Assignment -1

Python Programming

Assignment Date	27 th September 2022
Student Name	Muralidharan.M
Student Roll Number	820319205022
Maximum Marks	2 Marks

Basic Python **1.**

Split this string In

[3]:

s = "Hi there Rocky!" In [31]: s**.**split()

Output [31]:

['Hi', 'there', 'Rocky!']

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 [5]:

Print ("The diameter of {} is {} kilometers."Format (planet, diameter))

The diameter of Earth is 12742 kilometers.

3. In this nest dictionary grab the word "hello" In [6]:

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

3,'hello']}]}]} In [7]:

d ['k1'] [3]['tricky'] [3]['target'][3] Output [7]:

'hello' **Numpy** In [1]:

import numpy **as** np array=np**.**arange (30,

71, 2)

Print ("Array of all the even integers from 30 to 70") Print (array)

Array of all the even integers from 30 to 70

[30 32 34 36 38 40 42 44 46 48 50 52 54 56 58 60 62 64 66 68 70]

4.1 Create an array of 10 zeroes?

4.2 Create an array of 10 fives?

In [9]:

np.zeros(10)

Output [9]:

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

In [10]:

np.ones (10) * 5

Output [10]:

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

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

In [14]:

Print (np.arange (20, 35, and 2))

[20 22 24 26 28 30 32 34]

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

In [15]: np.arange (0, 9). Reshape ((3, 3))

Output [11]

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 [30]:

Print ('\n---Results of a ([1, 2, 3]) and b ([4, 5, 6]) ---')

---Results of a ([1, 2, 3]) and b ([4, 5, 6]) ---

Pandas

8. Create a data frame with 3 rows and 2

columns In [28]: **import** pandas **as** pd

In [6]:

import pandas **as** pd

record = {"Name": ["arunkumar ","vijay",], "marks": ["20","46",],

"status":["fail","pass"]} df =

pd**.**DataFrame(record) df Output [6]:

	Name	marks	status
0	arunkumar	20	fail
1	vijay	46	pass

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

In [16]:

import pandas **as** pd

dti = pd**.**date_range(start="2023-01-01",end="10-02-

2023")**.**to_pydatetime()**.**tolist() dti

Output[16]:

