Programming – Introduction Level

Group Project

Name: HelveticHockeyStats

Nathan Willemin, Benoît Gainon

22.12.23

Assignment overview:

This assignment focuses on the design, implementation, and testing of a Python program to process collections of data using data structures, as described below.

Assignment deliverable:

The deliverable for this assignment is the following file:

HelveticHockeyStats.py

Assignment background:

The National League is made up of 14 teams spread throughout Switzerland. In this project, you will be able to browse through the various statistics of the top tier ice hockey league in Switzerland.

The data was retrieved from the following csv file, which is regularly updated throughout the season:

https://data.sihf.ch/Statistic/api/cms/export?alias=player&searchQuery=1%2F%2F1&filterQuery=&filterBy=Season%2CPhase%2CTeam%2CPosition%2CLicence&orderBy=points&orderByDescending=true&format=csv

Each row from the dataset contains the following information:

- Ranking
- Player name
- Team of the player
- Player position (defender or forward)
- Number of games played
- Number of goals
- Number of assists

- Number of points
- Points per game
- Total penalty time (in minutes)
- Net number of goals scored/conceded by the player's team when he was on the ice

We removed this last column (named [+/-]) which, in our opinion, was lacking relevance and added three columns:

- Penalty time per game
- Total efficiency (representing the difference between the total points and the penalty time)
- Efficiency per game

Assignment specifications:

Using the code presented in this project, the user can choose from these ten options:

1. **Player**: The statistics of one specific player by entering his name. The data is displayed both in text and as a bar chart.
2. **Ranking:** Retrieve statistics for the player at a specific ranking position.
3. **Team:** List all players from a specific team along with their total points.
4. **Team Rankings**: Display summed up statistics for each team (goals, assists, total points, total penalties, total efficiency) and visualize the total points of each team in a bar chart.
5. **Top Rankings:** Show the top 'N' players based on total points or points per game, where both 'N' and the metric are chosen by the user.
6. **Defenders:** Display the top 'N' defenders, sorted by their initial ranking in the dataset, i.e. the total number points.
7. **Goals:** List the top 'N' players sorted by the total number of goals scored.
8. **Assists:** Provide a list of the top 'N' players sorted by the total number of assists.
9. **Efficiency:** Show the top 'N' players based on total efficiency or efficiency per game.
10. **Penalties:** Display the top 'N' players sorted by total penalties or penalties per game.

Assignment notes:

In this comprehensive Python script, we delve into the fascinating world of ice hockey statistics. The first step is to set up the environment by importing essential libraries like Pandas for data manipulation and Matplotlib and Seaborn for insightful visualizations.

The heart of the script lies in its ability to fetch and process data directly from a specified URL, where it retrieves a CSV file containing detailed player statistics. This data is then transformed: German terms are translated to English and “R” is changed to “Ranking” for better understanding and some columns are removed or added (see ‘Assignment Background’).

An interactive loop forms the core interaction point with the user. Here, users can search the dataset based on various criteria such as player details, team stats, rankings, and more. Each search criterion is initially addressed using conditional statements (if, elif, else) in order to provide insightful information:

- Looking up individual players reveals their statistics in both numerical and graphical forms.
- Team-based searches provide an aggregated view of the team's performance.
- Ranking searches enable quick access to statistics of players at specific ranking positions, with checks to ensure user inputs are valid.

Visual representations of data using Seaborn offer a more intuitive understanding of the statistics, making it easier to digest complex information. The script is also robust, equipped to handle invalid inputs smoothly and guide users towards correct usage.

To conclude, this script is a great tool for Swiss ice hockey fans and tries to differentiate itself from usual statistics websites with its user-friendly and interactive interface.