Assignment -1

Python Programming

Assignment Date	27 September 2022
Student Name	K.B DINESH ROHIT
Student Roll Number	412519104030
Maximum Marks	2 Marks

Question-1:

Split the string

Solution :

```
s="hi there sam!"

print(s.split())
```

```
s="hi there sam!"
print(s.split())
```

```
['hi', 'there', 'sam!']
```

Question-2:

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

Solution:

```
planet="Earth"
diameter=12742
star ="The diameter of {p} is {k} Kilometers."
print(star.format(p=planet,k=diameter))
```

```
planet="Earth"
diameter=12742
star ="The diameter of {p} is {k} Kilometers."
print(star.format(p=planet,k=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['k1'][3]['tricky'][3]['target'][3]
```

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

```
'hello'
```

Question-4:

Create an array of 10 zeros?

Create an array of 10 fives?

Solution:

```
array=np.zeros(10)
print("An array of 10 zeros")
print(array)
array=np.ones(10)*5
print("An array of 10 fives")
print(array)
```

```
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.]
```

```
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.]
```

Question-5:

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

Solution:

```
a=np.arange(20,35,2)
print(a)
```

```
a=np.arange(20,35,2)
print(a)
```

```
[20 22 24 26 28 30 32 34]
```

Question-6:

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

Solution:

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

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

```
[[0 1 2]
 [3 4 5]
 [6 7 8]]
```

Question-7:

Concatenate a and b

Solution:

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

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

```
[1 2 3]
[1 2 3]
```

Question-8:

Create a dataframe with 3 rows and 2 columns

Solution:

```
import pandas as pd
data=[['vamsi',10],['mahesh',20],['sai',30]]
a=pd.DataFrame(data,columns=['Name','Age',])
print(a)
```

```
import pandas as pd
```

```
data=[['vamsi',10],['mahesh',20],['sai',30]]
a=pd.DataFrame(data,columns=['Name','Age',])
print(a)
```

```
     Name  Age
0   vamsi   10
1  mahesh   20
2     sai   30
```

Question-9:

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

Solution:

```
from datetime import datetime, timedelta

def date_range(start, end):
    delta = end - start  # as timedelta
    days = [start + timedelta(days=i) for i in range(delta.days + 1)]
    return days

start_date = datetime(2023, 1, 1)
end_date = datetime(2023, 2, 10)

print(*date_range(start_date,end_date))
```

```
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
```

Question-10:

Create 2D list to DataFrame

Solution:

```
lists = [[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]]
df=pd.DataFrame(lists,columns=['Number','FName','Age'])
print(df)
```

```
   Number FName  Age
0       1   aaa   22
1       2   bbb   25
2       3   ccc   24
```