[datetime.datetime (2023, 1, 1, 0, 0),

datetime.datetime (2023, 1, 2, 0, 0),

datetime.datetime (2023, 1, 3, 0, 0),

datetime.datetime (2023, 1, 4, 0, 0),

datetime.datetime (2023, 1, 5, 0, 0),

datetime.datetime (2023, 1, 6, 0, 0),

datetime.datetime (2023, 1, 7, 0, 0),

datetime.datetime (2023, 1, 8, 0, 0),

datetime.datetime (2023, 1, 0, 0),

7,

datetime.datetime (2023, 1, 10, 0, 0),

datetime.datetime (2023, 1, 11, 0, 0),

datetime.datetime (2023, 1, 12, 0, 0),

datetime.datetime (2023, 1, 13, 0, 0),

datetime.datetime (2023, 1, 14, 0, 0),

datetime.datetime (2023, 1, 15, 0, 0),

datetime.datetime (2023, 1, 16, 0, 0),

datetime.datetime (2023, 1, 17, 0, 0),

datetime.datetime (2023, 1, 18, 0, 0),

datetime.datetime (2023, 1, 19, 0, 0),

datetime.datetime (2023, 1, 20, 0, 0),

datetime.datetime (2023, 1, 21, 0, 0),

datetime.datetime (2023, 1, 22, 0, 0),

datetime.datetime (2023, 1, 23, 0, 0),

datetime.datetime (2023, 1, 24, 0, 0),

datetime.datetime (2023, 1, 25, 0, 0),

datetime.datetime (2023, 1, 26, 0, 0),

datetime.datetime (2023, 1, 27, 0, 0),

datetime.datetime (2023, 1, 28, 0, 0),

datetime.datetime (2023, 1, 29, 0, 0),

datetime.datetime (2023, 1, 30, 0, 0),

datetime.datetime (2023, 1, 31, 0, 0),

datetime.datetime (2023, 0),

datetime.datetime (2023, 0),

datetime.datetime (2023, 0),

datetime.datetime (2023, 2, 0, 0),

datetime.datetime (2023, 0, 0),

datetime.datetime (2023, 0), datetime.datetime

(2023,

datetime.datetime (2023,

datetime.datetime (2023,

datetime.datetime (2023, 2, 1, 0, 0),

datetime.datetime (2023, 2, 2, 0, 0),

datetime.datetime (2023, 2, 3, 0, 0),

2, 4, 0,

2, 5, 0,

2, 6, 0,

7,

3, 8,

2, 9, 0,

2, 10, 0, 0),

2, 11, 0, 0),

2, 12, 0, 0),

datetime.datetime (2023, 2, 13, 0, 0),

datetime.datetime (2023, 2, 14, 0, 0),

datetime.datetime (2023, 2, 15, 0, 0),

datetime.datetime (2023, 2, 16, 0, 0),

datetime.datetime (2023, 2, 17, 0, 0),

datetime.datetime (2023, 2, 18, 0, 0),

datetime.datetime (2023, 2, 19, 0, 0),

datetime.datetime (2023, 2, 20, 0, 0),

datetime.datetime (2023, 0),

datetime.datetime (2023, 0),

datetime.datetime (2023, 0),

datetime.datetime (2023, 3, 0, 0),

datetime.datetime (2023, 2, 21, 0, 0),

datetime.datetime (2023, 2, 22, 0, 0),

datetime.datetime (2023, 2, 23, 0, 0),

datetime.datetime (2023, 2, 24, 0, 0),

datetime.datetime (2023, 2, 25, 0, 0),

datetime.datetime (2023, 2, 26, 0, 0),

datetime.datetime (2023, 2, 27, 0, 0),

datetime.datetime (2023, 2, 28, 0, 0),

datetime.datetime (2023, 3, 1, 0, 0),

datetime.datetime (2023, 3, 2, 0, 0),

datetime.datetime (2023, 3, 3, 0, 0),

datetime.datetime (2023, 3, 4, 0, 0),

3, 5, 0,

3, 6, 0,

3, 7, 0,

8,

datetime.datetime (2023, 0),

datetime.datetime (2023, 0),

datetime.datetime (2023, 0),

datetime.datetime (2023, 4, 0, 0),

```
datetime.datetime (2023, 0, 0),
    datetime.datetime (2023,
    datetime.datetime (2023,
    datetime.datetime (2023,
            datetime.datetime (2023,
                                    4, 9,
                                3, 10, 0, 0),
                                3, 11, 0, 0),
                                3, 12, 0, 0),
                                3, 13, 0, 0),
    datetime.datetime (2023, 3, 14, 0,
    0), datetime.datetime (2023, 3, 15,
    0, 0), datetime.datetime (2023, 3,
    16, 0, 0), datetime.datetime (2023,
    3, 17, 0, 0), datetime.datetime
    (2023, 3, 18, 0, 0),
    datetime.datetime (2023, 3, 19, 0,
    0), datetime.datetime (2023, 3, 20,
    0, 0), datetime.datetime (2023, 3,
    21, 0, 0), datetime.datetime (2023,
    3, 22, 0, 0), datetime.datetime
    (2023, 3, 23, 0, 0),
    datetime.datetime (2023, 3, 24, 0,

    datetime.datetime (2023, 0),
    datetime.datetime (2023, 0),
    datetime.datetime (2023, 0),

    datetime.datetime (2023, 5, 0, 0),
```

datetime.datetime (2023, 0, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023,
0), datetime.datetime (2023, 3, 25,
0, 0), datetime.datetime (2023, 3,
26, 0, 0), datetime.datetime (2023,
3, 27, 0, 0), datetime.datetime
(2023, 3, 28, 0, 0),
datetime.datetime (2023, 3, 29, 0,
0), datetime.datetime (2023, 3, 30,
0, 0), datetime.datetime (2023, 3,
31, 0, 0), datetime.datetime (2023,
4, 1, 0, 0), datetime.datetime (2023,
4, 2, 0, 0),
4, 3, 0,
4, 4, 0,
4, 5, 0,
6,
5, 7,
4, 8, 0,
4, 9, 0,
datetime.datetime (2023,
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 6, 0, 0),

