Logo, Dalarna University

A logo of Dalarna University

**GIK29B: PYTHON AND R PROGRAMMING**

**A PROGRAM TO CREATE AND ADMINISTER HANDBALL TEAMS**

**JAPHET ALFRED NTWALLE**

**2022-10-26**

Table of Contents

[1. INTRODUCTION 3](#_Toc117935382)

[2. DESCRPTION 3](#_Toc117935383)

[2.1 Team class 3](#_Toc117935384)

[2.1.1 Data attributes 3](#_Toc117935385)

[2.1.2 Setters, getters, string method 4](#_Toc117935386)

[2.2 Interface class 4](#_Toc117935387)

[2.3 Main Program 5](#_Toc117935388)

[3. User guide 5](#_Toc117935389)

[3.1 Running the program 5](#_Toc117935390)

# **INTRODUCTION**

This report describes an object-oriented program implemented in the Python programming language. The program allow the user to register and administer handball teams for a handball event. It allow the user to create teams, display team information based on team unique identifiers, name or type. It also allows the user to update different fields of a particuart team, delete a team from the list, save all the team information as string to a text file as well as restore back all the information into the program as objects.

The program contains 3 python script files: team.py, menu.py and script.py . The team.py script contains one class named Team which is used to create teams which are instances of the class Tram. methods to create, read, update,delete, save and restore teams are contained in the menu.py file. The script file contains the main program which allow the user to run the program.

The program is to be executed on the command line interface where the user will run the script-py file and be presented with a menu to perform all the required operations.

# DESCRPTION

In this section, a description of each class used in the program is presented. This includes the purpose of the class and hiw it interact with other classes in the the program

## Team class

The Team class is contained in the team.py script file. The purpose of this class is to allow the user to create a handball team object containing a team id, date of creation, name, fee status, fee amount and team participation status. The Team class is imported into the menu.py script to be used for creating, updating, reading, and deleting teams.

Text, website

a python class definition

The class Team contains a class variable which is used to increment the id attribute for each object created. This is done to make sure each object created has a unique identification.

### Data attributes

The Team class is initialized by defining the \_\_init\_\_ method and passing the data attributes as parameters. The id and date attributes are set automatically and they are read only as the user can not change them. The user only have access to team name, type and fee status. The fee amount is set automatically depending on the fee status, if a team has paid the fee, the fee amount will be set to 99 otherwise wit will be set to 0. The cancel attribute represents the participation status of the team. All the teams created will have a default value of None with the assumption that all teams will be participating in the event. If a team cancels the attribute will be updated to a date represting the date a team cancelled their participation in the event



### Setters, getters, string method

To restrict user access to the data attributes and validate the user input, the @property decorator is used. A string method is included to allow displaying team information in a human readable format

Graphical user interface, application

Description automatically generated

## Interface class

The Interface class is contained in the interface.py script file and mainly contain methods to perform all the operations. The class define the methods that will be used to create a team, read and update team information, delete a team from the list, save the team information as strings to a text file, restore the team information as objects using the saved text file. Also, the Interface class has a class attribute named teams which is list responsible for storing all the teams created.

The Interface class imports the Team class from the team.py file and use it to perform the following functionalities:

* create team objects (instances of the Team class), Allow the user to create a team by passing in the team name, type ans fee status, the method then create a team and append it to the list of teams
* Display team information: Allow the user to display team information using the team id. The user enters a team id and the methods display the team id, name, type, fee status, fee amount and participation status. The user can also choose to display teams by type or display all teams contained in the list
* Update team information: By using the team id, user can update the team name, type, fee status or participation status.
* Cancel participation: If a team withdraw from the event, the team information is update to include the date the team cancelled theor participation. By default all teams created have a default value of None meaning that they will be participating in the event
* Save data: After creating teams, a user can decide to save the teams information in a text file in order to have access to them even after the program session has ended. This will allow the user to restore the information the next time they run the program and be able to add ,display, update or delete teams
* Read data: if the user has previously created teams and saved them into a text file, this method allow them to restore the teams back into the program. By using the text file, the teams are recreated back into objects and then stored in a list. The user can then continue to perform other operations as usual
* Count teams:displays the total number of teams registered for the event. It also displays the percentage of teams which have paid the participation fee
* Display the menu: In order for the user to be able to run the program, two methods are defined to display the program menu, one to display the main menu and the second to display a sub menu when updatinh a team information
* Delete
  1. Main Program

The menu.py svript file is the main program/executing program. It imports the Interface class and call its methods accordingly. It also import the sys module that is used to exit from the program. The program has global constants for the main menu as well as the uodate sub menu. Once the user run the program, the menu will be displayed and user will be prompted to enter their choice according to the menu.

# User guide

This program allow the user to register and administer handball teams. The program contain three python script files named team.py, interface.py and menu.py. To be able to run the program make sure that all the files are in the same folder/working directory. This guide will assume the user has VSCode installed

## How to run the program

* Open the terminal/command line
* Navigate to the folder where the scripts are stored

Text

Description automatically generated

* Run the command python menu.py: After running the command a menu will be displayed as well as a prompt to enter a choiceText

  Description automatically generated
* Your choice should be an integer from the main menu
  1. Create a team : On the command line prompt enter 1 and press enter on your keybord. Fill in the name of the team to be created, the type of team ( B for boys and G for girls) and fee status( yes or no) to show if team have paid the participating fee
  2. Show team usingid: Display a team information using the team id, enter 2 followed by enter key on your keyboard. Enter the id of the team to display, hit enter. The team info will be displayed above the main menuA picture containing text

     Description automatically generated
  3. Show team by type: Display team information based on type. Enter 3 followed by the enter key. Fill in the type of team to show (b/boys, g/girls). This will display all teams of the specified type
  4. Show all teams: enter your choice followed by the enter key. This will display all the teams created so far.Since the teams are displayed in rows, make sure your display window is expanded to see all the teams
  5. Update team: