

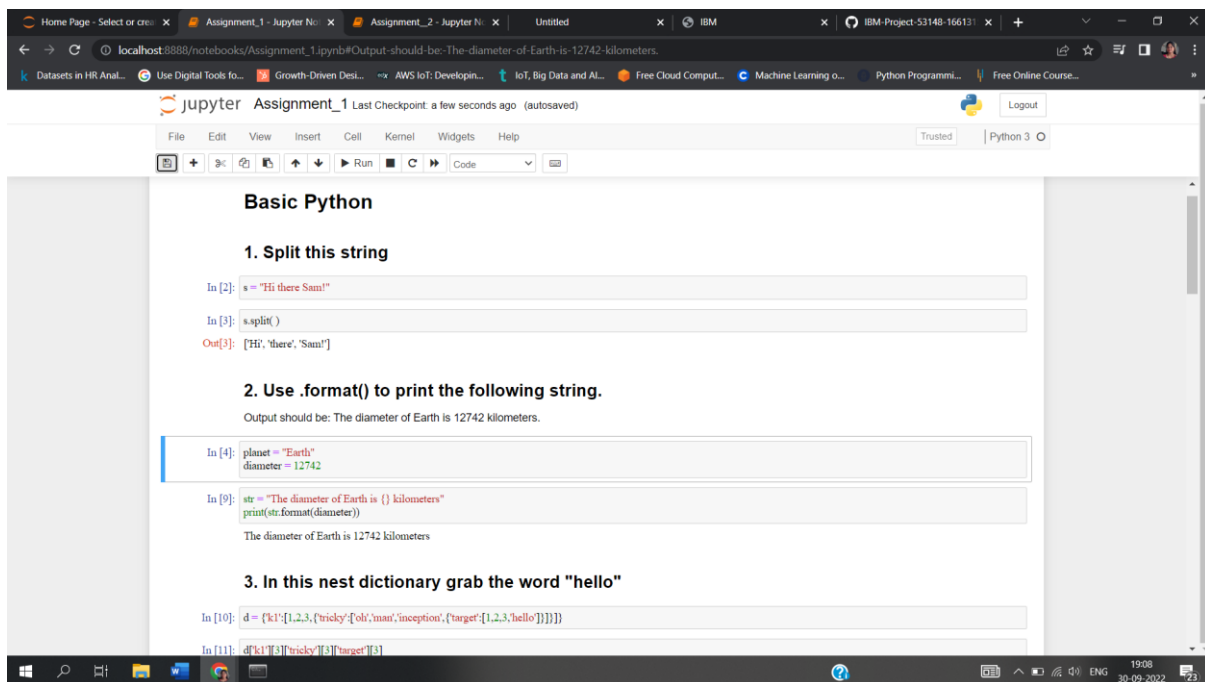
Assignment -1

Applied Data Science

| | |
|---------------------|----------------------------|
| Assignment Date | 19 September 2022 |
| Student Name | Ms. Navaneetha Krishnaveni |
| Student Roll Number | 721719104056 |
| Maximum Marks | 2 Marks |

1.Split the string.

