**BUAN 6320.008 – Database Foundations for Business Analytics**

# GROUP 10 PROJECT REPORT

# PERSONAL NUTRITION
# AND
# FITNESS MANAGEMENT SYSTEM


**ANANYA CANAKAPALLI – AXC230004**

**HARSHITHA REDDY DUGGANAPALLI – HXD230001**

**RESHMA LNU – RXL230009**

**UNMONA DAS – UXD230001**

**VISHRUTH ACHARYA – VXA220058**

**YASH TATED –YXT220026**

## PROBLEM STATEMENT

In the contemporary era of busy lifestyles, achieving optimal nutrition and fitness is a persistent challenge for many individuals. The traditional one-size-fits-all approach to dietary and fitness planning lacks the granularity to cater to the unique needs of each individual. This limitation is further exacerbated by variables such as dietary restrictions and personalized fitness objectives. Manually tracking and updating nutrition and exercise metrics is a time-consuming process that often leads to inconsistent or inaccurate data. With these challenges, there exists a pressing need for a comprehensive, automated system. This system should not only streamline the tracking and management of personal nutrition and fitness but also offer customized plans based on an individual's unique health parameters and preferences, potentially revolutionizing personal health management.

## BUSINESS PROPOSAL

We are an enterprise dedicated to crafting user-centric applications for people of all ages and backgrounds, with a primary focus on improving health and fitness journeys. Our flagship product, the Nutrition and Fitness Management System, redefines personal wellness. It offers tailored nutrition plans, customized workouts, and data-driven insights, all accessible via smartphones. Our team of innovative problem solvers is committed to enhancing user experiences and streamlining existing workflows. We understand that time is precious, and our application optimizes health management, ensuring that your wellness goals remain within reach. Furthermore, we believe in the power of community support and connectivity. By fostering a sense of belonging and motivation, our application goes beyond a mere tool—it becomes your trusted partner on your path to a healthier and happier life. Join us in this exciting journey toward a revitalized sense of well-being.

## TARGET AUDIENCE

This application is intended for people of all ages who want to manage their individualized nutrition and fitness goals to improve their general health and well-being. It is the most advantageous for:

- Anyone who wants to increase their sports performance, gain muscle, or lose weight
- Those who have particular dietary demands, such as those who have food allergies, chronic illnesses, or ethical and religious convictions
- Those who want assistance getting started in fitness
- Athletes who are looking to improve their performance
- Anyone who wants to keep their weight in check and stay active
- Parents who are looking to raise healthy children

The application is made to be available to users of all ages and technical skill levels. This approach is for everyone who wishes to manage their fitness and diet more effectively and enhance their health.

# FEATURES & CAPABILITIES

## 1. Personalized Nutrition Plans

- Based on customer preferences, health data, and exercise objectives, the system generates customized nutrition regimens. These plans include calorie totals, nutrient breakdowns, and suggested meals.
- This system generates can take in meal logs, has a hydration monitoring system and grocery list generator based on the user inputs.

## 2. Customized Exercise Regimens

- Users receive workout schedules based on their goals and fitness levels. The system suggests a range of routines, including flexibility, cardio, and strength training.
- The system supports exercise variety and emphasizes on rest and restoration. It can be integrated with wearable devices to track health while enabling users to post updates on a social platform.

## 3. Progress Tracking

- Users of the system will be able to monitor changes in their weight, muscular growth, and health indicators like high cholesterol and blood pressure over time.
- Health Milestones and Goal Setting: To provide further motivation and structure, our system allows users to set specific health milestones and goals. Users can define targets such as achieving a certain weight, reducing cholesterol levels, or maintaining blood pressure within a healthy range.
- Data-Driven Insights and Recommendations: Our system doesn't just provide users with raw data; it leverages advanced algorithms to offer data-driven insights and personalized recommendations.

## 4. Social Features

- Users can communicate with friends, compete against one another in fitness challenges, and discuss their advancement. This promotes motivation and a sense of belonging.
- This platform gives users the opportunity to share their fitness knowledge and experiences with peers in addition to motivation and a sense of purpose. It turns into a centre for sharing guidance and even assisting one another in overcoming challenges.

## 5. Health and Wellness Content

- The platform offers articles, videos, and tips on various topics related to nutrition and fitness.
- This educational content equips users with the knowledge they need to make informed decisions.

## 6. Reminders and Notifications

- The system sends timely reminders for meals, workouts, and health check-ups, ensuring that users stay on track.
- The system also notifies the user if their friends have reached any major milestones.

### 7. Virtual Coaching

- For our premium subscribers, we offer an additional feature of virtual coaching and guidance.
- The platform offers on-demand video training and access to our exclusive live streaming content for a variety of training and fitness routines.
- Personal coaching sessions that would help the customer ranging from planning a specific diet chat to schedule a one-on-one training session with certified nutritionists and fitness trainers.

## SOLUTION

In response to the prevailing challenges of fitness and nutrition management, we propose an innovative application that employs a user-friendly interface to deliver effective nutrition and exercise regimens. Recognizing that each individual's health and fitness needs are unique, our solution will provide tailored meal plans and exercise routines, ensuring users have a clear path to achieving their desired outcomes. To simplify tracking, our system seamlessly integrates with wearable devices, offering real-time progress feedback. Supplementing these core features, we will provide educational content that ensures users are well-informed in their health journey. Additionally, the virtual guidance and social platform features transform health management from a solitary endeavour to a community-driven experience. In essence, The Personal Nutrition and Fitness Management System serves as a comprehensive digital health companion, streamlining the often-complex process of managing personal health and fitness seamlessly.

# LIST OF ENTITIES

| ENTITY NO. | ENTITY | DESCRIPTION | ATTRIBUTES |
|---|---|---|---|
| 1 | User Profile Info | This table is designed to store all the user details enrolling into the fitness and nutrition management platform | 1. User ID (Primary Key)<br>2. Name<br>3. Age<br>4. Gender<br>5. Weight<br>6. Height<br>7. Dietary Restrictions (e.g., vegetarian, gluten-free)<br>8. Allergies<br>9. Activity Level (sedentary, moderately active, highly active)<br>10. Contacts (Mobile No.) |
| 2 | Nutrition Plan | This table is designed to store personalized nutrition plans for users, recommend daily calorie intake, macronutrient targets, meal schedules, and associated recipes and each plan is tailored to individual dietary requirements and goals. | 1. Plan ID (Primary Key)<br>2. User ID (Foreign Key referencing User Profile)<br>3. Recommended Calorie Intake<br>4. Macronutrient Targets (carbohydrates, proteins, fats)<br>5. Meal Schedule (breakfast, lunch, dinner, snacks)<br>6. Recipes (linked to Recipe Entity) |
| 3 | Recipes | This table serves as a repository for various recipes. This table provides users with specific recipes that match their nutritional plans and dietary preferences. | 1. Recipe ID (Primary Key)<br>2. Recipe Name<br>3. Ingredients<br>4. Preparation Instructions<br>5. Nutritional Information (calories, macronutrients, vitamins, minerals) |
| 4 | Exercise Plan | The table outlines individualized workout plans for users. It includes details like type of workout, duration of workouts, recommended frequency, difficulty level, and specific exercises and exercise plan is customized to the user's fitness level and goals. | 1. Plan ID (Primary Key)<br>2. User ID (Foreign Key referencing User Profile)<br>3. Workout Type (cardio, strength training, flexibility)<br>4. Workout Duration<br>5. Recommended Frequency (days per week)<br>6. Difficulty Level<br>7. Exercises (linked to Exercise Entity) |

| | | | |
|---|---|---|---|
| 5 | Fitness Goals | This table is dedicated to tracking the fitness goals of users. It includes type of goal, the target date for achieving the goal, and a field for tracking progress. This allows for a personalized approach to goal setting and progress tracking in fitness for each user. | 1. Goal ID<br>2. User ID<br>3. Goal Type<br>4. Target Achievement Date<br>5. Progress Tracking (e.g., distance run, weight lifted) |
| 6 | Group Community | Group Community in the context of a nutrition and fitness management system is likely a feature that allows users to form communities or groups with shared interests in health, wellness, and fitness. | 1. GroupID<br>2. Gname<br>3. Gdescription<br>4. Group_goals |
| 7 | Sharing Features | These sharing features enhance the group community experience, allowing users to exchange information, support each other, and create a collaborative environment within the nutrition and fitness management system. | 1. ShareID<br>2. ProgressID<br>3. Achievements_Desc |
| 8 | Training | This table consists of the details of the trainers like their names. their certification and the kind of specialization they have obtained. | 1. Trainer_ID_PK<br>2. TFirstName<br>3. TLName<br>4. Sex<br>5. Certification<br>6. Specialization<br>7. Contact |
| 9 | Sessions | This table consists of the session details between the users and the trainers along with the duration and the type of session that was hosted. | 1. User_ID_FK<br>2. Session_Type<br>3. Session_Date<br>4. Duration<br>5. Trainer_ID_FK<br>6. Rating |
| 10 | Notifications | The notifications entity is designed to store information about notifications sent to users, including the type of notification, when it was sent, and the content of the notification. | 1. Notification_ID<br>2. User_ID<br>3. Notification_Type<br>4. Timestamp<br>5. Notification_Message |

| | | | |
|---|---|---|---|
| 11 | Addresses | The addresses table stores information about the physical addresses of users. | 1. User_ID<br>2. Address_Line1<br>3. City<br>4. State<br>5. Zip |
| 12 | Workout Reminders | Manages workout reminders for users. Includes fields for Workout_Type to specify work out details and Reminder_Time for the scheduled reminder time. | 1. Reminder_ID<br>2. User_ID<br>3. Workout_Type<br>4. Reminder_Time |
| 13 | Progress Log | The Progress Log table is structured to store detailed logs of users' fitness activities. Key data points include the date of the workout, duration, calories burned, and attendance status, along with additional notes for each session. This table not only tracks individual progress but also provides valuable insights for personalized fitness recommendations and program adjustments. | 1. Progress_id_pk<br>2. User_id_fk<br>3. Date<br>4. Workout_duration<br>5. Calories_burned<br>6. Attendance<br>7. Notes |
| 14 | Content Table | The Content table serves as a repository for various types of content entries. It includes essential details such as the Title of the content, information about the Author, when the content was published, the type of content. This table facilitates the management and organization of different content forms contributed by various users within the system | 1.ContentID_pk<br>2.Title<br>3.Author<br>4.PublishedDate<br>5.ContentText<br>6.ContentType<br>7.User_id_fk |

| | | | |
|---|---|---|---|
| 15 | Updated User Table (Audit Table) | This table is the audit table that stores the previous data of the user whenever a user information is updated. This is associated with the BEFORE UPDATE trigger created on the user table | 1.User_id<br>2.Ufname<br>3.Ulname<br>4.Age<br>5.Sex<br>6.Weight<br>7.Height_cm<br>8.Diet_rest<br>9.Allergies<br>10.Contact<br>11.Modification<br>12.Timeofchange |
| 16 | Fitness Goals Audit Table | This table is the audit table that stores the new data of fitness goals user whenever a user adds a new fitness goal to the fitness table. This is associated with the AFTER UPDATE trigger created on the fitness goals table. | 1.Audit_ID<br>2.User_ID_fk<br>3.Action_Performed<br>4.Goal_Type<br>5.Date_Time |
| 17 | Trainers Audit Table | This table is the audit table that stores the information of a trainer whenever a trainer record is deleted. This is associated with the BEFORE DELETE trigger created on the trainer's table | 1.Action_id<br>2.Old_trainer_id<br>3.tTfname<br>4.Tlname<br>5.Sex<br>6.Certification<br>7.Specilisation<br>8.Contact<br>9.Action_description<br>10.Action_timestamp |

# QUERIES

1. Retrieving the addresses of the users that are enrolled for a training session to deliver the material required for training.

```
375
376     -- 3. Retrieving the address for the users that are enrolled for a training session (so that material re
377 •   SELECT Addresses_T10.User_ID_FK, Addresses_T10.Address_Line1, Addresses_T10.City, Addresses_T10.State, A
378     FROM Addresses_T10
379     JOIN Sessions_T3 ON Addresses_T10.User_ID_FK = Sessions_T3.user_id_fk
380     JOIN Trainers_T2 ON Sessions_T3.trainer_id_fk = Trainers_T2.trainer_id_pk;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| User_ID_FK | Address_Line1 | City | State | Zip |
|---|---|---|---|---|
| 1 | 123 Bailey St | Orlando | Florida | 12345 |
| 3 | 789 Frankford Ln | Albany | Oregon | 11223 |
| 3 | 789 Frankford Ln | Albany | Oregon | 11223 |
| 7 | 843 Battersby Blvd | Orlando | Florida | 69831 |
| 1 | 123 Bailey St | Orlando | Florida | 12345 |
| 12 | zyx lane | dallas | texas | 75080 |
| 5 | 121 Blustery Rd | Raleigh | North Carolina | 77889 |

Result 1 | Read Only

2. Send a notification to all the users in the same city to notify them that there is a marathon run being organized.

```
381
382     -- 4. Send a notification to all the users in the same city to notify them that there is a marathon run
383 •   INSERT INTO Notifications_T9 (User_ID_FK, Notification_Message, Notification_Type, Timestamp) SELECT Use
384
385 •   SELECT *
386     FROM notifications_t9;
387
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

| Notification_ID_PK | User_ID_fk | Notification_Type | Timestamp | Notification_Message |
|---|---|---|---|---|
| 3 | 3 | Update | 2023-12-07 18:01:39 | New diet plan available now! Check it out. |
| 4 | 4 | Reminder | 2023-12-07 18:01:39 | Remember to hydrate after your workout |
| 5 | 5 | Alert | 2023-12-07 18:01:39 | Please update your weight details for better rec... |
| 6 | 1 | Info | 2023-12-07 18:01:39 | Check out our new workout routines for the su... |
| 7 | 6 | Reminder | 2023-12-07 18:01:39 | Log your meals to get better diet insights |
| 8 | 7 | Alert | 2023-12-07 18:01:39 | You missed your workout session today. Stay o... |
| 9 | 2 | Info | 2023-12-07 18:01:39 | Join the community event this weekend! Fun an... |
| 10 | 8 | Reminder | 2023-12-07 18:01:39 | Time to check your weekly progress on the das... |
| 11 | 3 | Marathon Alert | 2023-12-07 18:01:51 | Marathon Run Alert: Join the marathon in your ... |
| 12 | 13 | Marathon Alert | 2023-12-07 18:01:51 | Marathon Run Alert: Join the marathon in your ... |
| NULL | NULL | NULL | NULL | NULL |

notifications_t9 30 | Apply | Revert

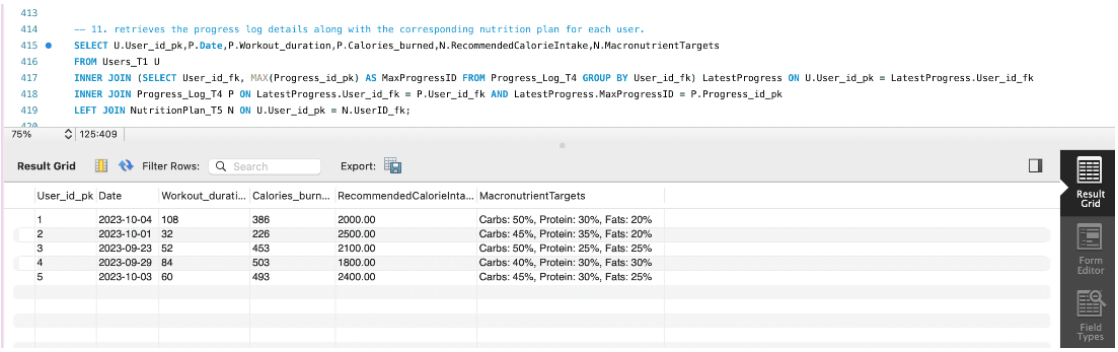
3. Find users who have set workout reminders but have not set any goals

```
412 •   SELECT Users_T1.User_id_pk, Users_T1.Ufname AS Username, Users_T1.Contact AS Contact
413     FROM Users_T1 JOIN WorkoutReminder_T11 ON Users_T1.User_id_pk = WorkoutReminder_T11.User_ID_fk
414     LEFT JOIN FitnessGoals_T8 ON Users_T1.User_id_pk = FitnessGoals_T8.UserID_fk
415     WHERE FitnessGoals_T8.GoalID_pk IS NULL;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| User_id_pk | Username | Contact |
|---|---|---|
| 11 | Katie | 7785412552 |
| 13 | Mia | 4587452635 |
| 14 | Noah | 7856985632 |
| 15 | Olivia | 4587458966 |

4. This query joins multiple tables to retrieve a comprehensive log of user workouts and their corresponding nutritional plans, including workout duration, calories burned, recommended calorie intake, and macronutrient targets, all sorted by user ID and date.



```
413
414     -- 11. retrieves the progress log details along with the corresponding nutrition plan for each user.
415     SELECT U.User_id_pk,P.Date,P.Workout_duration,P.Calories_burned,N.RecommendedCalorieIntake,N.MacronutrientTargets
416     FROM Users_T1 U
417     INNER JOIN (SELECT User_id_fk, MAX(Progress_id_pk) AS MaxProgressID FROM Progress_Log_T4 GROUP BY User_id_fk) LatestProgress ON U.User_id_pk = LatestProgress.User_id_fk
418     INNER JOIN Progress_Log_T4 P ON LatestProgress.User_id_fk = P.User_id_fk AND LatestProgress.MaxProgressID = P.Progress_id_pk
419     LEFT JOIN NutritionPlan_T5 N ON U.User_id_pk = N.UserID_fk;
```

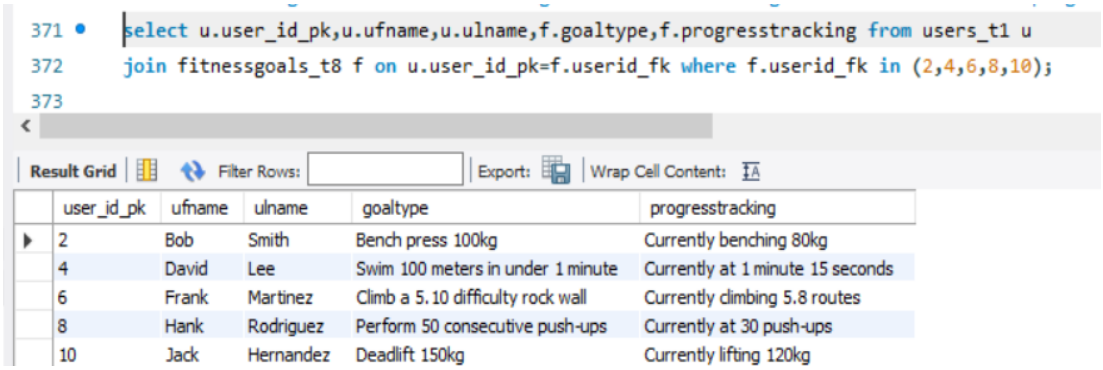
75% 125:409

Result Grid | Filter Rows: Search | Export:

| User_id_pk | Date | Workout_durati... | Calories_burn... | RecommendedCalorieInta... | MacronutrientTargets |
|---|---|---|---|---|---|
| 1 | 2023-10-04 | 108 | 386 | 2000.00 | Carbs: 50%, Protein: 30%, Fats: 20% |
| 2 | 2023-10-01 | 32 | 226 | 2500.00 | Carbs: 45%, Protein: 35%, Fats: 20% |
| 3 | 2023-09-23 | 52 | 453 | 2100.00 | Carbs: 50%, Protein: 25%, Fats: 25% |
| 4 | 2023-09-29 | 84 | 503 | 1800.00 | Carbs: 40%, Protein: 30%, Fats: 30% |
| 5 | 2023-10-03 | 60 | 493 | 2400.00 | Carbs: 45%, Protein: 30%, Fats: 25% |

Result Grid | Form Editor | Field Types

5. Retrieving the user details along with their fitness goals and their current progress from the Users_T1 table and the Fitnessgoals_T8 table.

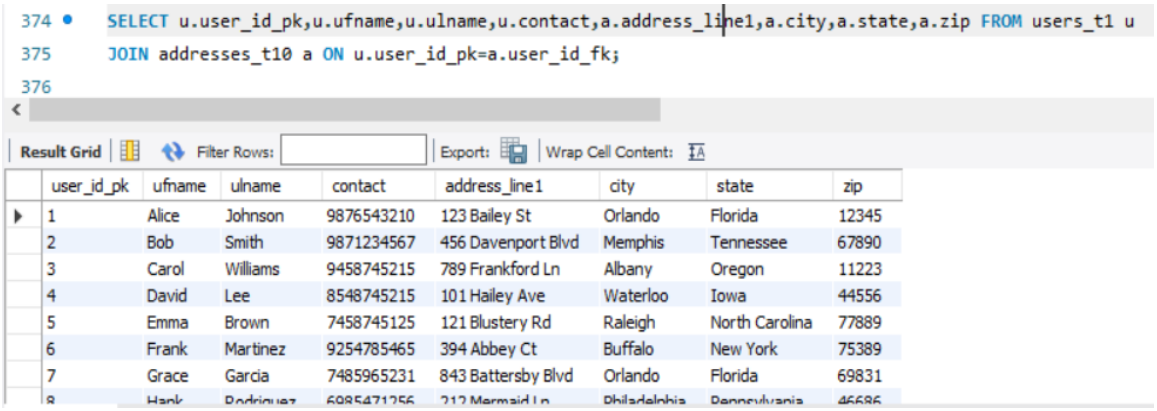


```
371 • select u.user_id_pk,u.ufname,u.ulname,f.goaltype,f.progresstracking from users_t1 u
372   join fitnessgoals_t8 f on u.user_id_pk=f.userid_fk where f.userid_fk in (2,4,6,8,10);
373
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| user_id_pk | ufname | ulname | goaltype | progresstracking |
|---|---|---|---|---|
| 2 | Bob | Smith | Bench press 100kg | Currently benching 80kg |
| 4 | David | Lee | Swim 100 meters in under 1 minute | Currently at 1 minute 15 seconds |
| 6 | Frank | Martinez | Climb a 5.10 difficulty rock wall | Currently climbing 5.8 routes |
| 8 | Hank | Rodriguez | Perform 50 consecutive push-ups | Currently at 30 push-ups |
| 10 | Jack | Hernandez | Deadlift 150kg | Currently lifting 120kg |

6. Retrieving the user details along with their addresses from Users_T1 table and Addresses_T10 table.

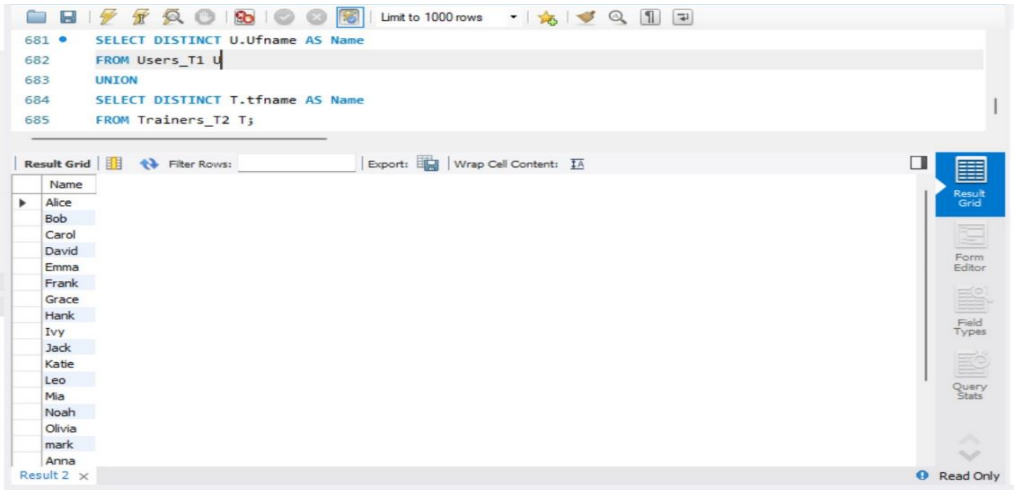


```
374 • SELECT u.user_id_pk,u.ufname,u.ulname,u.contact,a.address_line1,a.city,a.state,a.zip FROM users_t1 u
375   JOIN addresses_t10 a ON u.user_id_pk=a.user_id_fk;
376
```

