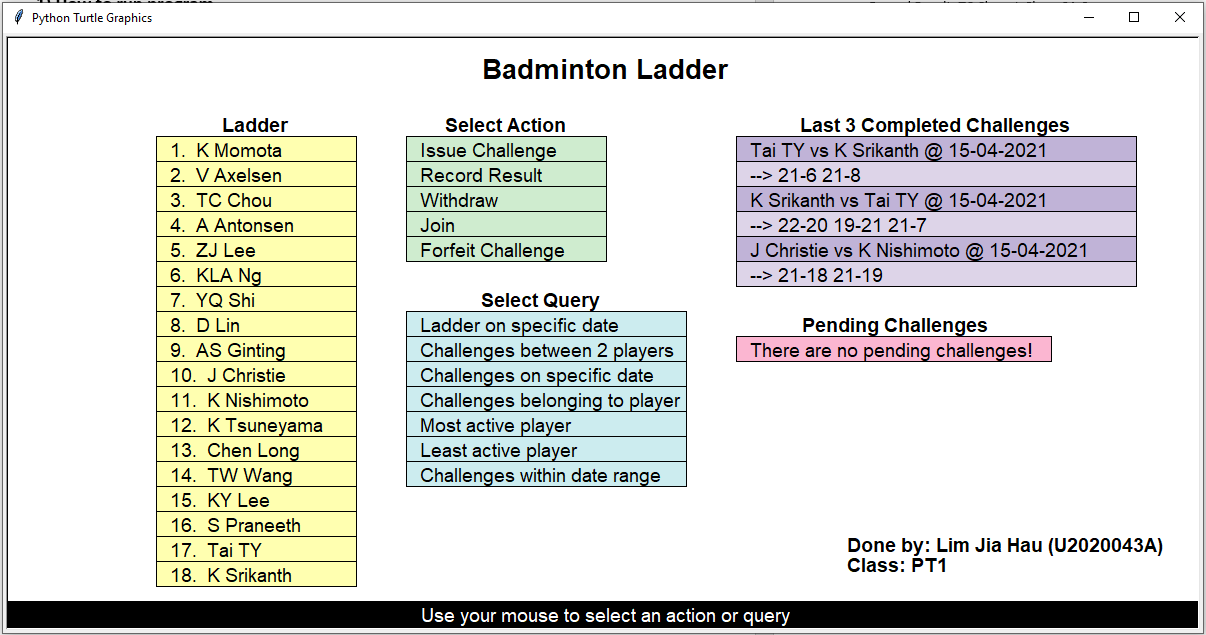
MA1008 Introduction to Computational Thinking

Mini Project: A Badminton Ladder

**1) How to run program**

My program consists of four python files. Running the “main.py” file will lead you to the **Main Selection Screen** as shown below. The helper files, “load.py”, “actions.py”, and “queries.py”, are additional code for the program to run as intended.

**Main Selection Screen**



The Main Selection Screen consists of “Ladder”, “Select Action”, “Select Query”, “Last 3 Completed Challenges”, and “Pending Challenges”.

**Ladder** – Displays the up-to-date ranking of players.

**Select Action** – Collects input from user to perform specific actions.

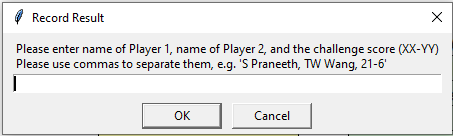
**Select Query** – Collects input values so that the data can be returned and displayed accordingly.

**Last 3 Completed Challenges** – Displays the three most recent completed challenges with their date and scores.

**Pending Challenges** – Displays challenges that are currently ongoing.

For the program to run as intended, it is critical that the input from the follows the correct formatting. For example, under **Collecting Input**, there is a standard format given which the user has to abide by when keying in results of a match. Not following the format will lead to an error message at the bottom of the main screen, preventing the user from progressing. Note that the inputs are case sensitive. For example, “D Lin” and “d lin” are not the same player. This was done to encourage users to be intentional with their inputs.

