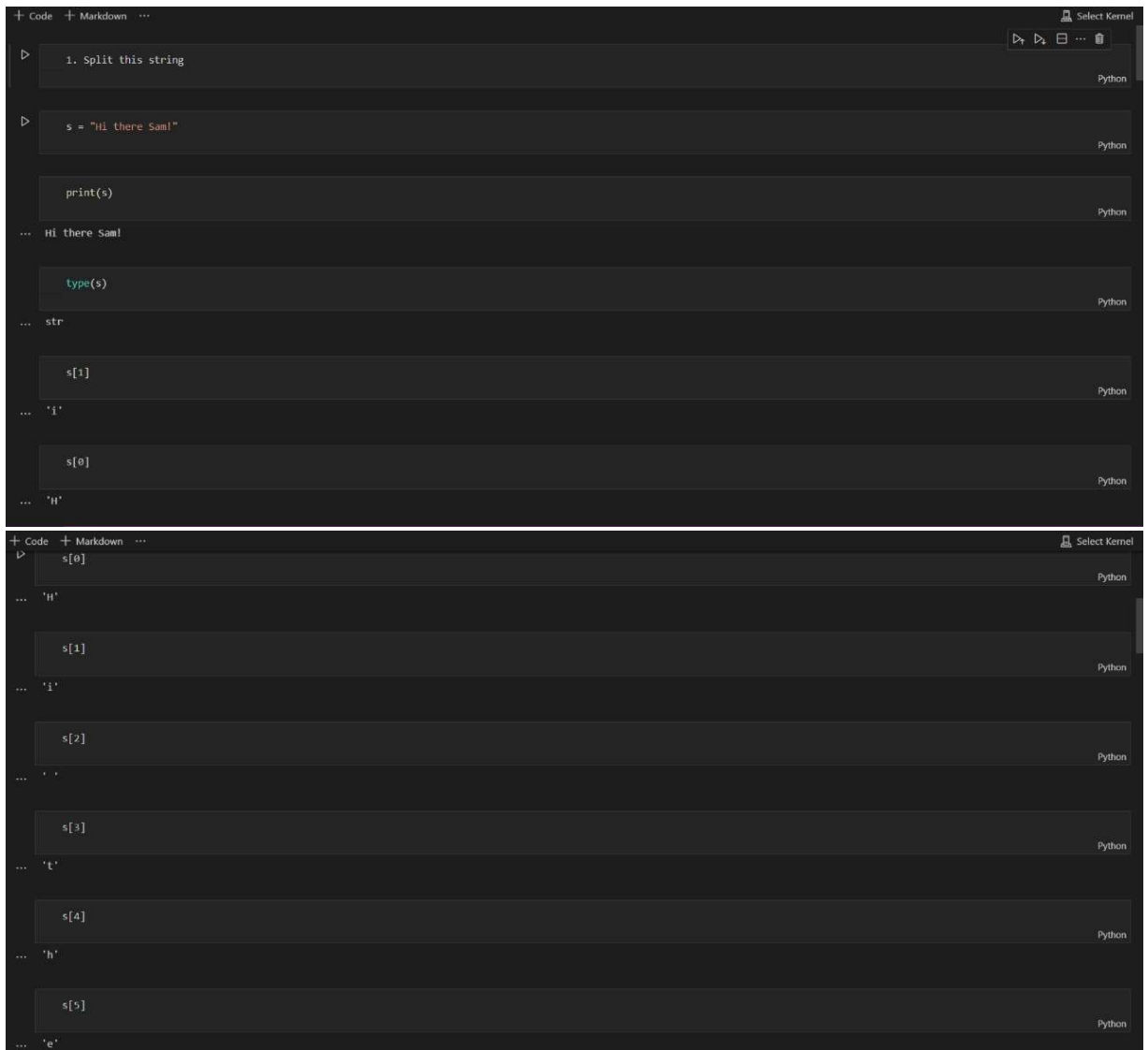


ASSIGNMENT – 1

1. Split this string



The image displays two screenshots of a Jupyter Notebook interface, showing the process of splitting a string and accessing its individual characters.

Top Screenshot:

- Cell 1: Contains the text "1. Split this string".
- Cell 2: Contains the code `s = "Hi there Sam!"`. The output is `Python`.
- Cell 3: Contains the code `print(s)`. The output is `Hi there Sam!`.
- Cell 4: Contains the code `type(s)`. The output is `str`.
- Cell 5: Contains the code `s[1]`. The output is `'i'`.
- Cell 6: Contains the code `s[0]`. The output is `'H'`.

Bottom Screenshot:

- Cell 7: Contains the code `s[0]`. The output is `'H'`.
- Cell 8: Contains the code `s[1]`. The output is `'i'`.
- Cell 9: Contains the code `s[2]`. The output is `' '`.
- Cell 10: Contains the code `s[3]`. The output is `'t'`.
- Cell 11: Contains the code `s[4]`. The output is `'h'`.
- Cell 12: Contains the code `s[5]`. The output is `'e'`.

```
+ Code + Markdown ... Select Kernel
... 'e'

s[6] Python

... 'p'

s[7] Python

... 'e'

s[8] Python

... ' '

s[9] Python

... 'S'

s[10] Python

... 'a'

s[11] Python

... 'm'

s[12] Python

... 'I'
```

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

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

txt="The diameter of {planet} is {diameter} kilometers".format(planet="Earth",diameter=12742) Python

print(txt) Python

... The diameter of Earth is 12742 kilometers

In this nest dictionary grab the word "hello" Python

+ Code + Markdown ... Select Kernel
... The diameter of Earth is 12742 kilometers

In this nest dictionary grab the word "hello" Python

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

print(d['k1'][3]['tricky'][3]['target'][3]) Python

... hello

import numpy as np Python
```

4.1 Create an array of 10 zeros?

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

```
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]
```

```
+ Code + Markdown ...
```

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

```
Create an array of 10 fives?
```

```
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

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

```
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

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

```
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])
```

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

Python

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

8. Create a dataframe with 3 rows and 2 columns

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

Python

```
import pandas as pd
import numpy as np
A = np.random.randint(10, size=(3,2))
df = pd.DataFrame(A)
```

Python

```
df
```

Python

```
...
0 1
0 9 8
1 3 1
2 0 6
```

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

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

Python

```
import pandas as pd
per1 = pd.date_range(start='1-1-2023',
                    end='10-2-2023', freq='M')
```

+ Code + Markdown ...

Select Kernel

```
import pandas as pd

per1 = pd.date_range(start='1-1-2023',
                    end='10-2-2023', freq='M')

for val in per1:
    print(val)
```

Python

```
... 2023-01-31 00:00:00
2023-02-28 00:00:00
2023-03-31 00:00:00
2023-04-30 00:00:00
2023-05-31 00:00:00
2023-06-30 00:00:00
2023-07-31 00:00:00
2023-08-31 00:00:00
2023-09-30 00:00:00
```

```
10. Create 2D list to DataFrame
lists = [[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]]
```

```
10. Create 2D list to DataFrame
lists = [[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]]

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

	column 1	column 2	column 3
0	1	aaa	22
1	2	bbb	25
2	3	ccc	24

```
+ Code + Markdown ...
2023-07-31 00:00:00
2023-08-31 00:00:00
2023-09-30 00:00:00

10. Create 2D list to DataFrame
lists = [[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]]

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

	column 1	column 2	column 3
0	1	aaa	22
1	2	bbb	25
2	3	ccc	24