Practical No. 03

AIM - Write a program to Compute Similarity between two text documents.

Source Code -

```
import numpy as np
import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.metrics.pairwise import cosine_similarity

def cosine_similarity(x,y):
    #ensure length of x and y are the same
    if len(x)!=len(y):
        return None
    #compute the dot product of x and y
    dot_product=np.dot(x,y)
    #compute the magnitude of x and y
    magnitude_x=np.sqrt(np.sum(x**2))
    magnitude_y=np.sqrt(np.sum(y**2))
    #compute the cosine similarity
    cosine_similarity=dot_product/(magnitude_x*magnitude_y)
    return cosine_similarity

corpus=['data science is one of the most important field of science',
        'this is one of the best data science courses',
        'data scientist analyse data']

#create a matrix to represent the corpus
x=CountVectorizer().fit_transform(corpus).toarray()
print(x)
cos_sin_1_2=cosine_similarity(x[0,:],x[1,:])
cos_sin_1_3=cosine_similarity(x[0,:],x[2,:])
cos_sin_2_3=cosine_similarity(x[1,:],x[2,:])
print('cosine similarity between:')
print('#t document1 and document2:',cos_sin_1_2)
print('#t document1 and document3:',cos_sin_1_3)
print('#t document2 and document3:',cos_sin_2_3)
```

OUTPUT -

```
In [1]: runfile('C:/Users/ckt/Documents/KUNAL-workspace/IR/P3.py',
wdir='C:/Users/ckt/Documents/KUNAL-workspace/IR')
[[0 0 0 1 1 1 1 1 2 1 2 0 1 0]
 [0 1 1 1 0 0 1 0 1 1 1 0 1 1]
 [1 0 0 2 0 0 0 0 0 0 0 1 0 0]]
cosine similarity between:
#t document1 and document2: 0.6885303726590962
#t document1 and document3: 0.21081851067789195
#t document2 and document3: 0.2721655269759087

In [2]:
```