**Collecting input**



**Example of Incorrect inputs**







**Select Actions**

**Issue Challenge** - Clicking on “Issue Challenge” will prompt user to input the player 1 (challenger), player 2 (challenged), and date (e.g., S Praneeth, TW Wang, 10-04-2021). The challenger has to be a player of lower rank than that of the challenged. Note that one player may only have one outstanding challenge, either as the challenger or the challenged. After issuing the challenge, it will be on hold in Pending Challenges until further action is carried out.

**Record Result -** Each input into “Record Result” counts as one match out of best of three matches.

**Withdraw** – Clicking on “Withdraw” and inputting a player name will remove said player from the ladder. Players ranked below will move up to fill the empty position.

**Join** – Clicking on “Join” will prompt a user input for the name of the player which doesn’t exist in the ladder to be added to the bottom of the ladder.

**Forfeit Challenge** - Clicking on “Forfeit Challenge” will prompt user to enter the names of players that are currently in “Pending Challenges” (e.g., S Praneeth, vs TW Wang). Once the challenge has been forfeited, it will be voided.

#Comments: The program has been coded such that if a player decides to forfeit a challenge or withdraw from the ladder while in a Pending Challenge, the match will be voided. As if the match never happened in the first place.

**Select Query**

The program gives the user the ability to extract data from the Data.txt file and display them in a graphic display. Click on the desired query, and input accordingly.

**Ladder on specific date -** Input the date desired to display rankings on a specified date. (e.g., rankings on 02-02-2021)

**Ladder on 02-02-2021**



**Challenges between 2 players -** Input the names of two players to see their match history (e.g., S Praneeth vs J Christie). The scores and dates of the matches will also be displayed.

**Challenges on specific date -** Input the date desired into the input prompt to receive data on the matches that happened on a specific date (e.g., 05-01-2021). The scores, date, and players that competed on that date will be display.

**Challenges belonging to player** -Input the name of a player to view their match history (e.g., ZJ Lee)**.** The date and score of the match will be displayed, including the competitors.

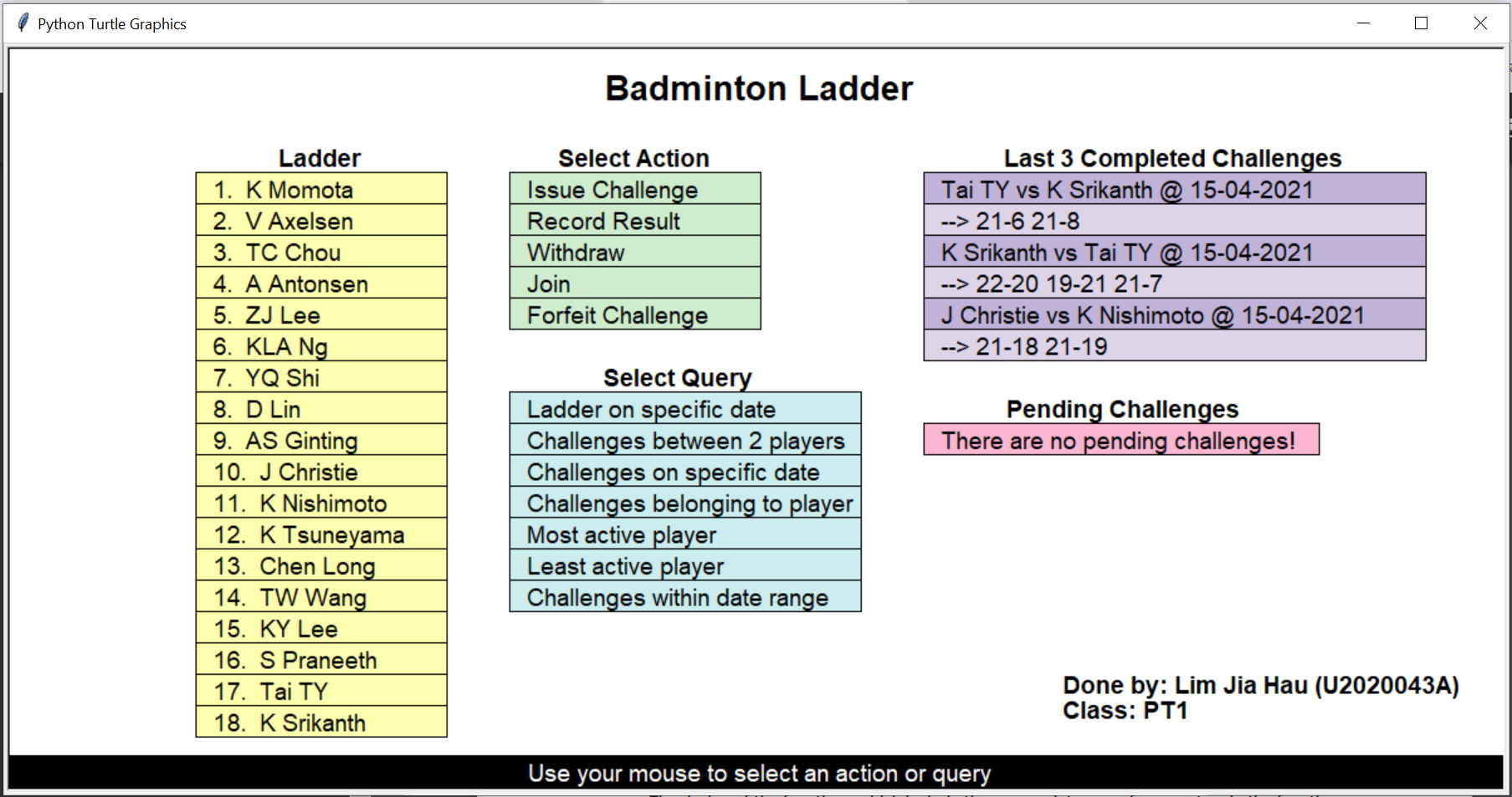
**Most and Least active player -** Clicking on the “Most active player” or “Least active player” will show the most active player and least active player respectively

