

ICT 162
Object Oriented Programming

Tutor-Marked Assignment

July 2022 Presentation

TUTOR-MARKED ASSIGNMENT (TMA)

This assignment is worth **24%** of the final mark for **ICT162, Object Oriented Programming**.

The cut-off date for this assignment is **Wednesday, 26 October 2022, 2355 hrs**.

Note to Students:

Submit your solution document in the form of a single MS Word file. You are to include the following particulars in your submission: Course Code, Title of the TMA, SUSS PI No., Your Name, and Submission Date. Put this information in the first page of your solution document. Use the template word document provided – **SUSS_PI_No-FullName_162TMA.docx**. Rename the file with your SUSS_PI_No, FullName, and read the submission details under Module\TMA01 in the ICT162 L01. **Do NOT submit as a pdf document.**

You should make only one submission for TMA.

You are to copy and paste Python source code into your solution document as text. If you submit source code as image, no marks will be awarded to your program. Submit screenshots for **only** output of your program, where applicable.

Assignment Requirements:

- Unless specified in the question, you CANNOT use packages not covered in this module, e.g., re, collections, numpy, pandas etc.
 - All classes must be documented. Provide sufficient comments to your code and ensure that your program adheres to good programming practices such as not using global variables.
 - Failing to do so can incur a penalty of as much as 50% of the mark allotted.
-

Read the following introduction before attempting this TMA.

In the game of Golf, golfers have a collection of tools that they used to get the golf ball from the tee box to the hole. These tools are called golf clubs. A golfer is allowed to carry these clubs in his golf bag and free to decide what the final makeup ultimately looks like. The diagram in Figure A shows the 3 categories of golf clubs.

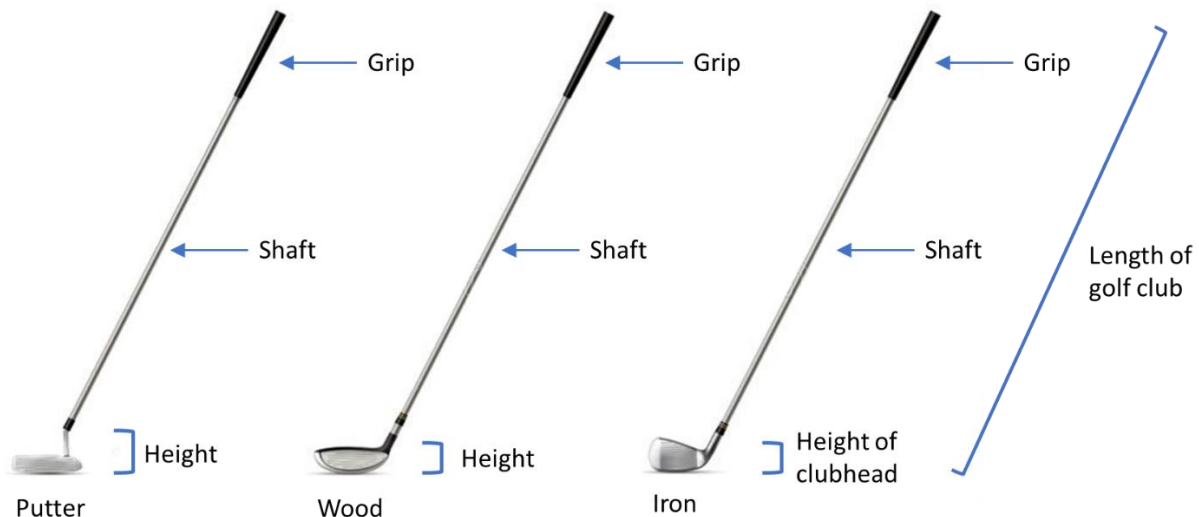





Figure A

Each club is made up of:

- Clubhead: This part of the club is located at the bottom.
 - All heads have loft and weight. Loft is the angle created between the face of the clubhead and the ground and will be different for every club. The loft and how fast the clubhead is swung will have a direct impact on the distance the ball travels.
 - The following are the 3 common types of clubhead:

Putter	Wood	Iron
		
Putter heads has different styles like blade, half-mallet, and mallet. The putter is used to roll the ball forward into the hole.	Woods are bigger and rounder clubheads that are designed to shoot the ball over long distances. The size of the head is measured in cubic centimetre (cc).	Made of hard (cast) or soft (forged) iron. Irons are the most represented clubs in a golfer's bag, numbering from 3-iron to 9-iron, pitching and sand wedge.

- Grip: Top part of the club, where golfers will hold. Can be rubber, leather or synthetic.
- Shaft: Middle part, either graphite or steel, with different flex. Bottom end of the shaft is connected to the clubhead, and the top part is installed with the grip.

Question 1 (20 marks)

Study the class diagram in Figure Q1, which focuses on the clubhead class and its subclasses.

Demonstrate your understanding of how class hierarchy and association are used to organize information, and then construct the class hierarchy and association according to specification.

