

###Lab: 05

Date: 25.02.222

###Program No.:05

###Program Name: Write a Python Program to find Inverted Index and
Process Boolean Query.

#ANINDYA NAG

ROLL : UG/02/BTCSE/2018/005

#CODE:3

Collection of documents

D1="I play tennis everyday"

D2="Everyday I go to play cricket"

D3="I like cricket and tennis"

D4="Dhoni won the world cup"

D5="I love a cup of tea everyday"

docs = [D1,D2,D3,D4,D5]

docs

```
[> ['I play tennis everyday',  
    'Everyday I go to play cricket',  
    'I like cricket and tennis',  
    'Dhoni won the world cup',  
    'I love a cup of tea everyday']
```

Gather the set of all unique terms

unique_terms = {term for doc in docs for term in doc.split()}

unique_terms

```
{'Dhoni',  
 'Everyday',  
 'I',  
 'a',  
 'and',  
 'cricket',  
 'cup',  
 'everyday',  
 'go',  
 'like',  
 'love',  
 'of',  
 'play',  
 'tea',  
 'tennis',  
 'the',  
 'to',
```

```
'won',  
'world']}
```

```
# Construct an inverted index  
# here as a Python dictionary for ease of interpretability
```

```
inverted_index = {}
```

```
for i, doc in enumerate(docs):  
    for term in doc.split():  
        if term in inverted_index:  
            inverted_index[term].add(i+1)  
        else: inverted_index[term] = {i+1}
```

```
inverted_index
```

```
{'Dhoni': {4},  
 'Everyday': {2},  
 'I': {1, 2, 3, 5},  
 'a': {5},  
 'and': {3},  
 'cricket': {2, 3},  
 'cup': {4, 5},  
 'everyday': {1, 5},  
 'go': {2},  
 'like': {3},  
 'love': {5},  
 'of': {5},  
 'play': {1, 2},  
 'tea': {5},  
 'tennis': {1, 3},  
 'the': {4},  
 'to': {2},  
 'won': {4},  
 'world': {4}}
```

```
def or_postings(posting1, posting2):  
    p1 = 0  
    p2 = 0  
    result = list()  
    while p1 < len(posting1) and p2 < len(posting2):  
        if posting1[p1] == posting2[p2]:  
            result.append(posting1[p1])  
            p1 += 1  
            p2 += 1  
        elif posting1[p1] > posting2[p2]:  
            result.append(posting2[p2])  
            p2 += 1  
        else:  
            result.append(posting1[p1])  
            p1 += 1
```

```

while p1 < len(posting1):
    result.append(posting1[p1])
    p1 += 1
while p2 < len(posting2):
    result.append(posting2[p2])
    p2 += 1
return result

```

```

wopl_1 = list(inverted_index[input()])
pl_2 = list(inverted_index[input()])

```

```

print(pl_1)
print(pl_2)
print('-----')

```

```
or_postings(pl_1, pl_2)
```

```

play
go
[1, 2]
[2]
-----
[1, 2]

```

```

def and_postings(posting1, posting2):
    p1 = 0
    p2 = 0
    result = list()
    while p1 < len(posting1) and p2 < len(posting2):
        if posting1[p1] == posting2[p2]:
            result.append(posting1[p1])
            p1 += 1
            p2 += 1
        elif posting1[p1] > posting2[p2]:
            p2 += 1
        else:
            p1 += 1
    return result

```

```

pl_1 = list(inverted_index[input()])
pl_2 = list(inverted_index[input()])

```

```

print(pl_1)
print(pl_2)
print('-----')

```

```
and_postings(pl_1, pl_2)
```

```
play
```

```
go  
[1, 2]  
[2]  
-----  
[2]
```

✓ 5s completed at 3:21 PM

