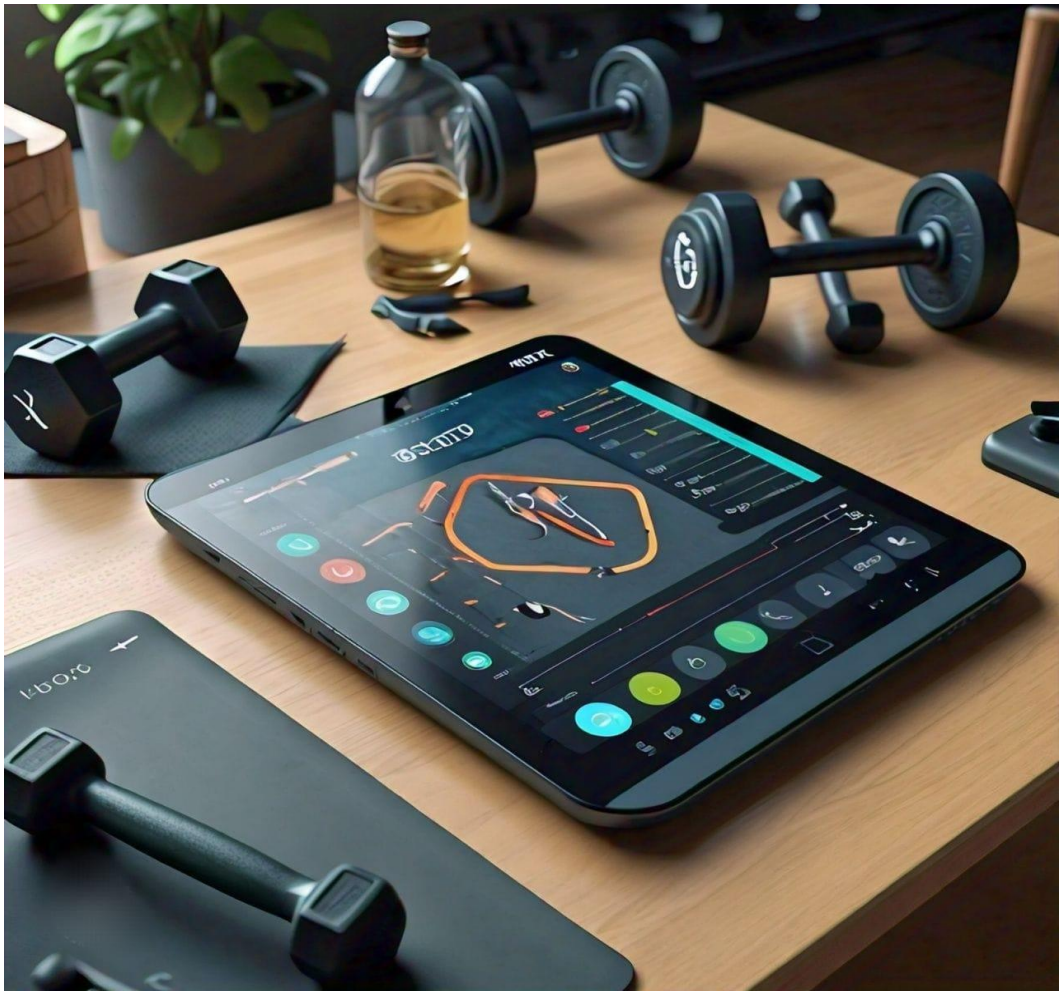
Case Study #1 - Fitness Tracking and Workout Planning System

Introduction:

A **Fitness Tracking and Workout Planning System** helps individuals achieve their fitness goals by monitoring workouts, analyzing progress, and providing personalized plans. Users can track activities, set goals, and receive insights to improve performance. The system adapts plans based on individual needs, fostering motivation and promoting a healthier lifestyle.



Problem Statement:

Many people struggle with maintaining fitness routines due to lack of motivation and progress tracking.