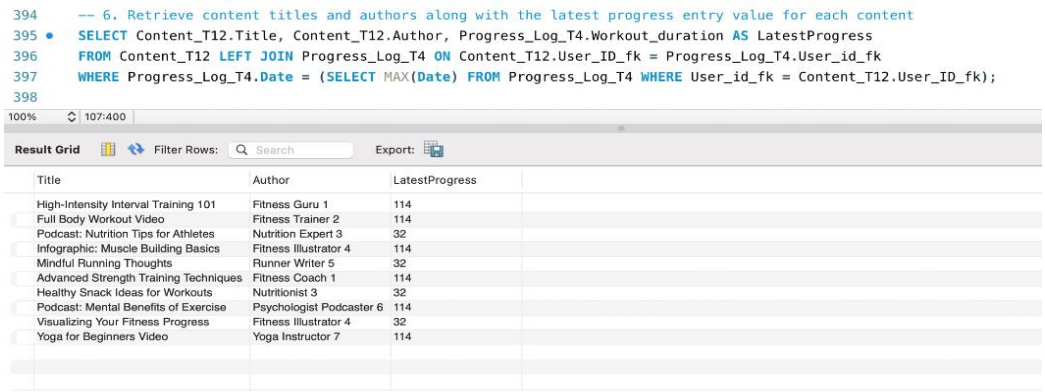
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| user_id_pk | ufname | ulname | contact | address_line1 | city | state | zip |
|---|---|---|---|---|---|---|---|
| 1 | Alice | Johnson | 9876543210 | 123 Bailey St | Orlando | Florida | 12345 |
| 2 | Bob | Smith | 9871234567 | 456 Davenport Blvd | Memphis | Tennessee | 67890 |
| 3 | Carol | Williams | 9458745215 | 789 Frankford Ln | Albany | Oregon | 11223 |
| 4 | David | Lee | 8548745215 | 101 Hailey Ave | Waterloo | Iowa | 44556 |
| 5 | Emma | Brown | 7458745125 | 121 Blustery Rd | Raleigh | North Carolina | 77889 |
| 6 | Frank | Martinez | 9254785465 | 394 Abbey Ct | Buffalo | New York | 75389 |
| 7 | Grace | Garcia | 7485965231 | 843 Battersby Blvd | Orlando | Florida | 69831 |
| 8 | Hank | Rodriguez | 6985471256 | 212 Mermaid Ln | Philadelphia | Pennsylvania | 46686 |

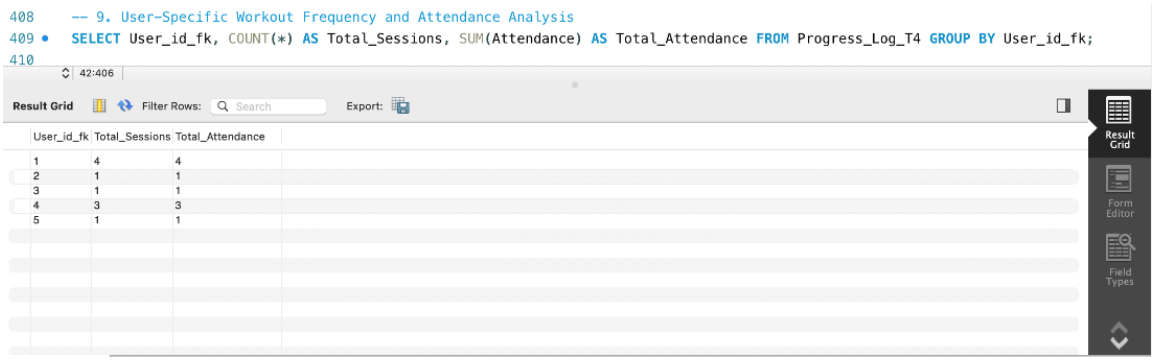
7. Combining and selecting distinct names from two different tables—Users_T1 and Trainers_T2.




8. Retrieving the latest workout progress entries for each user by joining the Content_T12 and Progress_Log_T4 tables, showing content titles, authors, and the duration of the latest workout session.




9. Aggregating data from the Progress_Log_T4 table to analyze workout frequency and attendance per user. It counts the total sessions and sums the attendance for each user, grouped by the user's ID.



10. Calculating the total calories burned per user by summing up the Calories_burned field from the Progress_Log_T4 table. The results are grouped by the user's ID to provide individual totals.

```
411    -- 10. Analysis of Total Calories Burned per User
412 •  SELECT User_id_fk, SUM(Calories_burned) AS Total_Calories_Burned FROM Progress_Log_T4 GROUP BY User_id_fk;
413
```

Result Grid | Filter Rows: Search | Export:

| User_id_fk | Total_Calories_Burned |
|---|---|
| 1 | 1823 |
| 2 | 226 |
| 3 | 453 |
| 4 | 1807 |
| 5 | 493 |

11. To check an overview of users and the count of content items they have viewed

```
423 •  SELECT U.User_id_pk AS User_ID,CONCAT(U.Ufname, ' ', U.Ulname) AS Username,U.Contact,COUNT(CT.ContentID_pk) AS Content_Count
424    FROM Users_T1 U LEFT JOIN Content_T12 CT ON U.User_id_pk = CT.User_ID_fk
425    GROUP BY U.User_id_pk, Username, U.Contact ORDER BY U.User_id_pk;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| User_ID | Username | Contact | Content_Count |
|---|---|---|---|
| 1 | Alice Johnson | 9876543210 | 6 |
| 2 | Bob Smith | 9871234567 | 4 |
| 3 | Carol Williams | 9458745215 | 0 |
| 4 | David Lee | 8548745215 | 0 |
| 5 | Emma Brown | 7458745125 | 0 |
| 6 | Frank Martinez | 9254785465 | 0 |
| 7 | Grace Garcia | 7485965231 | 0 |
| 8 | Hank Rodriguez | 6985471256 | 0 |
| 9 | Ivy Perez | 9584621452 | 0 |
| 10 | Jack Hernandez | 2844521458 | 0 |
| 11 | Katie Doe | 7785412552 | 0 |
| 12 | Leo Moore | 1234567 | 0 |

12. To determine the workout duration variability of users based on their different workouts and progress using the Progress_Log_T4 Table.

```
428 •  SELECT User_id_fk, MAX(Workout_duration) AS Longest_Workout, MIN(Workout_duration) AS Shortest_Workout, STDDEV(Workout_duration)
429    AS Duration_Variability FROM Progress_Log_T4 GROUP BY User_id_fk;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| User_id_fk | Longest_Workout | Shortest_Workout | Duration_Variability |
|---|---|---|---|
| 1 | 114 | 81 | 12.616952880945542 |
| 2 | 32 | 32 | 0 |
| 3 | 52 | 52 | 0 |
| 4 | 104 | 46 | 24.055491403558285 |
| 5 | 60 | 60 | 0 |

13. Getting the details of trainers who have the Fitness certification

```
445 •   SELECT * FROM trainers_t2
446     WHERE trainer_id_pk IN (SELECT trainer_id_fk FROM sessions_t3 WHERE certification = 'Fitness');
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

| trainer_id_pk | tfname | tlname | sex | certification | specilisation | contact |
|---|---|---|---|---|---|---|
| 2 | John | Doe | M | Fitness | Cardio | 9871234567 |
| 4 | Michael | Brown | M | Fitness | Yoga | 8546321578 |
| 5 | Sophia | Garcia | F | Fitness | Cardio | 9856321547 |
| 6 | David | Martinez | M | Fitness | Gym | 7854698523 |
| 8 | Daniel | Lee | M | Fitness | Yoga | 4587452122 |
| 9 | Emma | Hernandez | F | Fitness | Gym | 9965887452 |

14. Get users who have set workout reminders for yoga.

```
449 •   SELECT * FROM users_t1
450     WHERE user_id_pk IN (SELECT user_id_fk FROM workoutreminder_t11 WHERE workout_type = 'Yoga');
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

| User_id_pk | Ufname | Ulname | Age | Sex | Weight | Height_cm | Diet_rest | Allergies | Contact |
|---|---|---|---|---|---|---|---|---|---|
| 3 | Carol | Williams | 28 | F | 60 | 165 | Vegan | Lactose | 9458745215 |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

# Stored Procedures

### 1. Procedure to insert new user

- Purpose: Inserts new user details into the users_t1 table
- Parameters: ufname (user first name), ulname (user last name), age (age of the user), sex (gender), weight (weight of the user), height_cm (Height in cm)
- Functionality: Inserts a row into the users_t1 table with the provided information.

**OUTPUT**

```
505 •  CALL insert_new_user("mark","cubin",56, "M", 83, 178);
506 •  select * from users_t1 where ufname="mark";
507
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell

| User_id_pk | Ufname | Ulname | Age | Sex | Weight | Height_cm | Diet_rest | Allergies | Contact |
|---|---|---|---|---|---|---|---|---|---|
| 16 | mark | cubin | 56 | M | 83 | 178 | NULL | NULL | 0 |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

Outcome: There is a new record in the users_t1 table

### 2. Procedure to insert fitness goals

- Purpose: Inserts new fitness goals into the fitnessgoals_t8 table.
- Parameters: user_id (User's ID), goal_type (Type of fitness goal), target_date (Date to achieve the goal)
- Functionality: Inserts a row into the fitnessgoals_t8 table with the provided user ID, goal type, and target date.

**OUTPUT**

```
517 •  call insert_fitness_goals(12,"cardio","2023-11-03");
518 •  select * from fitnessgoals_t8;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

| GoalID_pk | UserID_fk | GoalType | TargetAchievementDate | ProgressTracking |
|---|---|---|---|---|
| 1 | 1 | Run a marathon | 2023-12-18 | Training for 10 kilometers currently |
| 2 | 2 | Bench press 100kg | 2023-11-15 | Currently benching 80kg |
| 3 | 3 | Daily yoga for a month | 2023-11-01 | Completed 10 days consecutively |
| 4 | 4 | Swim 100 meters in under 1 minute | 2023-12-01 | Currently at 1 minute 15 seconds |
| 5 | 5 | Achieve a 5-minute plank | 2024-01-10 | Currently holding for 3 minutes |
| 6 | 6 | Climb a 5.10 difficulty rock wall | 2023-12-20 | Currently climbing 5.8 routes |
| 7 | 7 | Cycle 50 kilometers non-stop | 2024-02-05 | Last longest ride was 35 kilometers |
| 8 | 8 | Perform 50 consecutive push-ups | 2023-12-15 | Currently at 30 push-ups |
| 9 | 9 | Master the crow yoga pose | 2023-11-20 | Can hold the pose for a few seconds |
| 10 | 10 | Deadlift 150kg | 2024-03-01 | Currently lifting 120kg |
| 11 | 12 | cardio | 2023-11-03 | NULL |

Outcome: There is a new record in the fitnessgoals_t8 table

NOTE: There is a Null value at column progress tracking as it is a new Fitness Goal assigned to a new user.

### 3. Procedure to  Update user address

- Purpose: Updates the address of a user in the addresses_t10 table.
- Parameters: userid, addline (Address Line), newcity (City), newstate (State), newzip (ZIP Code).
- Functionality: Updates the address details for the given user ID.