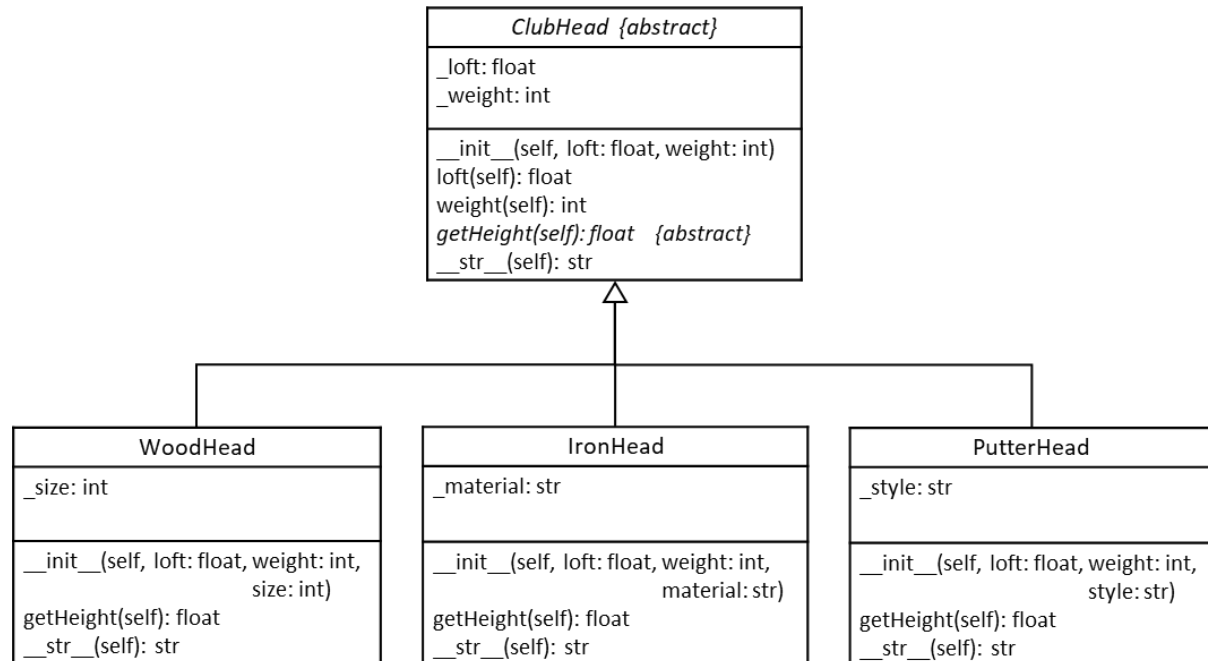


Figure Q1

(a) Implement the abstract class – ClubHead.

The ClubHead class has:

- Two instance variables:
 - `_loft` (float): loft of the clubhead in degrees.
 - `_weight` (int): weight of the clubhead in grams.
- Constructor initialises these 2 instance variables: `_loft` and `_weight`.
- Getter methods for the instance variables: `_weight` and `_loft`. Use the property decorator.
- One abstract method, `getHeight` which returns the height of the clubhead.
- The `__str__` method returns a string representation of a ClubHead object, in the following order: loft, weight, separated by commas:

```

9.0,200
37.0,250
3.0,380
  
```

(5 marks)

- (b) Write the Python class definitions for the WoodHead, IronHead and PutterHead.

The WoodHead class is a subclass of ClubHead, and it has:

- One additional instance variable: `_size` (int) representing the head size in cubic centimetre (cc).
- Implement the method `getHeight` that returns the height of the WoodHead. The height is calculated by dividing the size by 400. Hence, for head size 400cc, the height is 1 inch, while head size of 200cc is 0.5 inch.
- The `__str__` method returns a string representation of a WoodHead object, in the following order: “Wood”, loft, weight, size, separated by commas:

```
Wood,9.0,200,445
Wood,10.5,202,450
```

The IronHead class is also a subclass of ClubHead.

- One additional instance variable: `_material` (str) indicating if the head is made using cast or forged iron.
- For IronHead, the method `getHeight` returns 1 inch as iron design is limited.
- The `__str__` method returns a string representation of a IronHead object, in the following order: “Iron”, loft, weight, material, separated by commas:

```
Iron,37.0,250,Cast
Iron,56.0,280,Forged
```

The PutterHead class is also a subclass of ClubHead.

- One additional instance variable: `_style` (str) indicating if the putter is a “Blade”, “Half-Mallet” or “Mallet”.
- For PutterHead, the method `getHeight` returns 1 inch if the putter is a “Blade”. Otherwise, the method returns 0.5 inch.
- The `__str__` method returns a string representation of a PutterHead object, in the following order: “Putter”, loft, weight, style, separated by commas:

```
Putter,3.0,380,Blade
Putter,3.5,360,Mallet
```

(12 marks)

- (c) Write a main() function to do the following:

- (i) Create the appropriate ClubHead objects using the information below.

ClubHead	Loft (degrees)	Weight (grams)	Others
Putter	3.5	365	Style: Blade
Iron	37.5	285	Material: Forged
Wood	9.5	206	Size: 450cc

- (ii) Print the details (loft, weight etc.) and height of the created ClubHead objects in Q1(c)(i).

(3 marks)

Question 2 (25 marks)

As shown in Figure A, the golf club has 3 parts – the clubhead, shaft and grip. In Figure Q2, the class diagram is updated with Shaft, Grip and Club classes, to mimic the construction of golf club.

