

Assignment -1
Python Programming

Assignment Date	17 September 2022
Student Name	B.M.DON DAVIES
Student Roll Number	E1195016
Maximum Marks	2 Marks

Question-1:

1. Split this string

Solution:

`s.split()`

`s = "Hi there Sam!"`

`s="Hi there Sam"`

`s=s.split()`

`print(s);`

```
In [ ]: s = "Hi there Sam!"

In [ ]: s="Hi there Sam"
        s=s.split()
        print(s);

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

Question-2:

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

Output should be: The diameter of Earth is 12742 kilometers

Solution :

```
planet = "Earth"
```

```
diameter = 12742
```

```
planet = "Earth"
```

```
diameter = 12742
```

```
print("The diameter of {} is {}".format(planet, diameter))
```

The diameter of Earth is 12742 kilometers

```
In [ ]: planet = "Earth"
        diameter = 12742

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

The diameter of Earth is 12742 kilometers.
```

Question -3:

In this nest dictionary grab the word "hello"

Solution:

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

```
d= [1,2,[3,4],[5,[100,200,['hello']],23,11],1,7]
```

```
d[3][1][2][0]
```

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

[ ]: d= [1,2,[3,4],[5,[100,200,['hello']],23,11],1,7]
     d[3][1][2][0]

[ ]: 'hello'
```

Question-4.

Create an array of 10 zeros?

Create an array of 10 fives?

Solution:

4.1.

```
import numpy as np
```

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

```
print("An array of 10 zeros:")
```

```
print(array)
```

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

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

4.2.import numpy as np

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

```
print("An array of 10 fives:")
```

```
print(array)
```

```
In [ ]: 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

Solution:

```
import numpy as np
array=np.arange(20,35,2)
print("Array of all the even integers from 20 to 35")
print(array)
```

```
In [ ]: import numpy as np
        array=np.arange(20,35,2)
        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]
```

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

Solution:

```
import numpy as np
x = np.arange(0,9).reshape(3,3)
print(x)
```

```
In [1]: import numpy as np
        x = np.arange(0, 9).reshape(3,3)
        print(x)

[[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])
```

Solution:

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

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

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

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

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

Pandas

8. Create a dataframe with 3 rows and 2 columns

Solution:

```
import pandas as pd
```

```
import pandas as pd
```

```
# dictionary
```

```
record = {"Name": ["Ram", "Jack", "Rose"],
          "Marks": [29, 25, 23] }
```

```
# converting record into
```

```
# pandas dataframe
```

```
df = pd.DataFrame(record)
```

```
# select first 3 rows
```

```
# from the dataframe
```

```
df1 = df.head(3)
```

```
# show the dataframe
```

```
df1
```

```
In [ ]: import pandas as pd

In [3]: import pandas as pd

# dictionary
record = {"Name": ["Ram", "Jack", "Rose"],
          "Marks": [29, 25, 23]}

# converting record into
# pandas dataframe
df = pd.DataFrame(record)

# select first 3 rows
# from the dataframe
df1 = df.head(3)

# show the dataframe
df1
```

```
Out[3]:
```

	Name	Marks
0	Ram	29
1	Jack	25
2	Rose	23

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

Solution:

from datetime import date, timedelta

start_date = date(2023, 1, 1)

end_date = date(2023, 2, 10) # perhaps date.now()

delta = end_date - start_date # returns timedelta

for i in range(delta.days + 1):

day = start_date + timedelta(days=i)

```
In [4]: from datetime import date, timedelta

start_date = date(2023, 1, 1)
end_date = date(2023, 2, 10) # perhaps date.now()

delta = end_date - start_date # returns timedelta

for i in range(delta.days + 1):
    day = start_date + timedelta(days=i)
    print(day)
```

```
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
2023-02-10
```

10. Create 2D list to DataFrame

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

Solution:

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

List1

```
lst = [['aaa', 22], ['bbb', 25], ['ccc', 24]]
```

creating df object with columns specified

```
df = pd.DataFrame(lst)
```

```
print(df)
```

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

```
In [5]: import pandas as pd  
  
# List1  
lst = [['aaa', 22], ['bbb', 25], ['ccc', 24]]  
  
# creating df object with columns specified  
df = pd.DataFrame(lst)  
print(df)
```

```
   0  1  
0  aaa  22  
1  bbb  25  
2  ccc  24
```