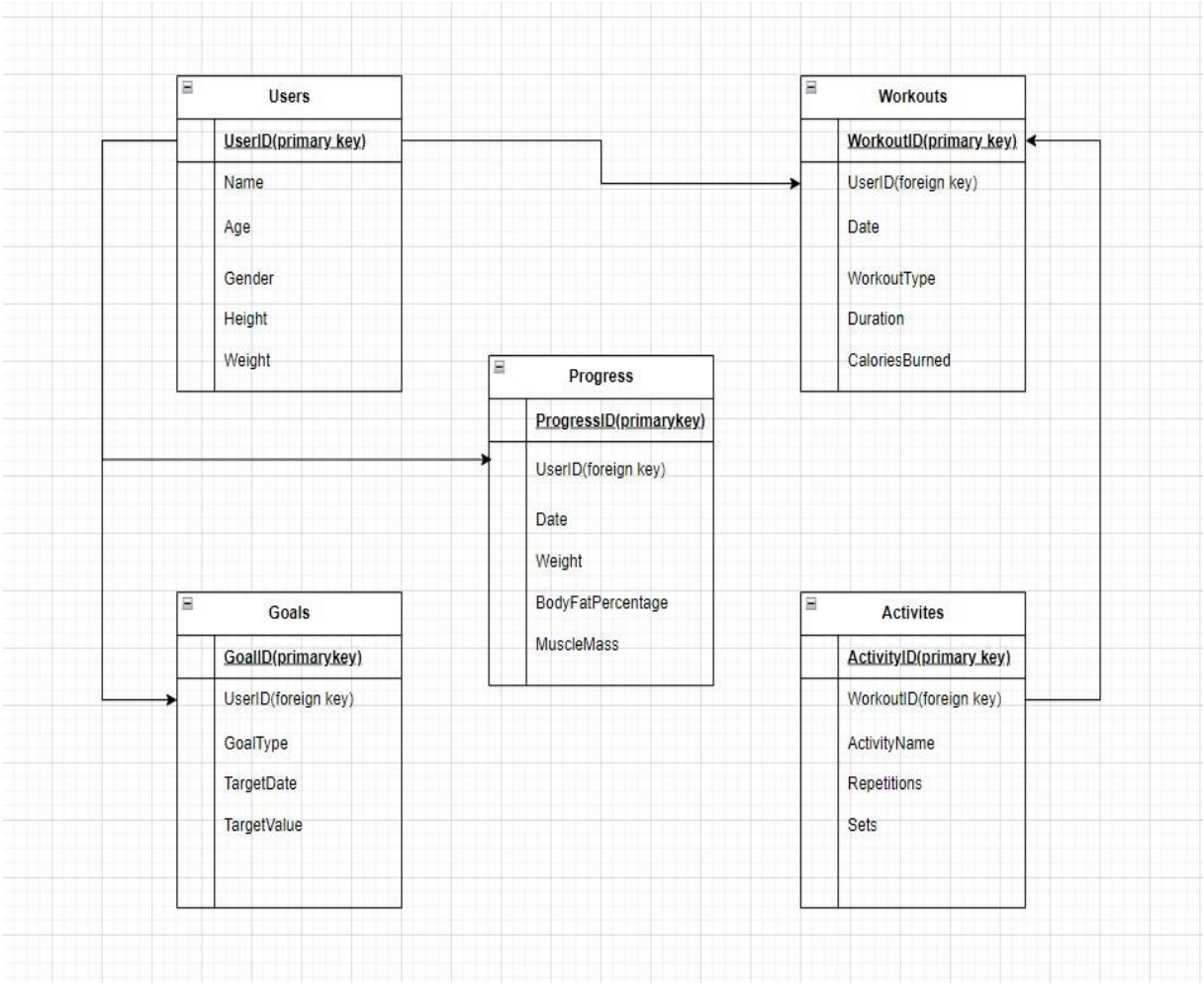
Solution:

Create a **Fitness Tracking and Workout Planning System** to:

- Track workouts and calories.
- Set and monitor goals.
- Analyze data for personalized plans.

This helps users stay motivated and achieve their fitness objectives effectively.

Entity Relationship Diagram:



Dataset:

```
CREATE DATABASE FitnessTracking;
USE FitnessTracking;

select*from users;
select*from workouts;
select * from Activities;
select*from goals;
select * from progress;
/* ---------------------------------------------- users----------------------------------------------*/
```

Create Table & Insert Data:

```
CREATE TABLE Users (
UserID INT PRIMARY KEY,
Name VARCHAR(100),
Age INT,
Gender VARCHAR(10),
Height DECIMAL(5,2),
Weight DECIMAL(5,2)
);

SELECT * FROM Users WHERE UserID =UserID;
SHOW COLUMNS FROM Users;
ALTER TABLE Users ADD COLUMN Height INT;
ALTER TABLE Users ADD COLUMN Weight INT;

INSERT INTO Users (UserID, Name, Age, Gender, Height, Weight) VALUES
(1, 'Alice', 30, 'Female', 165, 60),
(2, 'Bob', 25, 'Male', 180, 75),
(3, 'Charlie', 28, 'Male', 175, 70),
(4, 'Diana', 32, 'Female', 160, 55),
(5, 'Eve', 29, 'Female', 170, 65),
(6, 'Frank', 31, 'Male', 182, 85),
(7, 'Grace', 27, 'Female', 158, 54),
(8, 'Hank', 26, 'Male', 178, 78),
(9, 'Ivy', 34, 'Female', 162, 59),
(10, 'Jack', 33, 'Male', 185, 90),
(11, 'Kate', 30, 'Female', 168, 62),
(12, 'Leo', 35, 'Male', 174, 77),
(13, 'Mona', 29, 'Female', 167, 64),
(14, 'Nate', 28, 'Male', 179, 74),
(15, 'Olive', 31, 'Female', 161, 57),
```

(16, 'Paul', 36, 'Male', 180, 82),
(17, 'Quinn', 32, 'Female', 169, 63),
(18, 'Rick', 27, 'Male', 176, 73),
(19, 'Sara', 34, 'Female', 166, 58),
(20, 'Tom', 30, 'Male', 181, 86),
(21, 'Uma', 28, 'Female', 165, 61),
(22, 'Vic', 33, 'Male', 177, 79),
(23, 'Walt', 26, 'Male', 183, 88),
(24, 'Xena', 29, 'Female', 159, 56),
(25, 'Yara', 31, 'Female', 170, 66),
(26, 'Zane', 30, 'Male', 172, 81),
(27, 'Amy', 27, 'Female', 163, 60),
(28, 'Ben', 32, 'Male', 175, 76),
(29, 'Cleo', 28, 'Female', 168, 65),
(30, 'Dan', 33, 'Male', 179, 84),
(31, 'Ella', 31, 'Female', 164, 59),
(32, 'Finn', 30, 'Male', 173, 80),
(33, 'Gina', 34, 'Female', 162, 58),
(34, 'Hugo', 29, 'Male', 184, 87),
(35, 'Iris', 28, 'Female', 167, 64),
(36, 'Jake', 27, 'Male', 176, 72),
(37, 'Kara', 32, 'Female', 160, 55),
(38, 'Liam', 31, 'Male', 180, 83),
(39, 'Mia', 29, 'Female', 169, 63),
(40, 'Noah', 34, 'Male', 178, 75),
(41, 'Opal', 30, 'Female', 161, 57),
(42, 'Pete', 28, 'Male', 182, 85),
(43, 'Rita', 31, 'Female', 166, 62),
(44, 'Sam', 33, 'Male', 174, 78),
(45, 'Tina', 27, 'Female', 158, 54),
(46, 'Ugo', 35, 'Male', 181, 89),
(47, 'Vera', 30, 'Female', 170, 65),
(48, 'Will', 29, 'Male', 177, 77),
(49, 'Xavi', 32, 'Male', 183, 91),
(50, 'Yuri', 28, 'Female', 164, 60),
(51, 'Zoe', 31, 'Female', 168, 64),
(52, 'Ava', 27, 'Female', 162, 59),
(53, 'Bo', 34, 'Male', 180, 82),
(54, 'Cody', 30, 'Male', 175, 74),
(55, 'Demi', 29, 'Female', 159, 56),
(56, 'Eli', 33, 'Male', 178, 79),
(57, 'Fay', 31, 'Female', 167, 61),
(58, 'Gus', 32, 'Male', 172, 73),
(59, 'Hope', 28, 'Female', 165, 58),

(60, 'Ian', 30, 'Male', 181, 88);
/* -- **workouts**--*/
Create Table & Insert Data:

CREATE TABLE Workouts (
WorkoutID INT PRIMARY KEY,
UserID INT,
Date DATE,
WorkoutType VARCHAR(50),
Duration INT,
CaloriesBurned INT,
FOREIGN KEY (UserID) REFERENCES Users(UserID)
);
select*from workouts;
show columns from workouts;

INSERT INTO Workouts (WorkoutID, UserID, Date, WorkoutType, Duration, CaloriesBurned) VALUES
(1, 1, '2023-07-01', 'Cardio', 30, 300),
(2, 2, '2023-07-02', 'Strength', 45, 400),
(3, 3, '2023-07-03', 'Yoga', 60, 200),
(4, 4, '2023-07-04', 'Pilates', 50, 250),
(5, 5, '2023-07-05', 'Cycling', 40, 350),
(6, 6, '2023-07-06', 'Running', 30, 280),
(7, 7, '2023-07-07', 'Swimming', 60, 500),
(8, 8, '2023-07-08', 'HIIT', 45, 450),
(9, 9, '2023-07-09', 'Boxing', 50, 400),
(10, 10, '2023-07-10', 'CrossFit', 60, 600),
(11, 11, '2023-07-11', 'Cardio', 35, 310),
(12, 12, '2023-07-12', 'Strength', 50, 420),
(13, 13, '2023-07-13', 'Yoga', 55, 190),
(14, 14, '2023-07-14', 'Pilates', 45, 240),
(15, 15, '2023-07-15', 'Cycling', 42, 360),
(16, 16, '2023-07-16', 'Running', 32, 290),
(17, 17, '2023-07-17', 'Swimming', 62, 510),
(18, 18, '2023-07-18', 'HIIT', 48, 460),
(19, 19, '2023-07-19', 'Boxing', 52, 410),
(20, 20, '2023-07-20', 'CrossFit', 58, 590),
(21, 21, '2023-07-21', 'Cardio', 28, 290),
(22, 22, '2023-07-22', 'Strength', 47, 405),
(23, 23, '2023-07-23', 'Yoga', 63, 205),
(24, 24, '2023-07-24', 'Pilates', 53, 255),
(25, 25, '2023-07-25', 'Cycling', 38, 340),
(26, 26, '2023-07-26', 'Running', 29, 270),

(27, 27, '2023-07-27', 'Swimming', 58, 495),
(28, 28, '2023-07-28', 'HIIT', 44, 440),
(29, 29, '2023-07-29', 'Boxing', 48, 395),
(30, 30, '2023-07-30', 'CrossFit', 62, 605),
(31, 31, '2023-07-31', 'Cardio', 31, 295),
(32, 32, '2023-08-01', 'Strength', 49, 415),
(33, 33, '2023-08-02', 'Yoga', 57, 195),
(34, 34, '2023-08-03', 'Pilates', 47, 245),
(35, 35, '2023-08-04', 'Cycling', 41, 355),
(36, 36, '2023-08-05', 'Running', 33, 300),
(37, 37, '2023-08-06', 'Swimming', 64, 520),
(38, 38, '2023-08-07', 'HIIT', 46, 455),
(39, 39, '2023-08-08', 'Boxing', 54, 415),
(40, 40, '2023-08-09', 'CrossFit', 59, 595),
(41, 41, '2023-08-10', 'Cardio', 34, 305),
(42, 42, '2023-08-11', 'Strength', 52, 430),
(43, 43, '2023-08-12', 'Yoga', 60, 210),
(44, 44, '2023-08-13', 'Pilates', 51, 260),
(45, 45, '2023-08-14', 'Cycling', 37, 345),
(46, 46, '2023-08-15', 'Running', 28, 260),
(47, 47, '2023-08-16', 'Swimming', 57, 490),
(48, 48, '2023-08-17', 'HIIT', 43, 435),
(49, 49, '2023-08-18', 'Boxing', 50, 405),
(50, 50, '2023-08-19', 'CrossFit', 61, 610),
(51, 51, '2023-08-20', 'Cardio', 29, 285),
(52, 52, '2023-08-21', 'Strength', 46, 410),
(53, 53, '2023-08-22', 'Yoga', 62, 220),
(54, 54, '2023-08-23', 'Pilates', 54, 265),
(55, 55, '2023-08-24', 'Cycling', 39, 350),
(56, 56, '2023-08-25', 'Running', 30, 275),
(57, 57, '2023-08-26', 'Swimming', 55, 485),
(58, 58, '2023-08-27', 'HIIT', 42, 430),
(59, 59, '2023-08-28', 'Boxing', 49, 400),
(60, 60, '2023-08-29', 'CrossFit', 63, 615);

/* ---------- **Activities** ----------*/

Create Table & Insert Data:

CREATE TABLE Activities (
ActivityID INT PRIMARY KEY,
WorkoutID INT,
ActivityName VARCHAR(50),
Repetitions INT,

```
Sets INT,
FOREIGN KEY (WorkoutID) REFERENCES Workouts(WorkoutID)
);
select * from Activities;
show  columns from Activities;

INSERT INTO Activities (ActivityID, WorkoutID, ActivityName, Repetitions, Sets)
VALUES
(1, 1, 'Running', NULL, NULL),
(2, 2, 'Bench Press', 10, 3),
(3, 3, 'Sun Salutation', 5, 3),
(4, 4, 'Leg Raises', 15, 4),
(5, 5, 'Cycling Sprints', NULL, NULL),
(6, 6, 'Squats', 12, 4),
(7, 7, 'Butterfly Stroke', NULL, NULL),
(8, 8, 'Burpees', 20, 3),
(9, 9, 'Jab-Cross', 30, 4),
(10, 10, 'Deadlifts', 8, 4),
(11, 11, 'Elliptical', NULL, NULL),
(12, 12, 'Overhead Press', 10, 3),
(13, 13, 'Tree Pose', 5, 2),
(14, 14, 'Planks', NULL, 3),
(15, 15, 'Hill Climb', NULL, NULL),
(16, 16, 'Lunges', 12, 3),
(17, 17, 'Freestyle', NULL, NULL),
(18, 18, 'Mountain Climbers', 25, 3),
(19, 19, 'Hooks', 35, 4),
(20, 20, 'Snatches', 6, 5),
(21, 21, 'Treadmill', NULL, NULL),
(22, 22, 'Pull Ups', 8, 4),
(23, 23, 'Downward Dog', 6, 2),
(24, 24, 'Leg Press', 12, 4),
(25, 25, 'Road Cycling', NULL, NULL),
(26, 26, 'Box Jumps', 15, 3),
(27, 27, 'Backstroke', NULL, NULL),
(28, 28, 'Jumping Jacks', 30, 3),
(29, 29, 'Uppercut', 25, 4),
(30, 30, 'Kettlebell Swings', 15, 3),
(31, 31, 'Rower', NULL, NULL),
(32, 32, 'Push Ups', 20, 4),
(33, 33, 'Bridge Pose', 4, 2),
(34, 34, 'Russian Twists', 20, 3),
(35, 35, 'Stationary Bike', NULL, NULL),
(36, 36, 'High Knees', 25, 3),
```

(37, 37, 'Breaststroke', NULL, NULL),
(38, 38, 'Skaters', 20, 3),
(39, 39, 'Speed Bag', 40, 4),
(40, 40, 'Clean and Jerk', 7, 4),
(41, 41, 'Step Aerobics', NULL, NULL),
(42, 42, 'Bicep Curls', 12, 4),
(43, 43, 'Warrior Pose', 4, 3),
(44, 44, 'Crunches', 30, 3),
(45, 45, 'Trail Biking', NULL, NULL),
(46, 46, 'Tricep Dips', 10, 4),
(47, 47, 'Sidestroke', NULL, NULL),
(48, 48, 'Squat Thrusts', 20, 3),
(49, 49, 'Kicks', 35, 4),
(50, 50, 'Tire Flips', 10, 3),
(51, 51, 'Zumba', NULL, NULL),
(52, 52, 'Lat Pulldowns', 10, 4),
(53, 53, 'Cobra Pose', 5, 2),
(54, 54, 'Sit Ups', 25, 3),
(55, 55, 'Spinning', NULL, NULL),
(56, 56, 'Side Lunges', 15, 3),
(57, 57, 'Butterfly', NULL, NULL),
(58, 58, 'Lateral Jumps', 20, 3),
(59, 59, 'Shadow Boxing', 45, 4),
(60, 60, 'Farmer Walk', 8, 4);

/* ---------- **Goals**----------*/

Create Table & Insert Data:

CREATE TABLE Goals (
GoalID INT PRIMARY KEY,
UserID INT,
GoalType VARCHAR(50),
TargetDate DATE,
TargetValue DECIMAL(5,2),
FOREIGN KEY (UserID) REFERENCES Users(UserID)
);

INSERT INTO Goals (GoalID, UserID, GoalType, TargetDate, TargetValue) VALUES
(1, 1, 'Weight', '2023-08-01', 58),
(2, 2, 'Strength', '2023-08-15', 50),
(3, 3, 'Endurance', '2023-09-01', 120),
(4, 4, 'Flexibility', '2023-09-10', 75),
(5, 5, 'Balance', '2023-09-20', 65),
(6, 6, 'Agility', '2023-10-01', 30),

(7, 7, 'Speed', '2023-10-05', 25),
(8, 8, 'Power', '2023-10-15', 80),
(9, 9, 'Coordination', '2023-11-01', 70),
(10, 10, 'Stamina', '2023-11-10', 100),
(11, 11, 'Weight', '2023-08-05', 60),
(12, 12, 'Strength', '2023-08-20', 55),
(13, 13, 'Endurance', '2023-09-05', 130),
(14, 14, 'Flexibility', '2023-09-15', 80),
(15, 15, 'Balance', '2023-09-25', 70),
(16, 16, 'Agility', '2023-10-05', 35),
(17, 17, 'Speed', '2023-10-10', 30),
(18, 18, 'Power', '2023-10-20', 85),
(19, 19, 'Coordination', '2023-11-05', 75),
(20, 20, 'Stamina', '2023-11-15', 110),
(21, 21, 'Weight', '2023-08-10', 62),
(22, 22, 'Strength', '2023-08-25', 60),
(23, 23, 'Endurance', '2023-09-10', 140),
(24, 24, 'Flexibility', '2023-09-20', 85),
(25, 25, 'Balance', '2023-09-30', 75),
(26, 26, 'Agility', '2023-10-10', 40),
(27, 27, 'Speed', '2023-10-15', 35),
(28, 28, 'Power', '2023-10-25', 90),
(29, 29, 'Coordination', '2023-11-10', 80),
(30, 30, 'Stamina', '2023-11-20', 120),
(31, 31, 'Weight', '2023-08-15', 64),
(32, 32, 'Strength', '2023-08-30', 65),
(33, 33, 'Endurance', '2023-09-15', 150),
(34, 34, 'Flexibility', '2023-09-25', 90),
(35, 35, 'Balance', '2023-10-05', 80),
(36, 36, 'Agility', '2023-10-15', 45),
(37, 37, 'Speed', '2023-10-20', 40),
(38, 38, 'Power', '2023-10-30', 95),
(39, 39, 'Coordination', '2023-11-15', 85),
(40, 40, 'Stamina', '2023-11-25', 130),
(41, 41, 'Weight', '2023-08-20', 66),
(42, 42, 'Strength', '2023-09-05', 70),
(43, 43, 'Endurance', '2023-09-20', 160),
(44, 44, 'Flexibility', '2023-09-30', 95),
(45, 45, 'Balance', '2023-10-10', 85),
(46, 46, 'Agility', '2023-10-20', 50),
(47, 47, 'Speed', '2023-10-25', 45),
(48, 48, 'Power', '2023-11-05', 100),
(49, 49, 'Coordination', '2023-11-20', 90),
(50, 50, 'Stamina', '2023-11-30', 140),

(51, 51, 'Weight', '2023-08-25', 68),
(52, 52, 'Strength', '2023-09-10', 75),
(53, 53, 'Endurance', '2023-09-25', 170),
(54, 54, 'Flexibility', '2023-10-05', 100),
(55, 55, 'Balance', '2023-10-15', 90),
(56, 56, 'Agility', '2023-10-25', 55),
(57, 57, 'Speed', '2023-11-01', 50),
(58, 58, 'Power', '2023-11-10', 105),
(59, 59, 'Coordination', '2023-11-25', 95),
(60, 60, 'Stamina', '2023-12-05', 150);

/* ---------- **Progress**----------*/
Create Table & Insert Data:

CREATE TABLE Progress (
ProgressID INT PRIMARY KEY,
UserID INT,
Date DATE,
Weight DECIMAL(5,2),
BodyFatPercentage DECIMAL(5,2),
MuscleMass DECIMAL(5,2),
FOREIGN KEY (UserID) REFERENCES Users(UserID)
);

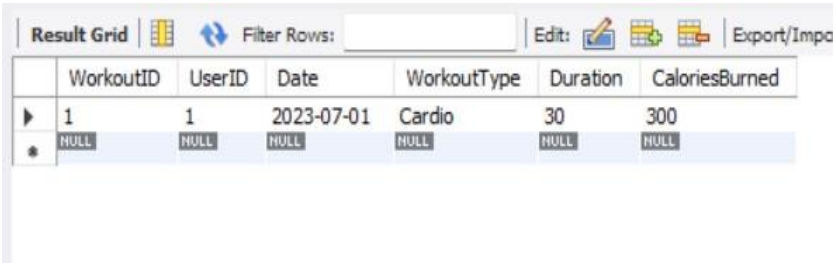
INSERT INTO Progress (ProgressID, UserID, Date, Weight, BodyFatPercentage, MuscleMass) VALUES
(1, 1, '2023-07-01', 60.00, 20.0, 30.0),
(2, 2, '2023-07-02', 75.00, 22.0, 32.0),
(3, 3, '2023-07-03', 70.00, 21.5, 31.0),
(4, 4, '2023-07-04', 55.00, 19.0, 28.0),
(5, 5, '2023-07-05', 65.00, 21.8, 29.5),
(6, 6, '2023-07-06', 85.00, 25.5, 35.0),
(7, 7, '2023-07-07', 54.00, 18.5, 27.0),
(8, 8, '2023-07-08', 78.00, 23.0, 33.0),
(9, 9, '2023-07-09', 59.00, 20.8, 28.5),
(10, 10, '2023-07-10', 90.00, 26.0, 37.0),
(11, 11, '2023-07-11', 62.00, 21.0, 30.5),
(12, 12, '2023-07-12', 77.00, 22.5, 32.5),
(13, 13, '2023-07-13', 64.00, 21.2, 31.0),
(14, 14, '2023-07-14', 74.00, 22.0, 32.0),
(15, 15, '2023-07-15', 60.00, 20.9, 30.0),
(16, 16, '2023-07-16', 82.00, 24.0, 33.5),
(17, 17, '2023-07-17', 63.00, 21.0, 31.0),
(18, 18, '2023-07-18', 73.00, 22.8, 32.5),

(19, 19, '2023-07-19', 58.00, 20.5, 29.0),
(20, 20, '2023-07-20', 88.00, 25.0, 34.5),
(21, 21, '2023-07-21', 61.00, 21.5, 30.5),
(22, 22, '2023-07-22', 80.00, 23.5, 33.0),
(23, 23, '2023-07-23', 62.00, 21.8, 31.5),
(24, 24, '2023-07-24', 57.00, 20.0, 30.0),
(25, 25, '2023-07-25', 68.00, 22.2, 32.0),
(26, 26, '2023-07-26', 56.00, 19.5, 29.0),
(27, 27, '2023-07-27', 65.00, 21.0, 31.0),
(28, 28, '2023-07-28', 60.00, 20.8, 30.5),
(29, 29, '2023-07-29', 74.00, 22.0, 32.5),
(30, 30, '2023-07-30', 85.00, 25.0, 33.5),
(31, 31, '2023-07-31', 63.00, 21.2, 31.0),
(32, 32, '2023-08-01', 79.00, 23.8, 33.0),
(33, 33, '2023-08-02', 68.00, 22.5, 32.0),
(34, 34, '2023-08-03', 66.00, 22.2, 31.5),
(35, 35, '2023-08-04', 70.00, 23.0, 32.0),
(36, 36, '2023-08-05', 75.00, 24.0, 33.0),
(37, 37, '2023-08-06', 62.00, 21.8, 30.5),
(38, 38, '2023-08-07', 81.00, 24.5, 34.0),
(39, 39, '2023-08-08', 64.00, 22.0, 31.0),
(40, 40, '2023-08-09', 86.00, 25.0, 34.5),
(41, 41, '2023-08-10', 60.00, 21.5, 30.0),
(42, 42, '2023-08-11', 78.00, 23.5, 32.5),
(43, 43, '2023-08-12', 69.00, 22.5, 31.0),
(44, 44, '2023-08-13', 55.00, 20.5, 29.0),
(45, 45, '2023-08-14', 64.00, 21.8, 30.0),
(46, 46, '2023-08-15', 77.00, 24.0, 32.0),
(47, 47, '2023-08-16', 68.00, 22.0, 31.0),
(48, 48, '2023-08-17', 73.00, 23.5, 32.0),
(49, 49, '2023-08-18', 60.00, 21.2, 30.5),
(50, 50, '2023-08-19', 85.00, 25.5, 33.5),
(51, 51, '2023-08-20', 69.00, 22.5, 31.5),
(52, 52, '2023-08-21', 62.00, 21.8, 30.0),
(53, 53, '2023-08-22', 80.00, 24.0, 32.5),
(54, 54, '2023-08-23', 66.00, 22.2, 31.5),
(55, 55, '2023-08-24', 72.00, 23.0, 32.0),
(56, 56, '2023-08-25', 58.00, 21.5, 30.0),
(57, 57, '2023-08-26', 67.00, 22.8, 31.0),
(58, 58, '2023-08-27', 74.00, 23.5, 32.0),
(59, 59, '2023-08-28', 60.00, 21.5, 30.5),
(60, 60, '2023-08-29', 82.00, 24.5, 33.0);