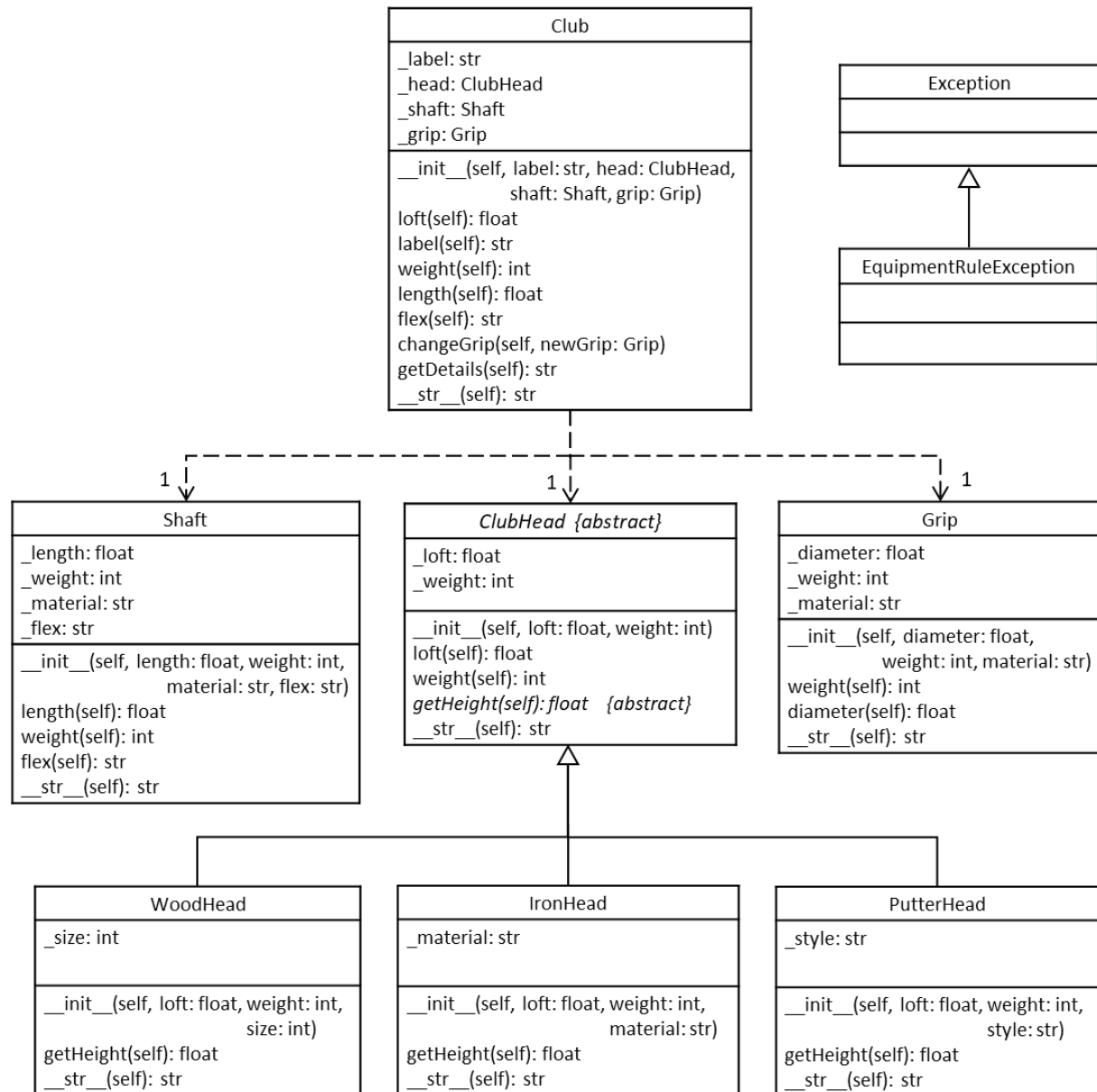


Figure Q2

(a) Implement the Shaft and Grip class.

The Shaft class has:

- Four instance variables:
 - `_length` (float): indicating the length of the shaft in inches, e.g., 42.5, 45.75, 40
 - `_weight` (int): weight of the shaft in grams.
 - `_material` (str): indicating if the shaft is made of graphite or steel.

- `_flex (str)`: a rating of a shaft's ability to bend during the golf swing, usually express in following terms: SR for Senior, R for Regular, S for Stiff, XS for Extra-Stiff.
- Constructor initialises the 4 instance variables using the given parameters.
- Getter method for instance variables length, weight, and flex. Use the property decorator.
- The `__str__` method returns a string representation of a Shaft object, in the following order: length, weight, material, flex, separated by commas:

```
42.5,79,Graphite,R
47.0,106,Steel,S
```

The Grip class has:

- Three instance variables:
 - `_diameter (float)`: represent the thickness (diameter) of the grip in inches, usually 0.6 or 0.58
 - `_weight (int)`: weight of the grip in grams.
 - `_material (str)`: indicating if the grip is made of rubber, leather or synthetic.
- Constructor initialises the 3 instance variables using the given parameters.
- Getter methods for instance variables weight and diameter. Use the property decorator.
- The `__str__` method returns a string representation of a Grip object, in the following order: diameter, weight, material, separated by commas:

```
0.60,62,Rubber
0.58,75,Leather
```

(6 marks)

- (b) Implement `EquipmentRuleException`, a subclass of the `Exception` class. This class has no additional attribute or method. When the application encounters an equipment rule violation, an exception from this class is raised.
- (1 mark)
- (c) Construct and Implement the `Club` class.

The `Club` class has:

- Four instance variables:
 - `_label (str)`: labelling or numbering in uppercase, to help golfers identify the club. For example, "7-IRON", "3-WOOD" and "4-HYBRID".
 - `_head (ClubHead)`: representing the clubhead object for this club.
 - `_shaft (Shaft)`: representing the shaft for this club.
 - `_grip (Grip)`: the grip that is installed for this club (onto the top end of the shaft).
- Constructor initialises the 4 instance variables using the given parameters. The constructor must validate the following rules:
 - Weight of head must be more than the combined weight of the shaft and grip. Otherwise, raise `EquipmentRuleException` with appropriate message.
 - Assembled club length must be within 18 to 48 inches. Otherwise, raise `EquipmentRuleException` with appropriate message

- Getter method for instance variables label. Use the property decorator.
- Define getter methods for loft, flex, length, and weight. Use the property decorator
 - Loft of the club refers to the loft of the clubhead.
 - Flex of the club refers to the shaft's flex.
 - Length of the club is the length of the shaft and the height of the clubhead.
 - Weight of the club is the total weight of clubhead, shaft and grip.
- The changeGrip method replaces the instance variable _grip with the given parameter Grip object, provided that the weight of head must be more than the combined weight of the shaft and this new grip. Otherwise, raise EquipmentRuleException with appropriate message.
- getDetails method returns a string presentation of the club's label, loft, length, flex, and weight. Below are two examples in the suggested format:

```
Club: DRIVER      Loft: 10.5 Length: 45.75in Flex: R Weight: 335g
Club: 4-iron      Loft: 23.0 Length: 39.50in Flex: S Weight: 422g
```

- The __str__ method returns the label of the club, follow by the string representations of clubhead, shaft and grip. Below are two examples:

```
3-Hybrid,20.5,250,220,40.0,90,Graphite,R,0.6,62,Rubber
4-iron,23.0,260,Cast,38.5,100,Steel,S,0.6,62,Rubber
```

(12 marks)

(d) Write a main() function with exception handling for the following:

- (i) Create **FOUR (4)** golf clubs using the information below.

Label	ClubHead	Loft (degrees)	Weight (grams)	Others
Driver	Wood	10.5	203	Size: 450cc
8-iron	Iron	34.5	268	Material: Cast
Sunset	Putter	3	380	Style: Mallet
5-wood	Wood	17.5	150	Size: 280cc

Label	Shaft Flex	Shaft Length (inches)	Shaft Weight (grams)	Shaft Material
Driver	Stiff	45	68	Graphite
8-iron	Regular	35.5	109	Steel
Sunset	Stiff	33	120	Steel
5-wood	Stiff	42.5	89	Steel

- All grips are 0.6 inches in diameter, 62g in weight and made of "Rubber".
 - Print the total weight of the clubs created.
- (ii) Replace the grips for all created golf clubs as they are worn out. The new grips are made of "Leather", 75g in weight and 0.58 inches in diameter.

(6 marks)

Question 3 (35 marks)

Golfit is the leader in golf club components and accessories, i.e., Grips, Shafts, ClubHeads, Custom Clubs, Clubmaking Tools and Supplies.

