Command based youth hockey cup management system (CYHCMS)

Author name: Lema Gebrie

Date: October 30/2024

Table of Contents

1. Introduction

The "Youth Hockey Cup Management System" is a Python project developed to help manage hockey teams for a youth hockey tournament. It stores essential team information, including team name, gender, age category, fee, payment status, and participation status (active or canceled). The program allows users to create, update, delete, and view team records and provides statistics on team registration and payment statuses. All the updated values will be saved automatically on the teams.txt file.

The result of the recorded lists of teams can be shown at the temas.txt file in which it automatically comes while we are opening the main.py program or it will create itself automatically while we create a new list after we run the main program.

2. Solution Description

The solution is structured into three primary components: **Team class, user_interface class, and main class** as the program's entry point.

2.1 Team Class (team.py)

- **Purpose and Responsibilities:**
  The Team class represents individual hockey teams. It is responsible for storing team data, including:
  - Team ID (unique identifier)
  - Name, gender, coach, and age category
  - Fee paid status (Yes/ No) and amount due if unpaid
  - Participation status and cancellation date (if applicable)
- **Collaborations:**
  The Team class collaborates closely with the User_Interface class, which instantiates Team objects and handles their interactions, including saving and retrieving data from the teams.txt file.

2.2 User_Interface Class (user-interface.py)

- **Purpose and Responsibilities:**

  The User_Interface class manages the user interaction for the program. It provides a menu-driven interface with options to create, view, update, list, and delete team information. It includes methods to:

  - Save teams to teams.txt, ensuring persistence across sessions.
  - Display statistics, such as the total registered team, percentage of teams which paid fees.
  - Handle user selections to perform various actions on team data.
  - Update or cancel team participation: In this case the historical data will not be deleted since we need for future analysis. Therefore cancelation means not deletion in this case.

- **Collaborations:**

  The **User_Interface** class relies on the **Team class** to store individual team details and the main program to start the interface. It reads from and writes to teams.txt to keep records update.

2.3 Main Program (main.py)

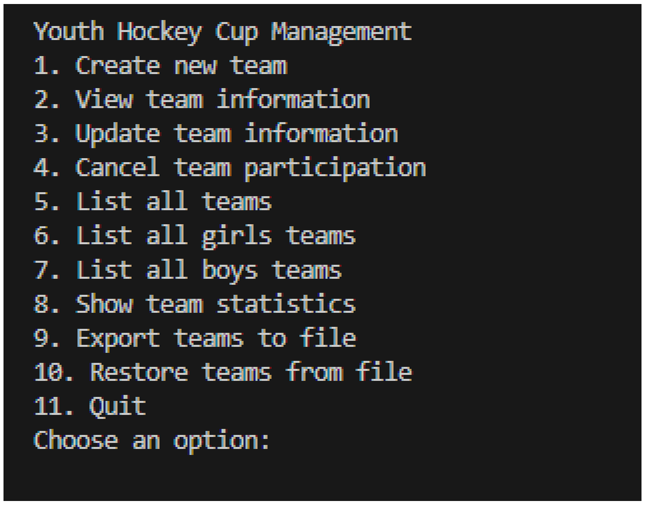
- **Purpose and Responsibilities:**

  The main program initializes the User_Interface and starts the application. It serves as the entry point, allowing users to interact with the menu options defined in User_Interface.

3. User Guide

To Run the Program please follow the following steps properly.

1. **Setup**: Ensure you have Python installed.
2. **Save all the three classes** (team, user_interface and main) in the **same folder**
3. Make sure all the three programs are error free
4. **Execute main.py**: Run the main.py script to start the program. This will display a menu with options



```
Youth Hockey Cup Management
1. Create new team
2. View team information
3. Update team information
4. Cancel team participation
5. List all teams
6. List all girls teams
7. List all boys teams
8. Show team statistics
9. Export teams to file
10. Restore teams from file
11. Quit
Choose an option:
```

(I run it in visual studio)

Menu Options:

- **Option 1**: Create a new team by providing team name, gender, age category, coach, and fee status (This is the first and most important menu that we should know because we are going to the other menu options after we register team information otherwise there will be no team to view, update, list, show statistics etc.)
- **Option 2**: View a specific team's details using the team ID, as we create each team then our program gives an automatic a 6 digit team ID and we should use this ID to view a specific team. We can copy or see the teams ID on the teams.txt file or by just displaying the team using option 4 (In which it can show all the teams info).

- **Option 3**: Update team details, including changing the participation status to canceled if needed (In this case also we are going to use the teams ID for updating if necessary)
- **Option 4**: cancel team participation, we are using ythis menu to cancel the teams participation
- **Option 5**: List all teams, this option displays all the registered and stored team details. Practically speaking "List all teams" option is equal to teams.txt lists because it shows all the teams which are saved in the teams.txt file.
- **Option 6 and 7**: List teams by gender (boys or girls), for analysis purpose if we need to see the lists based on their gender we can use option 5 to view all girls teams and option 6 to view al the boys teams.
- **Option 8**: Display statistics, including the total teams and the percentage with fees paid.
- **Option 9**: Save the current team data to teams.txt, this option is used to export the newly registered data to the teams.txt file but the files we are creating also automatically saved on the teams.txt file.
- **Option 10**: restore teams from a file: used to access or restore files saved in a file.
- **Option 11**: Quit: used to exit or terminate the main.py code

Note: We don't need to have the import option because we need only save a previously created text file with same column numbers as we have on the code and it will read as well as it can add to it accordingly.

Conclusion

The command based hockey team management system is sued to easily manage the registration, update, follow up and looking in o the statistics od teams (like how many teams registered, how many of them paid in percentage etc). To use the system properly we need to follow all the steps and instructions described above.