Practical No. 1

Aim - Write a program to demonstrate bitwise operation.

Source Code -

```
import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer

corpus = [
        'this is the first document',
        'this document is a second document',
        'and this is the third document',
        'is this the first document?'
    ]

vectorizer = CountVectorizer()
x = vectorizer.fit_transform(corpus)

print("FIT TRANSFORM -> ")
print(x.toarray())

df = pd.DataFrame(x.toarray(), columns=vectorizer.get_feature_names())
print("THE GENERATED DATA FRAME -> ")
print(df)

alldata = df[(df['this'] == 1) & (df['first'] == 1)]
print("indices where this terms are present in ", alldata.index.tolist())

ordata = df[(df['this'] == 1) | (df['first'] == 1)]
print("indices    where    either    of    this    terms    are    present    in    ",
ordata.index.tolist())

notdata = df[(df['and'] != 1)]
print("indices where and term is not present in ", notdata.index.tolist())
```

Output -

```
In [1]: runfile('C:/Users/ckt/Documents/KUNAL-workspace/P1.py', wdir='C:/
Users/ckt/Documents/KUNAL-workspace')
FIT TRANSFORM ->
[[0 1 1 1 0 1 0 1]
 [0 2 0 1 1 0 0 1]
 [1 1 0 1 0 1 1 1]
 [0 1 1 1 0 1 0 1]]
THE GENERATED DATA FRAME ->
   and  document  first  is  second  the  third  this
0    0         1      1   1       0    1      0     1
1    0         2      0   1       1    0      0     1
2    1         1      0   1       0    1      1     1
3    0         1      1   1       0    1      0     1
indices where this terms are present in  [0, 3]
indices where either of this terms are present in  [0, 1, 2, 3]
indices where and term is not present in  [0, 1, 3]

In [2]: |
```