Practical No 1

**Name :** Rahul Sampatrao Patil

**Roll No :** 4176

# Text File 1 :-

( Sample1.txt )

Taj Mahal is a historic place to look like a very attractive and famous natural scene. It is located in Agra, Uttar Pradesh. It is located in a very beautiful place in a very beautiful place, behind which is the river, it looks like a paradise on earth. This white marble is used, it attracts the attention of people all over the world, watching it every year for. For the Taj Mahal Mumtaz, this is a symbol of Shah Jahan's inner love, who made it in memory of his wife Mumtaz. Tajmahal is counted among the seven wonders of the world. This is the grave of the great emperor Mumtaz Mahal (Samadhi).

# Text File 2 :-

( Sample2.txt )

Taj Mahal is the beautiful and most interesting place in India. This is a cultural monument of India, which was built by King Shah Jahan in his memory after the death of his wife (Mumtaz Mahal). It has a lot of greenery, a river bank on the back, a large area in front of a lake and a lawn. It is located in Agra, it is one of the seven wonders of the world. It is the most beautiful building made of white marble, it seems like a paradise of dream. It is made in an attractive way and punished with royal beauty. This is the beauty of the wonderful nature on earth.

# Code :-

import math import string import sys

def read\_file(filename):

try:

with open(filename, 'r') as f: data = f.read()

return data except IOError:

print("Error opening or reading input file: ", filename) sys.exit()

translation\_table = str.maketrans(string.punctuation+string.ascii\_uppercase," "\*len(string.punctuation)+string.ascii\_lowercase)

def get\_words\_from\_line\_list(text): text = text.translate(translation\_table) word\_list = text.split()

return word\_list

def count\_frequency(word\_list): D = {}

for new\_word in word\_list:

if new\_word in D:

D[new\_word] = D[new\_word] + 1 else:

D[new\_word] = 1 return D

def word\_frequencies\_for\_file(filename): line\_list = read\_file(filename)

word\_list = get\_words\_from\_line\_list(line\_list)

freq\_mapping = count\_frequency(word\_list) print("File", filename, ":", ) print(len(line\_list), "lines, ", ) print(len(word\_list), "words, ", ) print(len(freq\_mapping), "distinct words")

return freq\_mapping def dotProduct(D1, D2):

Sum = 0.0

for key in D1:

if key in D2:

Sum += (D1[key] \* D2[key]) return Sum

def vector\_angle(D1, D2):

numerator = dotProduct(D1, D2)

denominator = math.sqrt(dotProduct(D1, D1)\*dotProduct(D2, D2)) return math.acos(numerator / denominator)

def documentSimilarity(filename\_1, filename\_2): sorted\_word\_list\_1 = word\_frequencies\_for\_file(filename\_1) sorted\_word\_list\_2 = word\_frequencies\_for\_file(filename\_2)

distance = vector\_angle(sorted\_word\_list\_1, sorted\_word\_list\_2)

print("The distance between the documents is: % 0.6f (radians)"% distance) documentSimilarity('sample1.txt', 'sample2.txt')

**Output :-**

File sample1.txt :

598 lines,

113 words,

66 distinct words File sample2.txt :

779 lines,

154 words,

89 distinct words

The distance between the documents is: 0.618456 (radians)