datetime.datetime (2023, 0, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023,
datetime.datetime (2023,
4, 10, 0, 0),
4, 11, 0, 0),
datetime.datetime (2023, 4, 12, 0,
0), datetime.datetime (2023, 4, 13,
0, 0), datetime.datetime (2023, 4,
14, 0, 0), datetime.datetime (2023,
4, 15, 0, 0), datetime.datetime
(2023, 4, 16, 0, 0),
datetime.datetime (2023, 4, 17, 0,
0), datetime.datetime (2023, 4, 18,
0, 0), datetime.datetime (2023, 4,
19, 0, 0), datetime.datetime (2023,
4, 20, 0, 0), datetime.datetime
(2023, 4, 21, 0, 0),
datetime.datetime (2023, 4, 22, 0,
0), datetime.datetime (2023, 4, 23,
0, 0), datetime.datetime (2023, 4,
24, 0, 0), datetime.datetime (2023,
4, 25, 0, 0), datetime.datetime

datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),

datetime.datetime (2023, 7, 0, 0),

datetime.datetime (2023, 0, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023,
(2023, 4, 26, 0, 0),
datetime.datetime (2023, 4, 27, 0,
0), datetime.datetime (2023, 4, 28,
0, 0), datetime.datetime (2023, 4,
29, 0, 0), datetime.datetime (2023,
4, 30, 0, 0), datetime.datetime
(2023, 5, 1, 0, 0),
5, 2, 0,
5, 3, 0,
5, 4, 0,
5,
6, 6,
5, 7, 0,
5, 8, 0,
5, 9, 0,
5, 10, 0, 0),
datetime.datetime (2023, 5, 11, 0,
0), datetime.datetime (2023, 5, 12,
datetime.datetime (2023,
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 8, 0, 0),

datetime.datetime (2023, 0, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023,
datetime.datetime (2023,
0, 0), datetime.datetime (2023, 5,
13, 0, 0), datetime.datetime (2023,
5, 14, 0, 0), datetime.datetime
(2023, 5, 15, 0, 0),
datetime.datetime (2023, 5, 16, 0,
0), datetime.datetime (2023, 5, 17,
0, 0), datetime.datetime (2023, 5,
18, 0, 0), datetime.datetime (2023,
5, 19, 0, 0), datetime.datetime
(2023, 5, 20, 0, 0),
datetime.datetime (2023, 5, 21, 0,
0), datetime.datetime (2023, 5, 22,
0, 0), datetime.datetime (2023, 5,
23, 0, 0), datetime.datetime (2023,
5, 24, 0, 0), datetime.datetime
(2023, 5, 25, 0, 0),
datetime.datetime (2023, 5, 26, 0,
0), datetime.datetime (2023, 5, 27,
0, 0), datetime.datetime (2023, 5,

datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),

datetime.datetime (2023, 9, 0, 0),

datetime.datetime (2023, 0, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023,
28, 0, 0), datetime.datetime (2023,
5, 29, 0, 0), datetime.datetime
(2023, 5, 30, 0, 0),
5, 31, 0, 0),
6, 1, 0,
6, 2, 0,

