Challenge Description: Web Scraping and Dataset Organization for Machine Learning

Objective:

In this project, you will demonstrate your skills in web scraping, data organization, and basic machine learning dataset preparation. Your task is to scrape a dataset of car images along with their corresponding labels from a given website or API. The labels provided are specific car models: 206, 207, 405, 504, Peride, Samand LX, Samand Soren, Tara, Dena, Rana, 206 SD, and L90. Once you have collected the data, you will organize it into a structured format suitable for machine learning tasks.

Project Requirements:

1. **Web Scraping:**
    - Identify a website or API that provides car images along with their labels (e.g., specific car models).
    - Write a Python script using libraries like requests, BeautifulSoup, or Selenium to scrape the images and their labels.
    - Ensure that you handle pagination, if necessary, to collect a sufficient number of images (at least 200 images per car model).
    - Store the images and their labels in a structured format (e.g., JSON, CSV).
2. **Data Organization:**
    - Organize the scraped images and labels into a directory structure that is suitable for machine learning tasks. Specifically, the dataset should be organized as follows:

```
dataset/
├── 206/
│   ├── image_1.jpg
│   ├── image_2.jpg
│   └── ...
├── 207/
│   ├── image_1.jpg
│   ├── image_2.jpg
│   └── ...
├── 405/
│   ├── image_1.jpg
│   ├── image_2.jpg
│   └── ...
├── 504/
│   ├── image_1.jpg
│   ├── image_2.jpg
│   └── ...
├── Peride/
│   ├── image_1.jpg
```

```
│   ├── image_2.jpg
│   └── ...
├── Samand_LX/
│   ├── image_1.jpg
│   ├── image_2.jpg
│   └── ...
├── Samand_Soren/
│   ├── image_1.jpg
│   ├── image_2.jpg
│   └── ...
├── Tara/
│   ├── image_1.jpg
│   ├── image_2.jpg
│   └── ...
├── Dena/
│   ├── image_1.jpg
│   ├── image_2.jpg
│   └── ...
├── Rana/
│   ├── image_1.jpg
│   ├── image_2.jpg
│   └── ...
├── 206_SD/
│   ├── image_1.jpg
│   ├── image_2.jpg
│   └── ...
└── L90/
    ├── image_1.jpg
    ├── image_2.jpg
    └── ...
```

- Each subdirectory should represent a unique car model.
- The images should be named in a way that allows easy mapping to their corresponding labels.

3. **Documentation and Submission:**
    - Provide a detailed README file that explains how to run your web scraping script and how to use the organized dataset.
    - Include any necessary dependencies and installation instructions.
    - **Include a link to your course in the README file.**
    - Submit your project as a GitHub repository with the following structure:

```
project/
├── scraper.py
├── README.md
├── requirements.txt
└── dataset/
    ├── 206/
    ├── 207/
    ├── 405/
    ├── 504/
    ├── Peride/
    ├── Samand_LX/
    ├── Samand_Soren/
    ├── Tara/
```

```
├── Dena/
├── Rana/
├── 206_SD/
└── L90/
```

Evaluation Criteria:

- **Functionality:** Does the web scraping script work as expected? Is the dataset correctly organized and usable for machine learning tasks?
- **Code Quality:** Is the code well-structured, readable, and documented? Are appropriate libraries and tools used?
- **Efficiency:** Is the web scraping process efficient? Is the dataset organized in a way that is easy to use for machine learning?
- **Creativity:** Did you come up with a unique approach to solving the problem? Did you handle edge cases or challenges in an innovative way?

Bonus Points:

- Implement data augmentation techniques using Python libraries to enhance the dataset.
- Write unit tests for your web scraping script.
- Deploy your dataset as a web service using Flask or FastAPI or Django allowing users to query the dataset via an API.

Deadline:

You have **[30 days]** to complete this project. Good luck, and happy coding!

This challenge is designed to test your ability to combine web scraping and data organization. It will give you hands-on experience with real-world data collection and preparation for machine learning tasks.