Workflow of Web Scraping Project

- Documented by Ansuman Parija

1. Database Creation (connectDb):

- Establishes a connection to a SQLite database named Quotes.db.
- Creates a table named Quotes if it does not already exist, with the following columns:
    - id (Primary Key, Auto-incremented)
    - theme (Text)
    - lines (Text)
    - author (Text)
- Commits changes to the database.

2. Data Fetching (fetchData):

- Sends an HTTP GET request to the specified URL (http://www.values.com/inspirational-quotes) using the requests library.
- Parses the HTML content using the BeautifulSoup library with the html.parser.
- Searches for relevant article elements (, <h2>, , <p>, <img> and <a> tags)
- Returns a list of extracted tuples (headline, summary, link).

3. Data Storage (storeData):

- Inserts the fetched data into the Quotes table in the SQLite database.
- Handles duplicate or other errors gracefully to ensure smooth operation.
- Commits the changes and closes the database connection.

4. Main Functionality (main):

- Calls connectDb to ensure the database is prepared.

- Fetches data from the provided URL using fetchData.

- Stores the fetched data into the database using storeData.

- Prints messages to indicate success or cancellation of the process.

Key Features

- **Error Handling:**

Database errors (e.g., connection issues, duplicate entries) are logged. HTTP request errors are captured and logged for debugging.

- **Code Modularity:**

Functions are well-structured for specific tasks: database setup, data fetching, and storage.

- **Customizable Scraping:**

The URL and HTML elements can be modified to accommodate other webpages and structures.