**Lab Assignment 4**

**Report On**

**CIS 612 Big Data Processing**

**November 4, 2022**

1. **Part One:** Building a Knowledge Base in Inverted Index for Big Data Applications such as web Page Categorization or Google Search Engine.

We build inverted index in two MySQL tables.

* + First Level Dictionary Table(Term, DocFreq, CollectionFreq)
  + Second Level Posting Table(Term, Doc\_Id, TermFreq)

A Python script created in the root folder (part\_1.py) is run to generate predefined tables as shown in the figure below.

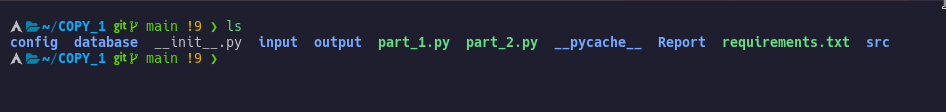
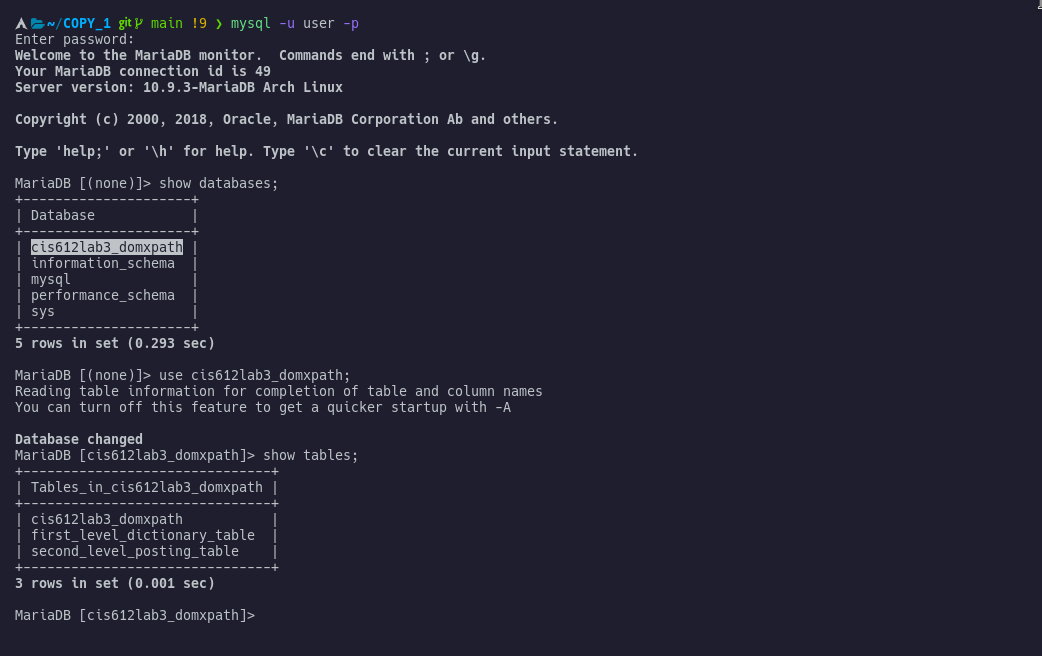
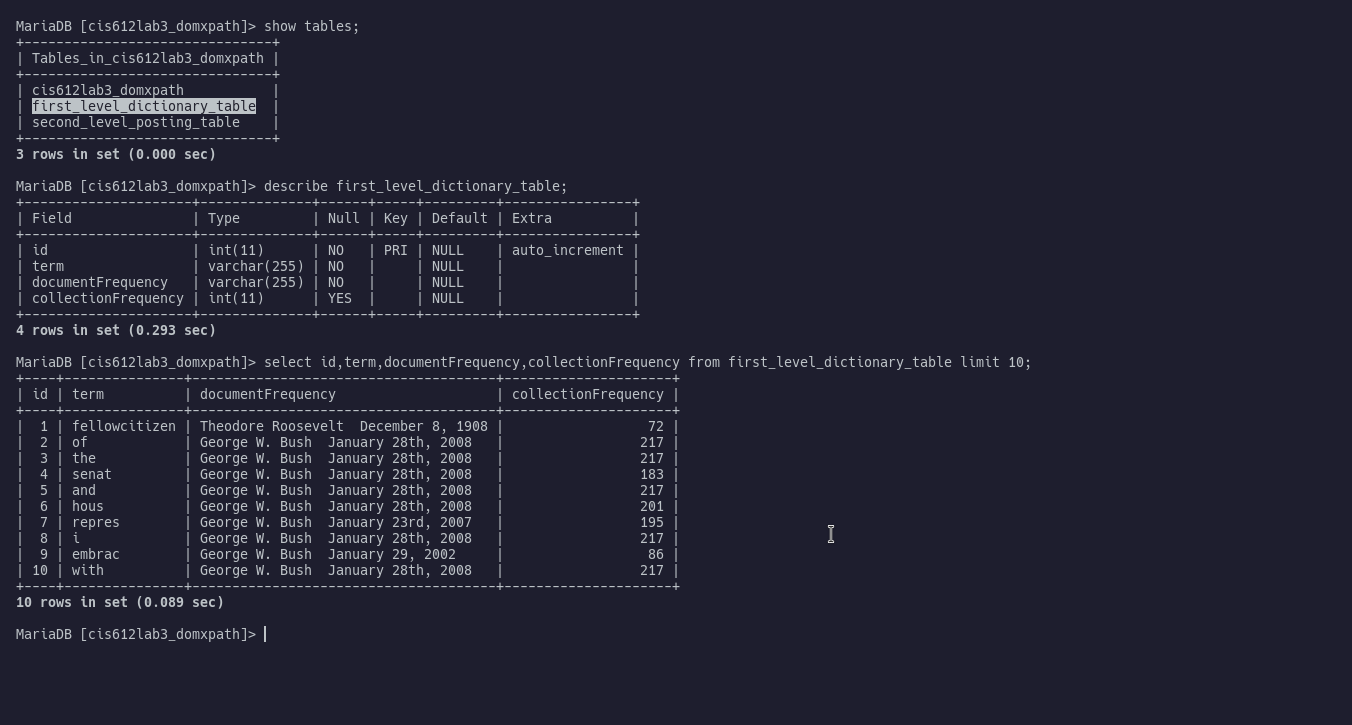
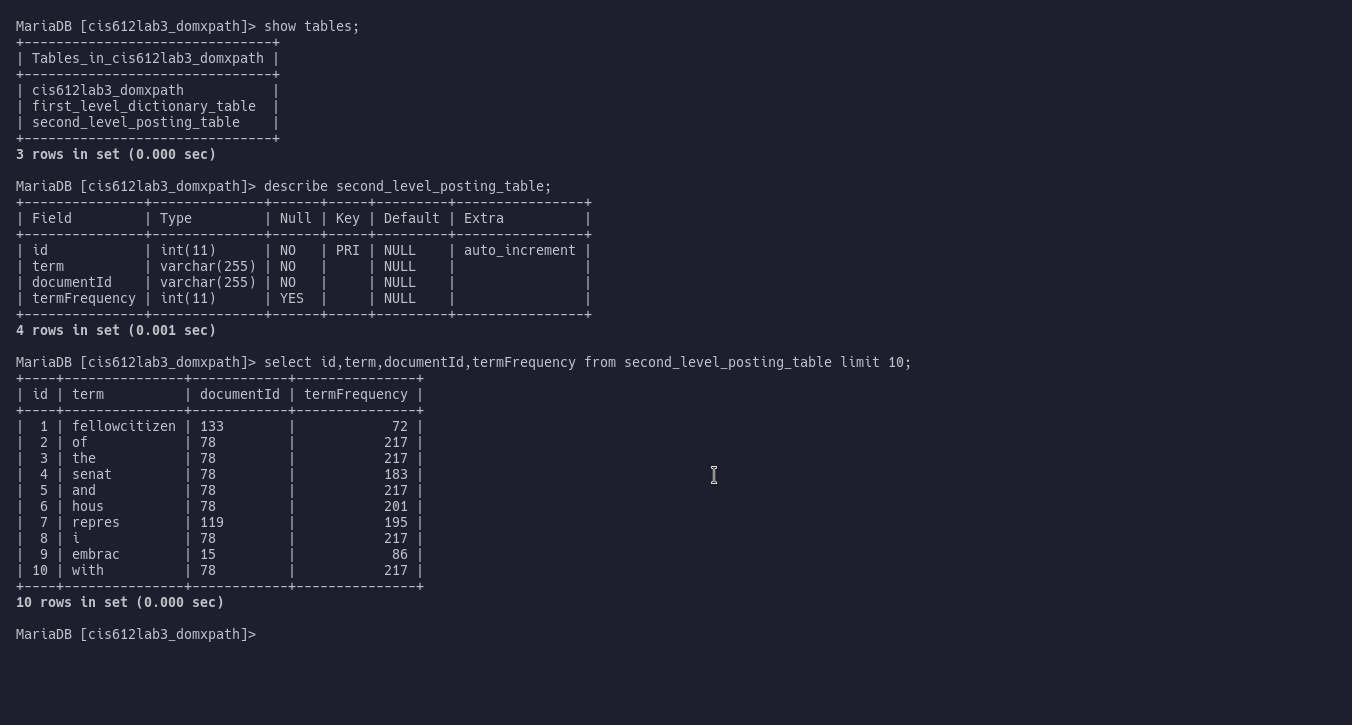


Fig 1.1: Files required.

Fig 1.2: Login MySQL, Database and Tables.

Fig 1.3: First Level dictionary table

Fig 1.4: Posting File in SQL table

Documents are preprocessed in a better way to enhance a word count in a systematic way,

Below are the NLP methods used in the project :-

* + Text punctuation
  + Tokenization
  + Convert numbers to words
  + Suffix trimming
  + Remove stop words
  + Lowercasing

The python script **part\_1.py**  contains the code below.



Fig 1.5 Python script part\_1.py

1. **Part Two:** Building Document Vectors for Web Page Content Analysis.

We are cheking for text similarities between documents in cosine similarity metric.

Using cosine similarity, we consider matrix vectors use. Part Two mostly consistes of document vectorization and cosine similarity computatios.

We use the TfidfVectorizer() class module to perform document vectorization

TfidfVectorizer() - Transform a count matrix to a normalized tf or tf-idf representation.

Tf means term-frequency while tf-idf means term-frequency times inverse

document-frequency. This is a common term weighting scheme in information retrieval, that has also found good use in document classification.

This is the vectorize function.



Fig 1.6: Vectorization

Next step is to find similarity between text, we use the dot method of numpy computation functions. 

Fig 1.7: Numpy dot method computation

Below is the code for python script **part\_2.py**

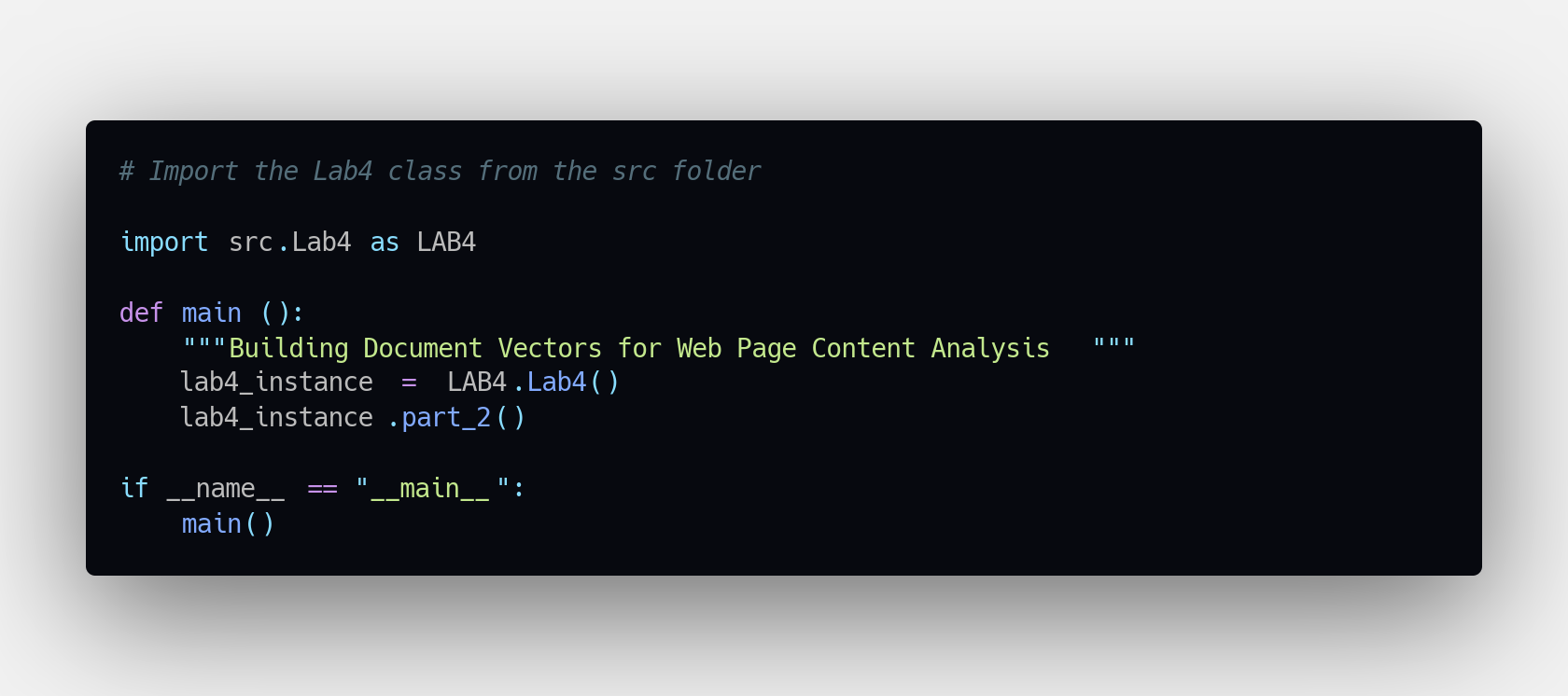


Fig 1.8: Python script part\_2.py

**Running and configuration**

Requirements.

Python3, pip modules in requirements.txt, MySQL server

1. The server credentials are located in .env file which is loaded and assigned to necessary variables during connection.

Edit this file to suit your local configuration.

1. Open your terminal and start your MySQL server, e.g. In Linux you would use this command

sudo systemctl start mysql.service

Note: Service names may be different, please confirm with your installation.

1. Unzip your project folder and change into the COPY directory.
2. Now we need pip modules installed

pip3 install –r requirements.txt

1. Now run the script part\_1.py and part\_2.py respectively

python3 part\_1.py and python3 part\_2.py

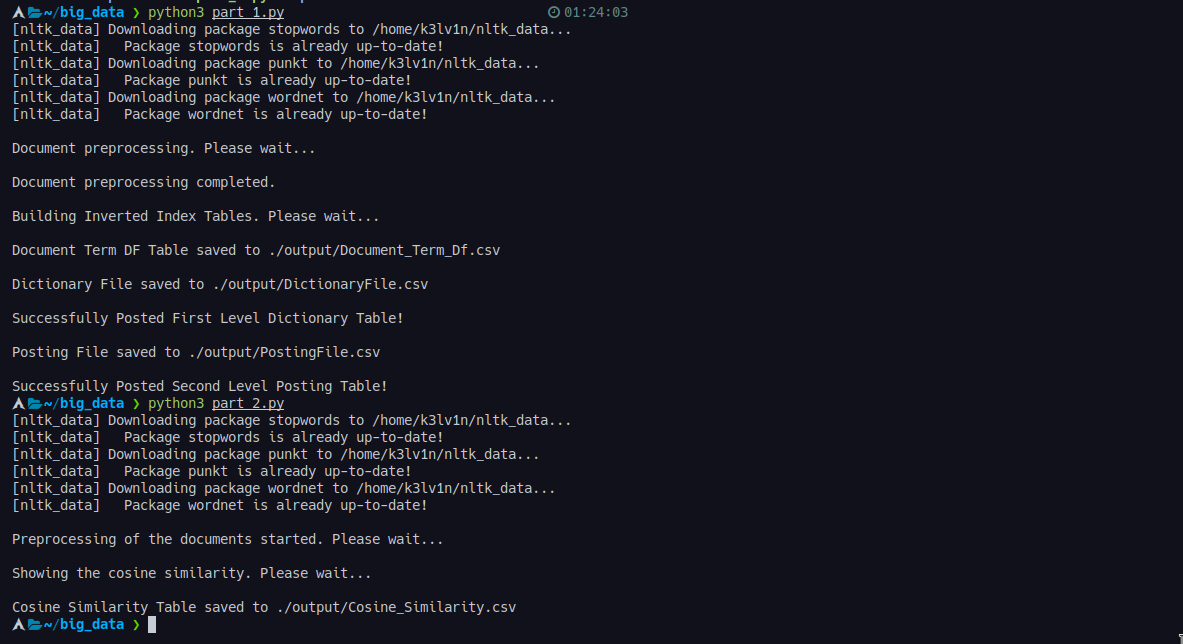


Fig 1.9: Output part\_1.py and part\_2.py