**OUTPUT**

```
529 •  call update_user_address(12,'zyx lane','dallas','texas',75080);
530 •  select * from addresses_t10;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| User_ID_FK | Address_Line1 | City | State | Zip |
|---|---|---|---|---|
| 1 | 123 Bailey St | Orlando | Florida | 12345 |
| 2 | 456 Davenport Blvd | Memphis | Tennessee | 67890 |
| 3 | 789 Frankford Ln | Albany | Oregon | 11223 |
| 4 | 101 Hailey Ave | Waterloo | Iowa | 44556 |
| 5 | 121 Blustery Rd | Raleigh | North Carolina | 77889 |
| 6 | 394 Abbey Ct | Buffalo | New York | 75389 |
| 7 | 843 Battersby Blvd | Orlando | Florida | 69831 |
| 8 | 212 Mermaid Ln | Philadelphia | Pennsylvania | 46686 |
| 9 | 709 Bentley Rd | Waterloo | Iowa | 10058 |
| 10 | 413 Hollywood Ave | Oakland | California | 76309 |
| 11 | 634 Graceland Ave | Princeton | New Jersey | 26899 |
| 12 | zyx lane | dallas | texas | 75080 |
| 13 | 863 Peachtree Ln | Albany | Oregon | 46387 |
| 14 | 742 Wayne Ct | Roswell | New Mexico | 95317 |
| 15 | 632 Woodman Ct | Mitchelle | South Dakota | 75214 |

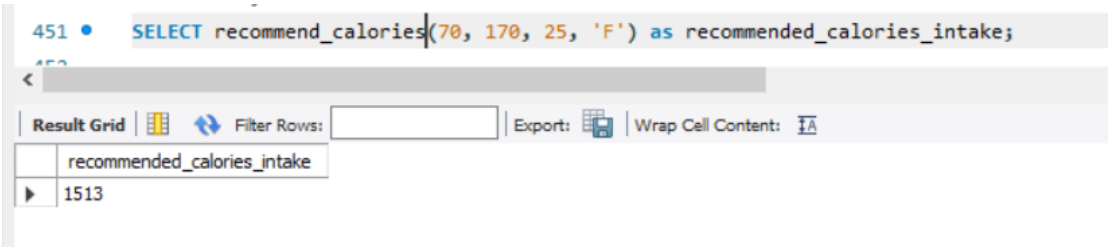
Outcome: The values assosiated with the row User_ID_FK =12 has been updated by calling the procedure update_user_address.

# Functions

### 1. Function recommend_calories

- Purpose: Calculates the recommended daily calorie intake based on the user's weight, height, age, and sex.
- Parameters: weight (in kg), height (in cm), age (in years), sex ('M' for male, 'F' for female).
- Functionality: Uses a formula to calculate Basal Metabolic Rate (BMR) which is a measure of the number of calories required to keep your body functioning at rest. The formula differs for males and females.

**OUTPUT**

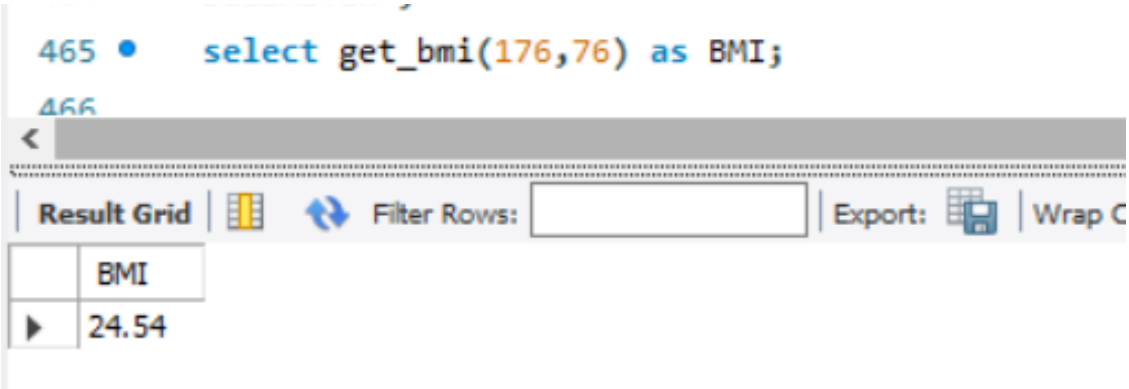



Based on the height, weight, age and gender, the functions gives the recommended calorie intake.

### 2. Function get_bmi

- Purpose: Calculates the Body Mass Index (BMI) of a user.
- Parameters: height (in cm), weight (in kg).
- Functionality: BMI is calculated as weight (kg) divided by the square of height (m). This function first converts height from cm to meters before calculating BMI.

**OUTPUT**

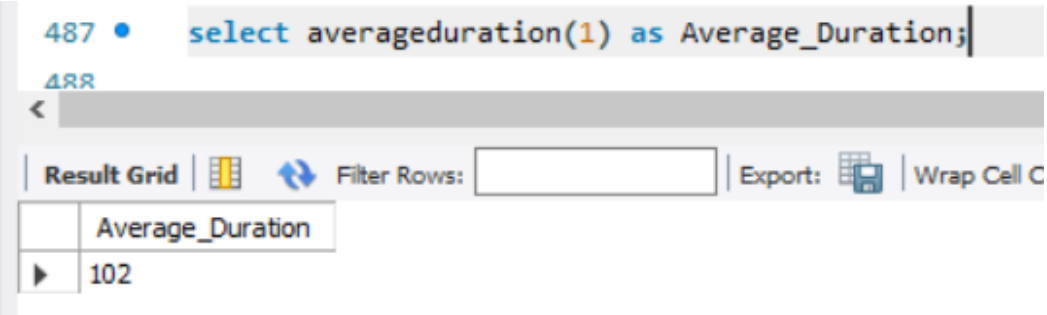



Here the BMI of an individual generated based on the values provided by the customer where the height is 176 cms and weight is 76 kgs would be 24.54.

### 3. Function AverageDuration:

- Purpose: Computes the average duration of workout sessions for a specific user.
- Parameters: userID (User's ID).
- Functionality: It sums the total workout duration and counts the number of sessions from the progress_log_t4 table for a given user and then divides the total duration by the number of sessions to find the average.

**OUTPUT**




The function returns the average workout duration for each user.

# Triggers

### 1. User Update Trigger:

- Purpose: Triggers whenever the user data is updated
- Parameters: User_id, Ufname, Ulname, Age, Sex, Weight, Height_cm, Diet_rest , Allergies, Contact, Modification, Timeofchange
- Functionality: Inserts a row into the updated_user_t15 table with the previous user data before updating.

**OUTPUT**

```
542 •  UPDATE users_t1 SET contact=1234567 WHERE user_id_pk=12;
543 •  select * from updated_user_T15;
544
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| User_id | Ufname | Ulname | Age | Sex | Weight | Height_cm | Diet_rest | Allergies | Contact | Modification | Timeofchange |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 12 | Leo | Moore | 23 | M | 68 | 176 | Low Carb | Gluten | 7458456325 | Before Update | 2023-12-07 20:19:29 |

### 2. Fitness Goals Trigger:

- Purpose: Triggers whenever the user inserts a new fitness goal
- Parameters: Audit_ID, User_ID_fk, Action_Performed, Goal_Type , Date_Time
- Functionality: Inserts a row into the fitnessgoals_audit_t16 table with the modification type and the type of goal set.

