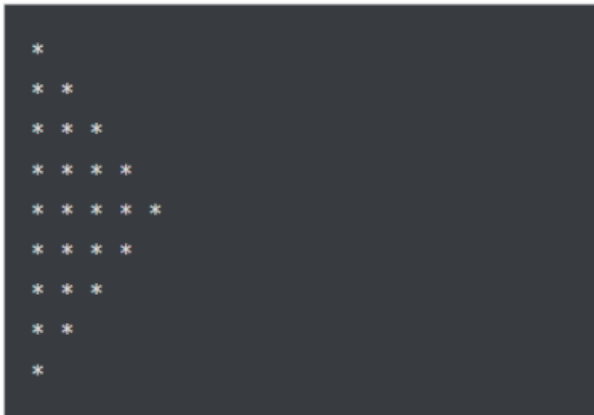


GitHub: <https://github.com/sandy100061/MachineLearningAssignment/tree/main/Assignment2>

Video Link: https://drive.google.com/file/d/13doTB3vm1p3kA6bJ4hmTn0UdQLh2ahak/view?usp=drive_link

Question 1

1. Use a python code to display the following star pattern using the for loop.



Code :-

```
max = 5
i = 0
while i < max:
    j = 0
    while j <= i:
        print("*", end = " ")
        j += 1
    print('')
    i += 1

i = max - 1
while i > 0:
    j = i
    while j > 0:
        print("*", end = " ")
        j -= 1
    print('')
    i -= 1
```

 [Open in Colab](#)

In [20]: *#1. Use a python code to display the following star pattern using the for loop*

```
max = 5
i = 0
while i < max:
    j = 0
    while j <= i:
        print("*", end = " ")
        j += 1
    print('\n')
    i += 1

i = max - 1
while i > 0:
    j = i
    while j > 0:
        print("*", end = " ")
        j -= 1
    print('\n')
    i -= 1
```


```
*
* *
* * *
* * * *
* * * * *
* * * * *
* * * *
* * *
* *
*
*
```

2. Use looping to output the elements from a provided list present at odd indexes. my_list = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]

Code :-

```
my_list = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]

for index in range(0, len(my_list), 2):
    print(my_list[index])
```

 Open in Colab

```
In [22]: #2. Use looping to output the elements from a provided list present at odd indexes.
my_list = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]

for index in range(0, len(my_list), 2):
    print(my_list[index])
```

```
10
30
50
70
90
```

3. Write a code that appends the type of elements from a given list.

Input

x = [23, 'Python', 23.98]

Expected output

[23, 'Python', 23.98]

[<class 'int'>, <class 'str'>, <class 'float'>]

Code :-

```
x = [23, 'Python', 23.98]
y = []
for item in x:
    y.append(type(item))

print(f'Input : {x}')
print(f'Expected Output : {y}')
```



```
In [26]: """3. Write a code that appends the type of elements from a given list.
Input
x = [23, 'Python', 23.98]
Expected output
[23, 'Python', 23.98]
[, , ]
"""
x = [23, 'Python', 23.98]
y = []
for item in x:
    y.append(type(item))

print(f'Input : {x}')
print(f'Expected Output : {y}')
```

```
Input : [23, 'Python', 23.98]
Expected Output : [<class 'int'>, <class 'str'>, <class 'float'>]
```

4. Write a function that takes a list and returns a new list with unique items of the first list.

Sample List: [1,2,3,3,3,3,4,5]

Unique List: [1, 2, 3, 4, 5]

Code :-

```
def getUniqueList(inputList: list):
    result = set()
    for item in inputList:
        result.add(item)
    return list(result)

x = [1,2,3,3,3,3,4,5]
result = getUniqueList(x)
print(f'Input list : {x}')
print(f'Unique list : {result}')
```

 Open in Colab

```
In [81]: """
4. Write a function that takes a list and returns a new list with unique items of the first list.
Sample List: [1,2,3,3,3,3,4,5]
Unique List: [1, 2, 3, 4, 5]
"""

def getUniqueList(inputList: list):
    result = set()
    for item in inputList:
        result.add(item)
    return list(result)

x = [1,2,3,3,3,3,4,5]
result = getUniqueList(x)
print(f'Input list : {x}')
print(f'Unique list : {result}')

Input list : [1, 2, 3, 3, 3, 3, 4, 5]
Unique list : [1, 2, 3, 4, 5]
```

5. Write a function that accepts a string and calculate the number of upper-case letters and lower-case

letters.

Input String: 'The quick Brow Fox'

Expected Output:


No. of Upper-case characters: 3

No. of Lower-case Characters: 12

Code :-

```
def calculateLetterCount(input: str):
    upperCount = 0
    lowerCount = 0
    for item in input:
        if item == ' ':
            continue
        if item.upper() == item:
            upperCount += 1
        else:
            lowerCount += 1
    print(f'No. of Upper-case characters: {upperCount}')
    print(f'No. of Lower-case characters: {lowerCount}')

inputStr = 'The quick Brow Fox'
print(f'Input String: '{inputStr}''')
calculateLetterCount(inputStr)
```

 [Open in Colab](#)

```
In [88]: """
5. Write a function that accepts a string and calculate the number of upper-case letters and lower-case
letters.
Input String: 'The quick Brow Fox'
Expected Output:
No. of Upper-case characters: 3
No. of Lower-case Characters: 12
"""
def calculateLetterCount(input: str):
    upperCount = 0
    lowerCount = 0
    for item in input:
        if item == ' ':
            continue
        if item.upper() == item:
            upperCount += 1
        else:
            lowerCount += 1
    print(f'No. of Upper-case characters: {upperCount}')
    print(f'No. of Lower-case characters: {lowerCount}')

inputStr = 'The quick Brow Fox'
print(f"Input String: '{inputStr}'")
calculateLetterCount(inputStr)
```

```
Input String: 'The quick Brow Fox'
No. of Upper-case characters: 3
No. of Lower-case characters: 12
```