A blue and white logo

AI-generated content may be incorrect.

**PGDM- Big Data Analytics**

**Big Data Management & Analytics**

**Project Report**

**ERD- Fitness\_tracking\_app**

**Submitted to: Prof. Amarnath Mitra**

**Submitted By: Group**

**Sejal Raj – 055041**

**Vidushi Rawat – 055056**

**Fitness Tracking System - Database Report**

**Introduction**

The Fitness Tracking System database is designed to help users monitor their fitness activities, track workouts, log progress, manage courses, and engage with instructors. This system integrates multiple components to provide a seamless experience for users looking to improve their health. The database is structured to store user information, track exercise routines, monitor dietary habits, and maintain records of achievements and goals.

**Entity and Column Descriptions**

This section provides a detailed breakdown of each table, including column descriptions and their relationships within the database.

**1. Person Table**

The Person table serves as a base entity to store general details about individuals in the system, whether they are users or instructors.

* **person\_id (PK, int)** – A unique identifier for each person.
* **name (string)** – The full name of the individual.
* **email (UNIQUE, string)** – Stores a unique email address for authentication and communication.
* **date\_of\_birth (date)** – The birth date of the individual.
* **gender (enum)** – Represents the gender of the person (e.g., Male, Female, Other).
* **role (enum)** – Specifies whether the person is a user, instructor, or another role in the system.

**Linkage:**

* This table is linked to the User table, meaning that a person can also be a registered user with additional attributes.

**2. User Table**

The User table contains detailed information about registered users and their fitness journey.

* **user\_id (PK, int)** – Unique identifier assigned to each user.
* **password\_hash (string)** – Securely stores the hashed password for authentication.
* **height\_cm (float)** – User's height in centimeters.
* **weight\_kg (float)** – User's weight in kilograms.
* **created\_at (timestamp)** – Records the timestamp when the user account was created.

**Linkage:**

* Inherits attributes from Person.
* Connected to ProgressLog (to track progress), ActivityTracking (to log daily activity), Workout (to manage workouts), NutritionLog (to monitor diet), and User\_Follows\_Instructor (to follow instructors).

**3. ProgressLog Table**

The ProgressLog table keeps track of the user's fitness progress over time.

* **progress\_id (PK, int)** – Unique identifier for each progress log entry.
* **log\_date (date)** – The date when the progress entry was recorded.
* **weight (float)** – User's weight at the time of the log.
* **bmi (float)** – Body Mass Index calculated based on height and weight.
* **user\_id (FK, int)** – Foreign key linking to User, indicating the user to whom the log belongs.

**4. Activity Tracking**

This table records various activities performed by users, allowing them to monitor their daily movements.

* **activity\_id (PK, int)** – Unique identifier for each activity record.
* **activity\_date (date)** – The date when the activity was recorded.
* **steps (int)** – The number of steps taken by the user.
* **calories\_burned (int)** – The amount of calories burned during the activity.
* **user\_id (FK, int)** – Links to the User table to associate activities with a user.

**5. Workout Table**

The Workout table stores information about users' workout sessions.

* **workout\_id (PK, int)** – Unique identifier for a workout session.
* **workout\_date (date)** – The date when the workout was performed.
* **user\_id (FK, int)** – Foreign key referencing the User who performed the workout.
* **course\_id (FK, int)** – Foreign key linking the workout to a specific fitness course.

**Linkage:**

* Connected to Workout\_Exercise to include details about exercises performed.
* Related to Course to allow workouts to be associated with specific courses.

**6.Workout\_Exercise Table**

This table establishes the many-to-many relationship between Workout and Exercise, defining the exercises included in each workout.

* **workout\_exercise\_id (PK, int)** – Unique identifier for the workout-exercise combination.
* **workout\_id (FK, int)** – References Workout to indicate which workout the exercise belongs to.
* **exercise\_id (FK, int)** – References Exercise to specify the exercise performed.
* **sets (int)** – The number of sets performed for this exercise.
* **reps (int)** – The number of repetitions per set.
* **weight\_kg (float)** – The weight used for resistance training.

**7. Exercise Table**

The Exercise table stores details about different exercises available in the system.