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



The screenshot shows a Jupyter Notebook titled 'Assignment_1' with the following content:

```
File Edit View Insert Cell Kernel Widgets Help
In [2]: s = "Hi there Sam!"
In [3]: s.split()
Out[3]: ['Hi', 'there', 'Sam!']

2. Use .format() to print the following string.
Output should be: The diameter of Earth is 12742 kilometers.

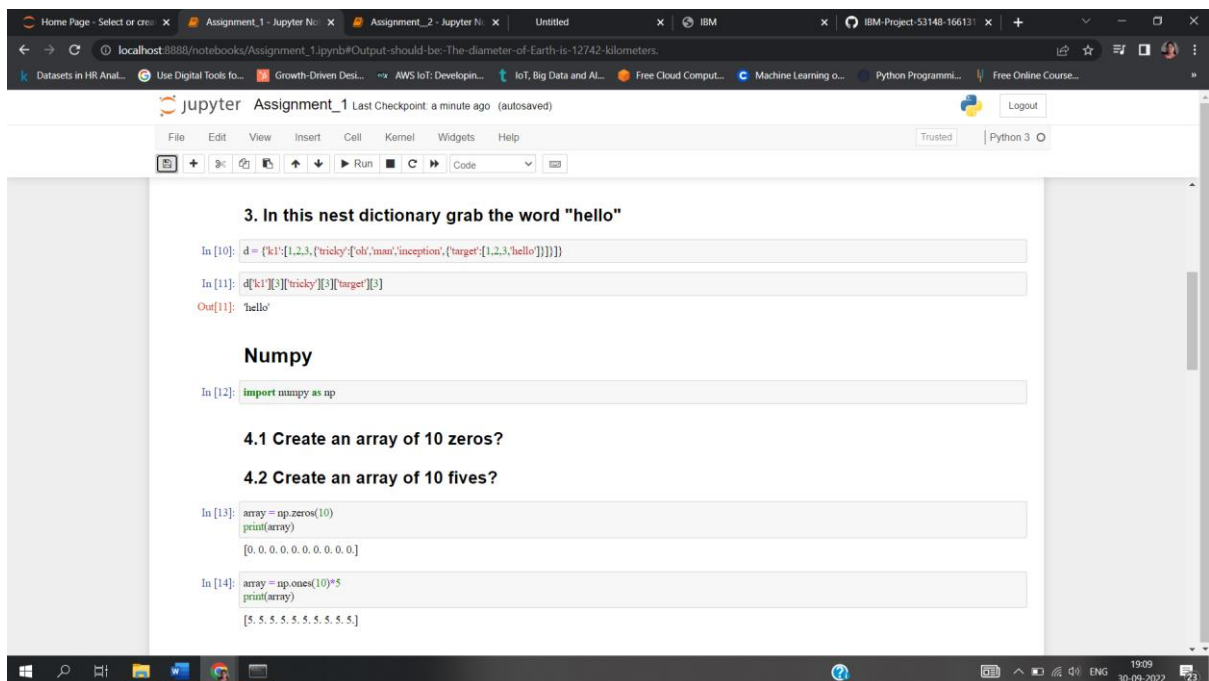
In [4]: planet = "Earth"
         diameter = 12742
In [9]: str = "The diameter of Earth is {} kilometers"
         print(str.format(diameter))
The diameter of Earth is 12742 kilometers

3. In this nest dictionary grab the word "hello"
In [10]: d = {'k1':{1,2,3,'tricky':{'oli','man','inception',{'target':[1,2,3,'hello']}}}}
In [11]: d['k1'][3]['tricky'][3]['target'][3]
```

3. In this nest dictionary grab the word “hello”.

4.1 Create an array of 10 zeroes.

4.2 Create an array of 10 fives.



The screenshot shows a Jupyter Notebook titled "Assignment_1" with the following content:

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

In [10]: d = {'k1': [1, 2, 3, {'tricky': {'ohi', 'man', 'inception', {'target': [1, 2, 3, 'hello']}}]}}
In [11]: d['k1'][3]['tricky'][3]['target'][3]
Out[11]: 'hello'

Numpy

In [12]: import numpy as np

4.1 Create an array of 10 zeros?

4.2 Create an array of 10 fives?

In [13]: array = np.zeros(10)
          print(array)
          [0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]

