**Final Assignment Report**

Programming Language 1

Standings Table

Mahde Al Jamil

Student Number: 2203822

Student ID: 1.2.246.562.24.53532288716

# Overview:

The Standing Table program is made to fulfill the requirement of keeping track of sports scores. The source code is written in Java language, and it's designed to work really well. This report talks about the details of the program, showing how it meets certain technical requirements.

# Technical Specifications:

## Code Structure:

The program's architecture is thoughtfully designed, featuring a main method for the user interface and additional methods for other specific functionalities. This modular approach enhances code organization, readability, and maintainability.

## Naming Conventions:

The names assigned to variables and methods are chosen to be clear and meaningful, aligning with established Java conventions. This enhances the program's readability, making it accessible to developers and users alike.

## Array Utilization:

## One big part of the program is how it uses arrays. There's a special grid called "pointTable" that holds all the info about each team. Each row in this grid has details like how many games a team played, how many goals they scored, and how many they let the other team score. This setup helps store and find important information more easily.

## Comments:

The code has lots of comments that act like a guide, explaining how things work. These comments show us what each part of the code does, also help us understand why it's written that way. Which makes the program easier to understand and helps if someone wants to change it later.

## Global Constants:

The program avoids using global variables as instructed but uses global constants. These constants like TEAM\_NAME\_INDEX and MATCHES\_PLAYED\_INDEX help organize and work with team data in a reliable way. This method keeps the code consistent and makes it easier to understand.

## Error Handling:

The program is capable of dealing with mistakes. If someone puts in the wrong information, the program doesn't stop working completely. Instead, it talks to the user and explains what went wrong, helping them fix it. This makes the program more dependable for users.

# Functionalities:

* Easy-to-Use Interface: The program has a menu that's easy to understand and helps users move around without any trouble. It gives clear instructions to make things simple.
* Team Management: Teams are handled automatically based on match results. New teams are added, or existing ones are updated, keeping the standings accurate and current.
* Score Calculation: The program uses a smart method to figure out team scores, considering wins, draws, and losses. This shows a good understanding of how sports leagues work.
* Standings Organization: Teams in the standings table are sorted dynamically by points and goal differences. This helps users quickly see how teams rank and makes it easy to analyze.
* Helpful Error Handling: When there's a mistake, the program doesn't just show an error—it helps users fix it by giving specific advice. This makes it easier to use and reduces mistakes.
* Quick Updates: The program swiftly updates teams so that changes in match results are shown right away in the standings. It responds in real-time, which is really efficient.
* Handling Lots of Data: By using arrays to store information, the program can manage a lot of teams and matches without slowing down. It's good at handling a growing amount of data.

# Conclusion:

The Standing Table program shows how well it follows specific technical requirements. I think it is well-organized, uses clear names, manages data using a 2-dimensional array, has lots of comments, and deals with user mistakes in a efficient way. Together, these things make it work well and dependable.