▾ 68_Adnan Shaikh

```python
from nltk.corpus import words
from random import sample, shuffle
from bloom_filter2 import BloomFilter
```

```python
N = 20
P = 0.05
```

```python
bloom_obj = BloomFilter(N,P)
print(f"Size of Bucket: {bloom_obj.num_bits_m}\nFalse Positive Probability: {bloom_
```

```
     Size of Bucket: 125
     False Positive Probability: 0.05
```

```python
word_present = sample(words.words(),N)
word_present
```

```
⊡→  ['protosalt',
     'cellular',
     'lagopous',
     'bijouterie',
     'fondlingly',
     'retrievable',
     'sheltery',
     'kiekie',
     'preferential',
     'platyope',
     'thunderproof',
     'polydermy',
     'Pondo',
     'restis',
     'Polyangium',
     'unconcessible',
     'befilch',
     'savingness',
     'harmoniousness',
     'stereotropism']
```

```python
word_absent = sample(set(words.words()).symmetric_difference(set(word_present)),N)
word_absent
```

```
     ['tergitic',
     'diaheliotropically',
     'Amatembu',
     'cuboides',
     'epedaphic',
     'reappearance',
     'untwirled',
     'honoree',
```

```
       'infrastapedial',
       'decolorize',
       'Agelaius',
       'Anacreon',
       'expletiveness',
       'spousally',
       'Anthropopithecus',
       'melancholia',
       'verdet',
       'Tenuirostres',
       'redintegration',
       'Purkinje']


   for word in word_present:
     bloom_obj.add(word)




   shuffle(word_present)
   shuffle(word_absent)
   test_words = sample(word_present,N//2) + sample(word_absent,N//2)
   test_words

       ['befilch',
        'thunderproof',
        'platyope',
        'savingness',
        'Polyangium',
        'harmoniousness',
        'sheltery',
        'preferential',
        'fondlingly',
        'unconcessible',
        'infrastapedial',
        'spousally',
        'epedaphic',
        'verdet',
        'Amatembu',
        'Anacreon',
        'diaheliotropically',
        'tergitic',
        'reappearance',
        'expletiveness']


   for word in test_words:
     if word in bloom_obj and word in word_present:
         print(f"{word} is present in DB")
     elif word in bloom_obj and word not in word_present:
       print(f"{word} is not present in DB. FALSE POSITIIIVE!!!!!")
     else:
       print(f"{word} is not present in DB")


       befilch is present in DB
       thunderproof is present in DB
       platyope is present in DB
       savingness is present in DB
```

```
Polyangium is present in DB
harmoniousness is present in DB
sheltery is present in DB
preferential is present in DB
fondlingly is present in DB
unconcessible is present in DB
infrastapedial is not present in DB
spousally is not present in DB
epedaphic is not present in DB
verdet is not present in DB
Amatembu is not present in DB
Anacreon is not present in DB
diaheliotropically is not present in DB
tergitic is not present in DB. FALSE POSITIIIVE!!!!!
reappearance is not present in DB
expletiveness is not present in DB
```

Colab paid products - Cancel contracts here