```
#Step 1 — Open Google Colab Open Colab and create a new notebook. We will use Colab because it already supports Pyth
medical_text = """
The patient was diagnosed with diabetes and hypertension.
She is taking medications including metformin and insulin.
The doctor recommended monitoring blood glucose levels regularly.
Patients suffering from cardiac diseases require continuous care.
"""
```

```
#Step 2 — Install and import libraries We install NLTK and spaCy, then import required modules. These libraries help

!pip install nltk spacy
!python -m spacy download en_core_web_sm
```

```
Requirement already satisfied: nltk in /usr/local/lib/python3.12/dist-packages (3.9.1)
Requirement already satisfied: spacy in /usr/local/lib/python3.12/dist-packages (3.8.11)
Requirement already satisfied: click in /usr/local/lib/python3.12/dist-packages (from nltk) (8.3.1)
Requirement already satisfied: joblib in /usr/local/lib/python3.12/dist-packages (from nltk) (1.5.3)
Requirement already satisfied: regex>=2021.8.3 in /usr/local/lib/python3.12/dist-packages (from nltk) (2025.11.3)
Requirement already satisfied: tqdm in /usr/local/lib/python3.12/dist-packages (from nltk) (4.67.1)
Requirement already satisfied: spacy-legacy<3.1.0,>=3.0.11 in /usr/local/lib/python3.12/dist-packages (from spacy)
Requirement already satisfied: spacy-loggers<2.0.0,>=1.0.0 in /usr/local/lib/python3.12/dist-packages (from spacy)
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (1
Requirement already satisfied: cymem<2.1.0,>=2.0.2 in /usr/local/lib/python3.12/dist-packages (from spacy) (2.0.13)
Requirement already satisfied: preshed<3.1.0,>=3.0.2 in /usr/local/lib/python3.12/dist-packages (from spacy) (3.0.12
Requirement already satisfied: thinc<8.4.0,>=8.3.4 in /usr/local/lib/python3.12/dist-packages (from spacy) (8.3.10)
Requirement already satisfied: wasabi<1.2.0,>=0.9.1 in /usr/local/lib/python3.12/dist-packages (from spacy) (1.1.3)
Requirement already satisfied: srsly<3.0.0,>=2.4.3 in /usr/local/lib/python3.12/dist-packages (from spacy) (2.5.2)
Requirement already satisfied: catalogue<2.1.0,>=2.0.6 in /usr/local/lib/python3.12/dist-packages (from spacy) (2.0
Requirement already satisfied: weasel<0.5.0,>=0.4.2 in /usr/local/lib/python3.12/dist-packages (from spacy) (0.4.3)
Requirement already satisfied: typer-slim<1.0.0,>=0.3.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (0.2
Requirement already satisfied: numpy>=1.19.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (2.0.2)
Requirement already satisfied: requests<3.0.0,>=2.13.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (2.32
Requirement already satisfied: pydantic!=1.8,!=1.8.1,<3.0.0,>=1.7.4 in /usr/local/lib/python3.12/dist-packages (fron
Requirement already satisfied: jinja2 in /usr/local/lib/python3.12/dist-packages (from spacy) (3.1.6)
Requirement already satisfied: setuptools in /usr/local/lib/python3.12/dist-packages (from spacy) (75.2.0)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (25.0)
Requirement already satisfied: annotated-types>=0.6.0 in /usr/local/lib/python3.12/dist-packages (from pydantic!=1.8
Requirement already satisfied: pydantic-core==2.41.4 in /usr/local/lib/python3.12/dist-packages (from pydantic!=1.8,
Requirement already satisfied: typing-extensions>=4.14.1 in /usr/local/lib/python3.12/dist-packages (from pydantic!=
Requirement already satisfied: typing-inspection>=0.4.2 in /usr/local/lib/python3.12/dist-packages (from pydantic!=1
Requirement already satisfied: charset_normalizer<4,>=2 in /usr/local/lib/python3.12/dist-packages (from requests<3
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.12/dist-packages (from requests<3.0.0,>=2.13.0
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.12/dist-packages (from requests<3.0.0,>=
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.12/dist-packages (from requests<3.0.0,>=
Requirement already satisfied: blis<1.4.0,>=1.3.0 in /usr/local/lib/python3.12/dist-packages (from thinc<8.4.0,>=8.3
Requirement already satisfied: confection<1.0.0,>=0.0.1 in /usr/local/lib/python3.12/dist-packages (from thinc<8.4.0
Requirement already satisfied: cloudpathlib<1.0.0,>=0.7.0 in /usr/local/lib/python3.12/dist-packages (from weasel<0
Requirement already satisfied: smart-open<8.0.0,>=5.2.1 in /usr/local/lib/python3.12/dist-packages (from weasel<0.5
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.12/dist-packages (from jinja2->spacy) (3.0
Requirement already satisfied: wrapt in /usr/local/lib/python3.12/dist-packages (from smart-open<8.0.0,>=5.2.1->weas
Collecting en-core-web-sm==3.8.0
  Downloading https://github.com/explosion/spacy-models/releases/download/en_core_web_sm-3.8.0/en_core_web_sm-3.8.0-
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 12.8/12.8 MB 38.4 MB/s eta 0:00:00
✓ Download and installation successful
You can now load the package via spacy.load('en_core_web_sm')
⚠ Restart to reload dependencies
If you are in a Jupyter or Colab notebook, you may need to restart Python in
order to load all the package's dependencies. You can do this by selecting the
'Restart kernel' or 'Restart runtime' option.
