

EXPERIMENT 2

AIM: Implementation word and sentence tokenization in Python.

SOURCE CODE:

English Language Tokenisation:

```
import re
from nltk.tokenize import word_tokenize,sent_tokenize
from pathlib import Path
text = Path('myfiles.txt').read_text()
tokens = re.findall("[\w]+", text)
print("Word Tokenization without NLTK:\n")
print(tokens)
print("\nWord TOkenization with NLTK:\n")
print(word_tokenize(text))
print("\nSentence Tokenization without NLTK\n")
sentences = text.split('. ')
print(sentences)
print("\nSentence Tokenization with NLTK\n")
print(sent_tokenize(text))
```

Hindi Language Tokenisation:

```
from nltk.tokenize import tokenize
hindi_text = """प्राचीन काल में विक्रमादित्य नाम के एक आदर्श राजा हुआ करते थे।
अपने साहस, पराक्रम और शौर्य के लिए राजा विक्रम मशहूर थे। """
tokenize(hindi_text, "hi")
from indicnlp.tokenize import sentence_tokenize
sentences=sentence_tokenize.sentence_split(hindi_text, lang='hi')
for t in sentences:
    print(t)
```

myfiles.txt

Founded in 2002, SpaceX's mission is to enable humans to become a spacefaring civilization and a multi-planet species by building a self-sustaining city on Mars. In 2008, SpaceX's Falcon 1 became the first privately developed liquid-fuel launch vehicle to orbit the Earth.

OUTPUT:

```
IDLE Shell 3.9.7
File Edit Shell Debug Options Window Help
Python 3.9.7 (tags/v3.9.7:1016ef3, Aug 30 2021, 20:19:38) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\admin\Desktop\SEM 8\Practicals\NLP\Exp2.py =====
Word Tokenization without NLTK:

['Founded', 'in', '2002', 'SpaceXs', 'mission', 'is', 'to', 'enable', 'humans', 'to', 'become',
'a', 'spacefaring', 'civilization', 'and', 'a', 'multi', 'planet', 'species', 'by', 'building',
'a', 'self', 'sustaining', 'city', 'on', 'Mars', 'In', '2008', 'SpaceXs', 'Falcon', '1', 'became',
'the', 'first', 'privately', 'developed', 'liquid', 'fuel', 'launch', 'vehicle', 'to', 'orbit',
'the', 'Earth']

Word Tokenization with NLTK:

['Founded', 'in', '2002', ',', 'SpaceXs', 'mission', 'is', 'to', 'enable', 'humans', 'to', 'beco',
me', 'a', 'spacefaring', 'civilization', 'and', 'a', 'multi-planet', 'species', 'by', 'building',
'a', 'self-sustaining', 'city', 'on', 'Mars', 'In', '2008', ',', 'SpaceXs', 'Falcon', '1',
'became', 'the', 'first', 'privately', 'developed', 'liquid-fuel', 'launch', 'vehicle', 'to',
'orbit', 'the', 'Earth', '.']

Sentence Tokenization without NLTK

['Founded in 2002, SpaceXs mission is to enable humans to become a spacefaring civilization and
a multi-planet \nspecies by building a self-sustaining city on Mars', 'In 2008, SpaceXs Falcon 1
became the first privately developed \nliquid-fuel launch vehicle to orbit the Earth.']

Sentence Tokenization with NLTK

['Founded in 2002, SpaceXs mission is to enable humans to become a spacefaring civilization and
a multi-planet \nspecies by building a self-sustaining city on Mars.', 'In 2008, SpaceXs Falcon
1 became the first privately developed \nliquid-fuel launch vehicle to orbit the Earth.']
>>> |
```

Fig1. Performing English Language Word and Sentence Tokenization with and without NLTK

```
In [3]: tokenize(hindi_text, "hi")
Out[3]: ['प्राचीन',
'_काल',
'_में',
'_विक्रमादित्य',
'_नाम',
'_के',
'_एक',
'_आदर्श',
'_राजा',
'_हुआ',
'_करते',
'_थे',
'_।',
'_अपने',
'_साहस',
'',
'_पराक्रम',
'_और',
'_शौर्य',
'_के',
'_लिए',
'_राजा',
'_विक्रम',
'_मशहूर',
'_थे',
'_।']

In [5]: for t in sentences:
        print(t)

प्राचीन काल में विक्रमादित्य नाम के एक आदर्श राजा हुआ करते थे।
अपने साहस, पराक्रम और शौर्य के लिए राजा विक्रम मशहूर थे।
```

Fig2. Performing Hindi Language Word and Sentence Tokenization