**Project Documentation: RedBus Government Bus Data Scraping and Streamlit Application**

**Overview**

This project involves scraping data from the RedBus website, specifically focusing on government bus routes. The scraped data is stored in a MySQL database, and a Streamlit application is developed to display this data with filtering and pagination functionalities.

**Project Components**

* Data Scraping with Selenium and Python: Automates the process of extracting government bus routes and associated details from the RedBus website.
* Data Storage in MySQL: Stores the scraped bus data in a MySQL database for easy access and management.
* Streamlit Application: A web interface that allows users to view and filter the bus data stored in the MySQL database.

1. **Data Scraping**
   1. **Requirements**

* Python Libraries: `selenium`, `pandas`, `pymysql`
* WebDriver: Chrome WebDriver for automating browser tasks.
  1. **Script Workflow**
* Input Parameters:
  + SQL host name, root name, and password are provided by the user.
* Web Scraping:
  + Website URL: `https://www.redbus.in/`
* Process:
  + The script navigates through the RedBus website, specifically targeting the government bus section.
  + It scrapes each bus route link and name from the website.
  + The data is stored in a CSV file for further processing.
* Error Handling:
  + The script includes error handling mechanisms to manage exceptions during web scraping and data extraction.
  1. **Storing Scraped Data in CSV**
* The scraped data (bus route links and names) is saved into a CSV file named `RB\_All\_Govt\_Bus\_Route\_Links.csv`.

1. **Data Storage in MySQL**
   1. **MySQL Database Setup**

* Database Creation:
  + The script creates a database named `All\_RedBus`.
  + A table named `All\_RB\_data` is created with columns for various bus attributes (e.g., route name, bus name, price, seats available, etc.).
  1. **Data Insertion**
* Inserting Scraped Data:
  + The script reads the CSV file and inserts the scraped bus data into the MySQL table.
  1. **Database Schema**
* Table Name: `All\_RB\_data`
  + Columns:
    1. `ID` (Primary Key)
    2. `Route\_Name`
    3. `Route\_Link`
    4. `Bus\_Name`
    5. `Bus\_Type`
    6. `Departing\_Time`
    7. `Duration`
    8. `Reaching\_Time`
    9. `Star\_Rating`
    10. `Price`
    11. `Seats\_Available`

1. **Streamlit Application**
   1. **Requirements**

* Python Libraries: `streamlit`, `pandas`, `pymysql`
  1. **Application Workflow**
* Data Loading:
  + The application connects to the MySQL database and loads the bus data into a Pandas DataFrame.
* Filtering Options:
  + Users can filter buses based on bus type, route, star rating, and price range.
* User Interface:
  + The application provides a user-friendly interface where data can be viewed, filtered, and sorted.
  1. **Streamlit Caching**
* @st.cache\_data: Used to cache the data load function to improve the application’s performance.
  1. **Session State Management**
* Filters: Session state is used to manage the selected filters for bus type, route, star rating, and price.

1. **Execution**
   1. **Running the Scraper**

* Ensure WebDriver is installed and available in your system's PATH.
* Run the Python script to start the scraping process. The script will prompt you to enter SQL credentials.
  1. **Running the Streamlit Application**
* Install Streamlit using `pip install streamlit`.
* Run the application using the command `streamlit run <script\_name.py>`.
* Interact with the application to filter and view the bus data.

1. **Conclusion**

This project automates the process of scraping government bus data from the RedBus website, storing it in a MySQL database, and providing an interactive interface for users to explore the data through a Streamlit application. The combination of web scraping, data management, and web interface development showcases a complete data pipeline from extraction to user interaction.