Project Title: *Premier League Insights: Scraping, Analyzing, and Visualizing Football Stats*

Project Description:
In this project, I will extract and analyze data from the official Premier League website. The goal is to gather rich football-related data such as player statistics, match results, team standings, and fixture details. This data will be cleaned, processed using regular expressions where needed, analyzed for insights, and finally visualized for better understanding.

Target Website & Data Type:
Website: https://www.premierleague.com
Data to be extracted includes:

- Match results (scores, dates, venues)
- Team statistics (wins, losses, points)
- Player stats (goals, assists, appearances)
- Fixture list and schedules

**** Overall Project Goal**:
To build a structured dataset from raw football data, derive meaningful statistics and trends, and present them visually. A final dashboard using Streamlit (bonus task) will allow interactive exploration of this information.

Approach:

1. **Data Extraction**:
   I will scrape data using Selenium or BeautifulSoup (due to the site's dynamic content). Focus will be on player profiles, fixtures, results, and standings. Raw data will be stored in CSV or JSON format.
2. **Data Cleaning & Regular Expressions**:
   Clean the raw text by removing HTML, handling nulls, and correcting inconsistent formats. Use regex to:
   a. Extract match scores (e.g., `(\d+)-(\d+)`)
   b. Parse dates (e.g., `\d{2}/\d{2}/\d{4}`)
   c. Extract structured player data such as height, position, or nationality
3. **Data Analysis**:
   Analyze team performance (goals scored/conceded, win ratios), identify top

players (goals, assists, appearances), and detect trends such as weekly form or head-to-head records.

4. **Data Visualization**:
   Visualize key insights using tools like matplotlib, seaborn, or plotly. Planned visuals include as EX:
   a. Bar charts for top scorers
   b. Line charts for team performance over the season
   c. Heatmaps for venue-wise or time-based patterns
5. **Data Storage**:
   Store the cleaned and structured data into a MongoDB database for organized access and scalability.
6. **(Bonus) Streamlit Web App with Prediction and Player Comparison**:
   **~ Develop a user-friendly, interactive dashboard using Streamlit that allows users to explore teams, players, and match stats with filters and dynamic graphs.
   Additionally, incorporate:
   ~A **match outcome prediction model (e.g., logistic regression or machine learning) that predicts upcoming match results based on team form, recent performance, and historical stats.
   ~A **player comparison tool, where users can select two players to view their stats side by side. The app will display comparative visuals like radar charts or bar graphs to highlight key differences and performance metrics.

Expected Outcomes:

- A complete end-to-end data pipeline: from web scraping and cleaning to analysis, visualization, and database storage
- A well-structured dataset containing detailed Premier League match, team, and player statistics
- Cleaned and processed data stored in a MongoDB database for efficient access and scalability
- Insightful visualizations that clearly communicate trends such as team performance over time, top scorers, or match frequency by venue
- *An interactive **Streamlit web application** for exploring match data, team stats, and player profiles
- A **match prediction feature** that provides users with predicted outcomes for upcoming fixtures using machine learning or rule-based models

- A **player comparison tool** that allows users to compare two players side-by-side using statistical data and visual charts
- Overall, a dynamic and interactive football analytics platform that showcases both technical data skills and domain knowledge