CS210 : ARTIFICIAL INTELLIGENCE LAB

LAB ASSIGNMENT 1: AI & Python

Date Assigned : 01/01/24 **Date Submitted : 07/01/24**

Submitted By:

Name: Akshat Sahu

Roll No: U22CS034

Branch: CSE

Semester: 4th Sem

Division : A

Submitted To: Dr. Chandra Prakash

Department of Computer Science and Engineering

SV NATIONAL INSTITUTE OF TECHNOLOGY SURAT

2024

```
[3] # Single line comment (this has no effect on your program)
    print("Hello MYSELF Akshat Sahu")

    Hello MYSELF Akshat Sahu
```

```
[2] print ('Welcome to the SVNIT Surat!')

    Welcome to the SVNIT Surat!
```

```
a=5
print (a)

5
```

```
a=5*4
a

20
```

```
[6] a
```

```
print("""this is
a multiple
line
python code
for Google Colab practice.
""")

This is
a multiple
line
python code
for Google Colab practice.
```

```
num = input("number :")
if int(num) > 0:
    print("Positive number")
elif int(num) == 0:
    print("Zero")
else:
    print("Negative number")

number :98
Positive number
```

```
[15] names = [ 'SVNIT Surat ','NIT Delhi', 'IIT Delhi', 'IIITM Gwalior ']
     print('Centrally Funded Technical Institute:')
     for i, name in enumerate(names):
         print ("{iteration} : {name}".format(iteration=i+1, name=name))

     Centrally Funded Technical Institute:
     1 : SVNIT Surat
     2 : NIT Delhi
     3 : IIT Delhi
     4 : IIITM Gwalior
```

```
# Write a function for multiplicatin :

## WRITE YOUR CODE HERE
def mul(a,b):
    print ('Mulitplication is : ', a*b)


mul(12,5)
```

```
Mulitplication is :  60
```

```
prices = {'T-Shirt': 299, 'Jeans': 999, 'Shirt': 699}
cart = {'T-Shirt': input('T-Shirt :'),'Jeans': input('Jeans :'), 'Shirt': input('Shirt :')}
shopping_bill = sum(prices[item] * int(cart[item])
                    for item in cart)
print ('Total Bill : %.2f' % shopping_bill +' rupees')
```

```
T-Shirt :399
Jeans :223
Shirt :109
Total Bill : 418269.00 rupees
```

```
a = math.sqrt(89)
print(a)
```

```
9.433981132056603
```

```
a = math.pow(98, 2)
print(a)
```

```
9604.0
```

```
x = 10
y = 1
for i in range(1, x):
  y *= i
print('Factorial of', x, 'is', y)
```

```
Factorial of 10 is 362880
```

```python
import time

vals = list(range(1, 100))
tic = time.time()
for x in vals:
  y = 1
  for i in range(1, x):
    y *= i
toc = time.time()
print('Elapsed time in secs without own function', toc - tic)

tic = time.time()
for x in vals:
  y = math.factorial(x)
toc = time.time()
print('Elapsed time in secs with own function', toc - tic)
```

```
Elapsed time in secs without own function 0.001436471939086914
Elapsed time in secs with own function 0.000263214111328125
```

```python
my_first_module.hello()
```

```
hello, i am living in a different file!!!
Hello NITD
```

21s

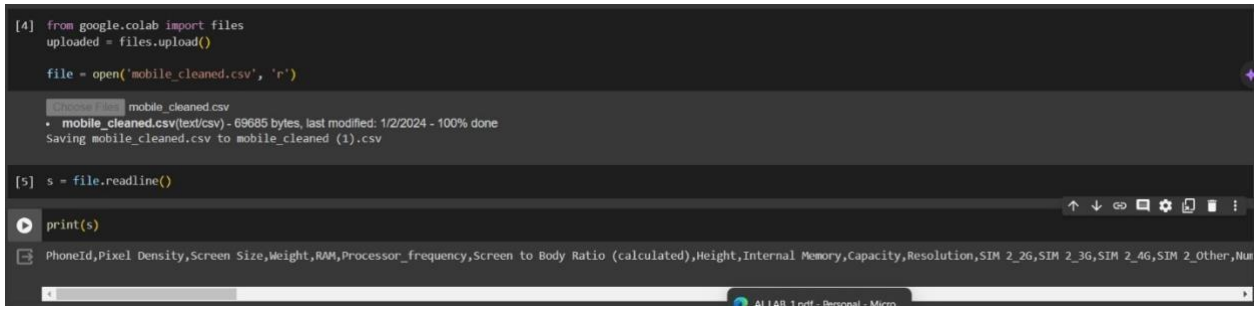
```python
from google.colab import files
uploaded = files.upload()

file = open('mobile_cleaned.csv', 'r')
```

Choose Files mobile_cleaned.csv

- **mobile_cleaned.csv**(text/csv) - 69685 bytes, last modified: 1/2/2024 - 100% done

```
Saving mobile_cleaned.csv to mobile_cleaned.csv
```



```python
# First you need to import the NumPy Library
import numpy as np

b1 = np.array([1,2,3,4,5,6])    #Declaring a NumPy Array
b2 = np.array([7,8,9,10,11,12])

print(b1+b2)

print(b2 * 3)
print("No. of dimensions: ", b1.ndim)  # Rows in array, considered as a matrix.
# Printing shape of array
print("Shape of array: ", b1.shape)  # Dimension

#reshaping an array
r_b1=b1.reshape(2,3)

print("Reshaped array: ", r_b1)
print("Shape of array: ", r_b1.shape)

# Printing size (total number of elements) of array
print("Size of array: ", b1.size) # elements in a row or column elements.

# Printing the datatype of elements in array
print("Array stores elements of type: ", b1.dtype)
```

```
[ 8 10 12 14 16 18]
[21 24 27 30 33 36]
No. of dimensions:  1
Shape of array:  (6,)
Reshaped array:  [[1 2 3]
 [4 5 6]]
Shape of array:  (2, 3)
Size of array:  6
Array stores elements of type:  int64
```

```python
# Creating array from a list with type float
A = np.array([[1, 2, 4], [5, 8, 7]], dtype = 'float')
print("A : ",A)
# Create a 3X4 array with all zeros. Please note, we have used double paranthesis.
B = np.zeros((3, 4))
print("B : ",B)
# Create an arrary of complex numbers
C = np.full((3, 3), 6, dtype = 'complex')
print("C : ",C)
# Create an array with random values
np.random.seed(1) # A seed is set to ensure that the results are consistent if you use this array in future computations also.

D = np.random.randn(2, 2)
print("D : ",D)
E = np.random.random((2, 2))
print("E : ",E)
F = np.random.randint(3, 15, size=(2, 4))
print("F : ",F)
```

```
A :  [[1. 2. 4.]
 [5. 8. 7.]]
B :  [[0. 0. 0. 0.]
 [0. 0. 0. 0.]
 [0. 0. 0. 0.]]
C :  [[6.+0.j 6.+0.j 6.+0.j]
 [6.+0.j 6.+0.j 6.+0.j]
 [6.+0.j 6.+0.j 6.+0.j]]
D :  [[ 1.62434536 -0.61175641]
 [-0.52817175 -1.07296862]]
E :  [[0.39676747 0.53881673]
 [0.41919451 0.6852195 ]]
```

```
D :  [[ 1.62434536 -0.61175641]
 [-0.52817175 -1.07296862]]
E :  [[0.39676747 0.53881673]
 [0.41919451 0.6852195 ]]
F :  [[14 13  5  7]
 [10 10 12  4]]
```

```python
# Reshaping 3X4 array to 2X2X3 array
A = np.array([[1, 2, 3, 4],
              [5, 6, 7, 8],
              [9, 10, 11, 12]])

new_A = A.reshape(2, 2, 3)

# Flatten array
B = np.array([[1, 2, 3], [4, 5, 6]])
flat_B= B.flatten()
# column_flat_B('C')

print ("\nOriginal array:\n", A)
print ("Reshaped array:\n", new_A)
print ("\nOriginal array:\n", B)
print ("Fattened array:\n", flat_B)
#print ("Column Fattened array:\n", column_flat_B)
```

```
Original array:
 [[ 1  2  3  4]
 [ 5  6  7  8]
 [ 9 10 11 12]]
Reshaped array:
 [[[ 1  2  3]
  [ 4  5  6]]

 [[ 7  8  9]
  [10 11 12]]]
```

```
Original array:
 [[1 2 3]
 [4 5 6]]
Fattened array:
 [1 2 3 4 5 6]
```

```python
a = np.array([[1,1,1],[1,1,1], [1,1,1]])
b = np.array([2,2,2])
c = a + b
c1= a*b #element wise multiplication. This will also be broadcasted accordingly.

print(c)
print(c1)
```

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

```python
# File open, read and write.

file1 = open("sample.txt", "w+")
file1.write("hello everyone");
file1.close()

# Open a file
file_read = open("sample.txt", "r")
str = file_read.read(80);
print("Name of the file: ", file_read.name)
print("Read String from file is : ", str)
# Close opend files
file_read.close()
```

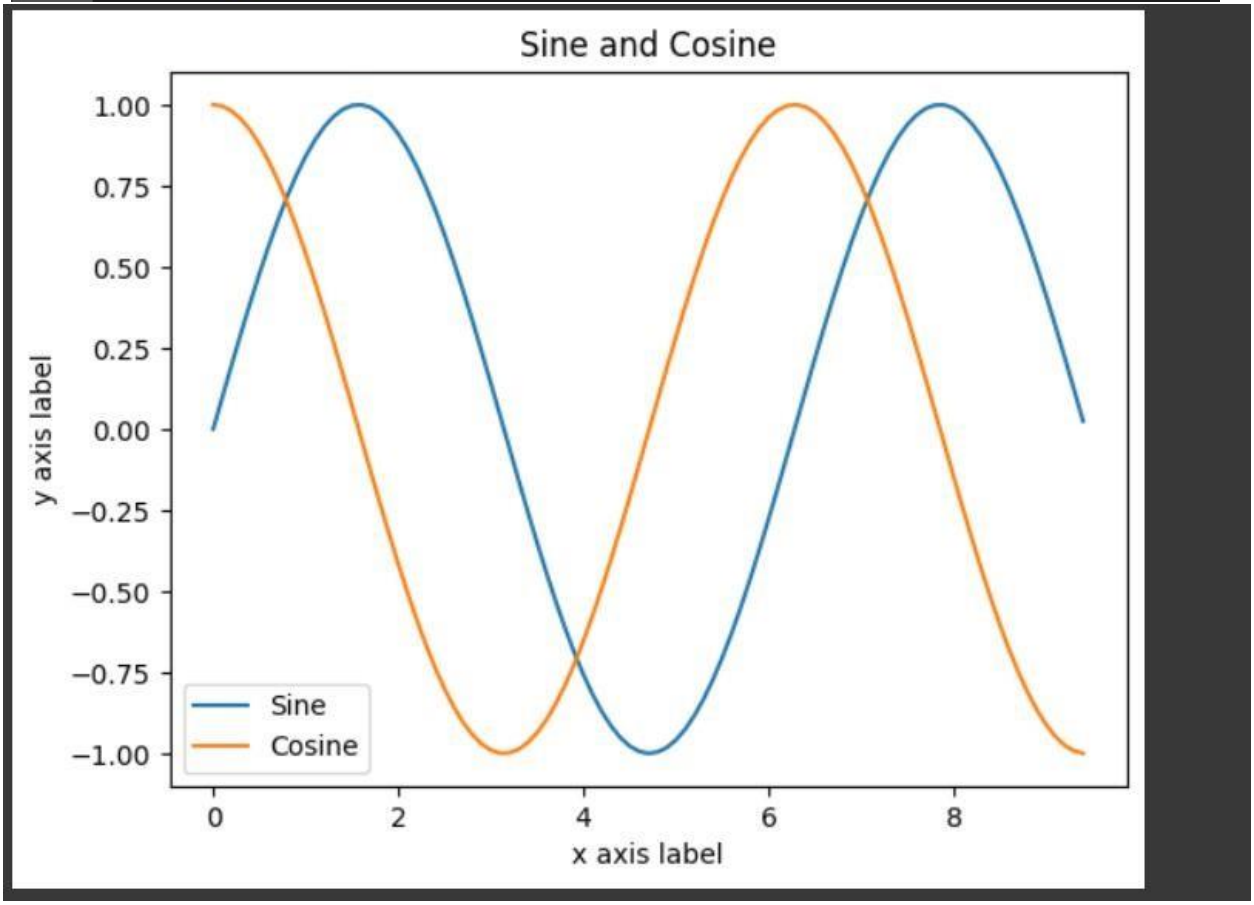
```
Name of the file:  sample.txt
Read String from file is :  hello everyone
```

```python
import numpy as np
import matplotlib.pyplot as plt

# Computes x and y coordinates for
# points on sine and cosine curves
x = np.arange(0, 3 * np.pi, 0.1)
y_sin = np.sin(x)
y_cos = np.cos(x)

# Plot the points using matplotlib
plt.plot(x, y_sin)
plt.plot(x, y_cos)
plt.xlabel('x axis label')
plt.ylabel('y axis label')
plt.title('Sine and Cosine')
plt.legend(['Sine', 'Cosine'])

plt.show()
```



Q Explore the large language models, make a write up on them answering the following question.

Ans What are they?

A large language model is a deep learning algorithm that can perform a variety of natural language processing (NLP) tasks. Large language models use a transformer models and are trained using massive datasets - hence, large. This enables them to recognize, translate, predict or generate text or other content.

How They WORK?

A large language model is based on a Transformer model and works by recieving an input, encoding it and decoding it to produce an output prediction.

Functions =>

- Natural language understanding and enabling enhanced communication between human and computer
- LLMs can be used for technical writting.
- LLMs can be employed to generate code snippet as well as it helps in debuging.