* **exercise\_id (PK, int)** – Unique identifier for each exercise.
* **name (string)** – The name of the exercise.
* **description (text)** – A detailed description of the exercise, including instructions.

**Linkage:**

* Connected to Workout\_Exercise to assign exercises to specific workouts.

**8. Course Table**

The Course table represents structured fitness programs available to users.

* **course\_id (PK, int)** – Unique identifier for each course.
* **course\_name (string)** – Name of the fitness course.
* **description (text)** – A detailed description of the course.

**Linkage:**

* Connected to User\_Course to manage user enrollments.
* Associated with Workout to categorize workouts within a course.

**9.User\_Course Table**

Tracks which users are enrolled in which courses.

* **user\_id (FK, int)** – Foreign key linking to User.
* **course\_id (FK, int)** – Foreign key linking to Course.
* **enrollment\_date (date)** – Date when the user enrolled in the course.

**10. NutritionLog Table**

Stores users' dietary intake to track their nutrition habits.

* **log\_id (PK, int)** – Unique identifier for a nutrition log entry.
* **log\_date (date)** – The date when the nutrition log was recorded.
* **meal\_type (enum)** – Specifies the type of meal (e.g., breakfast, lunch, dinner).
* **calories (float)** – The number of calories consumed in the meal.
* **user\_id (FK, int)** – References User to associate the log with a specific user.

**11. User\_Follows\_Instructor Table**

Tracks relationships between users and instructors, allowing users to follow specific fitness professionals.

* **follow\_id (PK, int)** – Unique identifier for the follow action.
* **follow\_date (date)** – Date when the follow relationship was established.
* **user\_id (FK, int)** – References User.
* **instructor\_id (FK, int)** – References Instructor.

**12. Instructor Table**

The Instructor table stores details about fitness instructors.

* **instructor\_id (PK, int)** – Unique identifier for each instructor.
* **specialization (string)** – The area of expertise of the instructor.

**Linkage:**

* Connected to User\_Follows\_Instructor to track which users follow which instructors.

**Relationships Overview**

1. **Person - User (One-to-One)**
   * A Person can have exactly one User account.
   * User inherits personal details from Person.
2. **User - ProgressLog (One-to-Many)**
   * A User can have multiple progress logs, tracking their weight, BMI, and other metrics over time.
   * The user\_id in ProgressLog acts as a foreign key referencing User.
3. **User - ActivityTracking (One-to-Many)**
   * A User can log multiple activity tracking records.
   * Tracks steps taken, calories burned, and the date of activity.
4. **User - Workout (One-to-Many)**
   * Each User can perform multiple workouts.
   * Workout contains workout\_id, workout\_date, and links to User.
5. **Workout - Workout\_Exercise (One-to-Many)**
   * A Workout consists of multiple exercises through Workout\_Exercise.
   * Workout\_Exercise links Workout with Exercise, specifying sets, reps, and weight used.
6. **Workout\_Exercise - Exercise (Many-to-One)**
   * Each Workout\_Exercise entry refers to a single Exercise.
   * Defines which exercises were performed in a workout.
7. **User - NutritionLog (One-to-Many)**
   * A User can log multiple nutrition records.
   * The log tracks meal type, calories consumed, and date.
8. **User - Course (Many-to-Many via User\_Course)**
   * A User can enroll in multiple Courses, and each Course can have multiple Users.
   * The User\_Course table maintains this many-to-many relationship.
9. **User - Instructor (Many-to-Many via User\_Follows\_Instructor)**
   * A User can follow multiple Instructors, and an Instructor can be followed by multiple Users.
   * The User\_Follows\_Instructor table manages this relationship.
10. **Instructor - Course (One-to-Many)**

* An Instructor teaches multiple Courses.
* Instructor is referenced in Course to show who teaches it.

**Key Many-to-Many Relationships**

* User & Course → User\_Course
* User & Instructor → User\_Follows\_Instructor
* Workout & Exercise → Workout\_Exercise

This relational structure ensures comprehensive fitness tracking, progress monitoring, and social engagement between users and instructors.

**Conclusion**

This database provides a comprehensive system for managing fitness-related data, ensuring effective tracking of workouts, exercises, nutrition, and progress. The relationships between tables enable users to get personalized insights into their fitness journey while interacting with courses and instructors.