3,

datetime.datetime (2023,
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),

datetime.datetime (2023, 10, 0, 0),

```
datetime.datetime (2023,  0,  0),
   datetime.datetime (2023,  0),
   datetime.datetime (2023,  0),
   datetime.datetime (2023,  0),
   datetime.datetime (2023,  0),
                                  7, 4,
                                 6, 5, 0,
                                 6, 6, 0,
                                 6, 7, 0,
                                 6, 8, 0,
    datetime.datetime (2023, 6, 9, 0, 0),
      datetime.datetime (2023, 6, 10, 0,
     0), datetime.datetime (2023, 6, 11,
      0, 0), datetime.datetime (2023, 6,
     12, 0, 0), datetime.datetime (2023,
          6, 13, 0, 0), datetime.datetime
                        (2023, 6, 14, 0, 0),
      datetime.datetime (2023, 6, 15, 0,
     0), datetime.datetime (2023, 6, 16,
      0, 0), datetime.datetime (2023, 6,
     17, 0, 0), datetime.datetime (2023,
          6, 18, 0, 0), datetime.datetime
                        (2023, 6, 19, 0, 0),
      datetime.datetime (2023, 6, 20, 0,

   datetime.datetime (2023,
   datetime.datetime (2023,
   datetime.datetime (2023,            0),


  datetime.datetime (2023, 11,     0, 0),
```

```
datetime.datetime (2023, 0, 0),
    datetime.datetime (2023, 0),
    datetime.datetime (2023, 0),
    datetime.datetime (2023, 0),
    datetime.datetime (2023, 0),
        0), datetime.datetime (2023, 6, 21,
         0, 0), datetime.datetime (2023, 6,
        22, 0, 0), datetime.datetime (2023,
              6, 23, 0, 0), datetime.datetime
                              (2023, 6, 24, 0, 0),
         datetime.datetime (2023, 6, 25, 0,
        0), datetime.datetime (2023, 6, 26,
         0, 0), datetime.datetime (2023, 6,
        27, 0, 0), datetime.datetime (2023,
                    6, 28, 0, 0), 6, 29, 0, 0),
                                    6, 30, 0, 0),
                                    7, 1, 0,


                                        2,
                                     8, 3,
                                    7, 4, 0,
                                    7, 5, 0,
                                    7, 6, 0,
                                    7, 7, 0,

    datetime.datetime (2023,
    datetime.datetime (2023,
    datetime.datetime (2023,


  datetime.datetime (2023, 12,
```

datetime.datetime (2023, 0, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 7, 8, 0, 0),
datetime.datetime (2023, 7, 9, 0, 0),
datetime.datetime (2023, 7, 10, 0, 0), datetime.datetime (2023, 7, 11, 0, 0), datetime.datetime (2023, 7, 12, 0, 0), datetime.datetime (2023, 7, 13, 0, 0), datetime.datetime (2023, 7, 14, 0, 0),
datetime.datetime (2023, 7, 15, 0, 0), datetime.datetime (2023, 7, 16, 0, 0), datetime.datetime (2023, 7, 17, 0, 0), datetime.datetime (2023, 7, 18, 0, 0), datetime.datetime (2023, 7, 19, 0, 0),
datetime.datetime (2023, 7, 20, 0, 0), datetime.datetime (2023, 7, 21, 0, 0), datetime.datetime (2023, 7, 22, 0, 0), datetime.datetime (2023, 7, 23, 0, 0), datetime.datetime

datetime.datetime (2023,
datetime.datetime (2023,
datetime.datetime (2023,

datetime.datetime (2023, 13,

datetime.datetime (2023, 0, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
(2023, 7, 24, 0, 0),
datetime.datetime (2023, 7, 25, 0,
0), datetime.datetime (2023, 7, 26,
0, 0), datetime.datetime (2023, 7,
27, 0, 0), 7, 28, 0, 0), 7, 29, 0, 0),
7, 30, 0, 0),

31, 0, 0),
9, 1,
8, 2, 0,
8, 3, 0,
8, 4, 0,
8, 5, 0,
datetime.datetime (2023, 8, 6, 0, 0),
datetime.datetime (2023, 8, 7, 0, 0),
datetime.datetime (2023, 8, 8, 0, 0),
datetime.datetime (2023, 8, 9, 0, 0),
datetime.datetime (2023, 8, 10, 0,
0), datetime.datetime (2023, 8, 11,