**OUTPUT**

```
569 •  call insert_fitness_goals(12,"cardio","2023-11-03");
570 •  select * from fitnessgoals_audit_t16;
571
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

| Audit_ID | User_ID_fk | Action_Performed | Goal_Type | Date_Time |
|---|---|---|---|---|
| 1 | 12 | Goal Set | cardio | 2023-12-07 20:19:29 |
| NULL | NULL | NULL | NULL | NULL |

### 3. Delete Trainer Trigger:

- Purpose: Triggers whenever the trainer record is deleted.
- Parameters: action_id, old_trainer_id, Tfname, Tlname, Sex, Certification, Specilisation, Contact, action_description, action_timestamp
- Functionality: Inserts a row into the trainers_audit_t17 table with the the previous trainer data before deletion.

**OUTPUT**

```
600 •  Delete from trainers_t2 where trainer_id_pk=10;
601 •  select * from trainers_audit_t17;
602 •  select * from trainers_t2;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

| action_id | old_trainer_id | tfname | tlname | sex | certification | Specilisation | contact | action_description | action_timestamp |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 10 | Liam | Perez | M | Fitness | Cardio | 2015458788 | Deleted | 2023-12-07 20:19:29 |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

# Indexes

## 1. Name Index

- Purpose: Created an Index on the first and last names of users from the users_t1 table for faster retrieval of data.
- Functionality: As the user table is references to majority of the tables, creating an index for the user table would be efficient.

**OUTPUT**

```
629 •  CREATE INDEX name_index ON users_t1 (Ufname,Ulname);
630 •  show index from nutritionplan_t5;
631
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| Table | Non_unique | Key_name | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type |
|---|---|---|---|---|---|---|---|---|---|---|
| nutritionplan_t5 | 0 | PRIMARY | 1 | PlanID_pk | A | 10 | NULL | NULL | | BTREE |
| nutritionplan_t5 | 1 | nutrition_index | 1 | UserID_fk | A | 10 | NULL | NULL | YES | BTREE |
| nutritionplan_t5 | 1 | nutrition_index | 2 | RecommendedCalorieIntake | A | 10 | NULL | NULL | | BTREE |

## 2. Nutrition Index

- Purpose: Created an Index on the user_id and recommendedcalorieintake of users from the nutritionplan_t5 table for faster retrieval of data.
- Functionality: As the nutrition plan is used in other tables like exercise and fitness goals, creating an index for the nutrition table, based on calorie intake would be efficient.

**OUTPUT**

```
633 •  CREATE INDEX nutrition_index ON nutritionplan_T5 (userid_fk, recommendedcalorieintake);
634 •  show index from users_t1;
635
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| Table | Non_unique | Key_name | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| users_t1 | 0 | PRIMARY | 1 | User_id_pk | A | 7 | NULL | NULL | | BTREE | | | YES |
| users_t1 | 1 | name_index | 1 | Ufname | A | 8 | NULL | NULL | YES | BTREE | | | YES |
| users_t1 | 1 | name_index | 2 | Ulname | A | 8 | NULL | NULL | YES | BTREE | | | YES |

# Views

## 1. View User_Goal_Status

- Purpose: Provides a quick view of each user's progress against their fitness goals.
- Functionality: Combines data from Users_T1 and FitnessGoals_T8 to show user details along with their fitness goals and progress.

**OUTPUT**

```
621 •   select * from user_goal_status;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| User_id_pk | Ufname | Ulname | GoalType | TargetAchievementDate | ProgressTracking |
|---|---|---|---|---|---|
| 1 | Alice | Johnson | Run a marathon | 2023-12-18 | Training for 10 kilometers currently |
| 2 | Bob | Smith | Bench press 100kg | 2023-11-15 | Currently benching 80kg |
| 3 | Carol | Williams | Daily yoga for a month | 2023-11-01 | Completed 10 days consecutively |
| 4 | David | Lee | Swim 100 meters in under 1 minute | 2023-12-01 | Currently at 1 minute 15 seconds |
| 5 | Emma | Brown | Achieve a 5-minute plank | 2024-01-10 | Currently holding for 3 minutes |
| 6 | Frank | Martinez | Climb a 5.10 difficulty rock wall | 2023-12-20 | Currently climbing 5.8 routes |
| 7 | Grace | Garcia | Cycle 50 kilometers non-stop | 2024-02-05 | Last longest ride was 35 kilometers |
| 8 | Hank | Rodriguez | Perform 50 consecutive push-ups | 2023-12-15 | Currently at 30 push-ups |
| 9 | Ivy | Perez | Master the crow yoga pose | 2023-11-20 | Can hold the pose for a few seconds |
| 10 | Jack | Hernandez | Deadlift 150kg | 2024-03-01 | Currently lifting 120kg |
| 12 | Leo | Moore | cardio | 2023-11-03 | NULL |
| 12 | Leo | Moore | cardio | 2023-11-03 | NULL |

## 2. View Trainer_Session_Summary:

- **Purpose**: Offers trainers a summary of their training sessions, including total sessions, average ratings, and session types.
- **Functionality**: Aggregates data from Trainers_T2 and Sessions_T3 to provide a summary of each trainer's sessions, categorized by session type.

**OUTPUT**

```
622 •   select * from trainer_session_summary;
623
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| trainer_id_pk | tfname | tlname | Total_Sessions | Average_Rating | session_type |
|---|---|---|---|---|---|
| 1 | Anna | Smith | 1 | 5.0000 | V |
| 2 | John | Doe | 1 | 4.0000 | P |
| 2 | John | Doe | 1 | 3.0000 | V |
| 3 | Laura | Johnson | 1 | 4.0000 | P |
| 3 | Laura | Johnson | 1 | 3.0000 | V |
| 4 | Michael | Brown | 1 | 5.0000 | P |
| 5 | Sophia | Garcia | 1 | 5.0000 | P |
| 6 | David | Martinez | 1 | 4.0000 | V |
| 7 | Olivia | Rodriguez | 1 | 4.0000 | V |
| 8 | Daniel | Lee | 1 | 3.0000 | P |
| 8 | Daniel | Lee | 1 | 3.0000 | V |
| 9 | Emma | Hernandez | 1 | 3.0000 | P |