In [14]: array = np.ones(10)*5
          print(array)
          [5. 5. 5. 5. 5. 5. 5. 5. 5. 5.]
```

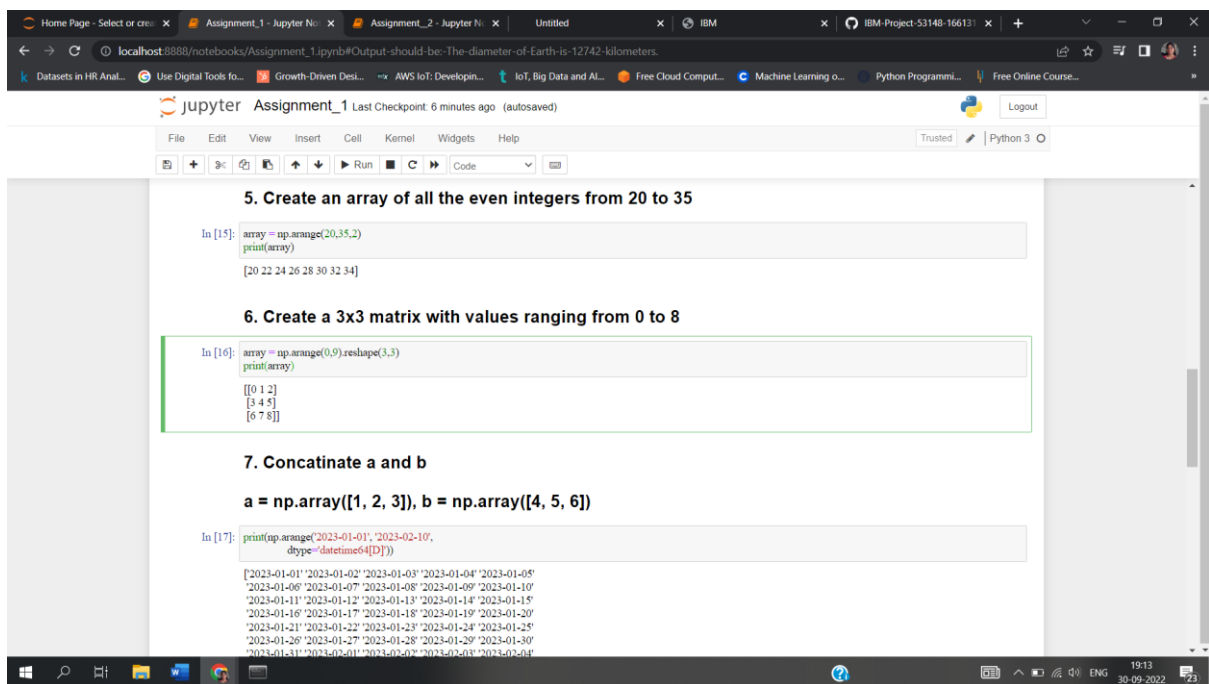
The interface includes a top toolbar with icons for file operations, a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help), and a status bar at the bottom showing the system date and time (30-09-2022, 19:09).

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

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

7. Concatenate a and b

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



The screenshot shows a Jupyter Notebook window titled "Assignment_1" with three tasks. Task 5: "Create an array of all the even integers from 20 to 35." The code is `array = np.arange(20,35,2)` and `print(array)`, resulting in `[20 22 24 26 28 30 32 34]`. Task 6: "Create a 3x3 matrix with values ranging from 0 to 8." The code is `array = np.arange(0,9).reshape(3,3)` and `print(array)`, resulting in `[[0 1 2] [3 4 5] [6 7 8]]`. Task 7: "Concatenate a and b". The code is `a = np.array([1, 2, 3]), b = np.array([4, 5, 6])` and `print(np.arange("2023-01-01", "2023-02-10", dtype='datetime64[D]'))`, resulting in a list of dates from "2023-01-01" to "2023-02-09".

```
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3
In [15]: array = np.arange(20,35,2)
         print(array)
         [20 22 24 26 28 30 32 34]

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

In [16]: array = np.arange(0,9).reshape(3,3)
         print(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])

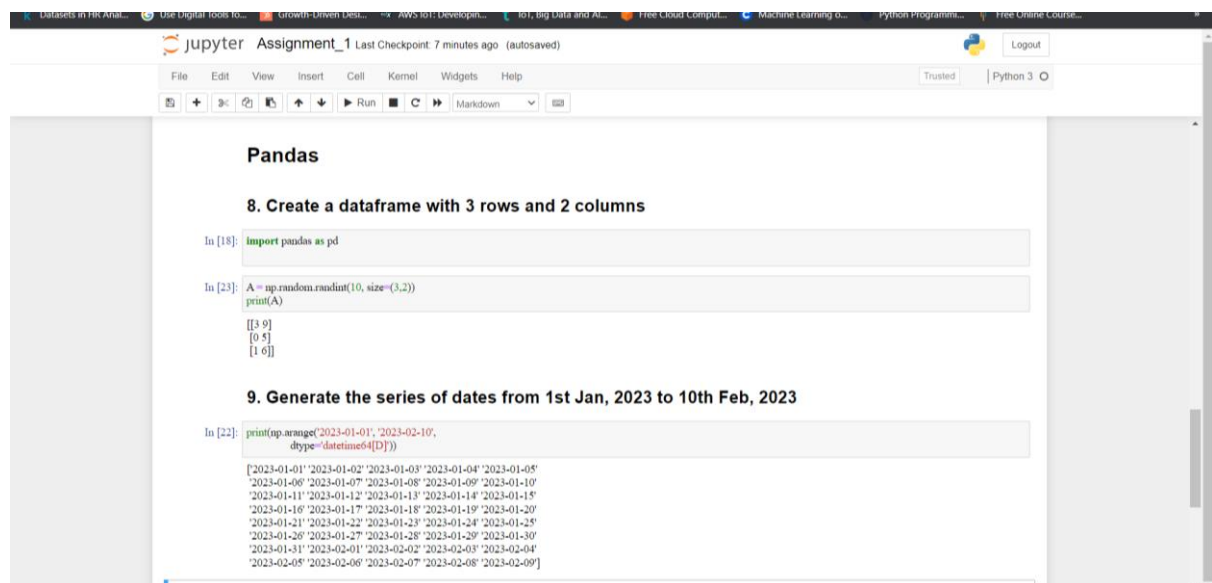
In [17]: print(np.arange("2023-01-01", "2023-02-10",
                        dtype='datetime64[D]'))
         ['2023-01-01' '2023-01-02' '2023-01-03' '2023-01-04' '2023-01-05'
          '2023-01-06' '2023-01-07' '2023-01-08' '2023-01-09' '2023-01-10'
          '2023-01-11' '2023-01-12' '2023-01-13' '2023-01-14' '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' '2023-02-03' '2023-02-04']
```

Pandas

8. Create a dataframe with 3 rows and 2 columns

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

10. Create 2D list to DataFrame



The screenshot shows a Jupyter Notebook titled 'Assignment_1' with the following content:

```
Pandas

8. Create a dataframe with 3 rows and 2 columns

In [18]: import pandas as pd

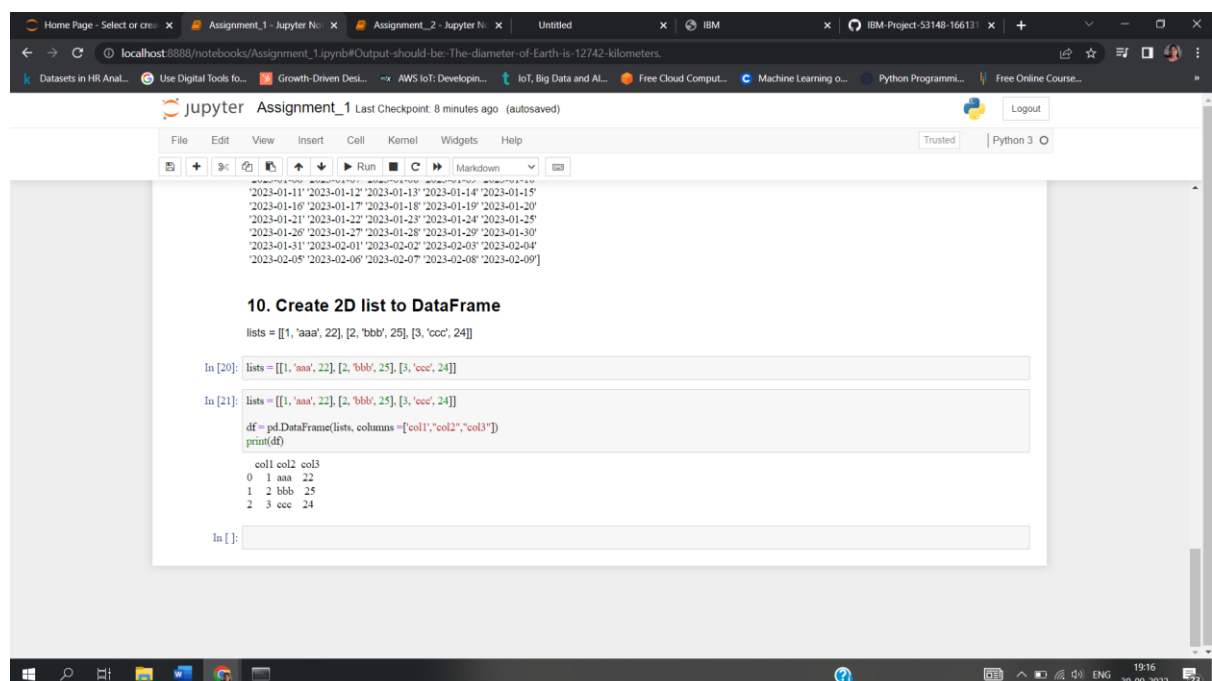
In [23]: A = np.random.randn(10, size=(3,2))
          print(A)

[[3. 9]
 [0. 5]
 [1. 6]]

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

In [22]: print(np.arange('2023-01-01', '2023-02-10',
                        dtype='datetime64[D]'))

['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' '2023-02-03' '2023-02-04'
 '2023-02-05' '2023-02-06' '2023-02-07' '2023-02-08' '2023-02-09']
```



The screenshot shows a Jupyter Notebook titled 'Assignment_1' with the following content:

```
10. Create 2D list to DataFrame

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

In [20]: lists = [[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]]

In [21]: lists = [[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]]
          df = pd.DataFrame(lists, columns=['col1', 'col2', 'col3'])
          print(df)

   col1 col2 col3
0     1  aaa   22
1     2  bbb   25
2     3  ccc   24

In []:
```