|  |  |
| --- | --- |
| Assignment Date | 28-09-2022 |
| Student Name | Ms Sujitha P |
| Student Roll Number | 4211191021098 |
| Maximum Marks | 2 Mark |

**Basic Python**

**1. Split this string**

In [ ]:s **=** "Hi there Sam!"

In [ ]:s**.**split()

Out[ ]:['Hi', 'there', 'Sam!']

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

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

In [ ]:

planet **=** "Earth"

diameter **=** 12742

In [ ]:print('The diameter of{} is {} kilometers.' **.** format(planet,diameter));

The diameter ofEarth is 12742 kilometers.

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

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

In [ ]:d['k1'][3]['tricky'][3]['target'][3]

Out[ ]:'hello'

**Numpy**

In [ ]:**import** numpy **as** np

**4.1 Create an array of 10 zeros?**

**4.2 Create an array of 10 fives?**

In [ ]:

a **=** np**.**zeros(10)

a

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

In [ ]:

b **=** np**.**ones(10)**\***5

b

Out[ ]: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 [ ]:

S **=** np**.**arange(20,35,2)

S

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

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

In [ ]:

b **=** np**.**arange(0,9)**.**reshape(3,3)

b

Out[ ]:array([[0, 1, 2],

[3, 4, 5],

[6, 7, 8]])

**7. Concatinate a and b**

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

In [ ]:

a **=** np**.**array([1,2,3])

b **=** np**.**array([4,5,6])

np**.**concatenate((a,b),axis**=**0)

Out[ ]:array([1, 2, 3, 4, 5, 6])

**Pandas**

**8. Create a dataframe with 3 rows and 2 columns**

In [ ]:

**import** pandas **as** pd

In [ ]:

d **=** {"names":["swetha","shaarmi","aswini"],"age":[20,19,20]}

df **=** pd**.**DataFrame(d)

df

Out[ ]:

|  | **Names** | **age** |
| --- | --- | --- |
| **0** | Sujitha.P | 21 |
| **1** | Swedha.M | 21 |
| **2** | Swetha.P | 20 |
| **3** | Yamini.P | 20 |

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

In [ ]:P **=** pd**.**date\_range(start**=**'1-1-2023',end**=**'10-2-2023')

**for** val **in** P:

print(val);

Out[ ]:

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

2023-01-20 00:00:00

2023-01-21 00:00:00

2023-01-22 00:00:00

2023-01-23 00:00:00

2023-01-24 00:00:00

2023-01-25 00:00:00

2023-01-26 00:00:00

2023-01-27 00:00:00

2023-01-28 00:00:00

2023-01-29 00:00:00

2023-01-30 00:00:00

2023-01-31 00:00:00

2023-02-01 00:00:00

2023-02-02 00:00:00

2023-02-03 00:00:00

2023-02-04 00:00:00

2023-02-05 00:00:00

2023-02-06 00:00:00

2023-02-07 00:00:00

2023-02-08 00:00:00

2023-02-09 00:00:00

2023-02-10 00:00:00

2023-02-11 00:00:00

2023-02-12 00:00:00

2023-02-13 00:00:00

2023-02-14 00:00:00

2023-02-15 00:00:00

2023-02-16 00:00:00

2023-02-17 00:00:00

2023-02-18 00:00:00

2023-02-19 00:00:00

2023-02-20 00:00:00

2023-02-21 00:00:00

2023-02-22 00:00:00

2023-02-23 00:00:00

2023-02-24 00:00:00

2023-02-25 00:00:00

2023-02-26 00:00:00

2023-02-27 00:00:00

2023-02-28 00:00:00

2023-03-01 00:00:00

2023-03-02 00:00:00

2023-03-03 00:00:00

2023-03-04 00:00:00

2023-03-05 00:00:00

2023-03-06 00:00:00

2023-03-07 00:00:00

2023-03-08 00:00:00

2023-03-09 00:00:00

2023-03-10 00:00:00

2023-03-11 00:00:00

2023-03-12 00:00:00

2023-03-13 00:00:00

2023-03-14 00:00:00

2023-03-15 00:00:00

2023-03-16 00:00:00

2023-03-17 00:00:00

2023-03-18 00:00:00

2023-03-19 00:00:00

2023-03-20 00:00:00

2023-03-21 00:00:00

2023-03-22 00:00:00

2023-03-23 00:00:00

2023-03-24 00:00:00

2023-03-25 00:00:00

2023-03-26 00:00:00

2023-03-27 00:00:00

2023-03-28 00:00:00

2023-03-29 00:00:00

2023-03-30 00:00:00

2023-03-31 00:00:00

2023-04-01 00:00:00

2023-04-02 00:00:00

2023-04-03 00:00:00

2023-04-04 00:00:00

2023-04-05 00:00:00

2023-04-06 00:00:00

2023-04-07 00:00:00

2023-04-08 00:00:00

2023-04-09 00:00:00

2023-04-10 00:00:00

2023-04-11 00:00:00

2023-04-12 00:00:00

2023-04-13 00:00:00

2023-04-14 00:00:00

2023-04-15 00:00:00

2023-04-16 00:00:00

2023-04-17 00:00:00

2023-04-18 00:00:00

2023-04-19 00:00:00

2023-04-20 00:00:00

2023-04-21 00:00:00

2023-04-22 00:00:00

2023-04-23 00:00:00

2023-04-24 00:00:00

2023-04-25 00:00:00

2023-04-26 00:00:00

2023-04-27 00:00:00

2023-04-28 00:00:00

2023-04-29 00:00:00

2023-04-30 00:00:00

2023-05-01 00:00:00

2023-05-02 00:00:00

2023-05-03 00:00:00

2023-05-04 00:00:00

2023-05-05 00:00:00

2023-05-06 00:00:00

2023-05-07 00:00:00

2023-05-08 00:00:00

2023-05-09 00:00:00

2023-05-10 00:00:00

2023-05-11 00:00:00

2023-05-12 00:00:00

2023-05-13 00:00:00

2023-05-14 00:00:00

2023-05-15 00:00:00

2023-05-16 00:00:00

2023-05-17 00:00:00

2023-05-18 00:00:00

2023-05-19 00:00:00

2023-05-20 00:00:00

2023-05-21 00:00:00

2023-05-22 00:00:00

2023-05-23 00:00:00

2023-05-24 00:00:00

2023-05-25 00:00:00

2023-05-26 00:00:00

2023-05-27 00:00:00

2023-05-28 00:00:00

2023-05-29 00:00:00

2023-05-30 00:00:00

2023-05-31 00:00:00

2023-06-01 00:00:00

2023-06-02 00:00:00

2023-06-03 00:00:00

2023-06-04 00:00:00

2023-06-05 00:00:00

2023-06-06 00:00:00

2023-06-07 00:00:00

2023-06-08 00:00:00

2023-06-09 00:00:00

2023-06-10 00:00:00

2023-06-11 00:00:00

2023-06-12 00:00:00

2023-06-13 00:00:00

2023-06-14 00:00:00

2023-06-15 00:00:00

2023-06-16 00:00:00

2023-06-17 00:00:00

2023-06-18 00:00:00

2023-06-19 00:00:00

2023-06-20 00:00:00

2023-06-21 00:00:00

2023-06-22 00:00:00

2023-06-23 00:00:00

2023-06-24 00:00:00

2023-06-25 00:00:00

2023-06-26 00:00:00

2023-06-27 00:00:00

2023-06-28 00:00:00

2023-06-29 00:00:00

2023-06-30 00:00:00

2023-07-01 00:00:00

2023-07-02 00:00:00

2023-07-03 00:00:00

2023-07-04 00:00:00

2023-07-05 00:00:00

2023-07-06 00:00:00

2023-07-07 00:00:00

2023-07-08 00:00:00

2023-07-09 00:00:00

2023-07-10 00:00:00

2023-07-11 00:00:00

2023-07-12 00:00:00

2023-07-13 00:00:00

2023-07-14 00:00:00

2023-07-15 00:00:00

2023-07-16 00:00:00

2023-07-17 00:00:00

2023-07-18 00:00:00

2023-07-19 00:00:00

2023-07-20 00:00:00

2023-07-21 00:00:00

2023-07-22 00:00:00

2023-07-23 00:00:00

2023-07-24 00:00:00

2023-07-25 00:00:00

2023-07-26 00:00:00

2023-07-27 00:00:00

2023-07-28 00:00:00

2023-07-29 00:00:00

2023-07-30 00:00:00

2023-07-31 00:00:00

2023-08-01 00:00:00

2023-08-02 00:00:00

2023-08-03 00:00:00

2023-08-04 00:00:00

2023-08-05 00:00:00

2023-08-06 00:00:00

2023-08-07 00:00:00

2023-08-08 00:00:00

2023-08-09 00:00:00

2023-08-10 00:00:00

2023-08-11 00:00:00

2023-08-12 00:00:00

2023-08-13 00:00:00

2023-08-14 00:00:00

2023-08-15 00:00:00

2023-08-16 00:00:00

2023-08-17 00:00:00

2023-08-18 00:00:00

2023-08-19 00:00:00

2023-08-20 00:00:00

2023-08-21 00:00:00

2023-08-22 00:00:00

2023-08-23 00:00:00

2023-08-24 00:00:00

2023-08-25 00:00:00

2023-08-26 00:00:00

2023-08-27 00:00:00

2023-08-28 00:00:00

2023-08-29 00:00:00

2023-08-30 00:00:00

2023-08-31 00:00:00

2023-09-01 00:00:00

2023-09-02 00:00:00

2023-09-03 00:00:00

2023-09-04 00:00:00

2023-09-05 00:00:00

2023-09-06 00:00:00

2023-09-07 00:00:00

2023-09-08 00:00:00

2023-09-09 00:00:00

2023-09-10 00:00:00

2023-09-11 00:00:00

2023-09-12 00:00:00

2023-09-13 00:00:00

2023-09-14 00:00:00

2023-09-15 00:00:00

2023-09-16 00:00:00

2023-09-17 00:00:00

2023-09-18 00:00:00

2023-09-19 00:00:00

2023-09-20 00:00:00

2023-09-21 00:00:00

2023-09-22 00:00:00

2023-09-23 00:00:00

2023-09-24 00:00:00

2023-09-25 00:00:00

2023-09-26 00:00:00

2023-09-27 00:00:00

2023-09-28 00:00:00

2023-09-29 00:00:00

2023-09-30 00:00:00

2023-10-01 00:00:00

2023-10-02 00:00:00

**10. Create 2D list to DataFrame**

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

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

In [ ]:

df **=** pd**.**DataFrame(lists)

df

Out[ ]:

|  | **0** | **1** | **2** |
| --- | --- | --- | --- |
| **0** | 1 | aaa | 22 |
| **1** | 2 | bbb | 25 |
| **2** | 3 | ccc | 24 |
| **3** | 4 | ddd | 26 |