In the game of golf, having the right Clubs and the right Fit can make a huge difference. Golfit employed many fitters, who are professionals performing golf club fitting for golfers. Fitters will measure and benchmark the golfer's current clubs, then provide new combinations of club heads and shafts to hit, eventually dialling in the perfect golf set (woods, irons, and putter) that will optimize the ball speed, spin rate, and launch angle for the golfer's game.

Golfit would like to develop an application to help record the specifications of the golf set, with objectives to ensure consistency in clubs, e.g., shafts' flex, length distribution, and also conforming to the equipment rules setup by R&A (Royal and Ancient Golf Club of St Andrews) and the USGA (United States Golf Association).

This question is a continuation of Question 1 & 2, hence will make use of classes in Question 1 & 2. Refer to the class diagram in Figure Q3.

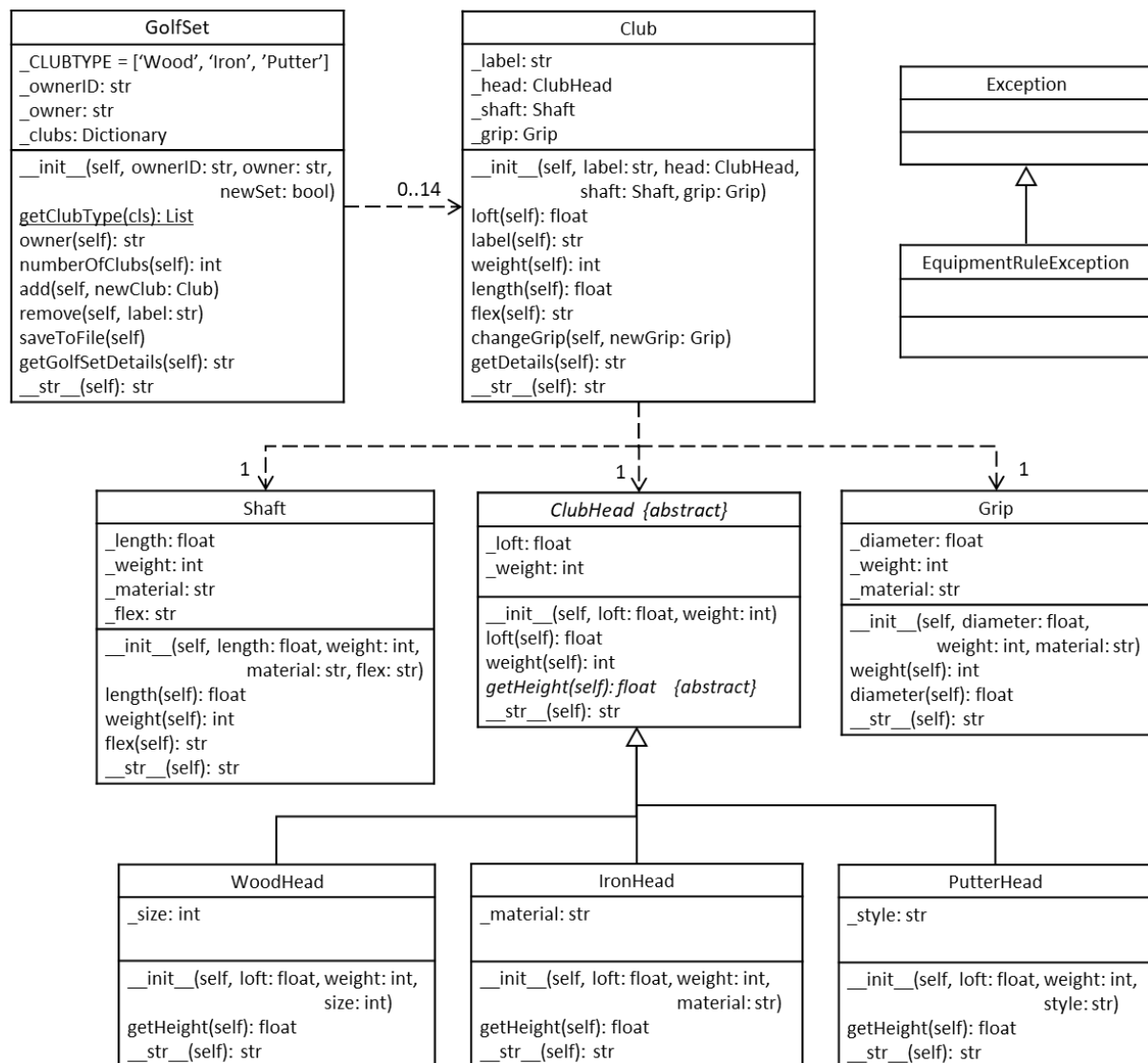


Figure Q3

(a) Implement the GolfSet class.

- It has a class variable: `_CLUBTYPE` that is a List containing the 3 categories of golf clubs, and a class method `getClubType` that returns the 3 categories in a List.
- There are 3 instance variables:
 - `_ownerID` (str): the golfer's ID. E.g., A17, B24, C20 etc.
 - `_owner` (str): the golfer's name.
 - `_clubs` (Dictionary): dictionary of lists, with each list containing clubs that made up this golf set.
- Constructor performs the following:
 - Initialises these 2 instance variables: `_ownerID` and `_owner`, using the given parameters.
 - Pre-set instance variable `_clubs` to `{"Wood": [], "Iron": [], "Putter": []}`
 - The keys for this dictionary are the club categories, and the values are lists of club objects.
 - If the parameter `newSet` is `True`, there is no need to load current golf set specification from file. Otherwise, construct the filename using "`ownerID-owner.txt`" (e.g., "`B24-Kobe.txt`") and populate `Club` objects into `_clubs` by reading lines from the file. Refer to Appendix A for file format.
 - Raise `EquipmentRuleException` with appropriate message if the file is not found.
 - Each line of the file contains information about the club, label, clubtype, clubhead, shaft and grip.
 - Based on clubtype, create the appropriate `Club` object, and add into matching clubtype category of `_clubs`.
- Getter methods for instance variable: `_owner`. Use the property decorator.
- Define getter methods for `numberOfClubs`. Use the property decorator
 - `numberOfClubs` is the sum of all clubs in the 3 categories of golf clubs.
- The `add` method has `clubType` (str) and a `newClub` (`Club` object) as parameters. The following equipment rules and quality checks need to be performed before adding this `newClub` into the golf set:
 - The maximum number of clubs in a golf set is 14.
 - Each club label should be unique, within the golf set.
 - The new `Club`'s shaft flex must match those in the same `clubType` category.
 - Within the same `clubType` category, there should not be clubs having the same loft.
 - Within the same `clubType` category, the `newClub` must not be longer than the next club with lower loft. E.g., 8-Iron (37.0 degree) must not be longer than 7-Iron (33.5 degree).
 - Within the same `clubType` category, the `newClub` must not be shorter than the next club with higher loft. E.g., 8-Iron (37.0 degree) must not be shorter than 9-Iron (40.5 degree).
 - If any of the above condition is not met, raise `EquipmentRuleException` with appropriate message.
- With `label` as parameter, the `remove` method will find the matching `Club` object and remove it from the collection `_clubs`.