datetime.datetime (2023,
datetime.datetime (2023,
datetime.datetime (2023,

datetime.datetime (2023, 14,

datetime.datetime (2023, 0, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
datetime.datetime (2023, 0),
0, 0), datetime.datetime (2023, 8,
12, 0, 0), datetime.datetime (2023,
8, 13, 0, 0), datetime.datetime
(2023, 8, 14, 0, 0),
datetime.datetime (2023, 8, 15, 0,
0), datetime.datetime (2023, 8, 16,
0, 0), datetime.datetime (2023, 8,
17, 0, 0), datetime.datetime (2023,
8, 18, 0, 0), datetime.datetime
(2023, 8, 19, 0, 0),
datetime.datetime (2023, 8, 20, 0,
0), datetime.datetime (2023, 8, 21,
0, 0), datetime.datetime (2023, 8,
22, 0, 0), datetime.datetime (2023,
8, 23, 0, 0), datetime.datetime
(2023, 8, 24, 0, 0),
datetime.datetime (2023, 8, 25, 0,
0), 8, 26, 0, 0),
8, 27, 0, 0),

datetime.datetime (2023,
datetime.datetime (2023,
datetime.datetime (2023,

datetime.datetime (2023, 15,

```
datetime.datetime   (2023,        0,   0),
    datetime.datetime     (2023,          0),
    datetime.datetime     (2023,          0),
    datetime.datetime     (2023,          0),
    datetime.datetime (2023,  0),
                                    8, 28, 0, 0),

                                         29, 0, 0),

    datetime.datetime (2023,
    datetime.datetime (2023,
    datetime.datetime (2023,

  datetime.datetime (2023, 16,
```

```
datetime.datetime    (2023,
   datetime.datetime   (2023,
   datetime.datetime   (2023,
   datetime.datetime   (2023,
   datetime.datetime (2023,
                                  10, 30, 0,
                                 8, 31, 0, 0),
                                 9, 1, 0, 0),
                                 9, 2, 0, 0),
                                 9, 3, 0, 0),
    datetime.datetime (2023, 9, 4, 0, 0),
    datetime.datetime (2023, 9, 5, 0, 0),
    datetime.datetime (2023, 9, 6, 0, 0),
    datetime.datetime (2023, 9, 7, 0, 0),
    datetime.datetime (2023, 9, 8, 0, 0),
    datetime.datetime (2023, 9, 9, 0, 0),
  datetime.datetime (2023, 9, 10, 0, 0),
  datetime.datetime (2023, 9, 11, 0, 0),
  datetime.datetime (2023, 9, 12, 0, 0),
  datetime.datetime (2023, 9, 13, 0, 0),
  datetime.datetime (2023, 9, 14, 0, 0),
  datetime.datetime (2023, 9, 15, 0, 0),
  datetime.datetime (2023, 9, 16, 0, 0),
  datetime.datetime (2023, 9, 17, 0, 0),
  datetime.datetime (2023, 9, 18, 0, 0),
  datetime.datetime (2023, 9, 19, 0, 0),
  datetime.datetime (2023, 9, 20, 0, 0),
  datetime.datetime (2023, 9, 21, 0, 0),
  datetime.datetime (2023, 9, 22, 0, 0),
   datetime.datetime (2023,  datetime.datetime
   (2023,  datetime.datetime (2023,

   datetime.datetime (2023, 17,
```

datetime.datetime (2023, 9, 23, 0, 0),

9, 24, 0, 0),

9, 25, 0, 0),

9, 26, 0, 0),

27, 0,

11, 28, 0,

9, 29, 0, 0),

9, 30, 0, 0),

10, 1, 0, 0),

10, 2, 0, 0)]

10. Create 2D list to Data Frame

Lists = [[1, 'acai', 22], [2, 'bob', 25], [3, 'ccc', 24]]

In [32]:

Lists = [[1, 'acai', 22], [2, 'bob', 25], [3, 'ccc', 24]]

In [4]:

lists ={"s.no": [1,2,3], "name":['aaa','bbb','ccc'], "value":[22,25,24] }

In [5]: pd.DataFrame(lists) Output[5]:

	s.no	name	value
0	1	aaa	22
1	2	bbb	25
2	3	ccc	24

In [7]:

pd.DataFrame(lists, index=["A","B","C"])

Output[7]:

	s.no	name	value

datetime.datetime (2023,

datetime.datetime (2023,

datetime.datetime (2023,

datetime.datetime (2023,

datetime.datetime (2023,

A	1	aaa	22
B	2	bbb	25

	s.no	**name**	**value**
C	3	ccc	24

In [ ]: