

The output of the code cell is displayed below the code:

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

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

Output should be:

The diameter of earth is 12742 kilometres.

```
planet = "Earth"
diameter = 12742
planet
= "Earth"
diameter = 12742
print('The diameter of {} is {} kilometres.'.format(planet,diameter));
```

OUTPUT



The screenshot shows a Jupyter Notebook interface with three code cells. The first cell contains import statements for pandas and numpy. The second cell contains the assignment of planet and diameter. The third cell contains the formatted string print statement.

```
[ ] import pandas as pd
import numpy as np

[1] planet = "Earth"
diameter = 12742

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

The output of the third code cell is displayed below the code:

```
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])
```

OUTPUT



The screenshot shows a Jupyter Notebook interface with a code cell containing the following Python code:

```
[ ] import pandas as pd
import numpy as np

[ ] 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])
```

The output of the code is displayed below the cell:

```
hello
```

4.Numpy

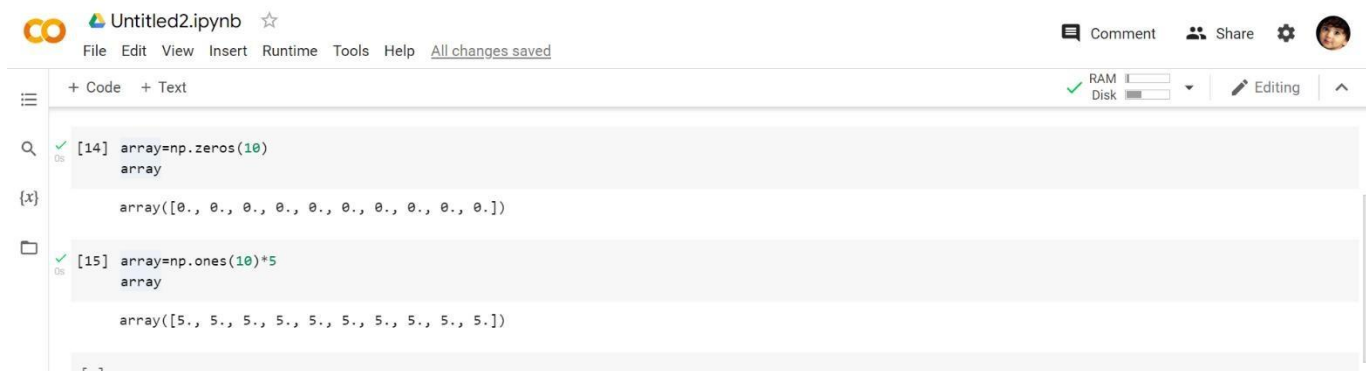
4.1 Create an array of 10 zeros?

4.2 Create an array of 10 fives?

```
4.1 array=np.zeros(10)
      Array
```

```
4.2 array=np.ones(10)*5
      Array
```

OUTPUT



The screenshot shows a Jupyter Notebook interface with two code cells. The first cell contains:

```
[14] array=np.zeros(10)
      array
```

The output is:

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

The second cell contains:

```
[15] array=np.ones(10)*5
      array
```

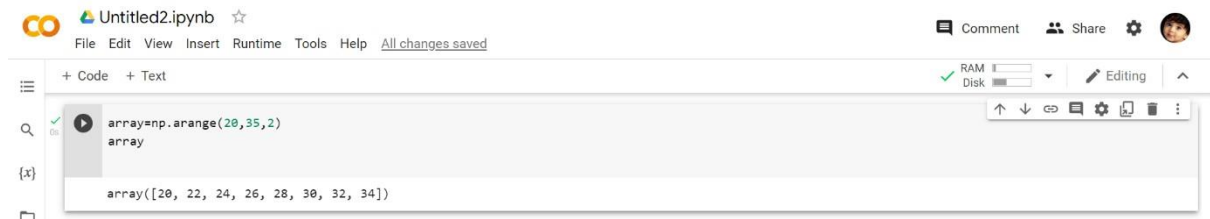
The output is:

```
array([5., 5., 5., 5., 5., 5., 5., 5., 5., 5.])
```

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

```
array=np.arange(20,35,2) array
```

OUTPUT



The screenshot shows a Jupyter Notebook interface with a file named 'Untitled2.ipynb'. The code cell contains the following Python code:

```
array=np.arange(20,35,2)
array
```


The output of the code is displayed below the code cell:

```
array([20, 22, 24, 26, 28, 30, 32, 34])
```

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

```
matrix=np.arange(0,9).reshape(3,3) matrix
```

OUTPUT



The screenshot shows a Jupyter Notebook interface with a file named 'Untitled2.ipynb'. The code cell contains the following Python code:

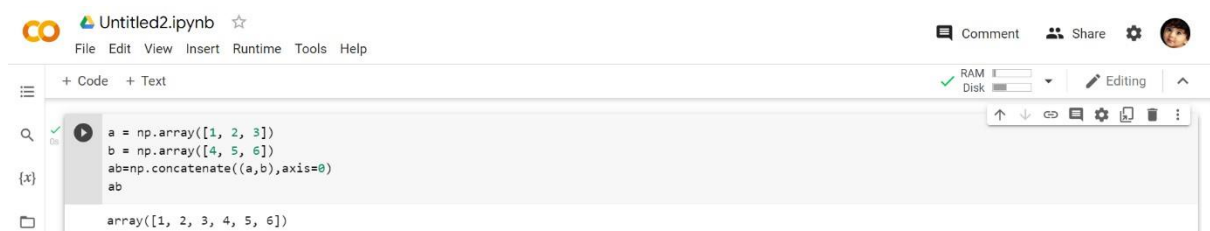
```
matrix=np.arange(0,9).reshape(3,3)
matrix
```

The output of the code is displayed below the code cell:

```
array([[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])

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



The screenshot shows a Jupyter Notebook interface with a file named 'Untitled2.ipynb'. The code cell contains the following Python code:

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

The output of the code is displayed below the code cell:

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

Pandas

8. Create a dataframe with 3 rows and 2 columns

```
import pandas as pd
data = [['vb', 10], ['hari', 15], ['prasath', 14]]
df = pd.DataFrame(data, columns=['Name', 'Age'])
df
```

OUTPUT



The screenshot shows a Jupyter Notebook interface with a code cell containing the following Python code:

```
[ ] import pandas as pd

data = [['vb', 10], ['hari', 15], ['prasath', 14]]
df = pd.DataFrame(data, columns=['Name', 'Age'])
df
```

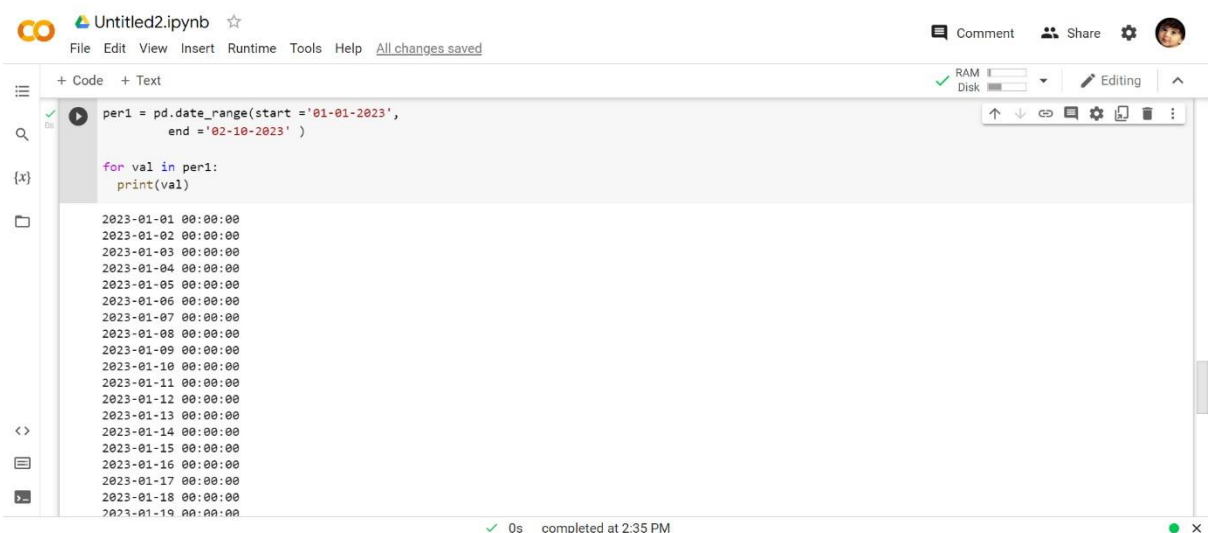
Below the code cell, the output of the DataFrame is displayed as a table:

	Name	Age
0	vb	10
1	hari	15
2	prasath	14

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

```
per1 = pd.date_range(start = '01-01-2023', end = '02-10-2023' )
for val in per1:
    print(val)
```

OUTPUT



The screenshot shows a Jupyter Notebook interface with a code cell containing the following Python code:

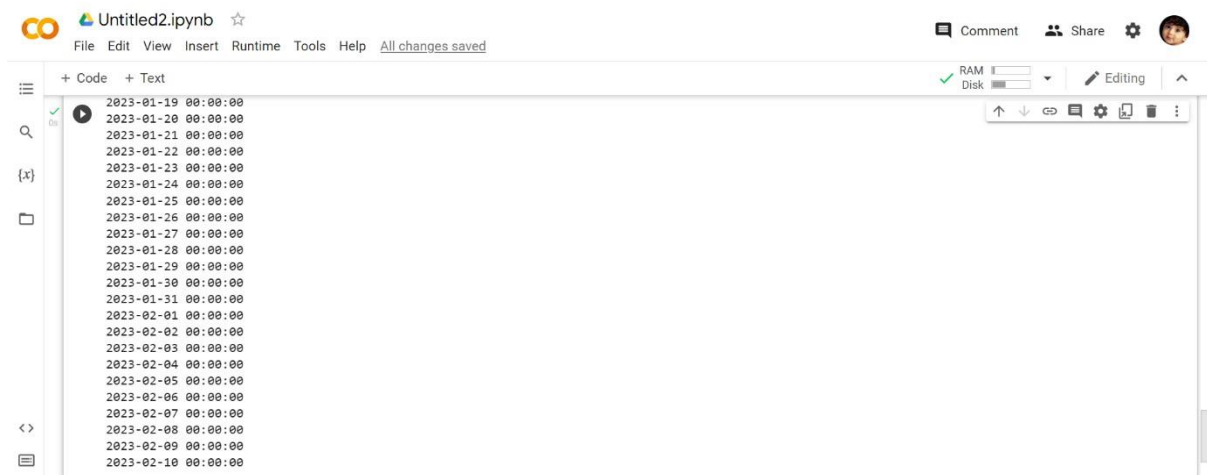
```
per1 = pd.date_range(start = '01-01-2023', end = '02-10-2023' )

for val in per1:
    print(val)
```

Below the code cell, the output of the date series is displayed as a list of timestamps:

```
2023-01-01 00:00:00
2023-01-02 00:00:00
2023-01-03 00:00:00
2023-01-04 00:00:00
2023-01-05 00:00:00
2023-01-06 00:00:00
2023-01-07 00:00:00
2023-01-08 00:00:00
2023-01-09 00:00:00
2023-01-10 00:00:00
2023-01-11 00:00:00
2023-01-12 00:00:00
2023-01-13 00:00:00
2023-01-14 00:00:00
2023-01-15 00:00:00
2023-01-16 00:00:00
2023-01-17 00:00:00
2023-01-18 00:00:00
2023-01-19 00:00:00
```

At the bottom of the notebook, a status bar indicates: "0s completed at 2:35 PM".



10. Create 2D list to DataFrame

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

In [35]:

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

In [58]:

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

```
# Create the pandas DataFrame df = pd.DataFrame(lists,
columns = ['s.no', 'name', 'Age'])
```

```
# print dataframe.
print(df )
```

OUTPUT



