CPSC 304 Project Cover Page

Milestone #:4				
Date:Dec 1, 2023				
Croup Number 21				
Group Number:21				

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Winnie Wu	26039453	b7i8b	Wwu947@student.ubc.ca
Nicole Lu	71679872	v7o6b	Nicolelu2011cat@gmail.com
Ethan Zhao	67073387	g3iok	Ethanzhao9876@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Project description

A short description of the final project, and what it accomplished.

Our project is a user-friendly GUI application designed to manage an Oracle database of sports tournaments. It is developed in Java and SQL scripts. There are scripts to set up database tables and fill it with entries so the database can be stored persistently. The database keeps track of tournaments, tournament matches, teams, team members (players, coaches, business members), sponsors etc. Users can interact with and view tables in the database, for example, update and view team and player information. To support our GUI, we developed functions to handle events that would respond to user inputs, and trigger warnings when prohibited actions happen (i.e., add Player with same ID as an existing Player).

To ensure that our tests are meaningful, we created and populated our tables with real-world data, including constraints required to model the ER diagram. We also allow the users to query the database with a few button clicks to see which entries fall under certain conditions. They do not have to write any code or the like. It is completely user-friendly in that aspect. Anyone without a computer science background should be able to use it easily.

A description of how your final schema differed from the schema you turned in. If the final schema differed, explain why.

Nothing changed in our final schema. It is the exact same as the original one we turned in.

Locations where each query can be found in the code:

ALL backend query logic can be found in this file:

https://github.students.cs.ubc.ca/CPSC304-2023W-T1/project_b7i8b_g3i0k_v7o6b/blob/main/CPSC304_Java_Project-master/src/ca/ubc/cs304/database/TournieDBHandler.java

Insert: lines 42-63 Update: lines 65-171 Delete: lines 173-192

Selection: IS NOT IMPLEMENTED

Projection: lines 194-270 (frontend uses `getTables` to get tables then sends back which table it wants, then asks for the columns of that table using `getTableColumns`, then sends back which columns it wants to project and projects

using 'project')
Join: lines 272-306

Aggregation Group By: lines 308-341

Aggregation Group By Having: lines 343-380 Aggregation Group By Nested: lines 382-409

Division: lines 422-465

Screenshots of query results

Screenshots of the sample output of the queries using the GUI (for example, for an insertion query you can show what data is in your table before you run the query, and then show another screenshot after running the query, from some kind of GUI input like a button).

- You need only to include screenshots for the required queries if you implemented more than what was required, screenshots are not needed for those extra queries.
- Each screenshot should be clearly labelled with which query (i.e., rubric item) it represents.