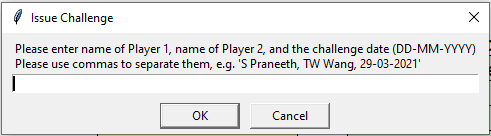
**Challenges within date range -** Input the desired range of dates (e.g., 01-03-2021, 31-03-2021) to view the matches that occurred during that period. Names of competing players, scores and dates of the matches will be displayed.

To check the ladder on a specific date, under “Select Query”, click “Ladder on a specific date”. Input “12-04-2021”. You will be able to see that D Trump was on the ladder before being removed, at position 15.

**2) Data of a Sequence of Runs**

**1) Issue Challenge and Record Result**





Input

Issue Challenge: Tai TY, K Srikanth, 15-04-2021

Record Result: Tai TY, K Srikanth, 21-6

Record Result: Tai TY, K Srikanth, 21-8

Output (display)

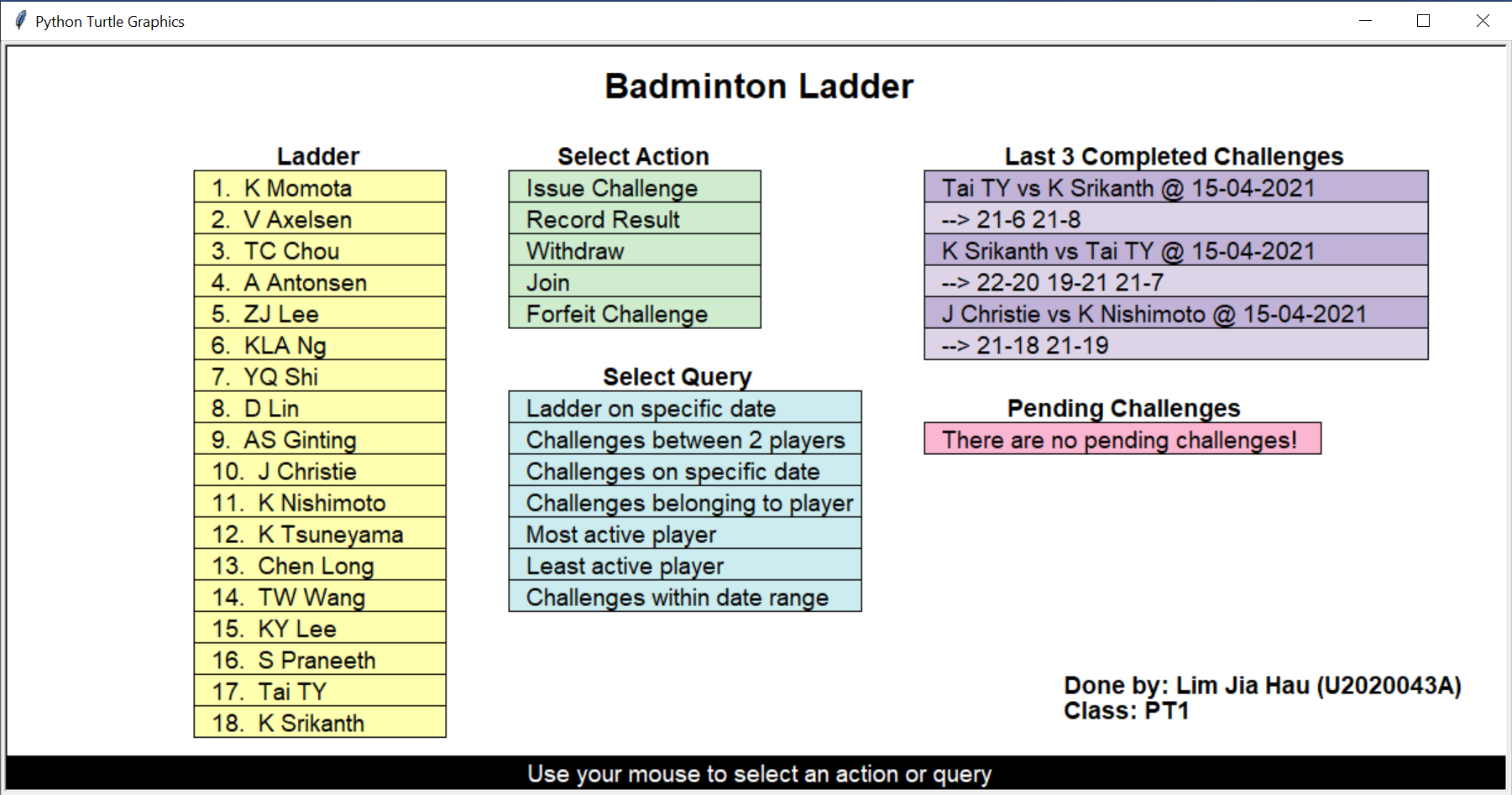
Last 3 Completed Challenges: Tai TY vs K Srikanth @ 15-04-2021

