

**BINUS UNIVERSITY**



**BINUS INTERNATIONAL**



**Assignment Cover Letter**

**(Individual Work** **)**

**Student Information :**

**Surname**  **Given Names Student ID Number**

Avatara  Ida Bagus Ryogassa (Ryo) 2440100323

**Course Name :** Program Design Methods

**Name of Lecturer(s) :** Jude Joseph Lamug Martinez

**Course Code**  **:** COMP6699

**Class**  **:** L1AC

**Major**   **:** Computer Science

**Title of Assignment** **:** Basketball Manager

**Type of Assignment** **:** Final Project

**Submission Pattern :**

**Due Date**  **:** 22nd June 2021   **Submission Date**  **:**  21st June 2021

The assignment should meet the below requirements.

1. Assignment (hard copy) is required to be submitted on clean paper, and (soft copy) as per lecturer’s instructions.
2. Soft copy assignment also requires the signed (hardcopy) submission of this form, which automatically validates the softcopy submission.
3. The above information is complete and legible.
4. Compiled pages are firmly stapled.
5. Assignment has been copied (soft copy and hard copy) for each student ahead of the submission.

**Plagiarism/Cheating**

BiNus International seriously regards all forms of plagiarism, cheating and collusion as academic offenses which may result in severe penalties, including loss/drop of marks, course/class discontinuity and other possible penalties executed by the university. Please refer to the related course syllabus for further information.

**Declaration of Originality**

By signing this assignment, I understand, accept and consent to BiNus International terms and policy on plagiarism. Herewith I declare that the work contained in this assignment is my own work and has not been submitted for the use of assessment in another course or class, except where this has been notified and accepted in advance.

Signature of Student:

Ida Bagus Ryogassa Avatara

‘

**TABLE OF CONTENTS**

* **PROJECT SPECIFICATION……………………………………………………………………………………………3**
* **SOLUTION DESIGN…………………………………………………………………………………………………….3**
* **EVIDENCE OF WORKING PROGRAM AND DISCUSSION……………………………………4 to last**
* **IMPROVEMENTS?....................................................................................................11**
* **REFERENCES…………………………………………………………………………………………………………..11**

**PROJECT SPECIFICATION**

* **Project Purpose:**

The purpose of this project is to make an simple yet complex project to solve. This project is a management system for a basketball league, where it allows to store played basketball matches depending on the date of the match (text-based).

* **Project Audience:**

People who are interested in basketball, or sport in general, can explore how to manage a basketball league. There are also several apps and games on sports managers.

* **Project Aim:**

The aim to create this project is to create a management system that people can explore and maybe have fun with.

* **Project Requirements:**
  + No Bugs
  + OOP Based
  + Using all methods that have been taught in class

**SOLUTION DESIGN**

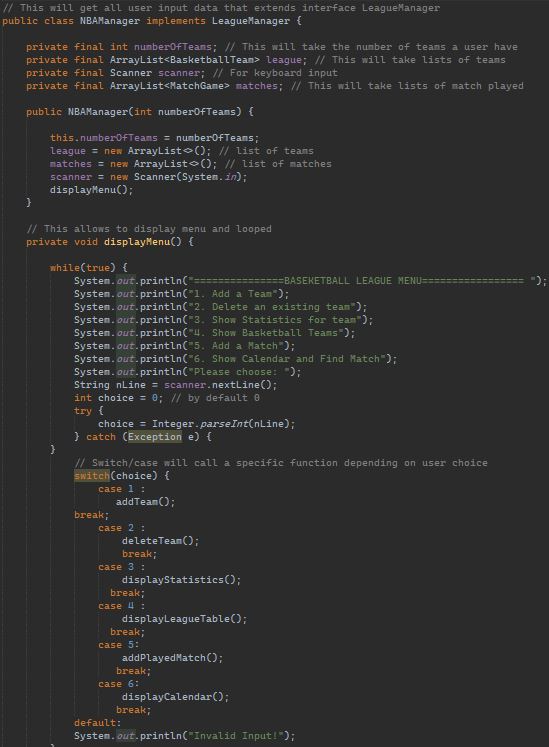
**Util Imports used:**

* ParseException
* SimpleDateFormat
* ArrayList
* Calendar
* Collectuibs
* Date
* Scanner
* Comparator

**EVIDENCE OF WORK:**

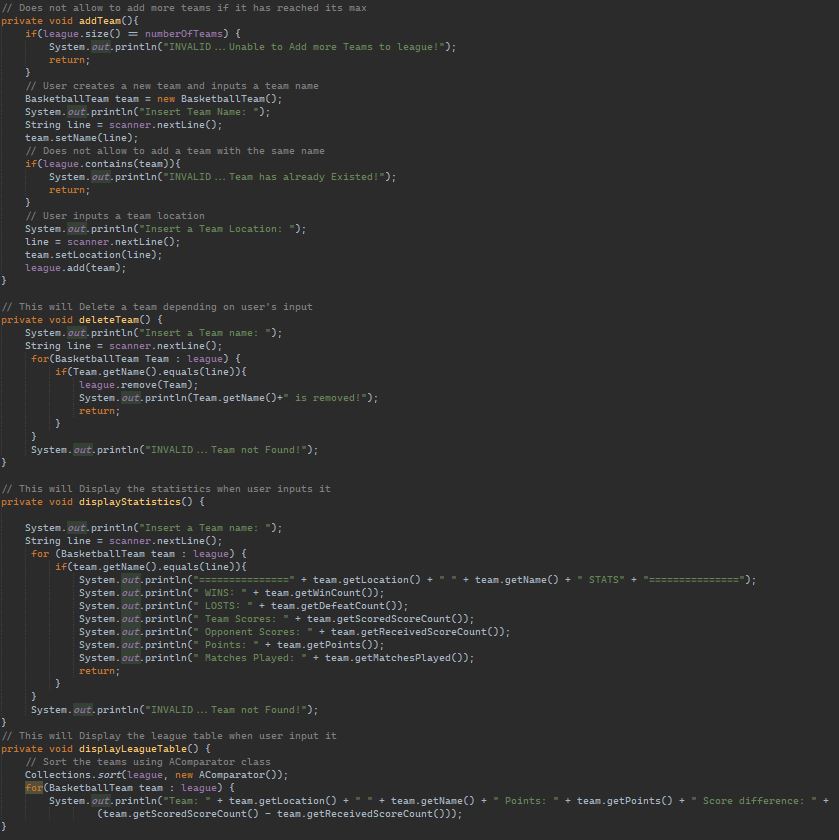
**NBAManager.java:**

This is the NBAManager.java that implements interface LeagueManager, where it houses the menu that calls different kinds of functionalities depending on user input.

****It also holds the array lists that holds information on the teams and matches based on the input of the user.

The menu can be seen in the picture which will use switch/case to call a function. The user have 6 different functions for them to play around.

Each functions provide different kinds of information. For example, addteam() allows a user to add and name a team freely.

