Q1. Create a new list named vegetables containing 'carrot', 'broccoli', and 'spinach'. Use

Ans:-

**Vegetables = [containing 'carrot', 'broccoli', 'spinach']**

**Print(Vegetables)**

Q2. Create a new tuple named 'more\_colors' containing 'yellow' and 'purple'. Concatenate

Ans:-

**# Define the tuples**

**more\_colors = ('yellow',)**

**more\_colors1 = ('purple',)**

**# Concatenate the tuples**

**concatenated\_colors = more\_colors + more\_colors1**

**print(concatenated\_colors)**

Q3. Unpack the elements of the concatenated tuple into individual variables (e.g., color1, color2, ...) and print them.

**Ans:-**

**more\_colors1 = (‘yellow’)**

**more\_colors2 = (‘purple’)**

**##Concatenate the tuples**

**Conactennated\_colors = more\_colors1 + more\_colors2**

**##unpack tuples**

**Tup1,tup2 = Conactennated\_colors**

**Print(Tup1)**

**Print(tup2)**

Q4. Using list slicing, reverse the order of the elements in the list 'colors = ['red', 'green', 'blue', 'yellow']'.

Ans:-

**colors = ['red', 'green', 'blue', 'yellow']**

**reversed\_colors = colors[::-1] #revserse**

**print(reversed\_colors)**

Q5. Using negative indexing, extract the last three elements from the list 'cities = ['New York', 'London', 'Paris', 'Tokyo']'.

Ans:-

**cities = ['New York', 'London', 'Paris', 'Tokyo']**

**# Extract the last three elements using negative indexing**

**last\_three\_cities = cities[-3:]**

**print(last\_three\_cities)**

**SQL**

Q1. Write an SQL query to create a table named 'image\_metadata' to store information about images. Include columns such as 'image\_id', 'file\_path', 'width', 'height', and 'label'.

Ans:-

CREATE TABLE image\_metadata (

image\_id INT PRIMARY KEY AUTO\_INCREMENT,

file\_path VARCHAR(255) NOT NULL,

width INT NOT NULL,

height INT NOT NULL,

label VARCHAR(100)

);

Q2. Insert records into the 'image\_metadata' table for three sample images, providing values for the columns.

Ans:-

INSERT INTO image\_metadata (file\_path, width, height, label) VALUES

('/images/sample1.jpg', 1920, 1080, 'Sample Image 1'),

('/images/sample2.jpg', 1280, 720, 'Sample Image 2'),

('/images/sample3.jpg', 800, 600, 'Sample Image 3');

Q3. Write an SQL query to count the number of missing values in the 'label' column of the 'image\_metadata' table.

Ans:-

SELECT COUNT(\*) AS missing\_labels

FROM image\_metadata

WHERE label IS NULL;

Q4. .Remove duplicate records from the 'image\_metadata' table based on the 'file\_path' column.

Ans:-

WITH RankedImages AS (

SELECT image\_id, file\_path, width, height, label,

ROW\_NUMBER() OVER (PARTITION BY file\_path ORDER BY image\_id) AS rn

FROM image\_metadata

)

DELETE FROM image\_metadata

WHERE image\_id IN (

SELECT image\_id

FROM RankedImages

WHERE rn > 1

);

Q5. Write an SQL query to update the 'width' and 'height' columns in the image\_metadata table. Double the values for all images.

Ans:-

UPDATE image\_metadata

SET width = width \* 2,

height = height \* 2;

DS

Q1. How will you handle missing data, errors, and inconsistencies in your text? What techniques will you use to normalize and prepare the text for analysis?

Ans:-

1. **Handling Missing Data**
2. Error Handling
3. Inconsistency Handling
4. Text Normalization
5. Tokenization
6. **Stopword Removal**
7. **Stemming/Lemmatization**
8. Punctuation and Special Characters
9. Preparing for Analysis

Q2. What features will you extract from the text that will be most relevant to your research question? (e.g., keywords, n-grams, sentiment scores, named entities)

Ans:-

**1. Keywords**

* **Frequency of Keywords**: Identify and count important terms that appear frequently in the text. Useful for topic modeling and understanding core themes.
* **TF-IDF (Term Frequency-Inverse Document Frequency)**: Measure the importance of a word in a document relative to a collection of documents.

**2. N-grams**

* **Unigrams**: Single words, useful for basic analysis and understanding word distribution.
* **Bigrams**: Pairs of consecutive words, which can reveal common phrases and context.
* **Trigrams**: Triplets of consecutive words, useful for capturing more context and phrases.

**3. Sentiment Scores**

* **Polarity**: Measure the overall sentiment of the text (positive, negative, neutral).
* **Subjectivity**: Measure the degree of subjectivity (opinion vs. fact) in the text.
* **Emotion Detection**: Identify specific emotions expressed in the text, such as joy, anger, or sadness.

**4. Named Entities**

* **Entity Recognition**: Extract names of people, organizations, locations, dates, and other proper nouns. Useful for understanding key subjects and topics.
* **Entity Linking**: Link named entities to external knowledge bases for more detailed information.

**5. Part-of-Speech (POS) Tags**

* **POS Tagging**: Identify the grammatical categories of words (nouns, verbs, adjectives, etc.). Useful for understanding sentence structure and word usage.

**6. Text Length Features**

* **Word Count**: Total number of words in the text.
* **Sentence Count**: Number of sentences, which can provide insights into text complexity and structure.

**7. Readability Scores**

* **Readability Indexes**: Metrics such as Flesch-Kincaid or Gunning Fog Index to assess how easy or difficult the text is to read.

**8. Contextual Embeddings**

* **Word Embeddings**: Vector representations of words (e.g., Word2Vec, GloVe) capturing semantic meaning and relationships.
* **Contextual Embeddings**: More advanced embeddings (e.g., BERT, GPT) that consider the context in which words appear.

**9. Topic Modeling**

* **Latent Dirichlet Allocation (LDA)**: Extract topics from the text based on word co-occurrence patterns.
* **Non-Negative Matrix Factorization (NMF)**: Another technique for identifying underlying topics in text.

Q3. How will you store and organize your collected data? (e.g., spreadsheets, databases, cloud storage)

Ans:-

**1. Databases**

* Relational Databases
* NoSQL Databases

**2. Cloud Storage**

* Cloud Services
* Data Warehousing

**3. Spreadsheets**

**4. Data Lakes**

**5. Data Management Systems**

* Data Catalogs
* Data Integration Tools

**6. Backup and Security**

* Regular Backups
* Security Measures

**7. Documentation**

* Data Dictionaries
* Version Control

The choice of storage and organization method depends on the volume of data, the complexity of data structures, and the specific needs of your analysis. Combining multiple methods can often provide the most robust solution, ensuring data integrity, accessibility, and scalability.