```

```
#Load Medical Text Corpus
medical_text = """
Diabetes is a chronic disease that affects how the body processes blood sugar.
Patients experiencing symptoms such as excessive thirst and frequent urination
should seek medical attention. Treatment options include insulin therapy and lifestyle changes.
"""
```

```
# STEP 4: Sentence Tokenization using NLTK

import nltk
from nltk.tokenize import sent_tokenize

# Download tokenizer model
nltk.download('punkt')

# Medical text corpus
medical_text = """
The patient was diagnosed with diabetes and hypertension.
She is taking medications including metformin and insulin.
The doctor recommended monitoring blood glucose levels regularly.
Patients suffering from cardiac diseases require continuous care.
"""

# Perform sentence tokenization
sentences = sent_tokenize(medical_text)

# Display sentences
for i, sentence in enumerate(sentences, 1):
    print(f"Sentence {i}: {sentence}")
```

```
Sentence 1:
The patient was diagnosed with diabetes and hypertension.
Sentence 2: She is taking medications including metformin and insulin.
Sentence 3: The doctor recommended monitoring blood glucose levels regularly.
Sentence 4: Patients suffering from cardiac diseases require continuous care.
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
```

```
#Step 5 – Word Tokenization (NLTK & spaCy) We break sentences into individual words (tokens). spaCy also recognizes
words = word_tokenize(medical_text)
print(words)
```

```
['The', 'patient', 'was', 'diagnosed', 'with', 'diabetes', 'and', 'hypertension', '.', 'She', 'is', 'taking', 'medic
```

```
nlp = spacy.load("en_core_web_sm")
doc = nlp(medical_text)

spacy_tokens = [token.text for token in doc]
print(spacy_tokens)
```

```
['\n', 'The', 'patient', 'was', 'diagnosed', 'with', 'diabetes', 'and', 'hypertension', '.', '\n', 'She', 'is', 'tak
```

```
#Step 6 – Apply Stemming (Porter Stemmer) Stemming cuts words to rough root forms. It is fast – but it may produce


stemmer = PorterStemmer()
stemmed_words = [stemmer.stem(word) for word in words if word.isalpha()]
print(stemmed_words)
```

```
['the', 'patient', 'wa', 'diagnos', 'with', 'diabet', 'and', 'hypertens', 'she', 'is', 'take', 'medic', 'includ', 'm
```

```
#Step 7 – Apply Lemmatization (NLTK & spaCy) Lemmatization converts words to meaningful dictionary base forms. Exam

lemmatizer = WordNetLemmatizer()
lemmatized_words = [lemmatizer.lemmatize(word.lower()) for word in words if word.isalpha()]
print(lemmatized_words)
```

```
['the', 'patient', 'wa', 'diagnosed', 'with', 'diabetes', 'and', 'hypertension', 'she', 'is', 'taking', 'medication
```

```
spacy_lemmas = [token.lemma_ for token in doc if token.is_alpha]
print(spacy_lemmas)
```

```
['the', 'patient', 'be', 'diagnose', 'with', 'diabete', 'and', 'hypertension', 'she', 'be', 'take', 'medication', 'i
```

```
# STEP 8: Comparison of Original, Stemmed, and Lemmatized Words

from nltk.stem import PorterStemmer
from nltk.stem import WordNetLemmatizer
from nltk.tokenize import word_tokenize
import nltk

nltk.download('punkt')
nltk.download('wordnet')
nltk.download('omw-1.4')

medical_text = """
The patient was diagnosed with diabetes and hypertension.
She is taking medications including metformin and insulin.
"""

# Tokenize words
words = word_tokenize(medical_text)

# Initialize stemmer and lemmatizer
stemmer = PorterStemmer()
lemmatizer = WordNetLemmatizer()

# Process only alphabetic words
original_words = [word.lower() for word in words if word.isalpha()]
stemmed_words = [stemmer.stem(word) for word in original_words]
lemmatized_words = [lemmatizer.lemmatize(word) for word in original_words]

# Display comparison
print("Original Word | Stemmed Word | Lemmatized Word")
print("-" * 45)

for o, s, l in zip(original_words, stemmed_words, lemmatized_words):
    print(f"{o:<13} {s:<14} {l}")
```

```
Original Word | Stemmed Word | Lemmatized Word
---------------------------------------------
the           the            the
patient       patient        patient
was           wa             wa
diagnosed     diagnos        diagnosed
with          with           with
diabetes      diabet         diabetes
and           and            and
hypertension  hypertens      hypertension
she           she            she
is            is             is
taking        take           taking
medications   medic          medication
including     includ         including
metformin     metformin      metformin
and           and            and
insulin       insulin        insulin
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
[nltk_data] Downloading package omw-1.4 to /root/nltk_data...
[nltk_data]   Package omw-1.4 is already up-to-date!
```