Case Study Questions & Answers:

1.Data Retrieval:Retrieve all workouts for a specific user within a date range

```
SELECT * FROM Workouts
WHERE UserID = 1
AND Date BETWEEN '2023-07-01' AND '2023-07-31';
```

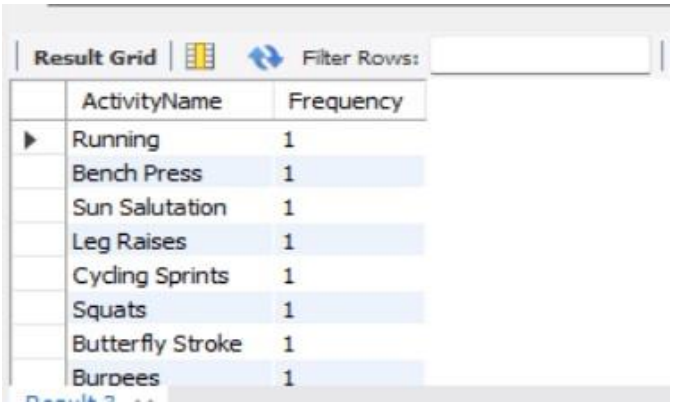


WorkoutID	UserID	Date	WorkoutType	Duration	CaloriesBurned
1	1	2023-07-01	Cardio	30	300
NULL	NULL	NULL	NULL	NULL	NULL

#1 rows returned.

2.Activity Insights:Most common activities across all users.

```
SELECT ActivityName, COUNT(*) AS Frequency
FROM Activities
GROUP BY ActivityName
ORDER BY Frequency DESC;
```



ActivityName	Frequency
Running	1
Bench Press	1
Sun Salutation	1
Leg Raises	1
Cycling Sprints	1
Squats	1
Butterfly Stroke	1
Burpees	1

60rows returned.

3.Calorie Burn:Find the total calories burned by each user in the last month.

```
SELECT DATE_FORMAT(Date, '%Y-%m') AS Month,
SUM(CaloriesBurned) AS  TotalCalories FROM Workouts
WHERE Date >= DATE_SUB(CURDATE(), INTERVAL )1 year GROUP BY Month;
```

Result Grid | Filter Rows:

Month	TotalCalories
2023-07	3995
2023-08	10980

#2rows returned.

4.Goal Completion:Identify users who have met their goals ahead of the target date.

```
SELECT g.UserID, g.GoalType
FROM Goals g
JOIN Workouts w ON g.UserID = w.UserID
GROUP BY g.UserID, g.GoalType, g.TargetDate
HAVING SUM(w.CaloriesBurned) >= MAX(g.TargetValue) AND MAX(w.Date) <=
MAX(g.TargetDate);
```

Result Grid | Filter Rows:

UserID	GoalType
1	Weight
2	Strength
3	Endurance
4	Flexibility
5	Balance

#60rows returned

5.User Progress:Calculate a user's progress towards their fitness goals.

```
SELECT g.GoalType, g.TargetValue, SUM(w.CaloriesBurned) AS TotalCaloriesBurned
FROM Goals g
JOIN Workouts w ON g.UserID = w.UserID
WHERE g.UserID = 1
GROUP BY g.GoalType, g.TargetValue;
```

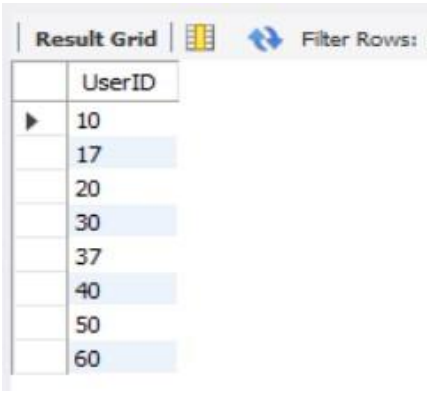
Result Grid | Filter Rows:

GoalType	TargetValue	TotalCalor
Weight	58.00	300

#1 row returned

6.Workout Intensity:List users who burned more than 500 calories in a single workout.

```
SELECT DISTINCT UserID
FROM Workouts
WHERE CaloriesBurned > 500;
```



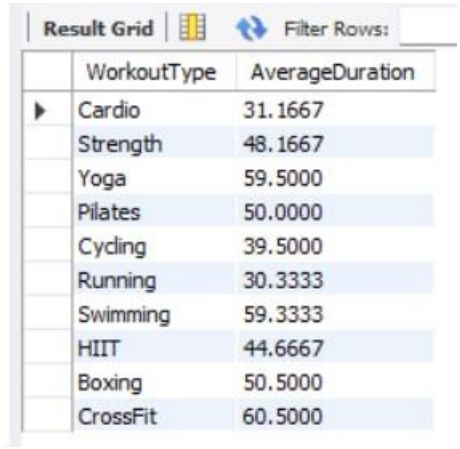
Result Grid | Filter Rows:

UserID
10
17
20
30
37
40
50
60

#8 rows returned.

7. Activity Duration:Calculate the average duration of each workout type.

```
SELECT WorkoutType, AVG(Duration) AS AverageDuration
FROM Workouts
GROUP BY WorkoutType;
```



Result Grid | Filter Rows:

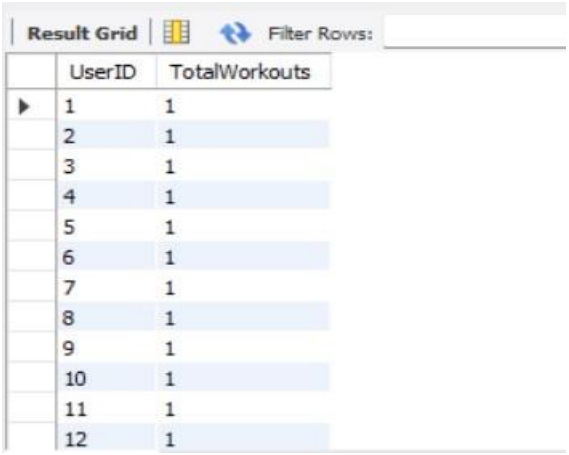
WorkoutType	AverageDuration
Cardio	31.1667
Strength	48.1667
Yoga	59.5000
Pilates	50.0000
Cycling	39.5000
Running	30.3333
Swimming	59.3333
HIIT	44.6667
Boxing	50.5000
CrossFit	60.5000

#10 rows returned.

8. Leaderboard:Create a leaderboard of users based on total workouts completed.

```
SELECT UserID, COUNT(*) AS TotalWorkouts
FROM Workouts
GROUP BY UserID
```

ORDER BY TotalWorkouts DESC;



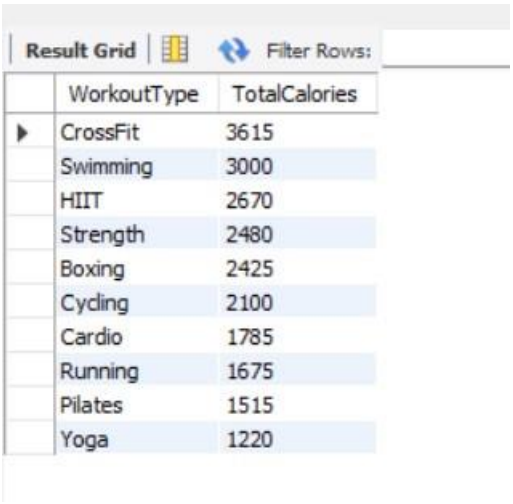
Result Grid | Filter Rows:

UserID	TotalWorkouts
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	1
12	1

#60 rows returned.

9.Retrieves the top 10 workout types based on calories burned.

```
SELECT WorkoutType, SUM(CaloriesBurned) AS TotalCalories
FROM Workouts
GROUP BY WorkoutType
ORDER BY TotalCalories DESC
LIMIT 10;
```



Result Grid | Filter Rows:

WorkoutType	TotalCalories
CrossFit	3615
Swimming	3000
HIIT	2670
Strength	2480
Boxing	2425
Cycling	2100
Cardio	1785
Running	1675
Pilates	1515
Yoga	1220

#10 rows returned.

10.Find the 05 users by muscle mass gain.

```
SELECT UserID, (MAX(MuscleMass) - MIN(MuscleMass)) AS MuscleGain
FROM Progress
GROUP BY UserID
```

```
ORDER BY MuscleGain DESC
LIMIT 05;
```

Result Grid | Filter Rows:

	UserID	MuscleGain
▶	1	0.00
	2	0.00
	3	0.00
	4	0.00
	5	0.00

#5 rows returned.

-SAFAHA PARVEEN S