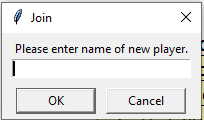
* 21-6 21-8

Data.txt file

Tai TY 18/K Srikanth 17/15-04-2021/21-6 21-8

**2) Insert players**





Input

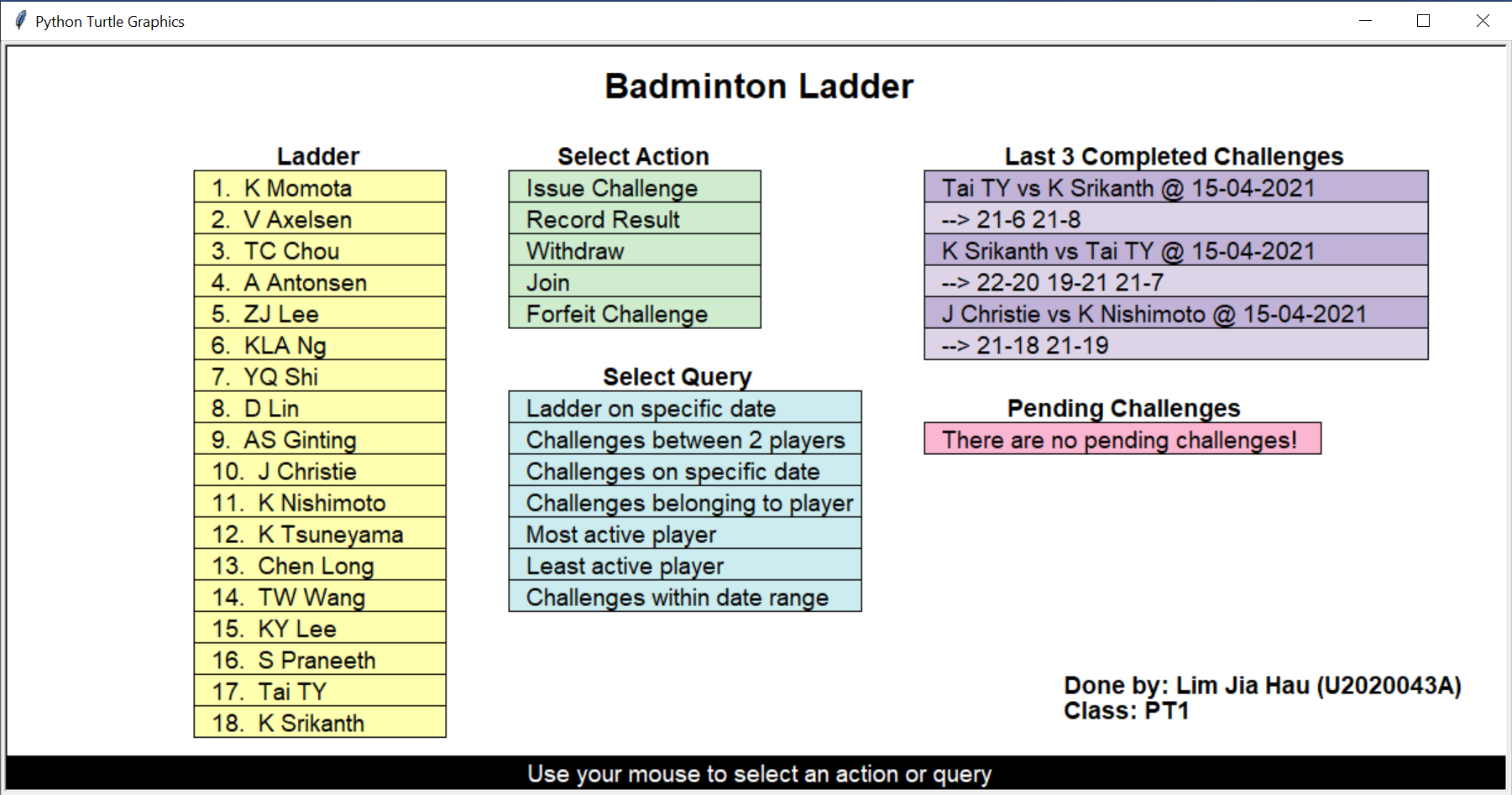
Join: K Srikanth

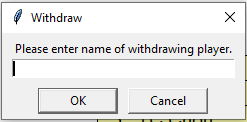
#Comments: After inputting K Srikanth into “Join” under “Select Action”, he will appear in the ladder as the bottom ranked player at the moment of insertion.

Data.txt file

+K Srikanth/13-04-2021

**3) Removal of players**





Input

Withdraw: D Trump

#Comments: After inputting D Trump, his name will no longer be on the ladder.

Data.txt file

-D Trump 15/13-04-2021

**#**Comments: In the Data.txt file, the numeric value after the name of the player indicates his position on the Ladder at the time when he was removed.

**3) Key Strengths and Limitations**

Strengths:

* Well-structured code, with multiple files to split into different sections. This helps with readability and understanding of the program.
* Program has good error checking and input validation to ensure that the user does not input something that is invalid (e.g., inputting an invalid date like 34-04-2021)
* The display is simple and easy to read while not straining to the user.

Limitations:

* Number and type of queries is limited to the programmer.