YouTube Data Analyzer

Overview:

The **YouTube Data Analyzer** is a powerful tool built using Streamlit, Python, and MySQL. It allows users to fetch, store, and analyze data from YouTube channels using the **YouTube Data API**. This tool provides an intuitive interface to explore channel details, videos, and comments, alongside executing predefined analytical queries to derive insights.

Key Features:

1. **Fetch YouTube Channel Data**:
   - Retrieve details like channel name, subscriber count, total videos, and description.
   - Store this data in a MySQL database while avoiding duplicate entries.
2. **View Stored Data**:
   - Access stored data for channels, videos, and comments directly from the database.
3. **Run Analytical Queries**:
   - Execute predefined queries to analyze channel performance:
     - Top videos by views, likes, or comments.
     - Channels with the most videos or comments.
     - Aggregated metrics such as total views or average video duration.
4. **User-Friendly Interface**:
   - Simple tab-based navigation.
   - Data displayed in interactive tables for easy analysis.

Installation and Setup:

Prerequisites

- **Python 3.7 or higher**: Ensure Python is installed on your system.
- **MySQL Database**: Install and configure MySQL.

Database Setup

1. Open your MySQL client and create a database:
   ```
   CREATE DATABASE youtube_db;
   ```

2.Use the following schema to set up the required tables:

```
CREATE TABLE channels (

    channel_id VARCHAR(50) PRIMARY KEY,

    channel_name VARCHAR(255),

    subscribers INT,

    views INT,

    total_videos INT,

    channel_description TEXT,

    playlist_id VARCHAR(50)

);


CREATE TABLE videos (

    video_id VARCHAR(50) PRIMARY KEY,

    channel_id VARCHAR(50),

    title VARCHAR(255),

    description TEXT,

    published_date DATETIME,

    view_count INT,

    like_count INT,

    comment_count INT,

    duration VARCHAR(50),

    FOREIGN KEY (channel_id) REFERENCES channels(channel_id)
);

CREATE TABLE comments (
    comment_id VARCHAR(50) PRIMARY KEY,
    video_id VARCHAR(50),
    comment_text TEXT,
    author VARCHAR(255),
    published_date DATETIME,
    FOREIGN KEY (video_id) REFERENCES videos(video_id)
);
```

Configure the Application

1.Replace the YouTube API key in the application:

```
api_key = "YOUR_YOUTUBE_API_KEY"
```

2.Update MySQL credentials in the create_db_connection function:

```
connection = mysql.connector.connect(
    host="localhost",
    user="YOUR_MYSQL_USERNAME",
    password="YOUR_MYSQL_PASSWORD",
    database="youtube_db"
     )
```

Run the Application

1.Start the application:

```
streamlit run app.py
```

2.Open your browser and navigate to:
http://localhost:8501

Application Usage

Tabs Overview

1. 🏠 **Home Tab**:
    - Enter a YouTube Channel ID to fetch channel data.
    - Data is stored in the database for further analysis.
2. 📊 **Fetched Data Tab**:
    - View stored data in the database for channels, videos, or comments.
3. 🔎 **Query Execution Tab**:
    - Run predefined analytical queries, including:
        - Top 10 most viewed videos.
        - Videos with the highest likes or comments.
        - Average duration of videos per channel.