- If there is no club matching the given label, then method should raise `EquipmentRuleException` with message stating there is no such club.
- Method `saveToFile` will construct a filename using “ownerID-owner.txt” (e.g., “B24-Kobe.txt”) and write the updated golf set specifications into the file. See Appendix A for file content and format.
- The `getGolfSetDetails` method returns a string presentation the golf set, including the count of clubs in this golf set. Here is an example of a golf set details:

```

Club: DRIVER      Loft: 10.5 Length: 45.38in Flex: R Weight: 333g
Club: 3-WOOD      Loft: 14.5 Length: 43.27in Flex: R Weight: 361g
Club: 5-WOOD      Loft: 17.5 Length: 42.20in Flex: R Weight: 387g
Club: 3-HYBRID    Loft: 20.5 Length: 40.55in Flex: R Weight: 402g
Club: 4-IRON      Loft: 23.0 Length: 39.50in Flex: R Weight: 422g
Club: 5-IRON      Loft: 26.5 Length: 39.00in Flex: R Weight: 426g
Club: 6-IRON      Loft: 30.0 Length: 38.50in Flex: R Weight: 430g
Club: 7-IRON      Loft: 33.5 Length: 38.00in Flex: R Weight: 434g
Club: 8-IRON      Loft: 37.0 Length: 37.50in Flex: R Weight: 438g
Club: 9-IRON      Loft: 40.5 Length: 37.00in Flex: R Weight: 442g
Club: PW          Loft: 45.0 Length: 36.50in Flex: R Weight: 444g
Club: SW          Loft: 54.0 Length: 36.25in Flex: R Weight: 446g
Club: LW          Loft: 60.0 Length: 36.00in Flex: R Weight: 448g
Club: ANSER       Loft: 3.0 Length: 34.50in Flex: S Weight: 562g
No of clubs: 14

```

- The `__str__` method returns a string representation of a `GolfSet` object, which consists of all clubs' label, follow by the string representations of clubhead, shaft and grip. Below is a sample:

```

DRIVER,Wood,10.5,203,450,44.25,68,Graphite,R,0.6,62,Rubber
3-WOOD,Wood,14.5,220,310,42.5,79,Graphite,R,0.6,62,Rubber
5-WOOD,Wood,17.5,240,280,41.5,85,Graphite,R,0.6,62,Rubber
3-HYBRID,Wood,20.5,250,220,40.0,90,Graphite,R,0.6,62,Rubber
4-IRON,Iron,23.0,260,Cast,38.5,100,Steel,R,0.6,62,Rubber
5-IRON,Iron,26.5,262,Cast,38.0,102,Steel,R,0.6,62,Rubber
6-IRON,Iron,30.0,264,Cast,37.5,104,Steel,R,0.6,62,Rubber
7-IRON,Iron,33.5,266,Cast,37.0,106,Steel,R,0.6,62,Rubber
8-IRON,Iron,37.0,268,Cast,36.5,108,Steel,R,0.6,62,Rubber
9-IRON,Iron,40.5,270,Cast,36.0,110,Steel,R,0.6,62,Rubber
PW,Iron,45.0,270,Cast,35.5,112,Steel,R,0.6,62,Rubber
SW,Iron,54.0,270,Cast,35.25,114,Steel,R,0.6,62,Rubber
LW,Iron,60.0,270,Cast,35.0,116,Steel,R,0.6,62,Rubber
Anser,Putter,3.0,380,Mallet,34.0,120,Steel,S,0.6,62,Rubber

```

- Bonus: within each club Type category, sort the list of club objects by their loft. This sorting can be done at any of these situations or whenever you believe is appropriate:
 - during loading of specification from file,
 - during or after adding a club,
 - within the `getGolfSetDetails()` method, or
 - within the `__str__()` method

(20 marks)

- (b) Write the application that allow Golfit fitters to manage the recording of club specifications after performing golf club fitting for golfers.

Upon starting the application, present the below main menu:

```
Golfit Main Menu
=====
1. Build a golf set
2. Load a golf set
0. Quit
Enter option:
```

In a nutshell, the application must allow:

- Building a golf set for new client
 - Obtain the golfer ID and name
 - Add or remove club to/from this golf set
 - Presenting the specifications of the new golf set
 - Save the specifications to file
 - Loading of golf set for existing client
 - Obtain the golfer ID and name
 - Read from file, and present the current specifications of the golf set
 - Add or remove club to/from the current golf set
 - Save the new specifications back to file
- (i) Building a golf set – to build a new golf set for a golfer, the fitter select option 1 in Golfit main menu.

Before presenting the sub menu, your application needs to do the following:

- Prompt fitter to enter golfer's ID and name.
- Create the GolfSet object, using golfers' ID, name and newSet as True.
- Display the golfers' ID, name, and then present the current golf set details, which should be empty at the start.
- In the sub-menu, the fitter can add or remove club into this golf set.
- When the fitter has completed building the golf set, the fitter can select choice 0 in the sub menu, to go back to the main menu. The application should save the golf set specification for the golfer before presenting the main menu.

```
Enter golfer's ID: B24
Enter golfer's name: Kobe
```

```
***** ← one-time message to fitter
Recommendation:
- Build the set from longest to shortest club
- i.e., from Driver to Putter
*****
```

```
B24 Kobe ← golfer's ID and name
No of clubs: 0 ← golf set specifications
```

```
Club Fitting for Kobe ← sub menu
=====
1. Add a club
2. Remove a club
```

0. Back to Main Menu
Enter choice:

Sub menu title should include the golfer's name, highlighted in Green.

- (ii) Loading a golf set – to load the golf set specifications for an existing golfer, the fitter select option 2 in Golfit main menu.