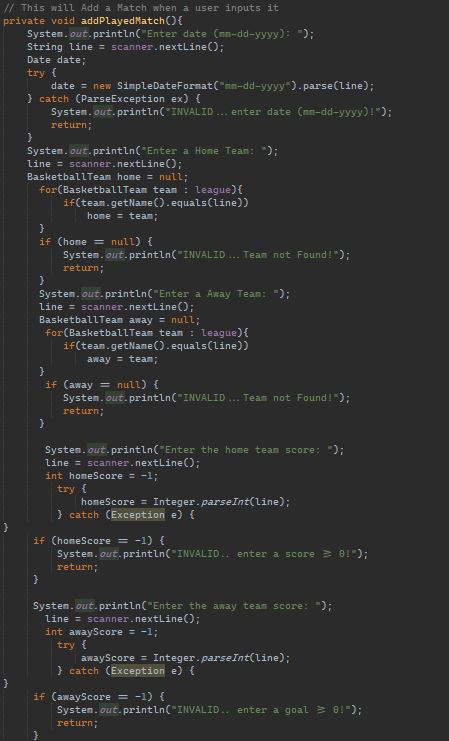
****

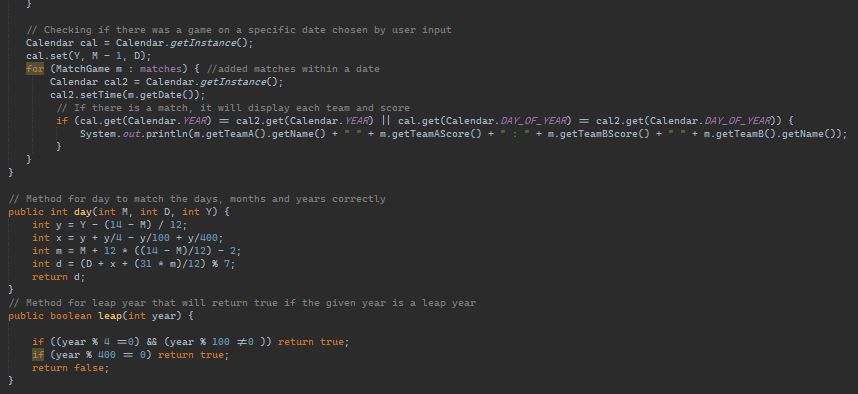
The addTeam() function allows a user to add a team and they can freely name their team and location. A user can add more than one team.

The deleteTeam() function allows a user to delete an existing team. The user will have to input the name of the team for deletion.

displayStatistics() function allows a user to see the stats of a team of their choice. The stats will change when a user set the scores for a match of two teams, if not then the stats will default at 0.

displayLeagueTable() function allows a user to see team points and score difference.

addPlayedMatch() function allows a user to add a match between two existing teams on a specific date. The user will have input 2 of the team and input a score of each team. Then the user can see the match when they choose the displayCalender() function. The displayCalendar() function allows a user to input a specific date to see if there are any added matches based on the selected date. User will have to input the year, month and day for it to work. The function also houses methods that an array of months and dates which can be accessed with user input. The function also makes a proper classic calendar that will be shown later. Furthermore, a method to check if it is a leap your is also added within the mix so it can be a proper calendar for users to experience.



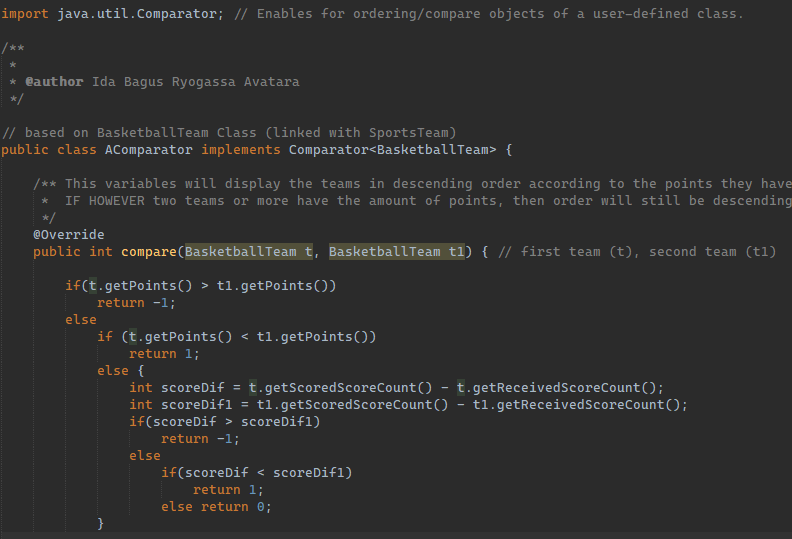
 ^^ This method checks if there is a match played on a specific date. If there is a match found then it will show the 2 teams and each of their scores.

Below that is another method for day to match the days, months and years correctly for it to be a proper calendar. It also has a method for leap year to return true if the given year is an actual leap year.

**<< BasketballTeam.java**

This houses a public class basketballteam that inherit an abstract class called SportsTeam.

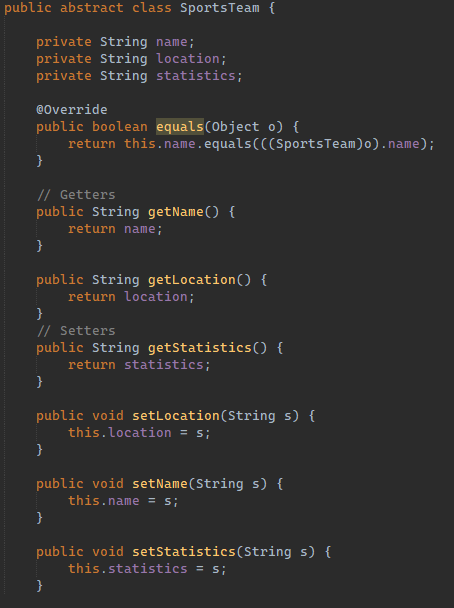
BasketballTeam class has a bunch of getters and setters that will receive different aspects like number of wins, loss, points, matches, etc depending on user input.

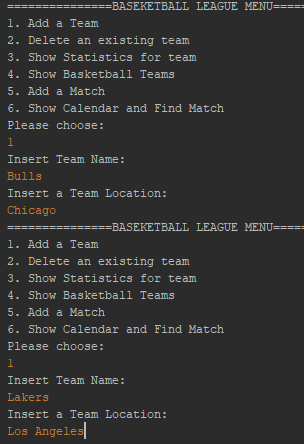
**AComparator.java**

This variables will display the teams in descending order according to the points they have scored (team with the most maximum points will appear first)

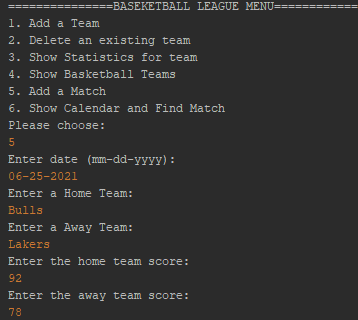
IF HOWEVER two teams or more have the amount of points, then order will still be descending based on SCORE DIFFERENCE (team with the best score difference will appear first)

**SportsTeam.java**

SportsTeam abstract class houses a getters and setters for the traits of a team. It will receive the name, location as well as the stats of a team.

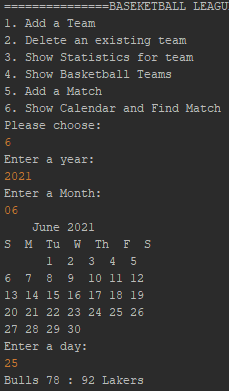
**Output:**

<< First, a user can input a team and they can freely name to their liking. In order for other functionalities to work, we, at least, must have two team added to the list. In this case, Chicago bulls and LA Lakers are added to the league list of teams.



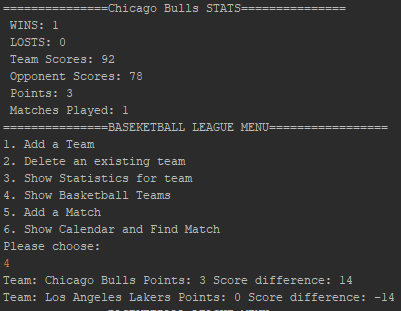
When two teams are created, a user can add a match to a specific date and the scores of each team.

In this example, a game was added on 06-25-2021 and having the bulls win against the lakers. >>



<< Then, a user can find the game that was added on 06-25-2021 with the 6th function. The user will have to input the year, month and day of their choice.

If there are no games, a reminder will show that are no games on that specific date.



Lastly, then a user can see the stats of the team.

**What can be improved?**

Honestly, using a GUI with either javafx or swing along side with mysql would be ideal because it looks better and more interactive for the user. Even with or without mysql, implementing fx and swing separately all I get are code errors and weird bugs that I cannot seem to find the solution for.

Therefore, I just had to go with plan b and use the console instead.

**References:**

<https://www.youtube.com/watch?v=xnc3PAA5TSU> (Implementing Calendar)