CP129 Final Project
Score: _____/100 v1.0

The fitness tracker application, named "FitTrack", was developed by a small startup called "FitWorks". FitWorks was founded by a group of fitness enthusiasts who wanted to create a platform that would help people achieve their fitness goals and connect with like-minded individuals. As the user base grew, FitWorks realized they needed to expand their development team to keep up with demand. FitWorks hired you as a contract worker to design and create a new database for the FitTrack application. Your task is to create a scalable and efficient database that will help FitTrack users achieve their fitness goals and connect with others in the fitness community.

Specifications

You will design and create a database for “FitTrack”. You will begin by creating dependency diagrams to identify the relationships between different data elements and to ensure that the database is properly normalized. From there, you will use the diagrams to create an entity-relationship diagram (ERD) that accurately represents the structure of the database. The ERD should include no fewer than 12 entities, such as users, workouts, exercises, goals, achievements, and social connections. Once the ERD is complete, you will use it to create the physical database schema and implement the necessary SQL statements to create the tables, relationships, and constraints in the database.

You will need to come up with your own business rules to create entities and relationships. Here are some example business rules that you can use:

- Each user can have multiple workout plans, and each workout plan can contain multiple exercises.
- Each exercise can be associated with one or more muscle groups, and each muscle group can be associated with multiple exercises.
- Each workout can be assigned a difficulty level, which can range from beginner to advanced.
- Each user can set and track multiple fitness goals, such as weight loss, muscle gain, or endurance improvement.
- Each goal can be associated with one or more metrics, such as weight, body fat percentage, or distance.
- A workout can include multiple exercises and use specific equipment.

When converting your ERD entities to tables, insert no fewer than 5 rows of data in each table. Here is a website you can use to generate data for your tables: https://www.mockaroo.com/

Once you have completed the creation of your fitness tracker database, you should also create at least five views that display key information from the database. Views are useful because they allow users to see information from multiple tables in a single place and can help simplify complex data relationships.

One view idea could be a "Top Exercises" view that displays the most popular exercises performed by users across the platform. This view could show the top 10 exercises ranked by the total number of sets or reps completed across all users in the database. Other potential view ideas could include a leaderboard of top performers in a given fitness challenge or a breakdown of workout data by muscle group.

Constraints & Clarifications

- All entities must be in 3NF.
- Entities and attributes should be named using the same naming scheme we have been using in class (all uppercase and snake case).
- ERDs should show weak/strong relationships, PKs and FKs, optionality, named relationships, and be well formatted (your ERD should not look like I dropped spaghetti on the floor).
- No M:N relationships can exist.
- No multi-valued attributes can exist.
- Ensure foreign keys are established in your database.

Grading

The grading rubric can be found on Canvas in the Final Project assignment.

Submission

There are three parts to your submission: 1) your dependency diagrams, 2) your ERD, and 3) your database.

Your dependency diagrams and ERD must be submitted digitally. I recommend using LibreOffice Draw or Microsoft Visio to create these. Alternatively, you can draw them on paper and scan them. However, if you choose this option, your writing must be legible. Illegible writing cannot be graded.

If you use LibreOffice Draw, ensure the autosave feature is on (and set the auto save time to 5 minutes or less). If you do not turn it on and LibreOffice crashes, you could end up losing hours of work. Here is a guide to ensure it is enabled: https://ask.libreoffice.org/t/what-are-the-autosave-options-and-where-is-work-saved/35203/5

When submitting your database, you will export it as a self-contained file and upload it. It should be a .sql file. Here is a guide to export your database: https://dev.mysql.com/doc/workbench/en/wb-admin-export-import-management.html

Tips

- Before starting, come up with your business rules first. This will help you identify entities and relationships. After you have your business rules, translate them into a dependency diagram. Ensure you have no multi-valued attributes and your tables are in 3NF. From there, translate your dependency diagram(s) into an ERD and then translate your ERD into a database.
- Do not wait to start this project. It will take at least 5 hours coming up with good business rules. You will lose points if your database does not make sense.
- Remember, creating databases is an iterative process. Your first ERD will likely not be your final ERD.