Before presenting the sub menu, your application needs to do the following:

- Prompt fitter to enter golfer's ID and name.
- Create the GolfSet object, using golfers' ID, name and newSet as False.
- Display the golfers' ID, name, and then present the current golf set details.
- The fitter can use the "Add a club" or "Remove a club" to alter the golf set for the golfer.
- When the fitter has completed the fitting process, the fitter can select choice 0 in the sub menu, to go back to the main menu. The application should save the updated golf set specification for the golfer before presenting the main menu.

Enter golfer's ID: **B17**
Enter golfer's name: **Robin**

```

B17 Robin                                     ← golfer's ID and name
Club: DRIVER      Loft: 10.5  Length: 45.38in  Flex: R  Weight: 333g
Club: 3-WOOD       Loft: 14.5  Length: 43.27in  Flex: R  Weight: 361g
Club: 5-WOOD       Loft: 17.5  Length: 42.20in  Flex: R  Weight: 387g
Club: 4-IRON       Loft: 23.0  Length: 39.50in  Flex: R  Weight: 422g
Club: 5-IRON       Loft: 26.5  Length: 39.00in  Flex: R  Weight: 426g
Club: 6-IRON       Loft: 30.0  Length: 38.50in  Flex: R  Weight: 430g
Club: 7-IRON       Loft: 33.5  Length: 38.00in  Flex: R  Weight: 434g
Club: 8-IRON       Loft: 37.0  Length: 37.50in  Flex: R  Weight: 438g
Club: 9-IRON       Loft: 40.5  Length: 37.00in  Flex: R  Weight: 442g
Club: PW           Loft: 45.0  Length: 36.50in  Flex: R  Weight: 444g
Club: SW           Loft: 54.0  Length: 36.25in  Flex: R  Weight: 446g
Club: LW           Loft: 60.0  Length: 36.00in  Flex: R  Weight: 448g
Club: ANSER        Loft:  3.0  Length: 34.50in  Flex: S  Weight: 562g
No of clubs: 13                                ← golf set specifications

Club Fitting for Robin                         ← sub menu
=====
1. Add a club
2. Remove a club
0. Back to Main Menu
Enter choice:

```

Sub menu title should include the golfer's name, highlighted in Green.

- (iii) Add a club – this is choice 1 of the sub menu, where fitter will enter the specification of the new club to be added into the golf set.

- To add a club, fitter must first provide the clubtype, label, clubhead, shaft and grip information.
- You program should create the appropriate ClubHead, Shaft, Grip objects, and then the Club object that represents the new club.

- Invoke the appropriate method of the GolfSet class to add this new club into the golf set.
- Display any error or confirmation messages for this add request, then return to the sub menu.
- Below showcases an example of successful adding:

```
B17 Robin
Club: DRIVER      Loft: 10.5  Length: 45.38in  Flex: R  Weight: 333g
Club: 3-WOOD      Loft: 14.5  Length: 43.27in  Flex: R  Weight: 361g
Club: 5-WOOD      Loft: 17.5  Length: 42.20in  Flex: R  Weight: 387g
No of clubs: 3
```

```
Club Fitting for Robin
=====
1. Add a club
2. Remove a club
0. Back to Main Menu
Enter choice: 1
Which club type to add: wood
Enter the new club label: 3-hybrid
Enter clubhead loft: 20.5
Enter clubhead weight: 250
Enter wood size: 22
Enter length of shaft: 40
Enter weight of shaft: 90
Enter shaft material: graphite
Enter shaft flex: r
Enter diameter of grip: 0.6
Enter weight of grip: 62
Enter grip material: rubber
New Club added
```

```
B17 Robin
Club: DRIVER      Loft: 10.5  Length: 45.38in  Flex: R  Weight: 333g
Club: 3-WOOD      Loft: 14.5  Length: 43.27in  Flex: R  Weight: 361g
Club: 5-WOOD      Loft: 17.5  Length: 42.20in  Flex: R  Weight: 387g
Club: 3-HYBRID    Loft: 20.5  Length: 40.05in  Flex: R  Weight: 402g
No of clubs: 4
```

```
Club Fitting for Robin
=====
1. Add a club
2. Remove a club
0. Back to Main Menu
Enter choice:
```

- Present screenshots of successful and unsuccessful adding.
- (iv) Remove a club – this is choice 2 of the sub menu, where fitter determined that a club is no longer required and want to remove it from the golf set.
- To remove a club, fitter just need to provide the club's label.
 - Your program should invoke the appropriate method of the GolfSet class to remove this club from the golf set.
 - Display any error or confirmation messages for this removal request, then return to the sub menu,
 - Below showcases example of unsuccessful and successful removal:

```

B17 Robin
Club: DRIVER      Loft: 10.5  Length: 45.38in  Flex: R  Weight: 333g
Club: 3-WOOD      Loft: 14.5  Length: 43.27in  Flex: R  Weight: 361g
Club: 5-WOOD      Loft: 17.5  Length: 42.20in  Flex: R  Weight: 387g
Club: 3-HYBRID    Loft: 20.5  Length: 40.05in  Flex: R  Weight: 402g
No of clubs: 4

```

```

Club Fitting for Robin
=====
1. Add a club
2. Remove a club
0. Back to Main Menu
Enter choice: 2
Enter the club label to remove: 3-HYBIRD
Cannot remove as 3-HYBIRD is not in set

```

```

B17 Robin
Club: DRIVER      Loft: 10.5  Length: 45.38in  Flex: R  Weight: 333g
Club: 3-WOOD      Loft: 14.5  Length: 43.27in  Flex: R  Weight: 361g
Club: 5-WOOD      Loft: 17.5  Length: 42.20in  Flex: R  Weight: 387g
Club: 3-HYBRID    Loft: 20.5  Length: 40.05in  Flex: R  Weight: 402g
Club: ANSER       Loft: 3.0   Length: 34.50in  Flex: S  Weight: 562g
No of clubs: 4

```

```

Club Fitting for Robin
=====
1. Add a club
2. Remove a club
0. Back to Main Menu
Enter choice: 2
Enter the club label to remove: 3-Hybrid
Removal done...

```

```

B17 Robin
Club: DRIVER      Loft: 10.5  Length: 45.38in  Flex: R  Weight: 333g
Club: 3-WOOD      Loft: 14.5  Length: 43.27in  Flex: R  Weight: 361g
Club: 5-WOOD      Loft: 17.5  Length: 42.20in  Flex: R  Weight: 387g
No of clubs: 3

```

```

Club Fitting for Robin
=====
1. Add a club
2. Remove a club
0. Back to Main Menu
Enter choice:

```

- Present screenshots of successful and unsuccessful removal.

(v) Errors and exceptions handling

Your application must handle all exceptions including input error. Type/Value error should be handled by allowing the user to re-enter. For all menu options, handle any raised EquipmentRuleException or Exception by printing the error messages.

Some examples of range/value validation:

```
Golfit Main Menu
=====
1. Build a golf set
2. Load a golf set
0. Quit
Enter option: 3
Sorry, please re-enter within range (0-2)
Enter option:

...
...
Club Fitting for Kobe
=====
1. Add a club
2. Remove a club
0. Back to Main Menu
Enter choice: 1
Which club type to add: Tail
Sorry, please select from ['Wood', 'Iron', 'Putter']
Which club type to add:
```

Some examples of exception handlings below:

```
Club Fitting for Robin
=====
1. Add a club
2. Remove a club
0. Back to Main Menu
Enter choice: 1
Which club type to add: wood
Enter the new club label: DRIVER
Enter clubhead loft: 10
Enter clubhead weight: 203
Enter wood size: 450
Enter length of shaft: 44
Enter weight of shaft: 65
Enter shaft material: Graphite
Enter shaft flex: R
Enter diameter of grip: 0.62
Enter weight of grip: 65
Enter grip material: rubber
Should not have same club label DRIVER within the set

...
...
Club Fitting for Robin
=====
1. Add a club
2. Remove a club
0. Back to Main Menu
Enter choice: 1
Which club type to add: Iron
Enter the new club label: 7.5-Iron
Enter clubhead loft: 35
Enter clubhead weight: 268
Enter iron material: Cast
Enter length of shaft: 37.5
```



```

Enter weight of shaft: 106
Enter shaft material: steel
Enter shaft flex: r
Enter diameter of grip: 0.6
Enter weight of grip: 62
Enter grip material: rubber
New club's length must not be shorter than next higher loft

...
...
Club Fitting for Robin
=====
1. Add a club
2. Remove a club
0. Back to Main Menu
Enter choice: 1
Which club type to add: wood
Enter the new club label: 3-Hybrid
Enter clubhead loft: 20
Enter clubhead weight: 250
Enter wood size: 220
Enter length of shaft: 39
Enter weight of shaft: 90
Enter shaft material: Graphite
Enter shaft flex: S
Enter diameter of grip: 0.6
Enter weight of grip: 62
Enter grip material: rubber
Not advisable to have different shaft flex within the Wood

```

(15 marks)

Question 4 (20 marks)

A Golf Distance Calculator (GDC) GUI allowed golfers to check how far they can hit the golf ball when they provide their swing speed, club loft and club length.

- (a) Develop and implement the GDC GUI application as shown in Figure 4.

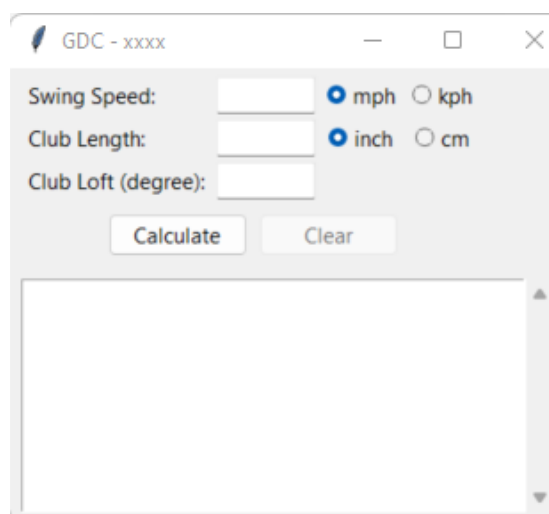


Figure 4

- Title of the GUI **MUST** include your name (replace XXXX with your name).
- The layout is recommended, but you can define any suitable design.
- These are the proposed widgets:
 - Label “Swing Speed:” to guide user input (Entry) on the swing speed.
 - Radiobuttons to allow user to select the metric for the swing speed.
 - Label “Club Length:” to guide user input (Entry) on the club length.
 - Radiobuttons to allow user to select the metric for the length.
 - Label “Club Loft (degree):” to guide user input (Entry) on the club loft.
 - 2 Buttons “Calculate” (enabled) and “Clear” (disabled).
 - Scrolled Text (disabled) to display the distance calculated or any messages.

(9 marks)

- (b) Implement event handling so that the GDC GUI can respond to left button clicks on both buttons.

To calculate distance, the formula is:

$$\text{Distance} = (280 - \text{abs}(\text{length}-48)*10 - \text{abs}(\text{loft}-10)*1.25) * \text{SwingSpeed}/96$$

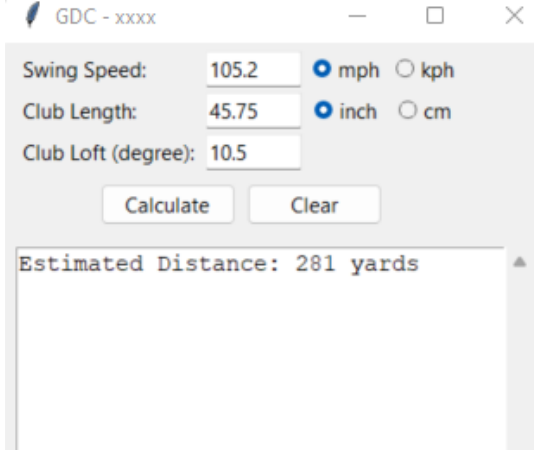
where length and SwingSpeed are in inches and mph respectively. For metric conversion,

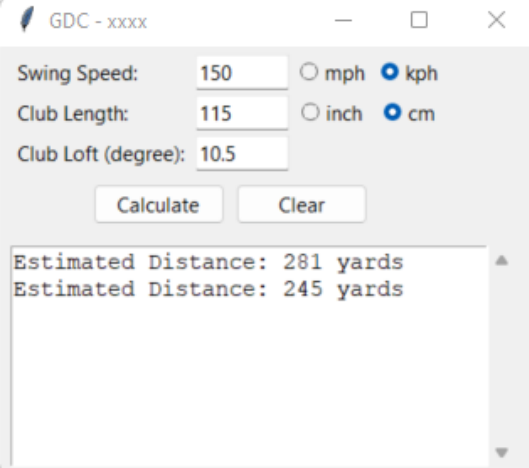
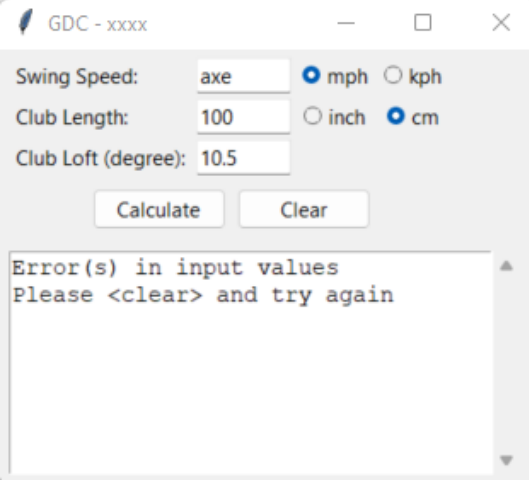
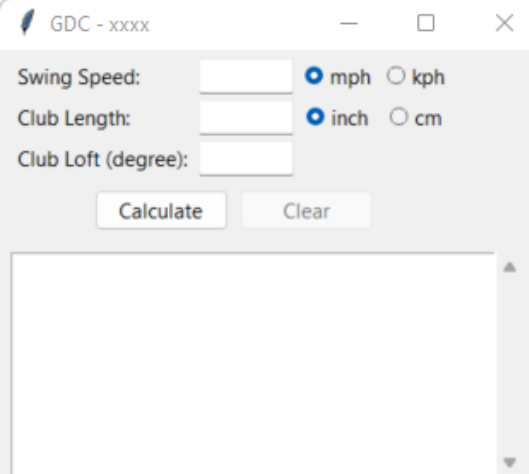
- use 1 kph to 0.621371 mph, and
- use 1 cm to 0.393701 inch

In addition, your GUI program should also perform the following input checks:

- Ensure swing speed is a positive number.
- Ensure length must be within 30 to 48 inches.
- Ensure loft must be within 8 to 64 degrees.

Below are demonstrations of the GDC GUI application:

	<p>When user enters swing speed, loft, length and click the “Calculate” button, your program will compute the distance in yards based on the formula give.</p> <p>Round/format the computed distance to nearest whole number before displaying.</p> <p>“Clear” button is also enabled whenever the ScrolledText is populated with any text.</p>
---	---

	
	<p>Your program should display error messages in the scrolledText when either of the swing speed, club loft and club length data entry contain:</p> <ul style="list-style-type: none"> - empty or no data value - non-numeric values - negative numbers - values not within the range shown in the input checks above
	<p>When “Clear” button is clicked, your program should:</p> <ul style="list-style-type: none"> - clear the 3 text fields - clear the ScrolledText - default selection of the radiobuttons to “mph” and “inch” - disable the “Clear” button

Apply Object oriented principles when writing the GUI program. The program must be written as a class, and you can submit your codes for Q4(a) together with Q4(b).

Submit at least 4 set of screenshots: displaying distance calculated for 3 different set of inputs, and one showing error data entry for swing speed.

(11 marks)

Appendix A

The following 2 files are samples to be used for creating GolfSet objects.

File 1: "B17-Robin.txt"

```
DRIVER,Wood,10.5,203,450,44.25,68,Graphite,R,0.6,62,Rubber
3-WOOD,Wood,14.5,220,310,42.5,79,Graphite,R,0.6,62,Rubber
5-WOOD,Wood,17.5,240,280,41.5,85,Graphite,R,0.6,62,Rubber
4-IRON,Iron,23.0,260,Cast,38.5,100,Steel,R,0.6,62,Rubber
5-IRON,Iron,26.5,262,Cast,38.0,102,Steel,R,0.6,62,Rubber
6-IRON,Iron,30.0,264,Cast,37.5,104,Steel,R,0.6,62,Rubber
7-IRON,Iron,33.5,266,Cast,37.0,106,Steel,R,0.6,62,Rubber
8-IRON,Iron,37.0,268,Cast,36.5,108,Steel,R,0.6,62,Rubber
9-IRON,Iron,40.5,270,Cast,36.0,110,Steel,R,0.6,62,Rubber
PW,Iron,45.0,270,Cast,35.5,112,Steel,R,0.6,62,Rubber
SW,Iron,54.0,270,Cast,35.25,114,Steel,R,0.6,62,Rubber
LW,Iron,60.0,270,Cast,35.0,116,Steel,R,0.6,62,Rubber
Anser,Putter,3.0,380,Mallet,34.0,120,Steel,S,0.6,62,Rubber
```

File 2: "A20-Marvin.txt"

```
DRIVER,Wood,10.5,203,450,44.25,68,Graphite,R,0.6,62,Rubber
5-WOOD,Wood,17.5,240,280,41.5,85,Graphite,R,0.6,62,Rubber
5-IRON,Iron,26.5,262,Cast,38.0,102,Steel,S,0.6,62,Rubber
6-IRON,Iron,30.0,264,Cast,37.5,104,Steel,S,0.6,62,Rubber
7-IRON,Iron,33.5,266,Cast,37.0,106,Steel,S,0.6,62,Rubber
8-IRON,Iron,37.0,268,Cast,36.5,108,Steel,S,0.6,62,Rubber
9-IRON,Iron,40.5,270,Cast,36.0,110,Steel,S,0.6,62,Rubber
PW,Iron,45.0,270,Cast,35.75,112,Steel,S,0.6,62,Rubber
GW,Iron,50.0,270,Cast,35.5,113,Steel,S,0.6,62,Rubber
SW,Iron,54.0,270,Cast,35.25,114,Steel,S,0.6,62,Rubber
LW,Iron,58.0,270,Cast,35.0,115,Steel,S,0.6,62,Rubber
ODYSSEY#7,Putter,3.0,365,Mallet,34.0,120,Steel,S,0.6,62,Rubber
```

Format of the file:

Each line of the file contains information in this order:

```
<label>,<clubtype>,<loft>,<weight>,<size/material/style>,<length>,<weight>,<material>,<flex>,<diameter>,<weight>,<material>
```

```
ClubHead information, Shaft information, Grip information
```

Note: <size/material/style> depends on <clubtype>

Line 1 of the file represent club 1, line 2 represent club 2 and so on.

---- END OF ASSIGNMENT ----