Springboard Data Scraping to DBMS- Readme

1. **Overview**:
   This project demonstrates a Python script that automates the process of uploading and downloading images using the Cloudinary cloud-based image and video management service. Cloudinary provides a convenient way to manage and store images, making it easier to integrate with various applications and services.

   The script performs the following key functionalities:
   a. **Image Upload**: The script uploads a set of images from a local dataset folder to Cloudinary. It iterates through predefined folder names (e.g., "Bank Statement", "Check", "ITR Form 16", "Salary Slip", "Utility") and uploads all the images within those folders to Cloudinary.

   b. **Image Download**: After the images are uploaded, the script then downloads the same set of images from Cloudinary and saves them in the respective folders on the local machine. This allows for easy access and integration of the images in various applications or further processing.

   c. **Error Handling**: The script includes error handling to address potential issues, such as incorrect file paths or Cloudinary credential errors, ensuring a robust and reliable execution of the image upload and download processes.

   d. **Customization**: The script allows users to customize the dataset folder path and the number of images to download from Cloudinary, providing flexibility to adapt the script to different project requirements.

   e. **Documentation**: The project includes a detailed "report file" that outlines the step-by-step process of the project, including the setup, implementation, and testing. Additionally, the "readme file" provides an overview of the project, installation instructions, and troubleshooting guidance.

   By using this script, developers and researchers can streamline the process of managing image datasets, making it easier to integrate Cloudinary's powerful image management capabilities into their applications.

Saladi V S S Siva Hemanth Kumar , Hemanthsaladi2004@gmail.com

2. **Prerequisites:**

- Python 3.x
- Cloudinary account with valid credentials (cloud name, API key, and API secret)
- Google Drive account (if using a dataset folder on Google Drive)

3. **Installation:**

   a. Install the required Python libraries:

```
Copy

pip install cloudinary
```

4. **Usage:**

   a. Update the Cloudinary credentials in the script:

```python
cloudinary.config(
    cloud_name = "your_cloud_name",
    api_key = "your_api_key",
    api_secret = "your_api_secret"
)
```

   b. Specify the path to your dataset folder on Google Drive:

```python
dataset_folder =
"/content/drive/MyDrive/your_dataset_folder"
```