#Ex9iNLP

```python
import re
from collections import Counter
import math
def calculate_cosine_similarity(text1, text2):
    def tokenize(text):
        words=re.findall(r'\w+',text.lower())
        return Counter(words)
    vec1=tokenize(text1)
    vec2=tokenize(text2)
    intersection=set(vec1.keys()) & set(vec2.keys())

    dot_product=sum(vec1[word]*vec2[word] for word in intersection)
    magnitude1=math.sqrt(sum(vec1[word] ** 2 for word in vec1.keys()))
    magnitude2=math.sqrt(sum(vec2[word] ** 2 for word in vec2.keys()))
    cos_similarity=dot_product / (magnitude1*magnitude2)
    return cos_similarity

text1="I am in Kollam"
text2="I am in Sivakasi"
similarity=calculate_cosine_similarity(text1, text2)
print("cosine similarity:", similarity)
```
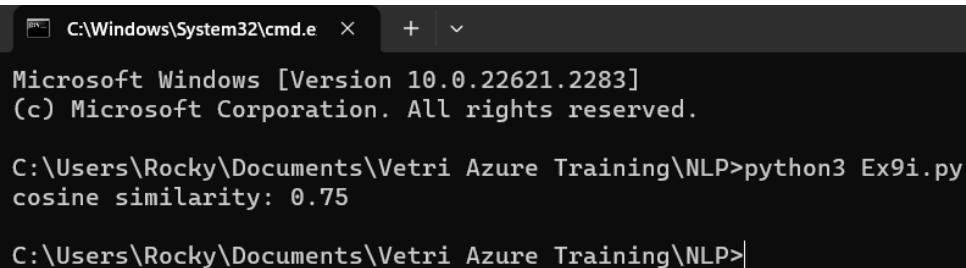
Output:

#Ex9iiNLP

```python
import re

def calculate_jaccard_similarity(text1,text2):
    def tokenize(text):
        words = re.findall(r'\w+', text.lower())
        return set(words)

    set1=tokenize(text1)
    set2=tokenize(text2)

    intersection = len(set1.intersection(set2))
    union = len(set1.union(set2))
    jaccard_similarity=intersection/union

    return jaccard_similarity

text1="I am in Kollam"
text2="I am in Sivakasi"
similarity=calculate_jaccard_similarity(text1, text2)
print("Jaccard Similarity:", similarity)
```
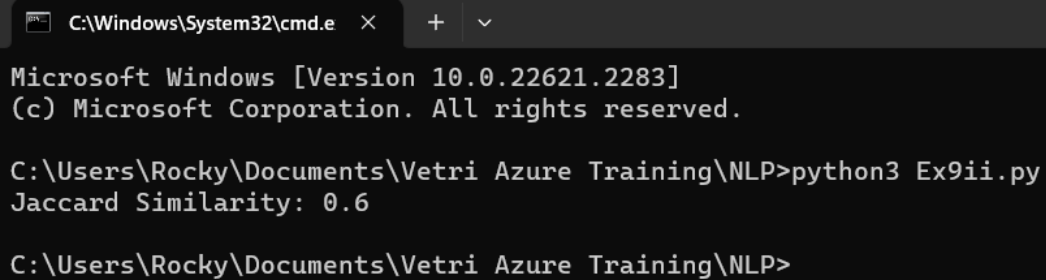
Output: